Oracle® Identity Governance

Configuring the Amazon Web Services Application

12c (12.2.1.3.0)
F34940-01
February 2021

Oracle Identity Governance Configuring the Amazon Web Services Application, 12c (12.2.1.3.0)

F34940-01

Primary Author: Gowri.G.R

Contributors: Syam Kumar Battu

Contents

List of Figures

List of Tables

Preface

This guide describes the connector that is used to onboard the Amazon Web Services application to Oracle Identity Governance.

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 Governance 12.2.1.3.0, visit the following Oracle Help Center page:

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

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

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

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.

Convention	Meaning
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

About the Connector

The Amazon Web Services connector integrates Oracle Identity Governance with the Amazon Web Services target system.
The following topics provide a high-level overview of the Amazon Web Services connector:

1.1 Introduction to the 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 Amazon Web Services connector lets you create and onboard AWS (Amazon Web Services) 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.

> **Note:**
>
> At some places in this guide, Amazon Web Services is sometimes referred to as the **target system**.

1.2 Certified Components

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

Table 1-1 Certified Components

Component	Requirement for AOB Application
Oracle Identity Governance	You can use any one of the following releases: • Oracle Identity Governance 12*c* PS3 (12.2.1.3.0) • Oracle Identity Governance 12*c* PS4 (12.2.1.4.0)
Oracle Identity Governance JDK	JDK 1.8 and later
Target systems	AWS SDK for Java API Reference - 2.13.76
Connector Server	11.1.2.1.0 or 12.2.1.3.0
Connector Server JDK	JDK 1.8 and later

1.3 Usage Recommendation

This is the recommendation for the Amazon Web Services connector version that you can deploy and use depending on the Oracle Identity Governance or Oracle Identity Manager version that you are using.

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.

1.4 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

1.5 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

Table 1-2 (Cont.) Supported Connector Operations

Operation	Supported
Reset Password	Yes
Policy Management	
Add and Remove Policies to Users	Yes
Group Management	
Add and Remove Groups to Users	Yes
Tag Management	
Add and Remove Tags to Users	Yes

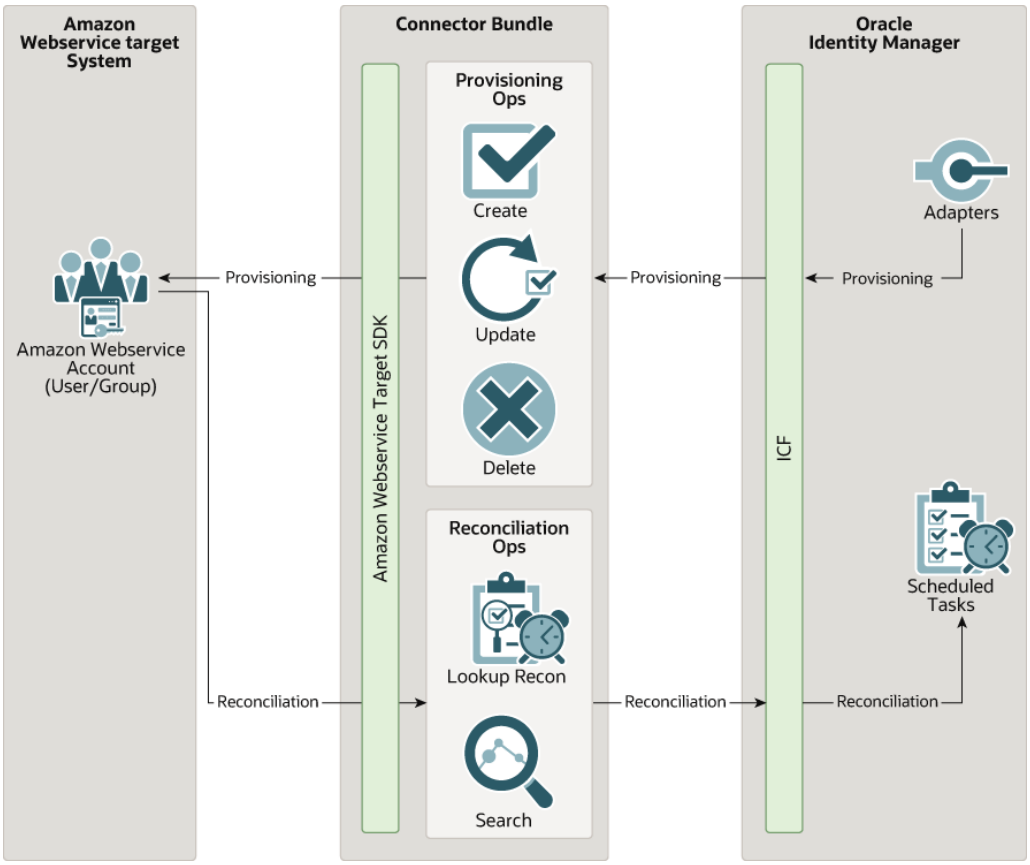
1.6 Connector Architecture

The Amazon Web Services connector 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.

Figure 1-1 shows the architecture of the Amazon Web Services connector.

Figure 1-1 Connector Architecture



The connector is configured to run in the Account management mode. 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 inturn invokes create operation on the Amazon Web Services Identity Connector Bundle and then the bundle calls the Amazon Web Service SDK for provisioning operations. The SDK 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 inturn invokes a search operation on the Amazon Web Services Identity Connector Bundle and then the bundle calls the Amazon Web Service SDK for the reconciliation operation. The SDK 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 Amazon Web Services resources that are already provisioned to OIM Users. If a match is found, then the update made to the Amazon Web Services record from the target system is copied to the Amazon Web Services resource in Oracle Identity Governance. If no match is found, then the userPrincipalName 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 an Amazon Web Services resource to the OIM User.

See Also:

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

1.7 Supported Connector Features Matrix

Provides the list of features supported by the AOB application.

Table 1-3 Supported Connector Features Matrix

Feature	AOB Application
Full reconciliation	Yes
Incremental reconciliation	Yes
Limited reconciliation	Yes
Reconcile deleted user records	Yes
Provide secure communication to the target system through SSL	Yes
Use connector server	Yes
Clone applications or create new application instances	Yes
Transformation and validation of account data	Yes
Support for pagination	Yes
Test connection	Yes

1.8 Features of the Connector

The features of the connector include full and incremental reconciliation, limited reconciliation, transformation and validation of account data and so on.

- Support for Full and Incremental Reconciliation
- Support for Limited (Filtered) Reconciliation
- Reconciliation of Deleted User Records
- Reconciliation of Lookup Definitions
- Support for the Connector Server

- Transformation and Validation of Account Data
- Support for Cloning Applications and Creating Instance Applications
- Secure Communication to the Target System
- Configuring Action Scripts
- Support for Enabling and Disabling Accounts

1.8.1 Support for Full and Incremental Reconciliation

In full reconciliation, all records are fetched from the target system to Oracle Identity Governance. In incremental reconciliation, only records that are added or modified after the last reconciliation run are fetched into Oracle Identity Governance.

You can switch from incremental to full reconciliation at any time after you deploy the connector. See Performing Full Reconciliation and Incremental Reconciliation for more information on performing full and incremental reconciliation runs.

1.8.2 Support for Limited (Filtered) Reconciliation

You can reconcile records from the target system based on a specified filter criterion.

You can set a reconciliation filter as the value of the Filter Query attribute of the user reconciliation scheduled job. This filter specifies the subset of newly added and modified target system records that must be reconciled. The Filter Query attribute helps you to assign filters to the webservices based on which you will get a filtered response from the target system.

See Performing Limited Reconciliation for more information on performing limited reconciliation.

1.8.3 Reconciliation of Deleted User Records

You can configure the connector for reconciliation of deleted user records. In target resource mode, if a user record is deleted on the target system, then the corresponding Exchange User resource is revoked from the OIM User.

For information about the Delete User reconciliation job, see Reconciliation Jobs.

1.8.4 Reconciliation of Lookup Definitions

You can configure the connector for reconciliation of groups and policies in the target system to be populated as entitlements in the lookup definitions on Oracle Identity Governance.

For detailed information about the jobs that are available for reconciling these entitlements, see Reconciliation Jobs.

1.8.5 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 wish 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.

For information about installing, configuring, and running the Connector Server, and then installing the connector in a Connector Server, see Using an Identity Connector Server in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance*.

1.8.6 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 Attributes in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

1.8.7 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 an Instance Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

1.8.8 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 information on SSL, see Configuring SSL.

1.8.9 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 the create, update, or delete an account provisioning operations. For example, you can configure a script to run before every user creation operation.

For more information about configuring these scripts, see Configuring Action Scripts.

1.8.10 Support for Enabling and Disabling Accounts

Enabling User accounts from Oracle Identity Governance will make the Console and Programmatic Access active in the target system if the enableProgrammaticAccess configuration parameter is set to `true`. Only Console access will be active if the configuration parameter is set to `false`.

Disabling user accounts from Oracle Identity Governance makes the Console access and Programmatic Access deactivated in the target system irrespective of the **enableProgrammaticAccess** configuration parameter value. This disables user accounts in Oracle Identity Governance thereby prohibiting them from performing any operation.

Enabling and disabling Oracle Identity Governance account status during reconciliation operation: Oracle Identity Governance account status will be disabled if both the Console access and Programmatic access are deactivated in the target. If either Console access or Programmatic access is actived, Oracle Identity Governance account status will be enabled.

> **Note:**
>
> For disable/enable operations to work, remove/create Login Profile in AWS IAM user respectively.

2

Creating an Application by Using the Connector

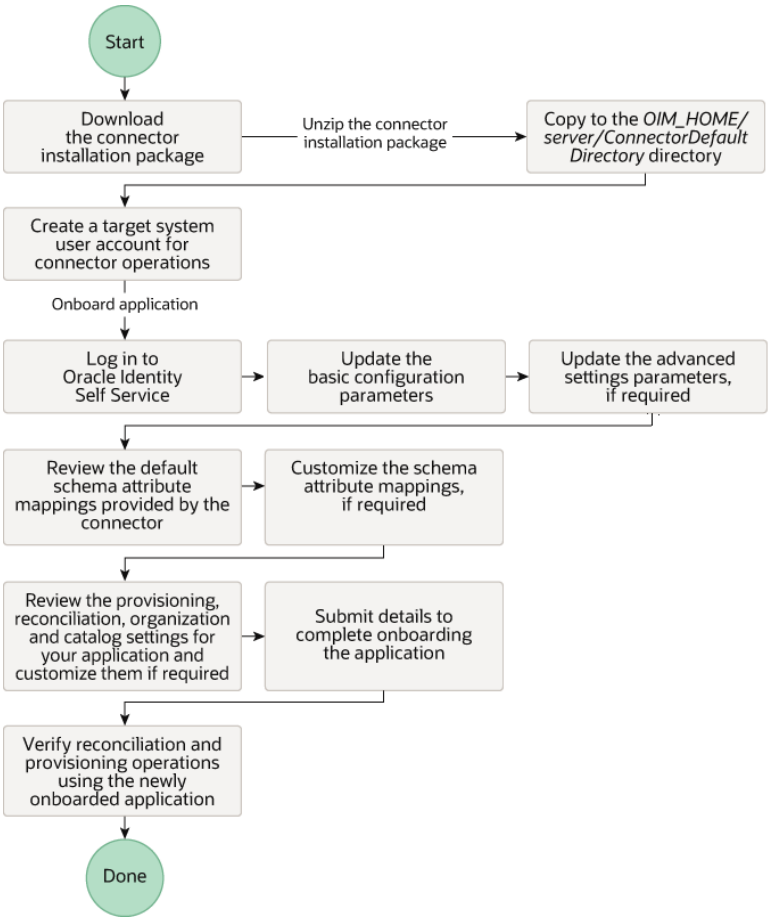
Learn about onboarding applications using the connector and the prerequisites for doing so.

2.1 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



2.2 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.

2.3 Downloading and Copying Third-Party Jar Libraries

You can either use third-party jars from the `AmazonWebServices-12.2.1.3.0 /lib` folder shipped with the connector package or download any latest, stable, and secure version. Please follow the below procedure to include third-party jars:

1. Create a directory named AmazonWebservices-*RELEASE_NUMBER* under the `OIM_ORACLE_HOME/server/ConnectorDefaultDirectory/targetsystems-lib/` directory.
2. Copy the third-party library jars for the Amazon Web Services Apps connector to the computer hosting Oracle Identity Governance present in `OIM_ORACLE_HOME/server/ConnectorDefaultDirectory/targetsystems-lib/AmazonWebServices-RELEASE_NUMBER` directory.

   For example, if you are using release 12.2.1.3.0 version of this connector, then create a directory named `AmazonWebServices-12.2.1.3.0 in the OIM_ORACLE_HOME/server/ConnectorDefaultDirectory/targetsystems-lib/ directory`.

> **Note:**
>
> If you are using Connector Server, copy Amazon Web Services Apps third-party libraries to the `CONNECTOR_SERVER_HOME/lib` directory.

If you are looking for latest third party jar libraries, use the following link to download them:

Table 2-1 Third-Party Jars

Jar Name	Download Link
auth-[Version].jar	https://mvnrepository.com/artifact/software.amazon.awssdk/auth/
iam-[Version].jar	https://mvnrepository.com/artifact/software.amazon.awssdk/iam/
aws-core-[Version].jar	https://mvnrepository.com/artifact/software.amazon.awssdk/aws-core/
sdk-core-[Version].jar	https://mvnrepository.com/artifact/software.amazon.awssdk/sdk-core
regions-[Version].jar	https://mvnrepository.com/artifact/software.amazon.awssdk/regions/
profiles-[Version].jar	https://mvnrepository.com/artifact/software.amazon.awssdk/profiles
utils-[Version].jar	https://mvnrepository.com/artifact/software.amazon.awssdk/utils
organizations-[Version].jar	https://mvnrepository.com/artifact/software.amazon.awssdk/organizations/

Table 2-1 (Cont.) Third-Party Jars

Jar Name	Download Link
sts-[Version].jar	https://mvnrepository.com/artifact/software.amazon.awssdk/sts/
cloudtrail-[Version].jar	https://mvnrepository.com/artifact/software.amazon.awssdk/cloudtrail/
apache-client-[Version].jar	https://mvnrepository.com/artifact/software.amazon.awssdk/apache-client/
http-client-spi-[Version].jar	https://mvnrepository.com/artifact/software.amazon.awssdk/http-client-spi/
aws-json-protocol-[Version].jar	https://mvnrepository.com/artifact/software.amazon.awssdk/aws-json-protocol/
aws-query-protocol-[Version].jar	https://mvnrepository.com/artifact/software.amazon.awssdk/aws-query-protocol/
metrics-spi-[Version].jar	https://mvnrepository.com/artifact/software.amazon.awssdk/metrics-spi/
protocol-core-[Version].jar	https://mvnrepository.com/artifact/software.amazon.awssdk/protocol-core/

2.4 Creating an Application By Using the Amazon Webservice 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 on each of the steps 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.

See Also:

- Configuring the Connector 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

2.5 Creating a Target System User Account for the AWS Target

The following topics describe the procedures to create a target system user account for the AWS target:

- Signing in with Root User Credentials
- Creating an IAM user in the AWS account
- Adding Inline Policy to an IAM User

2.5.1 Signing in with Root User Credentials

To sign in to an AWS account as the root user, perform the following steps:

> **Note:**
>
> To sign in to an AWS account as a root user, ensure you know the email address used to create the AWS account and the password for the root user.

1. Open https://console.aws.amazon.com/.
2. If you have not signed in previously using this browser, select **Root user**, enter the email address associated with your account, click **Next**, enter the password and choose **Sign in**.
3. If you have signed in as a root user previously using this browser, your browser might remember the email address for the AWS account. If so, you just need to enter password and select **Sign in**.

2.5.2 Creating an IAM user in the AWS account

To create an IAM user in the AWS account, perform the following steps:

1. If you are already signed in, under Services, search for **IAM**.
2. From the left navigation pane, under Access Management, select **Users**, and click **Add user**.

   Create a simple user without any permissions.
3. Perform the following steps to add a user in the Add User page:
   a. In the Set user details section, enter the user name (sign-in name for AWS).
   b. In the Select AWS access type section, under **Access type**, select **Programmatic access** and **AWS Management Console access** checkboxes.
   c. To manually enter the user password, under **Console Password**, select **Custom password**.
   d. Select the **Require password reset** checkbox and then select **Next: Permissions**.
4. From the Add User page, select **Set Permissions** and click **Next: Tags** without making any change.
5. From the Add tags (optional) page, click **Next: Review** and review all details used for creating the user and then click on **Create user**. You will receive a success message on the screen after you creating the user.
6. Click **Close**.

2.5.3 Adding Inline Policy to an IAM User

To add inline policies to IAM users, perform the following steps:

1. Using the search field, select the previosly created user from the User name list.
2. From the Summary page, select the **Permissions** tab and then select **Permissions policies**.
3. Click **Add inline policy**. You will be redirected to the **Create Policy** page.
4. Expand **Service** to define Actions and Resources for IAM, Organizations, CloudTrail and STS Services.
5. Click **Choose a service** and search for **IAM**.
6. Expand **Actions**, and then expand **Access level** to assign various access levels.

   From the List access level section, select the following checkboxes:

   - GetLoginProfile
   - ListGroupPolicies
   - ListUserPolicies
   - ListAccessKeys
   - ListGroups
   - ListUsers
   - ListAttachedGroupPolicies
   - ListGroupForUser
   - ListUserTags
   - ListAttachedUserPolicies
   - ListPolicies

   From the Read access level section, select the following checkboxes:

   - GetAccountAuthorizationDetails
   - GetGroup
   - GetPolicy
   - GetUser

   From the Tagging access level section, select the following checkboxes:

   - TagUser
   - UntagUser

   From the Write access level section, select the following checkboxes:

   - AddUserToGroup
   - DeleteLoginProfile
   - UpdateAccessKey
   - CraeteLoginProfile
   - DeleteUser
   - UpdateLoginProfile
   - CreateUser
   - RemoveUserFromGroup
   - UpdateUser

From the Permissions Management access level section, select the following checkboxes:

- AttachUserPolicy
- DeleteUserPolicy
- DetachUserPolicy
- DetachGroupPolicy

7. From the Resources section, select **All resources**, and click **Review policy**.

   In the Review Policy page, ensure to enter a name for your policy and click the **Create policy** button. The policy will be added to the user in permission tab.

8. To define Actions and Resources for Organizations, repeat steps 1 to 5 of this section with a minor change. While choosing a service in step 5, select **Organisations** instead of IAM.

9. Expand **Actions**, and then expand **Access level** to assign various access levels. Select the following checkboxes List and Read access level sections:

   - List access level section: **ListPoliciesForTarget**
   - Read access level section: **DescribeAccount** checkbox

10. From the Resources section, select **All resources**, and click **Review policy**.

    In the Review Policy page, ensure to enter a name for your policy and click the **Create policy** button. The policy will be added to the user in permission tab.

11. To define Actions and Resources for CloudTrial, repeat steps 1 to 5 of this section with a minor change. While choosing a service in step 5, select **CloudTrial** instead of IAM.

12. Expand **Actions**, and then expand **Access level** to assign the access level. From the Read access level section, select the Lookup Events chekbox.

13. From the Resources section, select **All resources**, and click **Review policy**.

    In the Review Policy page, ensure to enter a name for your policy and click the **Create policy** button. The policy will be added to the user in permission tab.

14. To define Actions and Resources for STS, repeat steps 1 to 5 of this section with a minor change. While choosing a service in step 5, select **STS** instead of IAM.

15. Expand **Actions**, and then expand **Access level** to assign the access level. From the Read access level section, select the GetCallerIdentity chekbox.

With this, you have successfully created an IAM user with four inline policies for each service. The same IAM user can be used as a communication user in Oracle Identity Governance to perform all the connector operations.

3
Configuring the Connector

While creating a target application, you must configure connection-related parameters that the connector uses to connect to 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.

3.1 Basic Configuration Parameters

These are the connection-related parameters that Oracle Identity Governance requires to connect to an Amazon Web Services application.

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

Table 3-1 Parameters in the Basic Configuration

Parameter	Mandatory ?	Description
userName	Yes	Enter the user name of the target system that you create for performing connector operations. **Sample value:** `johndoe`
accessKeyId	Yes	Enter the access key identifier (a unique string) issued by the authorization server to your client application during the registration process. You would have obtained the access key while configuring the newly registered application. **Sample value:** `AKIA33FL36M3OIF5C7N2`

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

Parameter	Mandatory ?	Description
secretAccessKey	Yes	Enter the secret Access Key used to authenticate the identity of your client application. You obtained the secret Access Key while performing the procedure described in Configuring the Newly Added Application. **Sample value:** `HWuvCMIptAhT5YmBx8ee0GpVVkyMBWLmqxJcf621`
proxyPassword	No	Enter the proxy password if you are using proxy server to access internet.
proxyHostPort	No	Enter the proxy host or IP and port if you are using proxy server to access internet. **Sample value:** `http://host:port`
proxyUsername	No	Enter the proxy username if you are using proxy server to access internet.

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 below table are mandatory.

Table 3-2 Advanced Settings Parameters

Parameter	Description
Bundle Name	This entry holds the name of the connector bundle. **Default value**: `org.identityconnectors.aws`
Bundle Version	This entry holds the version of the connector bundle. **Default value**: `12.3.0`

Table 3-2 (Cont.) Advanced Settings Parameters

Parameter	Description
Connector Name	This entry holds the name of the connector class. **Default value**: `org.identityconnectors.aws.AWSConnector`
pageSize	Specify the number of objects to return in a page from the target system in a paged search. **Default value**: `25`
passwordLastUsed	Enter `true` to display the Last activity attribute value in the parent form. **Default value**: `False`
region	Enter the Region for IAM and Organization. **Default value**: `aws-global`
cloudTrailRegion	Enter the region for Cloudtrail service used for incremental reconciliation **Default value**: `us-east-2`
policyGroup	Enter `true` to fetch inherit policies **Default value**: `False`
changePasswordNextSignIn	Enter `true` to force password change on next login. **Default value**: `False`
enableProgrammaticAccess	Enter `true` to enable the programmatic access. **Default value**: `False`
timeZone	This parameter displays the Oracle Identity Manager timezone. **Default value**: `IST`

3.3 Attribute Mappings

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.

Default Attributes for Amazon Web Services Target Application

Table 3-3 lists the user-specific attribute mappings between the process form fields in Oracle Identity Governance and Amazon Web Services 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.

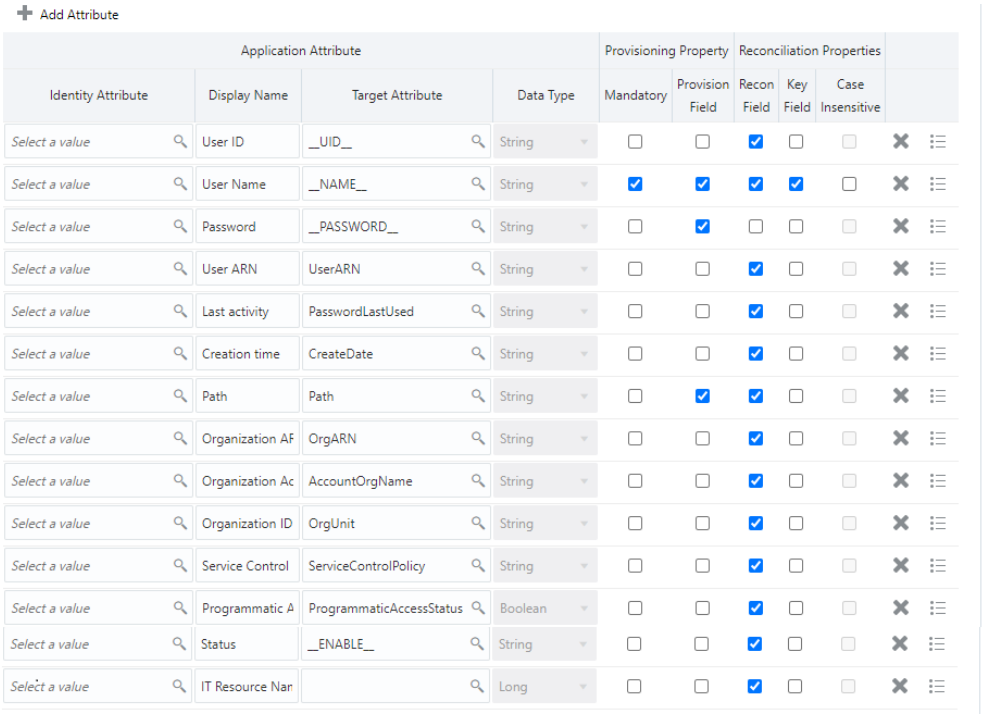
If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described in Creating a Target Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-3 Default Attributes for Amazon Web Services Target Application

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Provision Field?	Recon Field?	Key Field?	Case Insensitive?	Advanced Flag Settings
User ID	__UID__	String	No	No	Yes	No	No	Length:256
User Name	__NAME__	String	Yes	Yes	Yes	Yes	No	Length:64
Password	__PASSWORD__	String	No	Yes	No	No	No	Length:128
User ARN	UserARN	String	No	No	Yes	No	No	WriteBack; Length:2048
Last Activity	PasswordLastUsed	String	No	No	Yes	No	No	WriteBack; Length:256
Creation Time	CreateDate	String	No	No	Yes	No	No	WriteBack; Length:256
Path	Path	String	No	Yes	Yes	No	No	Length:512
Organization ARN	OrgARN	String	No	No	Yes	No	No	WriteBack; Length:2048
Organization Account Name	AccountOrgName	String	No	No	Yes	No	No	WriteBack; Length:50
Organization ID	OrgUnit	String	No	No	Yes	No	No	WriteBack; Length:50
Service Control Policies	ServiceControlPolicy	String	No	No	Yes	No	No	WriteBack; Length:256
Status	__ENABLE__	String	No	No	Yes	No	No	WriteBack; Length:256
Programmatic Access Status	ProgrammaticAccessStatus	Boolean	No	No	Yes	No	No	WriteBack
IT Resource Name		Long	No	No	Yes	No	No	

Table 3-3 shows the default User account attribute mappings.

Figure 3-1 Default Attribute Mappings for Amazon Web Services User Account



> **Note:**
>
> Ensure that the path begins and ends with `/`.
>
> Default value: `/`
>
> Example: `/Oracle/`

Groups Attribute

Table 3-4 lists the group forms attribute mappings between the process form fields in Oracle Identity Governance and Amazon Web Services 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.

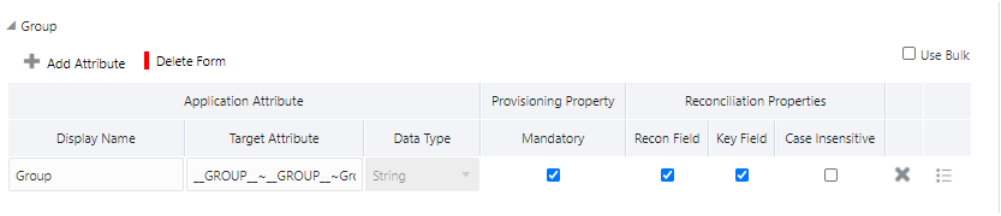
If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described in Creating a Target Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-4 Default Attribute Mappings for Groups

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Recon Field?	Key Field?	Case Insensitive?	Entitlement	Advanced settings
Group	__GROUP__~__GROUP__~GroupName	String	Yes	Yes	Yes	No	True	List of Values: Lookup.AWS.Group Length:256

Figure 3-2 shows the default attribute groups mapping.

Figure 3-2 Default Attribute Mappings for Groups



Policies Attribute

Table 3-5 lists the policy attribute mappings between the process form fields in Oracle Identity Governance and Amazon Web Services 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.

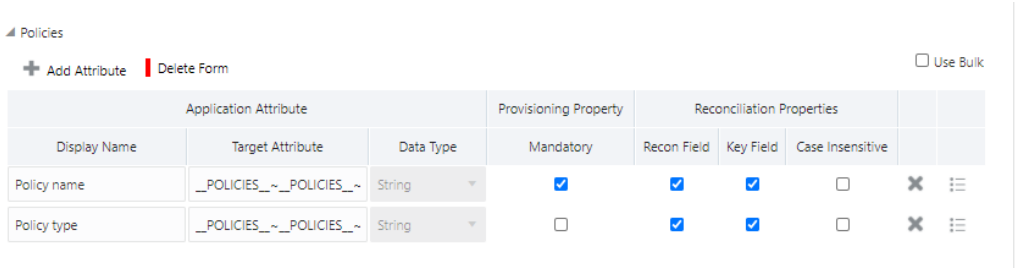
If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described in Creating a Target Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-5 Default Attribute Mappings for Policies

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Recon Field?	Key Field?	Case Insensitive?	Entitlement	Advanced Settings
Policy name	__POLICIES__~__POLICIES__~policyName	String	Yes	Yes	Yes	No	True	List of Values: Lookup.AWS.Policy Length:256
Policy type	__POLICIES__~__POLICIES__~policyType	String	No	Yes	Yes	No		Length:256

Figure 3-3 shows the default attribute policy mapping.

Figure 3-3 Default Attribute Mappings for Policies



Tags Attribute

Table 3-6 lists the tag attribute mappings between the process form fields in Oracle Identity Governance and Amazon Web Services 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.

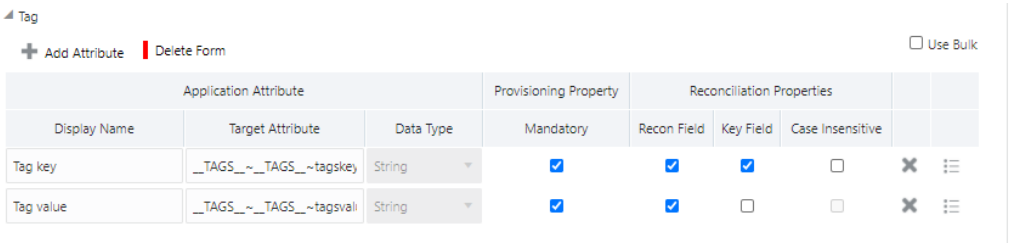
If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described in Creating a Target Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-6 Default Attribute Mappings for Tags

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Recon Field?	Key Field?	Case Insensitive?	Advanced Settings
Tag key	__TAGS__~__TAGS__~tagskey	String	Yes	Yes	Yes	No	Length:256
Tag value	__TAGS__~__TAGS__~tagsvalue	String	Yes	Yes	No	No	Length:256

Figure 3-4 shows the default attribute tag mapping.

Figure 3-4 Default Attribute Mappings for Tags



Inline Policy Attribute

Table 3-7 lists the inline policy attribute mappings between the process form fields in Oracle Identity Governance and Amazon Web Services 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.

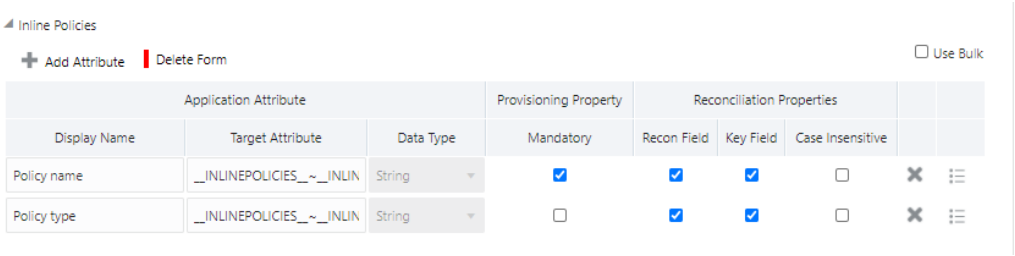
If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described in Creating a Target Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-7 Default Attribute Mappings for Inline Policies

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Recon Field?	Key Field?	Case Insensitive?	Advanced Settings
Policy Name	__INLINEPOLICIES__~__INLINEPOLICIES__~InlinePolicyName	String	Yes	Yes	Yes	No	Length:256
Policy type	__INLINEPOLICIES__~__INLINEPOLICIES__~InlinePolicyType	String	No	Yes	Yes	No	Length:256

Figure 3-5 shows the default attribute inline policiy mapping.

Figure 3-5 Default Attribute Mappings for Inline Policies



3.4 Correlation Rules, Situations, and Responses for a 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 Amazon Web Services 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.

Table 3-8 lists the default simple correlation rule for Amazon Web Services connector. If required, you can edit the default correlation rule or add new rules. You can create

simple correlation rules also. For more information about adding or editing simple or complex correlation rules, see Updating Identity Correlation Rule in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-8 Predefined Identity Correlation Rule for a Amazon Web Services Target Application

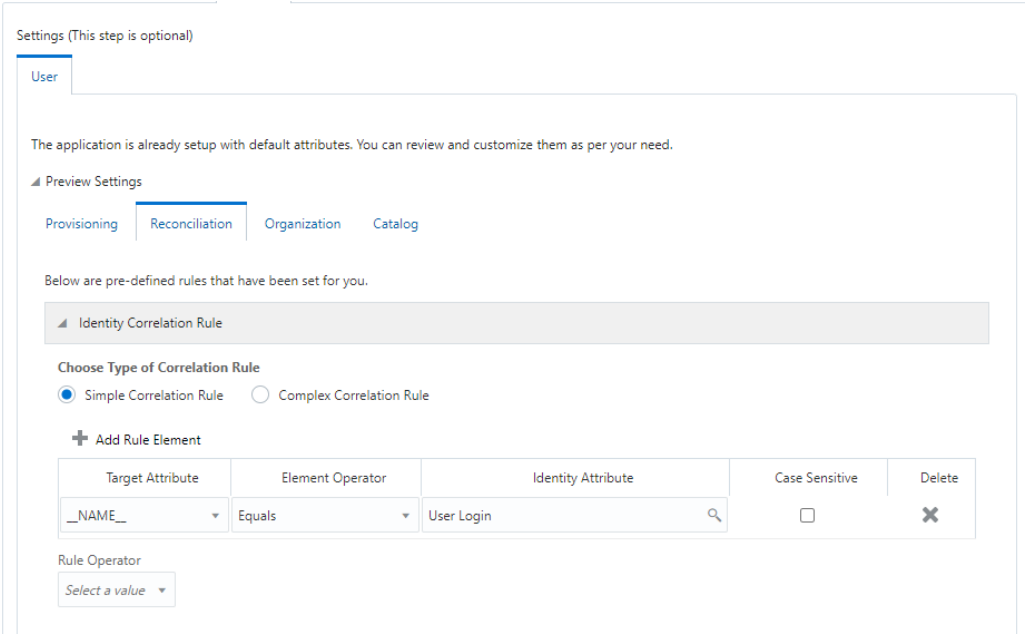
Target Attribute	Element Operator	Identity Attribute	Case Sensitive?	Rule Operator
__NAME__	Equals	User Login	No	

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.

Figure 3-6 shows the simple correlation rule for a Amazon Web Services target application.

Figure 3-6 Simple Correlation Rule for a Amazon Web Services Target Application



Predefined Situations and Responses

The Amazon Web Services 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.

Table 3-9 lists the default situations and responses for Amazon Web Services 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

Creating a Target Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-9 Predefined Situations and Responses for a Amazon Web Services Target Application

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

3.5 Reconciliation Jobs

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

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*.

User Reconciliation Job

The Amazon Webservice Target Resource User Reconciliation job is used to reconcile user data from a target application.

Table 3-10 Parameters of the Amazon Webservice Target Resource User Reconciliation Job

Parameter	Description
Application Name	Name of the application you created for your target system. This value is the same as the value that you provided for the Application Name field while creating your target application. Do not modify this value.
Scheduled Task Name	This parameter holds the name of the scheduled job. **Note:** For the scheduled job included with this connector, you must not change the value of this parameter. However, if you create a new job or create a copy of the job, then enter the unique name for that scheduled job as the value of this parameter. Default value: *APP_NAME* AWS Target Resource User Reconciliation
Filter Query	Enter the search filter for fetching user records from the target system during a reconciliation run. See Performing Limited Reconciliation for more information about this attribute.

Table 3-10 (Cont.) Parameters of the Amazon Webservice Target Resource User Reconciliation Job

Parameter	Description
Object Type	This attribute holds the name of the object type for the reconciliation run. Default value: `User` Do not change the default value.
Sync Token	This attribute holds the date and time stamp at when the last full or incremental reconciliation run started. **Default value**: `<String>0</String>` Note: • If you are running a schedule job with incremental reconciliation, sync token will be updated automatically. • If you know a valid value for sync token, you can enter it in the following example format: `<String>2020-05-19T18:29:49</String>` • This attribute stores values in an XML serialized format.

Delete User Reconciliation Job

The Amazon Web Services Target Resource Delete User Reconciliation job is used to reconcile deleted user data from a target application.

Table 3-11 Parameters of the Amazon Web Services Target Resource Delete User Reconciliation Job

Parameter	Description
Application Name	Name of the application you created for your target system. This value is the same as the value that you provided for the Application Name field while creating your target application. Do not modify this value.
Object Type	This attribute holds the name of the object type for the reconciliation run. Default value: `User` Do not change the default value.

Reconciliation Jobs for Entitlements

The following jobs are available for reconciling entitlements:

- Amazon Web Services Group Lookup Reconciliation
- Amazon Web Services Policy Lookup Reconciliation

The parameters for both the reconciliation jobs are the same.

Table 3-12 Parameters of the Reconciliation Jobs for Entitlements

Parameter	Description
Application Name	Name of the application you created for your target system. This value is the same as the value that you provided for the Application Name field while creating your target application. Do not modify this value.
Code Key Attribute	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). Default value: `__UID__` **Note:** Do not change the value of this attribute.
Decode Attribute	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). Default value: `__NAME__` **Note:** Do not change the value of this attribute.
Lookup Name	This parameter 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 reconciliation job you are using, the default values are as follows: • For Amazon Web Services Group Lookup Reconciliation - `Lookup.AWS.Group` • For Amazon Web Services Policy Lookup Reconciliationn - `Lookup.AWS.Policy`
Object Type	Enter the type of object whose values must be synchronized. Depending on the reconciliation job you are using, the default values are as follows: • For Amazon Web Services Group Lookup Reconciliation - `__GROUP__` • For Amazon Web Services Policy Lookup Reconciliation - Reconciliation - `__POLICY__` **Note:** Do not change the value of this attribute.

4

Performing Postconfiguration 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 Log Levels
- Enabling Logging

4.3.1 Understanding Log Levels

When you enable logging, Oracle Identity Manager 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 Manager 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 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 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
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_SEVER` are the domain name and server name specified during the installation of Oracle Identity Manager.

4.3.2 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='AWS-handler'
level='[LOG_LEVEL]'class='oracle.core.ojdl.logging.ODLHandlerFact
ory'>
    <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.AWS" level="[LOG_LEVEL]"
useParentHandlers="false">
    <handler name="AWS-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 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='AWS-handler'
level='NOTIFICATION:1'class='oracle.core.ojdl.logging.ODLHandlerF
actory'>
    <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.AWS" level="NOTIFICATION:1"
useParentHandlers="false">
    <handler name="AWS-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.

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

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 Resource of *Oracle Fusion Middleware Administering Oracle Identity Governance*. While creating the IT resource, ensure to use to select **Connector Server** from the **IT Resource Type** list.

In addition, specify values for the parameters of the IT resource for the Connector Server listed in Table 4-2

Table 4-2 Parameters of the IT Resource for the 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 Oracle Identity Governance 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.xl
   f
   ```

   > **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 AWS 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_NAME__c_description']}">
<source>Username</source><target/>
</trans-unit>
<trans-unit
id="$
{adfBundle['oracle.adf.businesseditor.model.util.BaseRuntimeResou
rceBundle']
['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity
.userEO.UD_AWSApp_USER_NAME__c_description']}"><source>User
Name</source><target/></trans-unit><trans-unit
id="sessiondef.oracle.iam.ui.runtime.form.model.AWSApp.entity.AWS
AppEO.UD_AWSApp_USER_NAME__c_LABEL"><source>User
Name</source><target/>
</trans-unit>
```

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

```
global.udf.UD_AWS_USER_NAME=\u30E6\u30FC\u30B6\u30FC\u540D
```

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

```
<trans-unit id="$
{adfBundle['oracle.adf.businesseditor.model.util.BaseRuntimeResou
rceBundle']
['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity
.userEO.UD_USER_NAME__c_description']}">
<source>Username</source>
<trans-unit
id="$
{adfBundle['oracle.adf.businesseditor.model.util.BaseRuntimeResou
rceBundle']
['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity
.userEO.UD_AWSApp_USER_NAME__c_description']}"><source>User
Name</source> <target>\u30E6\u30FC\u30B6\u30FC\u540D</target></
trans-unit><trans-unit
id="sessiondef.oracle.iam.ui.runtime.form.model.AWSApp.entity.AWS
AppEO.UD_AWSApp_USER_NAME__c_LABEL"><source>User
Name</source> <target>\u30E6\u30FC\u30B6\u30FC\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 AWS 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 AWS.
2. Copy the public key certificate of AWS 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 <password>
   ```

   The following are sample values for this command:

   - ```
     keytool -import -keystore <JAVA_HOME>/jre/lib/security/cacerts -
     file <Cert_Location>/AmazonRootCA1.crt -storepass <password> -alias
     AmazonRootCA1
     ```
   - ```
     keytool -import -keystore <WL_HOME>/server/lib/DemoTrust.jks
     -file <Cert_Location>/AmazonRootCA1.crt -storepass
     DemoTrustKeyStorePassPhrase -alias AmazonRootCA1
     ```

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
- 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 it to meet your requirements.

This chapter is discusses the following topics:

> **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
- Configuring Reconciliation Jobs
- Performing Provisioning Operations
- Performance Recommendation for the Amazon Web Services connector
- Uninstalling the Connector

5.1 Configuring Reconciliation

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

This section provides information on the following topics related to configuring reconciliation:

- Performing Full Reconciliation and Incremental Reconciliation
- Performing Limited Reconciliation

5.1.1 Performing Full Reconciliation and Incremental 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.

At the end of the reconciliation run, the connector automatically sets the Latest Token parameter of the job for user record reconciliation to the time stamp at which the run ended. From the next run onward, the connector considers only records created or modified after this time stamp for reconciliation. This is incremental reconciliation.

You can switch from incremental reconciliation to full reconciliation whenever you want to ensure that all target system records are reconciled in Oracle Identity Governance. To perform a full reconciliation run, ensure that no values are specified for the following parameters of the jobs for reconciling user records:

- Filter

- Latest Token

> **Note:**
>
> Incremental reconiliation leverages AWS CloudTrail capability. Hence, there can be a slight delay for the changes to reflect on CloudTrail.

5.1.2 Performing Limited Reconciliation

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

Limited or filtered reconciliation is the process of limiting the number of records being reconciled based on a set filter criteria. 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 Query parameter (a reconciliation job parameter) that allows you to use various filter conditions to filter the target system records. When you specify a value for the Filter Query parameter, the connector reconciles only the target system records that match the filter criterion into Oracle Identity Governance.

The following are filters that are supported by the Amazon Web Services connector:

- Filter Account using UserName

  For example, UserName=Alex

  Here any user with UserName Alex is reconciled.
- Filter Account using Path
  - For example, Path=/

    Here all users with path as / is reconciled.
  - For example, Path=/Oracle/

    Here all users with path under Oracle folder and sub folder are reconciled.
  - For example, Path=/Ora

    Here all users with path, folder starting with Ora (for example Oracle or OracleAdmin) and sub folder users are reconciled.

> **Note:**
>
> Amazon Web Services connector does not support any other filters.

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

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

   In addition to modifying the job details, you can enable or disable a job.
5. On the **Job Details** tab, in the Parameters region, specify values for the attributes of the scheduled task.

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

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

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

5.3 Performing Provisioning Operations

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

To perform provisioning operations in Oracle Identity Governance:

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

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

5.4 Performance Recommendation for the Amazon Web Services connector

You can improve the performace of full and incremental reconciliation operations.

To improve the full reconciliation performance, in the **Advanced configuration** settings, set the value for **PolicyGroup** and **PasswordLastUsed** configuration attributes to `False`. With this configuration change, the values for **Inherited policies in the child policy table** and **Password Last Used** attributes in the account form will show as blank.

To improve the filter reconciliation performance, it is recommended to use **USERNAME** as the filter value. Path filter will take more time due to extra calls based on the users for the specified path.

5.5 Uninstalling the Connector

Uninstalling the connector deletes all the account-related data associated with its resource objects.

If you want to uninstall the connector for any reason, then run the Uninstall Connector utility. Before you run this utility, ensure that you set values for `ObjectType` and `ObjectValues` properties in the ConnectorUninstall.properties file. For example, if you want to delete resource objects, scheduled tasks, and scheduled jobs associated with the connector, then enter `"ResourceObject", "ScheduleTask", "ScheduleJob"` as the value of the `ObjectType` property and a semicolon-separated list of object values corresponding to your connector as the value of the `ObjectValues` property.

> **Note:**
>
> If you set values for the `ConnectorName` and `Release` properties along with the `ObjectType` and `ObjectValue` properties, then the deletion of objects listed in the `ObjectValues` property is performed by the utility and the Connector information is skipped.

For more information, see Uninstalling Connectors in *Oracle Fusion Middleware Administering Oracle Identity Governance*.

6
Extending the Functionality of the Connector

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

This chapter contains the following sections:

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 the create, update, or delete an account provisioning operations. 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 Tenants

You must clone the application of your base application to configure it for multiple tenants.

The following example illustrates this requirement:

XYZ corporation has multiple tenants including an independent schema. 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
Frequently Asked Questions

Use these Frequently Asked Questions (FAQs) as guidelines and to troubleshoot connector issues.

1. What is Programmatic Access status attribute?

   Answer: Programmatic Access status attribute is a checkbox which shows the status of the programmatic access in the AWS target system. If the **enableProgrammaticAccess** configuration parameter is set to `true`, this checkbox will be updated during enable operation.

   > **Note:**
   >
   > - Since the **Programmatic Access Status** attribute is a write-back field, do not manually update it from Oracle Identity Governance.
   > - This checkbox is updated during reconciliation irrespective of the configuration parameter value.

2. What happens if the **ChangePasswordNextSignIn** flag is set to `true`?

   Answer: If the **ChangePasswordNextSignIn** flag is set to `true`, the **IAMUserChangePassword** policy will be added by default.

3. Why does enablement from Oracle Identity Governance fail for user accounts created through reconciliation?

   Answer: Password will be out of syncronization between the AWS target system and Oracle Identity Governance for user accounts created through reconciliation. So, after completion of initial reconciliation, perform reset password to ensure that password is synchronized between AWS and the Oracle Identity Governance systems.

   Note: This is applicable to enabling user by creating **Login Profile** in AWS and performing subsequent reconciliation to Oracle Identity Governance.

4. Why reset password dysfunctions if the path value is set to anything other than the default path(/)?

   Answer: It is a limitation from AWS. To reset password from Oracle Identity Governance, follow any one of the below steps in the AWS target system:

   a. Attach the **iam:ChangePassword** policy to the user.

   b. Select the **allow user to change their own password** check box from the set of rules customized inside the password policy from **Account settings**.

5. How are AWS inline policies supported from Oracle Identity Governance?

   Answer: Inline policies are fetched from AWS through reconciliation to the Oracle Identity Governance account. Connector supports detachment of inline policies from the account. Inline polices can be attached only from AWS.

6. When is the Policy Type value reflected in Oracle Identity Governance?

   Answer: Policy type value will be reflected in Oracle Identity Governance only after user reconciliation operation is completed.

7. What are the AWS Password-non-alphanumeric supported characters?

   Answer: (!@#$%^&*()_+-=[]{}|') are the AWS Password-non-alphanumeric supported characters.

8. Can you remove group inherited AWS policies from Oracle Identity Governance?

   Answer: No, you cannot remove Group inherited policy attached to user accounts from Oracle Identity Governance. However, removing a group from the Oracle Identity Governance account will remove the corresponding group and inherited policies of the user in the AWS target system. To reflect changes in Oracle Identity Governance, run the account reconciliation job.

8
Known Issues and Workarounds

These are the known issues and workarounds associated with this release of the connector.

8.1 Oracle Identity Governance Issue

This is an issue associated with Oracle Identity Governance.

Retry Create User Account Does Not Work

Retry create user account does not work when the password does not satisfy the Target password policy.

Workaround: There is no workaround for this issue currently.

8.2 AWS Target System Issue

This is an issue associated with AWS target system.

Incremental Reconciliation Does Not Fetch Updated Users

Incremental reconciliation operation uses Cloud Trail Events where updated user details are not fetched immediately.

Workaround: Before performing an incremental reconciliation operation, wait for a few minutes for users to be reflected in Cloud Trail Events.

A

Files and Directories in the Connector Installation Package

These are the components of the connector installation package that comprise the Amazon Web Services connector.

Table A-1 Files and Directories in the Amazon Web Services Connector Installation Package

File in the Installation Package	Description
/bundle/org.identityconnectors.aws-12.3.0.jar	This JAR is the ICF connector bundle.
configuration/AmazonWebservices-CI.xml	This file is used for installing a CI-based connector. This XML file contains configuration information that is used by the Connector Installer during connector installation.
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/AmazonWebservices-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.
/lib/	This folder is used to list all AWS Connector dependency jars. The following is a list of dependency jards shipped out of the box: • auth-2.13.76.jar • iam-2.13.76.jar • aws-core-2.13.76.jar • sdk-core-2.13.76.jar • regions-2.13.76.jar • profiles-2.13.76.jar • utils-2.13.76.jar • organizations-2.13.76.jar • sts-2.13.76.jar • cloudtrail-2.13.76.jar • apache-client-2.13.76.jar • http-client-spi-2.13.76.jar • aws-json-protocol-2.13.76.jar • aws-query-protocol-2.13.76.jar • metrics-spi-2.13.76.jar • protocol-core-2.13.76.jar