Oracle® Cloud

Administering Oracle Identity Cloud Service

Release 22.4.96
E55882-91
October 2024

Oracle Cloud Administering Oracle Identity Cloud Service, Release 22.4.96

E55882-91

Contents

2 Understand Application Integration

Part II Perform Identity Administration

3 Manage Oracle Identity Cloud Service Users

4 Manage Oracle Identity Cloud Service Groups

6 Manage Oracle Identity Cloud Service Jobs

7 Run Oracle Identity Cloud Service Reports

8 Manage Oracle Identity Cloud Service Secondary Instances

Part III Configure Administrator Settings

9 Change Oracle Identity Cloud Service Default Settings

10 Manage User Settings in Oracle Identity Cloud Service

11 Manage Oracle Identity Cloud Service Trusted Partner Certificates

12 Customize Oracle Identity Cloud Service Notifications

13 Manage Oracle Identity Cloud Service Password Policies

17 Manage Microsoft Active Directory (AD) Bridges for Oracle Identity Cloud Service

28 Manage Oracle Identity Cloud Service Network Perimeters

29 Manage Oracle Identity Cloud Service App Gateways

43 Use the E-Business Suite Asserter to Enable SSO for Oracle E-Business Suite with Oracle Identity Cloud Service

Preface

Welcome to *Administering Oracle Identity Cloud Service*.

This document is primarily for users who perform administrative functions with Oracle Identity Cloud Service. These users are responsible for managing user accounts, groups, applications, jobs, reports, default settings, user settings, trusted partner certificates, notifications, password policies, bridges, session settings, self-registration profiles, SDKs, schemas, Terms of Use, Adaptive Security, identity providers, identity provider policies, sign-on policies, App Gateways, account recovery, Multi-Factor Authentication, OAuth settings, and delegated authentication in Oracle Identity Cloud Service.

Topics:

Audience

This document is primarily for users who perform administrative functions with Oracle Identity Cloud Service.

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.

Feature Limitations

This guide documents the complete set of Oracle Identity Cloud Service features. Your localized version of Oracle Identity Cloud Service might contain a subset of these features. Therefore, you might find features in this documentation that are not available in your localized version of Oracle Identity Cloud Service.

Related Resources

- Using Oracle Identity Cloud Service
- Integrating Oracle Identity Cloud Service
- Known Issues for Oracle Identity Cloud Service
- REST API for Oracle Identity Cloud Service
- What's New for Oracle Identity Cloud Service
- Oracle Identity Cloud Service Infographics
- Oracle Identity Cloud Service Sample Applications
- Oracle Identity Cloud Service Solutions
- Oracle Identity Cloud Service Tutorials
- Oracle Identity Cloud Service Videos

Conventions

The following text conventions are used in this guide:

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

Part I
Get Started

Learn how to get started with Oracle Identity Cloud Service.

Chapters:

1

Get Started with Oracle Identity Cloud Service

The following sections describe how to get started with Oracle Identity Cloud Service for Oracle Cloud administrators and users. Familiarity with Oracle Cloud services is assumed.

Topics:

About Oracle Identity Cloud Service

Oracle Identity Cloud Service provides identity management, single sign-on (SSO), and identity governance for applications on-premise, in the cloud, or for mobile devices. Employees and business partners can access applications at any time, from anywhere, and on any device in a secure manner.

Oracle Identity Cloud Service integrates directly with existing directories and identity management systems, and makes it easy for users to get access to applications. It provides the security platform for Oracle Cloud, which allows users to securely and easily access, develop, and deploy business applications such as Oracle Human Capital Management (HCM) and Oracle Sales Cloud, and platform services such as Oracle Java Cloud Service, Oracle Business Intelligence (BI) Cloud Service, and others.

Administrators and users can use Oracle Identity Cloud Service to help them effectively and securely create, manage, and use a cloud-based identity management environment without worrying about setting up any infrastructure or platform details.

Using Oracle Identity Cloud Service, you can:

- **Manage your users, groups, and applications.** Tailor the relationships that your users and groups have with your cloud-based Oracle applications and custom applications. See About Oracle Identity Cloud Service User Accounts and Groups and About Oracle Identity Cloud Service Applications and Application Roles.

- **Manage jobs.** Bulk load data from other repositories into Oracle Identity Cloud Service, view jobs and job details, and export job errors. See Managing Oracle Identity Cloud Service Jobs.
- **Run reports.** Run operational or historical reports that capture data about Oracle Identity Cloud Service. See Running Oracle Identity Cloud Service Reports.
- **Manage default settings.** Change the default and session settings for your identity domain. See Change Oracle Identity Cloud Service Default Settings.
- **Manage user settings.** Change settings for user accounts. See Manage User Settings in Oracle Identity Cloud Service.
- **Manage certificates for your trusted partners.** Oracle Identity Cloud Service uses trusted partner certificates that have Distinguished Encoding Rules (DER) file extensions. See Manage Oracle Identity Cloud Service Trusted Partner Certificates.
- **Customize notifications.** Create and send notifications to administrators and users using the supplied email templates. Tailor the recipients and content of these notifications to meet your business and security requirements. See Customize Oracle Identity Cloud Service Notifications.
- **Manage password policies.** Create and manage password policies for an identity domain and assign them to groups. A password policy is applicable to all users in the group it is associated with. For all new users, Oracle Identity Cloud Service validates their passwords against your password policy to verify that those passwords meet the criteria for the policy. Adjust the strength of your password policies as needed to reflect different priorities and ensure a strong, secure environment. See Managing Oracle Identity Cloud Service Password Policies.
- **Customize the UI.** In addition to notifications and password policies, you can customize the **Sign In** page and Identity Cloud Service console. See Customizing the Oracle Identity Cloud Service Interface.
- **Manage Provisioning Bridges.** If you use on-premises apps such as Oracle Internet Directory as authoritative sources for your company's users and groups, then the Provisioning Bridge provides a link between these apps and Oracle Identity Cloud Service. The Provisioning Bridge can leverage Identity Connector Framework (ICF) connectors to synchronize with the associated apps so that any new, updated, or deleted user or group records are transferred into Oracle Identity Cloud Service. The state of each record is synchronized between the apps and Oracle Identity Cloud Service. See Manage Provisioning Bridges for Oracle Identity Cloud Service.
- **Manage Microsoft Active Directory (AD) Bridges.** If you use Microsoft Active Directory as an authoritative source, then the bridge provides a link between your on-premises Microsoft Active Directory and Oracle Identity Cloud Service. Oracle Identity Cloud Service can synchronize with Microsoft Active Directory so that any new, updated, or deleted user or group records are transferred into Oracle Identity Cloud Service. The state of each record is synchronized between Microsoft Active Directory and Oracle Identity Cloud Service. See Manage Microsoft Active Directory (AD) Bridges for Oracle Identity Cloud Service.
- **Manage session settings.** Define session expiration, logout and logout redirect URLs, and configure Allow Cross-Origin Resource Sharing (CORS) to allow client applications that run on one domain to obtain data from another domain. See Managing Oracle Identity Cloud Service Session Settings.
- **Manage self-registration profiles.** Create self-registration profiles to manage different sets of users, approval policies, and applications in Oracle Identity Cloud Service. See Manage Self-Registration Profiles in Oracle Identity Cloud Service.

- **Download software development kits (SDKs) and applications.** Oracle Identity Cloud Service provides you with a centralized location in the Identity Cloud Service console where you can download SDKs and applications. See Download Oracle Identity Cloud Service SDKs and Applications.
- **Customize schemas.** Create, edit, and delete custom schema attributes. You may need to create a custom schema attribute, for example, when you are creating your own user interface and can't find a schema attribute that you need in the base Oracle Identity Cloud Service schema attributes. See Customize Schemas in Oracle Identity Cloud Service.
- **Manage Adaptive Security.** Manage default and custom risk providers that Oracle Identity Cloud Service uses to evaluate risk-based activity for Oracle Identity Cloud Service users, and generate a risk score for these users, based on this activity. This risk score is a number that varies from risk provider to risk provider, reflecting user threat. See Manage Adaptive Security in Oracle Identity Cloud Service.
- **Manage identity providers.** Add SAML 2.0 and social identity providers so that users can interact with Oracle Identity Cloud Service using websites that are external to Oracle Identity Cloud Service. See Manage Oracle Identity Cloud Service Identity Providers.
- **Manage identity provider policies.** Create identity provider policies to restrict which identity providers appear on the **Sign In** page when users are accessing particular apps. See Manage Oracle Identity Cloud Service Identity Provider Policies.
- **Manage sign-on policies.** Create sign-on policies to define criteria that Oracle Identity Cloud Service uses to allow or deny access to users for apps that are assigned to them. See Manage Oracle Identity Cloud Service Sign-On Policies.
- **Manage network perimeters.** Define network perimeters to restrict the IP addresses that users can use to log in to Oracle Identity Cloud Service. See Manage Oracle Identity Cloud Service Network Perimeters.
- **Manage App Gateway.** to integrate web applications hosted either on a compute instance in a cloud infrastructure, or in an on-premises server with Oracle Identity Cloud Service for authentication purposes.
- **Manage account recovery.** Configure account recovery in Oracle Identity Cloud Service to help users regain access to their accounts if they have trouble signing in, they're locked out, or they forget their passwords. See Manage Account Recovery in Oracle Identity Cloud Service.
- **Manage Multi-Factor Authentication settings.** Change the security settings such as Multi-Factor Authentication (MFA) for your identity domain. See Managing Oracle Identity Cloud Service Multi-Factor Authentication Settings.
- **Manage OAuth Settings.** Configure OAuth settings for your environments. See Managing Oracle Identity Cloud Service OAuth SettingsConfiguring Oracle Identity Cloud Service OAuth Settings.
- **Manage delegated authentication.** Use delegated authentication to enable users to use their Microsoft Active Directory passwords to sign in to Oracle Identity Cloud Service to access resources and applications protected by Oracle Identity Cloud Service. See Configure Delegated Authentication in Oracle Identity Cloud Service.
- **Transfer configuration data.** Import and export configurations, entities, and customizations as an integral part of migrating an Oracle Identity Cloud Service environment. See Transferring Oracle Identity Cloud Service Configurations.
- **Manage account recovery.** Configure account recovery in Oracle Identity Cloud Service. See Manage Account Recovery in Oracle Identity Cloud Service.

Standard License Tier Features for Oracle Identity Cloud Service

Learn more about License Tiers.

Most features are already enabled for Standard Tier License tenants. See About Oracle Identity Cloud Service Pricing Models. If you don't see any of these features in Oracle Identity Cloud Service and want to use them, you must file a Service Request with My Oracle Support.

Category	Feature	Description
Application Gateway	App Gateway	Use App Gateway to integrate applications hosted either on a compute instance, in a cloud infrastructure, or in an on-premises server with Oracle Identity Cloud Service for authentication purposes. See Understand App Gateway.
Applications	Authorization Policy for Enterprise Applications	Enterprise applications that are protected using App Gateway can now make use of authorization policies. Administrators can define, allow or deny authorization policies using authenticated IdP, group membership, network perimeter, day and time of day as authorization conditions See Configure an Authorization Policy.
Device Fingerprint	Device Fingerprint	User device attributes are processed and the fingerprint is stored in a browser cookie to uniquely identify a user's system. See: About Device Fingerprints.
EBS Asserter	EBS Asserter	Use the Oracle Identity Cloud Service E-Business Suite Asserter component from Oracle Identity Cloud Service to integrate your Oracle E-Business Suite environment with other cloud and non-cloud services using Oracle Identity Cloud Service Single Sign-On (SSO). See Use the E-Business Suite Asserter to Enable SSO for Oracle E-Business Suite with Oracle Identity Cloud Service.

Category	Feature	Description
Identity Provisioning	Provisioning Bridge	The Provisioning Bridge provides synchronization of users and groups between your on-premises apps and Oracle Identity Cloud Service. Learn how you can create, manage, and remove Provisioning Bridges in Oracle Identity Cloud Service. See Understand the Provisioning Bridge and Why Use the Provisioning Bridge?.
Identity Provisioning	Lifecycle Rules	Manage the complete user life cycle and automate the process of the joiner, mover and leaver. If there is any change in a User attribute, you can propagate that to the downstream application (for example, if a user gets disabled, then all accounts owned by this user would be disabled automatically).
Security	IDP Discovery Rules	Identity Provider (IDP) Discovery enables you to organize the login page based on the username, for example, if you want corporate SSO login for some users and you want them to be logged in using social Identity Providers. Depending on the application being accessed and who is accessing it you can completely customize the way user can login. See: • Change Session Settings. • Understand Identity Provider Policies. • Add an Identity Provider Policy. • View Details About an Identity Provider Policy. • Modify an Identity Provider Policy. • Add Identity Provider Rules to the Policy. • Change the Priority of an Identity Provider Rule for the Policy. • Edit an Identity Provider Rule for the Policy. • Remove Identity Provider Rules from the Policy.

Category	Feature	Description
LDAP	LDAP2SCIM Proxy	The LDAP2SCIM proxy will allow application clients to integrate with Oracle Identity Cloud Service using LDAP protocol. This is a beta only feature currently available on invitation basis.
Passwordless Login	Tired of resetting passwords? Passwordless authentication is available.	Instead of passwords, proof of identity can be verified based on possession of something that uniquely identifies the user (for example, a one-time password (OTP), a registered mobile device, or a hardware token). Once enabled, users can access protected resources either by using a user name and password or passwordless authentication. Users use self-service to set up passwordless authentication. See Manage Passwordless Authentication.
SAML	Just-In-Time (JIT) Provisioning	Using SAML, JIT provisioning automates user account creation for target service providers when the user first tries to perform SSO and the user does not exist. In addition to automatic user creation, JIT implementation allows granting and revoking group memberships as part of provisioning. JIT implementation also updates provisioned users so the users' attributes in the Service Provider store can be kept in sync with the Identity Store user store attributes. See Understand SAML Just-In-Time Provisioning. SAML JIT Provisioning uses Oracle Identity Cloud Service REST APIs. See Create an Identity Provider. For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.
Security	AD Bridge High Availability	Set up High Availability and Load Sharing so that you don't have a single point of failure for your AD Bridge architecture. See About Multiple AD Bridges for High Availability and Load Balancing.

Category	Feature	Description
AD Bridge	AD Bridge – Sync Only	Synchronize users and groups from selected organizational units (OUs) in Microsoft Active Directory (AD) into Oracle Identity Cloud Service. You can perform either an incremental sync or a full sync. Learn about syncing new OUs and read some example use cases. See Understand Full and Incremental Sync.
Security	Delegated Authentication	With delegated authentication, identity domain administrators and security administrators don't have to synchronize user passwords between an on-premises Microsoft Active Directory (AD) enterprise directory structure and Oracle Identity Cloud Service. Users can use their AD passwords to sign in to Oracle Identity Cloud Service to access resources and applications protected by Oracle Identity Cloud Service. See Understand Delegated Authentication.
Security	Duo as an authentication factor.	Use Duo Security factors to securely authenticate and to sign into apps secured by Oracle Identity Cloud Service. See Configure Duo Security Settings.
Security	X.509 Certificate Authentication for Identity Providers	Use an X.509 authenticated identity provider with certificate-based authentication to comply with Personal Identity Verification (PIV) card requirements. See Enable X.509 Certificate Authentication, Import a Trusted Partner Certificate, and Add an X.509 Authenticated Identity Provider.
Security	Phone call as an authentication factor.	Use a phone call to securely authenticate and to sign into apps secured by Oracle Identity Cloud Service. See Configure Multi-Factor Authentication Settings and Configure One-Time Passcode Text Messages.

Category	Feature	Description
Security	FIDO Security	Use FIDO Authentication as an MFA Factor so that users use platform authentication, such as Windows Hello or Mac Touch ID, or cross platform authentication, using devices such as Yubikeys. See Configure FIDO Security.
Security	Group-Based Password Policies	You can have multiple password policies in Oracle Identity Cloud Service and associate them with different groups and set the priorities. Group password policies allow you to define password policies and associated rules to enforce password settings on the group level. You can create multiple policies with more- or less-restrictive rules. See • Set the Password Policies for Your Identity Domain.
Security	Network Perimeters	For security purposes, identity domain administrators, security administrators, and application administrators can define network perimeters in Oracle Identity Cloud Service. A network perimeter contains a list of IP addresses. See Understand Network Perimeters.
Security	Secure Oracle Database with RADIUS Proxy	Enterprises can now secure their Oracle Database instances with two-factor authentication using RADIUS Proxy. Using RADIUS Proxy, Oracle Identity Cloud Service can: • Manage all database Administrators and all database Users. • Define access controls using Database Roles to be managed by using Identity Cloud Service Groups. See • Setup RadiusProxy. • REST API for Oracle Identity Cloud Service.

Category	Feature	Description
User Experience	Customize the sign in page by creating your own HTML code and translations.	Instead of using the default sign in page, administrators can create a Hosted Sign In page to change the look and feel of the sign-in experience. You create a Hosted Sign In page by adding a background image as well as designing custom HTML code and specifying translations (specifying translations is optional.). See Create Hosted Sign In Pages.

About Oracle Identity Cloud Service Pricing Models

There are two pricing models for Oracle Identity Cloud Service.

- User Per Month: Beginning with version 18.4.2, Oracle Identity Cloud Service has a new pricing model for its customers. This pricing model bills users on the activity that they perform with Oracle Identity Cloud Service on a monthly basis. This not only streamlines projected billing calculations, but also helps customers to more-accurately predict how much money they will spend for any given month.
- Active User Per Hour: This pricing model is no longer available for new customers. The information below on this pricing model is included only for existing customers with active contracts that specify this pricing model.

See Oracle Platform as a Service and Infrastructure as a Service – Public Cloud Service Descriptions-Metered & Non-Metered for a complete list of public cloud service descriptions.

Understand the User Per Month Pricing Model

Learn about the pricing tiers for Oracle Identity Cloud Service for the User per Month pricing model and the features associated with each pricing tier.

For this pricing model, Oracle Identity Cloud Service has two pricing tiers:

- Oracle Identity Cloud Service Foundation: Oracle provides this free version of Oracle Identity Cloud Service for customers that subscribe to Oracle Software-as-a-Service (SaaS), Oracle Platform-as-a-Service (PaaS), and Oracle Cloud Infrastructure only.

  A customer can use this version to provide basic identity management functions, including user management, group management, password management, and basic reporting. For additional features, as indicated in the table below, a subscription to Oracle Identity Cloud Service Standard is required.

  A customer can’t use this version to integrate with third-party SaaS, PaaS, custom web or mobile applications, programmatic clients or On-Premises applications, even if those applications are hosted on Oracle Cloud Infrastructure. Those use cases require Oracle Identity Cloud Service Standard.
- Oracle Identity Cloud Service Standard: This licensed edition provides customers with an additional set of Oracle Identity Cloud Service features to integrate with other Oracle Cloud services, including Oracle Cloud SaaS and PaaS, custom applications hosted on-premises, on Oracle Cloud, or on a third-party cloud, as well as third-party SaaS

applications. Features listed in this pricing tier are applicable for both Enterprise users and Consumer users.

An incentive of the Standard tier for the User per Month pricing model is the Bring Your Own License (BYOL) program. If you're an Oracle customer who's using certain Oracle identity management on-premises technologies and is paying support for these technologies, then you can subscribe to the BYOL Standard tier and use the features of this tier at the BYOL rate.

See Buy an Oracle Cloud Subscription for more information about the payment plans available with Oracle Identity Cloud Service.

> **Note:**
>
> Once you have decided on the pricing model, you're reminded of which one you purchased by **Licence Type: Foundation** or **Licence Type: Standard** shown in the top right of the Identity Cloud Service console.

The following table illustrates the features associated with each Oracle Identity Cloud Service pricing tier:

Feature	Description	Foundation	Standard
License Types	**Available**: Oracle Identity Cloud - Enterprise User - User Per Month, Oracle Identity Cloud - Consumer User - User Per Month, Oracle Identity Cloud - Enterprise User - BYOL - User Per Month, Oracle Identity Cloud - Consumer User - BYOL - User Per Month, Oracle Identity Foundation Cloud Service **Default** (for primordial instance): Oracle Identity Foundation Cloud Service **Options**: Customer intending to use paid IDCS features should update the instance to one of the paid SKUs (based on usage, on-premises license, and other factors).		
Group-Based Password Policies	You can create multiple password policies in Oracle Identity Cloud Service, set the priority of these policies to determine in which order they apply, and then attach them to groups.		✓
User and Group Management	Manage the lifecycle of users and groups in Oracle Identity Cloud Service. Users and groups can be onboarded manually or can be imported in bulk from a CSV file.	✓	✓
User and Group Management	Grant user access to various applications by assigning users to the applications directly, or by assigning users to groups and groups to applications.	✓	✓
Self-Service Profile Management	Perform self-service capabilities to update user profile attributes and change passwords.	✓	

Feature	Description	Foundation	Standard
Advanced Self-Service Profile Management	Perform self-service capabilities to update user profile attributes, change passwords, manage linked social login accounts, view and manage devices registered for second-factor verification, and generate second-factor bypass codes.		✓
Self-Service Password Reset	Perform self-service reset of users' forgotten passwords.	✓ (using challenge questions and answers)	✓ (using all factors including email, SMS and push notifications)
SSO for Oracle Cloud Services	Authenticate to Oracle Identity Cloud Service and gain single-click access to Oracle Cloud services. This includes SSO between two Oracle Identity Cloud Service instances.	✓	✓
External Identity Provider Federation	Configure a SAML 2.0 external identity provider such as Active Directory Federation Services (AD FS) for federated SSO to Oracle Identity Cloud Service.	✓ (for one SAML identity provider)	✓ (for more than one SAML identity provider)
Basic User Provisioning and Synchronization for Oracle Cloud Apps	Provision user accounts to multiple Oracle SaaS and Oracle PaaS applications. You can also enable account synchronization to detect and synchronize any changes made directly on these target applications. Although you can use the provisioning templates, you can't change the default attribute mappings for provisioning and synchronization, or make any configuration changes to them.	✓	✓

Feature	Description	Foundation	Standard
Sign-on Policies	Use these policies to define criteria that Oracle Identity Cloud Service uses to determine whether to allow a user to sign in to Oracle Identity Cloud Service or prevent a user from accessing Oracle Identity Cloud Service. By defining this criteria, you control access that users have to your applications based on conditions such as the identity providers that will be used to authenticate the users, the groups to which the users belong, whether the users are assigned to administrator roles in Oracle Identity Cloud Service, or whether the users are accessing Oracle Identity Cloud Service using an IP address that's contained in a network perimeter. Oracle Identity Cloud Service provides you with a default sign-on policy. In addition to the default sign-on policy, you can add sign-on policies and associate them with specific apps. When a user uses one of these apps to attempt to sign in to Oracle Identity Cloud Service, Oracle Identity Cloud Service checks to see if the app has any sign-on policies associated with it. If so, then Oracle Identity Cloud Service evaluates the criteria of the sign-on rules assigned to the policy. If there are no sign-on policies for the app, then the default sign-on policy is evaluated by Oracle Identity Cloud Service.	✓ (for the default sign-on policy)	✓ (for any sign-on policies that you add)
Application Development SDKs	Enable your mobile and web applications to authenticate to Oracle Identity Cloud Service by using software development kits (SDKs).	✓	✓
Security and Usage Reports	Execute and view operational or historical reports that capture usage data about Oracle Identity Cloud Service users, and applications, and diagnostic level logs.	✓	✓
Oracle Identity Manager Connector for Oracle Identity Cloud Service	Use this connector in Oracle Identity Manager to manage the complete lifecycle of users and groups in Oracle Identity Cloud Service from Oracle Identity Manager. This connector also enables access certification of SaaS resources, Segregation of Duties (SoD) violation checks during the request and approval process, and reports on SaaS app usage in Oracle Identity Manager.	✓	✓

Feature	Description	Foundation	Standard
App Catalog	The App Catalog is a collection of partially configured application templates for thousands of SaaS applications, such as Amazon Web Services and Google Suite. Using the templates, you can define an application, configure SSO, and configure provisioning. Oracle creates and maintains the App Catalog for you, and provides step-by-step instructions that will help you to configure your applications. **Note:** For Oracle SaaS application SSO and provisioning, refer to the descriptions in the *SSO for Oracle Cloud Services* and the *Basic User Provisioning and Synchronization for Oracle Cloud Apps* rows above.		✓
Active Directory Synchronization	Use one or more Microsoft Active Directory bridges to synchronize identities and groups with Oracle Identity Cloud Service.		✓
User Self-Registration	Enable Business-to-Business (B2B) and Business-to-Consumer (B2C) users to register themselves to Oracle Identity Cloud Service. You can also create multiple self-registration profiles to manage different sets of users and access to applications.		✓
Self-Service Access Request	Enable users to request access to groups and applications from the App Catalog.		✓
SSO for Third-Party Cloud Services	Authenticate to Oracle Identity Cloud Service and gain single-click access to third-party SaaS services configured using the App Catalog. The App Catalog is a collection of pre-seeded applications for popular SaaS applications, such as Amazon Web Services, Google Suite, Office 365, and so on, that support federation standards such as SAML 2.0 and OAuth 2.0. It also allows you to configure Secure Form Fill for applications that don't support these standards. Using the App Catalog, you can define the application, configure SSO, and configure provisioning. Oracle creates and maintains the App Catalog for you.		✓
SSO for Custom Applications	For custom applications developed using Oracle Cloud services and deployed on Oracle Cloud (PaaS and IaaS), authenticate to Oracle Identity Cloud Service and gain single-click access to these applications.		✓
RADIUS Proxy	Remote Authentication Dial In User Service (RADIUS) is a network protocol—a system that defines rules and conventions for communication between network devices—for remote user authentication and accounting.		✓

Feature	Description	Foundation	Standard
Delegated Authentication with Password Writeback	Remove the need to synchronize user passwords between an on-premises Microsoft Active Directory enterprise directory structure and Oracle Identity Cloud Service. Users can use their Microsoft Active Directory passwords to sign in to Oracle Identity Cloud Service to access resources and applications protected by Oracle Identity Cloud Service.		✓
Linux-PAM Module	Use the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) to integrate your Linux environment with Oracle Identity Cloud Service to facilitate authentication to Linux hosts.		✓
Multi-Factor Authentication (MFA)	Enable strong authentication by configuring Multi-Factor Authentication (MFA) during user authentication. Configure device compliance policies and a wide variety of second factors, such as SMS, OTP, push notifications, and knowledge-based questions and answers.	✓ (Limited Use - Allowed for the Oracle Cloud Console only; Allowed factors: Mobile app passcode, Mobile app notification, Bypass code, FIDO, and Duo. Also, one additional sign-on policy is allowed to protect the Oracle Cloud Console [Application name - *OCI-V2-App-<TenancyName>*])	✓
Adaptive Security	Analyze contextual, risk, and threat information about the user, device, and network, and provide an intelligent, secure, and user-friendly way of providing access to corporate applications and resources. This also reduces the likelihood of online identity theft and fraud, which secures business applications even if the user's device or the user's account password is compromised.		✓
Social Authentication	Configure one or more social identity providers so that users can log in to Oracle Identity Cloud Service with their social credentials.		✓

Feature	Description	Foundation	Standard
Advanced User Provisioning and Synchronization for Oracle Cloud Apps	Support interactive provisioning to allow administrators to grant entitlements and specify values for application account attributes. Administrators can also synchronize entitlements and other application data from the application into Oracle Identity Cloud Service. In addition to interactive provisioning and synchronization, you can customize the pre-configured provisioning templates in the App Catalog by changing the default attribute mappings for provisioning and synchronization and making configuration changes to them.		✓
User Provisioning and Synchronization for Third-Party Cloud Apps	Configure provisioning of user accounts to multiple third-party cloud apps, such as Google Suite, Office 365, and so on, from a list of pre-configured provisioning templates in the App Catalog. Enable account synchronization to detect and synchronize any changes made directly on these target applications.		✓
Just In Time Provisioning	Just in time (JIT) provisioning automates the process of creating user accounts in connected applications. It uses the SAML protocol to provide necessary information from the identity provider (IDP) to the application.		✓
EBS Asserter	Integrate your Oracle E-Business Suite environment with Oracle Identity Cloud Service for authentication and password management purposes by using a lightweight Java application known as the Oracle E-Business Suite (EBS) Asserter. The right to use Oracle E-Business Suite Asserter also includes the right to use WebLogic Server Enterprise Edition solely for the purposes of running the asserter application in accordance with all terms and conditions as described in the Oracle Fusion Middleware Licensing Information User Manual.		✓
Terms of Use	Present disclaimers and acceptable use policies, also known as Terms of Use, to your users. Terms of Use helps you set the terms and conditions for your users to access your applications, based on user consent. This feature allows identity domain administrators to set relevant disclaimers for legal or compliance requirements and enforce the terms by refusing the service. You can configure Terms of Use on an application basis and collect consent from users before allowing them access to the application.		✓

Feature	Description	Foundation	Standard
App Gateway	The Oracle Identity Cloud Service App Gateway is a software appliance that you can use to provide Single Sign-On (SSO) and authorization for your on-premises applications. This enables you to use one appliance to provide SSO for multiple applications by allowing external users to access internal applications securely without the need for a VPN client. From the **App Gateway for Identity Cloud Service** application, you can access the documentation for the App Gateway. You can find this application on the **Downloads** page of the Identity Cloud Service console. To access this page, in the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Downloads**.		
WebGate	WebGate is a web-server plug-in that Oracle Access Management uses to protect on-premises web applications. It can be deployed on different web applications and web servers including, but not limited to, the Apache HTTP Server and Microsoft's Internet Information Services (IIS) web server. Instead of relying on Oracle Access Manager as an authentication service, WebGate can now interact to protect these applications by authenticating users to access the applications. When an unauthenticated user tries to access any applications that are protected by Oracle Identity Cloud Service, the user is redirected to the **Sign In** page of Oracle Identity Cloud Service for authentication.		
Schema Extension	If you're creating your own UI, and can't find a schema attribute that you need from the base Oracle Identity Cloud Service schema attributes, then you can add your own custom attributes using the Identity Cloud Service console.		

Feature	Description	Foundation	Standard
Generic SCIM App Template	With this template, you can provision or synchronize users between your applications and Oracle Identity Cloud Service. You can use this template to configure your applications so that the SCIM APIs are exposed, and you don't have to develop a single line of code. All that's required is to go to the App Catalog and search for a SCIM-managed app template. To use this template, you only have to provide your endpoint URL and the details that Oracle Identity Cloud Service requires to connect to your application, and then map the attributes between your application and Oracle Identity Cloud Service. Using the SCIM template to sync users between Oracle Identity Cloud Service and non-Oracle end points is a paid tier feature.		✓
Generic SCIM App Template	Using the SCIM template to sync users between two Oracle Identity Cloud Service instances.	✓	
SMS Messaging	The total SMS message count is a pool based on the total number of users who have enabled MFA with SMS multiplied by the number of messages per user per month. Enterprise users are limited to 10 messages per user per month. Consumer users are limited to three messages per user per month. Any additional SMS messaging used beyond the limit is billed as additional Monthly users.		✓
Advanced OAuth Capabilities	Use advanced capabilities such as Custom Claims, Token Issuance Policies and apply Sign-On Policies to custom OAuth applications to control token issuance.		✓
Social Login	Enable consumers to access applications using out-of-the-box social providers, define custom social providers (using the metadata-driven declarative providers feature), enable explicit and and social data capture using Oracle Identity Cloud Service.		✓
Provisioning Bridge	Use one or more Provisioning Bridges to provision and synchronize identities, groups, and application user accounts with applications with Oracle Identity Cloud Service.		✓
Create custom mobile, desktop, and web applications using OAuth 2.0 and OpenID Connect	Develop web, desktop and mobile applications using OAuth 2.0 and OpenID Connect to secure APIs and to integrate with API Gateways. Use Custom Claims to enrich claims and policies to control token issuance.		✓

API Rate Limits

Understand API rate limits for Foundation edition, and Enterprise users and Consumer users (Standard edition).

Oracle APIs are subject to rate limiting to protect the API service usage for all of Oracle's customers. If you reach the API limit for Foundation, Enterprise, or Consumer, then a 429 error code is returned.

This table shows the API rate limits for the different editions.

	Foundation Edition	Standard Edition - Enterprise	Standard Edition - Consumer
AuthN / sec	50	95	90
AuthN / min	1000	4500	3100
Token Mgmt / sec	40	65	60
Token Mgmt / min	1000	3400	2300
Others / sec (excluding bulk, import and export)	50	90	80
Others / min (excluding bulk, import and export)	1500	5000	4000
Bulk / sec	1	2	2
Bulk / min	2	6	6
Import and export / day	2	5	5

Understand the Active User Per Hour Pricing Model

Learn about the pricing tiers for Oracle Identity Cloud Service for the Active User per Hour pricing model and the features associated with each pricing tier. This pricing model is no longer available for new customers. The information below on this pricing model is included only for existing customers with active contracts that specify this pricing model.

For this pricing model, Oracle Identity Cloud Service has three pricing tiers:

- Oracle Identity Cloud Service Foundation: Oracle provisions the Enterprise version of Oracle Identity Cloud Service for customers that subscribe to Oracle Software-as-a-Service (SaaS), Oracle Platform-as-a-Service (PaaS), and Infrastructure-as-a-Service (IaaS) applications. Customers are then charged based on the features they use.

  A customer can use this version to provide basic identity management functions, including user management, group management, password management, and basic reporting. For additional features, as indicated in the table below, a subscription to Oracle Identity Cloud Service Basic or Oracle Identity Cloud Service Standard is required.

  A customer can’t use this version to integrate with third-party SaaS, PaaS, custom web or mobile applications, programmatic clients or On-Premises applications, even if those applications are hosted on Oracle Cloud Infrastructure. Those use cases require Oracle Identity Cloud Service Standard.
- Oracle Identity Cloud Service Basic: This licensed edition provides all of the features of Oracle Identity Cloud Service Foundation plus the ability to synchronize Microsoft Active Directory user identities and groups into Oracle Identity Cloud Service.
- Oracle Identity Cloud Service Standard: This licensed edition provides customers with an additional set of Oracle Identity Cloud Service features to integrate with other Oracle Cloud services, including Oracle Cloud SaaS and PaaS, custom applications hosted on-premises, on Oracle Cloud, or on a third-party cloud, as well as third-party SaaS

applications. Features listed in this pricing tier are applicable for both Enterprise users and Consumer users.

See Buy an Oracle Cloud Subscription for more information about the payment plans available with Oracle Identity Cloud Service.

> **Note:**
>
> Once you have decided on the pricing model, you're reminded of which one you purchased by **License Type: Foundation** or **License Type: Standard** shown in the top right of the Identity Cloud Service console.

The following table illustrates the features associated with each Oracle Identity Cloud Service pricing tier:

Feature	Description	Foundation	Basic	Standard
License Types	**Available**: Enterprise, B2C **Default**: Enterprise **Options**: Customer can change to B2C if using the IDCS instance to manage external users. Foundation is just a tier of features.			
Group-Based Password Policies	You can create multiple password policies in Oracle Identity Cloud Service, set the priority of these policies to determine in which order they apply, and then attach them to groups.			✓
User and Group Management	Manage the life cycle of users and groups in Oracle Identity Cloud Service. Users and groups can be onboarded manually or can be imported in bulk from a CSV file. You can grant user access to various applications by assigning users to the applications directly, or by assigning users to groups and groups to applications.	✓	✓	✓
Self-Service Profile Management	Perform self-service capabilities to update user profile attributes and change passwords.	✓		
Advanced Self-Service Profile Management	Perform self-service capabilities to update user profile attributes, change passwords, manage linked social login accounts, view and manage devices registered for second-factor verification, and generate second-factor bypass codes.		✓	✓

Feature	Description	Foundation	Basic	Standard
Self-Service Password Reset	Perform self-service reset of users' forgotten passwords.	✓ (using challenge questions and answers)	✓	✓ (using all factors including email, SMS and push notifications)
SSO for Oracle Cloud Services	Authenticate to Oracle Identity Cloud Service and gain single-click access to Oracle Cloud services. This includes SSO between two Oracle Identity Cloud Service instances.	✓	✓	✓
Basic User Provisioning and Synchronization for Oracle Cloud Apps	Provision user accounts to multiple Oracle SaaS and Oracle PaaS applications. You can also enable account synchronization to detect and synchronize any changes made directly on these target applications. Although you can use the provisioning templates, you can't change the default attribute mappings for provisioning and synchronization, or make any configuration changes to them.	✓	✓	✓
Oracle Identity Manager Connector for Oracle Identity Cloud Service	Use this connector in Oracle Identity Manager to manage the complete life cycle of users and groups in Oracle Identity Cloud Service from Oracle Identity Manager. This connector also enables access certification of SaaS resources, Segregation of Duties (SoD) violation checks during the request and approval process, and reports on SaaS app usage in Oracle Identity Manager.	✓	✓	✓
Application Development SDKs	Enable your mobile and web applications to authenticate to Oracle Identity Cloud Service by using software development kits (SDKs).	✓	✓	✓
Security and Usage Reports	Execute and view operational or historical reports that capture usage data about Oracle Identity Cloud Service users, and applications, and diagnostic level logs.	✓	✓	✓
External Identity Provider Federation	Configure a SAML 2.0 external identity provider such as Active Directory Federation Services (AD FS) for federated SSO to Oracle Identity Cloud Service.	✓ (for one SAML identity provider)		✓ (for more than one SAML identity provider)

Feature	Description	Foundation	Basic	Standard
Sign-on Policies	Use these policies to define criteria that Oracle Identity Cloud Service uses to determine whether to allow a user to sign in to Oracle Identity Cloud Service or prevent a user from accessing Oracle Identity Cloud Service. By defining this criteria, you control access that users have to your applications based on conditions such as the identity providers that will be used to authenticate the users, the groups to which the users belong, whether the users are assigned to administrator roles in Oracle Identity Cloud Service, or whether the users are accessing Oracle Identity Cloud Service using an IP address that's contained in a network perimeter. Oracle Identity Cloud Service provides you with a default sign-on policy. In addition to the default sign-on policy, you can add sign-on policies and associate them with specific apps. When a user uses one of these apps to attempt to sign in to Oracle Identity Cloud Service, Oracle Identity Cloud Service checks to see if the app has any sign-on policies associated with it. If so, then Oracle Identity Cloud Service evaluates the criteria of the sign-on rules assigned to the policy. If there are no sign-on policies for the app, then the default sign-on policy is evaluated by Oracle Identity Cloud Service.	✓ (for the default sign-on policy)		✓ (for any sign-on policies that you add)
Active Directory Synchronization	Use one or more Microsoft Active Directory bridges to synchronize identities and groups with Oracle Identity Cloud Service.		✓	✓
App Catalog	The App Catalog is a collection of partially configured application templates for thousands of SaaS applications, such as Amazon Web Services and Google Suite. Using the templates, you can define an application, configure SSO, and configure provisioning. Oracle creates and maintains the App Catalog for you, and provides step-by-step instructions that will help you to configure your applications. **Note:** For Oracle SaaS application SSO and provisioning, refer to the descriptions in the *SSO for Oracle Cloud Services* and the *Basic User Provisioning and Synchronization for Oracle Cloud Apps* rows above.			✓

Feature	Description	Foundation	Basic	Standard
User Self-Registration	Enable Business-to-Business (B2B) and Business-to-Consumer (B2C) users to register themselves to Oracle Identity Cloud Service. You can also create multiple self-registration profiles to manage different sets of users and access to applications.			✓
Self-Service Access Request	Enable users to request access to groups and applications from the App Catalog.			✓
SSO for Third-Party Cloud Services	Authenticate to Oracle Identity Cloud Service and gain single-click access to third-party SaaS services configured using the App Catalog. The App Catalog is a collection of pre-seeded applications for popular SaaS applications, such as Amazon Web Services, Google Suite, Office 365, and so on, that support federation standards such as SAML 2.0 and OAuth 2.0. It also allows you to configure Secure Form Fill for applications that don't support these standards. Using the App Catalog, you can define the application, configure SSO, and configure provisioning. Oracle creates and maintains the App Catalog for you.			✓
SSO for Custom Applications	For custom applications developed using Oracle Cloud services and deployed on Oracle Cloud (PaaS and IaaS), authenticate to Oracle Identity Cloud Service and gain single-click access to these applications.			✓
RADIUS Proxy	Remote Authentication Dial In User Service (RADIUS) is a network protocol—a system that defines rules and conventions for communication between network devices—for remote user authentication and accounting.			✓
Delegated Authentication with Password Writeback	Remove the need to synchronize user passwords between an on-premises Microsoft Active Directory enterprise directory structure and Oracle Identity Cloud Service. Users can use their Microsoft Active Directory passwords to sign in to Oracle Identity Cloud Service to access resources and applications protected by Oracle Identity Cloud Service.			✓
Linux-PAM Module	Use the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) to integrate your Linux environment with Oracle Identity Cloud Service to facilitate authentication to Linux hosts.			✓

Feature	Description	Foundation	Basic	Standard
Multi-Factor Authentication (MFA)	Enable strong authentication by configuring Multi-Factor Authentication (MFA) during user authentication. Configure device compliance policies and a wide variety of second factors, such as SMS, OTP, push notifications, and knowledge-based questions and answers.	✓ (Limited Use - Allowed for the Oracle Cloud Console only; Allowed factors: Mobile app passcode, Mobile app notification, Bypass code, FIDO, and Duo. Also, one additional sign-on policy is allowed to protect the Oracle Cloud Console [Application name - *OCI-V2-App-<TenancyName>*])		✓
Adaptive Security	Analyze contextual, risk, and threat information about the user, device, and network, and provide an intelligent, secure, and user-friendly way of providing access to corporate applications and resources. This also reduces the likelihood of online identity theft and fraud, which secures business applications even if the user's device or the user's account password is compromised.			✓
Social Authentication	Configure one or more social identity providers so that users can log in to Oracle Identity Cloud Service with their social credentials.			✓
Advanced User Provisioning and Synchronization for Oracle Cloud Apps	Support interactive provisioning to allow administrators to grant entitlements and specify values for application account attributes. Administrators can also synchronize entitlements and other application data from the application into Oracle Identity Cloud Service. In addition to interactive provisioning and synchronization, you can customize the pre-configured provisioning templates in the App Catalog by changing the default attribute mappings for provisioning and synchronization and making configuration changes to them.			✓

Feature	Description	Foundation	Basic	Standard
User Provisioning and Synchronization for Third-Party Cloud Apps	Configure provisioning of user accounts to multiple third-party cloud apps, such as Google Suite, Office 365, and so on, from a list of pre-configured provisioning templates in the App Catalog. Enable account synchronization to detect and synchronize any changes made directly on these target applications.			
Just In Time Provisioning	Just in time (JIT) provisioning automates the process of creating user accounts in connected applications. It uses the SAML protocol to provide necessary information from the identity provider (IDP) to the application.			
EBS Asserter	Integrate your Oracle E-Business Suite environment with Oracle Identity Cloud Service for authentication and password management purposes by using a lightweight Java application known as the Oracle E-Business Suite (EBS) Asserter. The right to use Oracle E-Business Suite Asserter also includes the right to use WebLogic Server Enterprise Edition solely for the purposes of running the asserter application in accordance with all terms and conditions as described in the Oracle Fusion Middleware Licensing Information User Manual.			
Terms of Use	Present disclaimers and acceptable use policies, also known as Terms of Use, to your users. Terms of Use helps you set the terms and conditions for your users to access your applications, based on user consent. This feature allows identity domain administrators to set relevant disclaimers for legal or compliance requirements and enforce the terms by refusing the service. You can configure Terms of Use on an application basis and collect consent from users before allowing them access to the application.			

Feature	Description	Foundation	Basic	Standard
App Gateway	The Oracle Identity Cloud Service App Gateway is a software appliance that you can use to provide Single Sign-On (SSO) and authorization for your on-premises applications. This enables you to use one appliance to provide SSO for multiple applications by allowing external users to access internal applications securely without the need for a VPN client. From the **App Gateway for Identity Cloud Service** application, you can access the documentation for the App Gateway. You can find this application on the **Downloads** page of the Identity Cloud Service console. To access this page, in the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Downloads**.			
WebGate	WebGate is a web-server plug-in that Oracle Access Management uses to protect on-premises web applications. Instead of relying on Oracle Access Manager as an authentication service, WebGate can now interact with Oracle Identity Cloud Service to protect these applications by authenticating users to access the applications. When an unauthenticated user tries to access any applications that are protected by Oracle Identity Cloud Service, the user is redirected to the **Sign In** page of Oracle Identity Cloud Service for authentication.			
Schema Extension	If you're creating your own UI, and can't find a schema attribute that you need from the base Oracle Identity Cloud Service schema attributes, then you can add your own custom attributes using the Identity Cloud Service console.			

Feature	Description	Foundation	Basic	Standard
Generic SCIM App Template	With this template, you can provision or synchronize users between your applications and Oracle Identity Cloud Service. You can use this template to configure your applications so that the SCIM APIs are exposed, and you don't have to develop a single line of code. All that's required is to go to the App Catalog and search for a SCIM-managed app template. To use this template, you only have to provide your endpoint URL and the details that Oracle Identity Cloud Service requires to connect to your application, and then map the attributes between your application and Oracle Identity Cloud Service. Using the SCIM template to sync users between Oracle Identity Cloud Service and non-Oracle end points is a paid tier feature.			✓
Generic SCIM App Template	Using the SCIM template to sync users between two Oracle Identity Cloud Service instances.	✓		
SMS Messaging	The total SMS message count is a pool based on the total number of users who have enabled MFA with SMS multiplied by the number of messages per user per month. Enterprise users are limited to 10 messages per user per month. Consumer users are limited to three messages per user per month. Any additional SMS messaging used beyond the limit is billed as additional Active users.			✓
Advanced OAuth Capabilities	Use advanced capabilities such as Custom Claims, Token Issuance Policies and apply Sign-On Policies to custom OAuth applications to control token issuance.			✓
Social Login	Enable consumers to access applications using out-of-the-box social providers, define custom social providers (using the metadata-driven declarative providers feature), enable explicit registration and social data capture using Oracle Identity Cloud Service.			✓
Provisioning Bridge	Use one or more Provisioning Bridges to provision and synchronize identities, groups, and application user accounts with applications with Oracle Identity Cloud Service.			✓

Feature	Description	Foundation	Basic	Standard
Create custom mobile, desktop, and web applications using OAuth 2.0 and OpenID Connect	Develop web, desktop and mobile applications using OAuth 2.0 and OpenID Connect to secure APIs and to integrate with API Gateways. Use Custom Claims to enrich claims and policies to control token issuance.			✓

API Rate Limits

Understand API rate limits for Active User Per Hour tiers.

Oracle APIs are subject to rate limiting to protect the API service usage for all of Oracle's customers. If you reach the API limit for Foundation, Enterprise, or B2C, then a 429 error code is returned.

This table shows the API rate limits for the different editions.

	Foundation	Enterprise	B2C
AuthN / sec	50	95	90
AuthN / min	1000	4500	3100
Token Mgmt / sec	40	65	60
Token Mgmt / min	1000	3400	2300
Others / sec (excluding bulk, import and export)	50	90	80
Others / min (excluding bulk, import and export)	1500	5000	4000
Bulk / sec	1	2	2
Bulk / min	2	6	6
Import and export / day	2	5	5

About Multiple Instances

Customers want to have separate environments for a single cloud service or application (for example, one environment for development and one for production).

Each environment may have different identity and security requirements so customers need to create separate environments to meet this criteria. You can create and manage multiple instances of Oracle Identity Cloud Service to protect your applications and Oracle Cloud services.

There are several benefits of using multiple instances of Oracle Identity Cloud Service. By having separate Oracle Identity Cloud Service environments, the users who work in one environment won't impact the work of users in another environment. Using multiple instances can help you maintain the isolation of administrative control over each environment. This is necessary if, for example, your security standards prevent development user IDs from existing in the production environment, or require that different administrators have control over different environments.

When multiple instances are utilized, you will have a **primary** instance, the instance which comes with your Oracle Cloud account, and one or more **secondary** (additional) instances. The cloud account administrator is the owner of the primary instance. This administrator can:

- Create secondary instances and be the identity domain administrator for them.

- Create secondary instances and, as part of the instance creation process, assign users to be identity domain administrators of the instances.
- Delegate the creation of secondary instances to other administrators.

The identity domain administrator is assigned to the secondary instance during the creation of the instance. Although the identity domain administrator of a secondary instance may have the same user name as a user in the primary instance, they are different users who might have different privileges in each instance, and will have separate passwords. This administrator can switch between the primary and secondary instances to work in each instance. See Identify and Switch Instances for more information about how to switch instances.

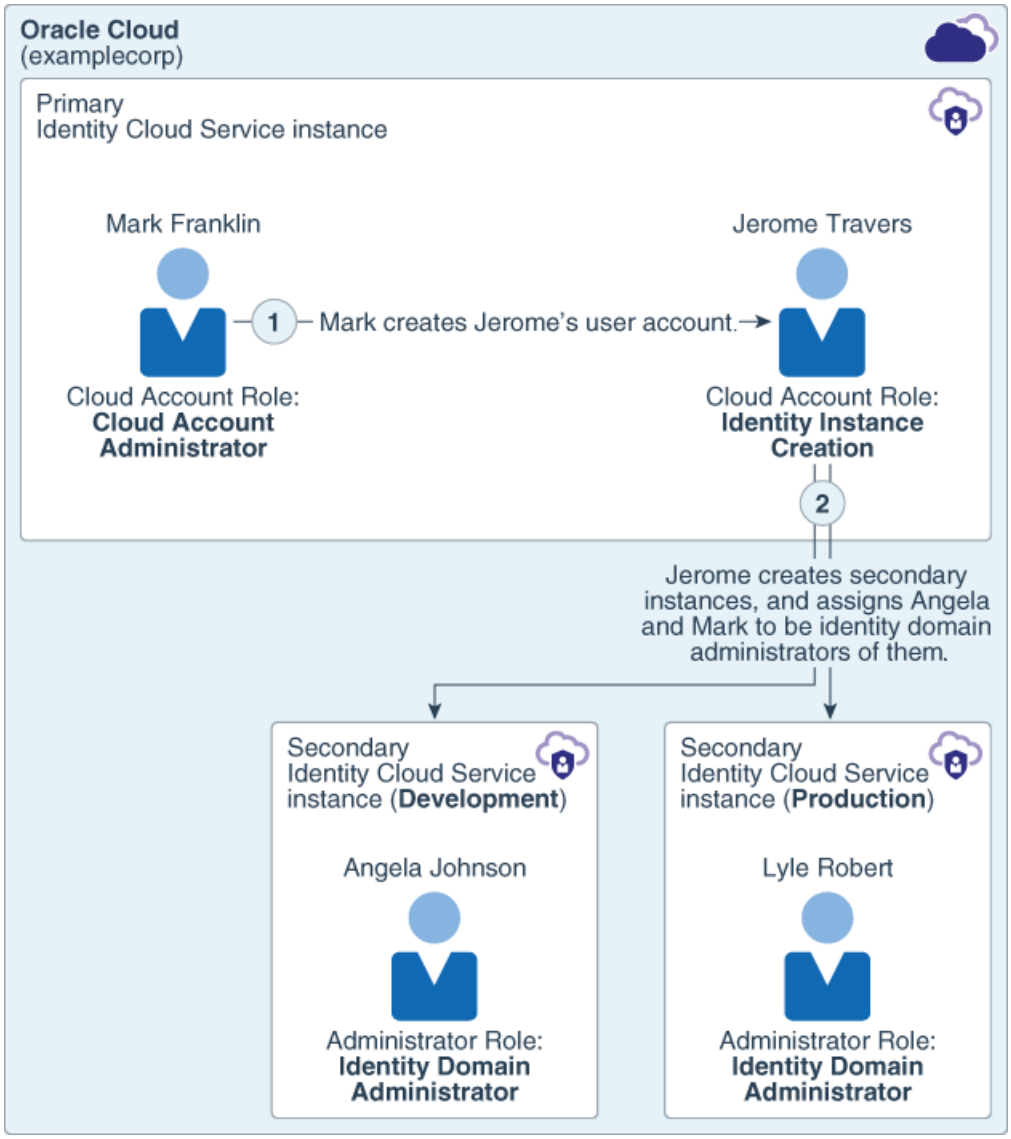
The identity domain administrator of a secondary instance has superuser privileges of that instance and can use the Oracle Identity Cloud Service feature set of the instance. In the secondary instance, the identity domain administrator can:

- Manage users, groups, applications, system configuration, and security settings.
- Perform delegated administration by assigning users to different administrative roles.
- Enable and disable Multi-Factor Authentication (MFA), configure MFA settings, and configure authentication factors.
- Create self-registration profiles to manage different sets of users, approval policies, and applications.

Regarding secondary instances, there are no new administrator or user processes to learn. The process to perform any administrator or user task in a secondary instance is identical to the process for performing it in the primary instance.

Important: The identity domain administrator of a secondary instance can't create a secondary instance of Oracle Identity Cloud Service from their instance. There can't be a parent-child relationship between secondary instances. All secondary instances must be created from the cloud account, either by the cloud account administrator or by another administrator (provided the cloud account administrator gives them permissions to do so). In addition to the cloud account administrator creating the primary instance, this administrator or another administrator can create up to nine secondary instances.

The figure below shows an example of the relationship among various administrators of multiple instances.

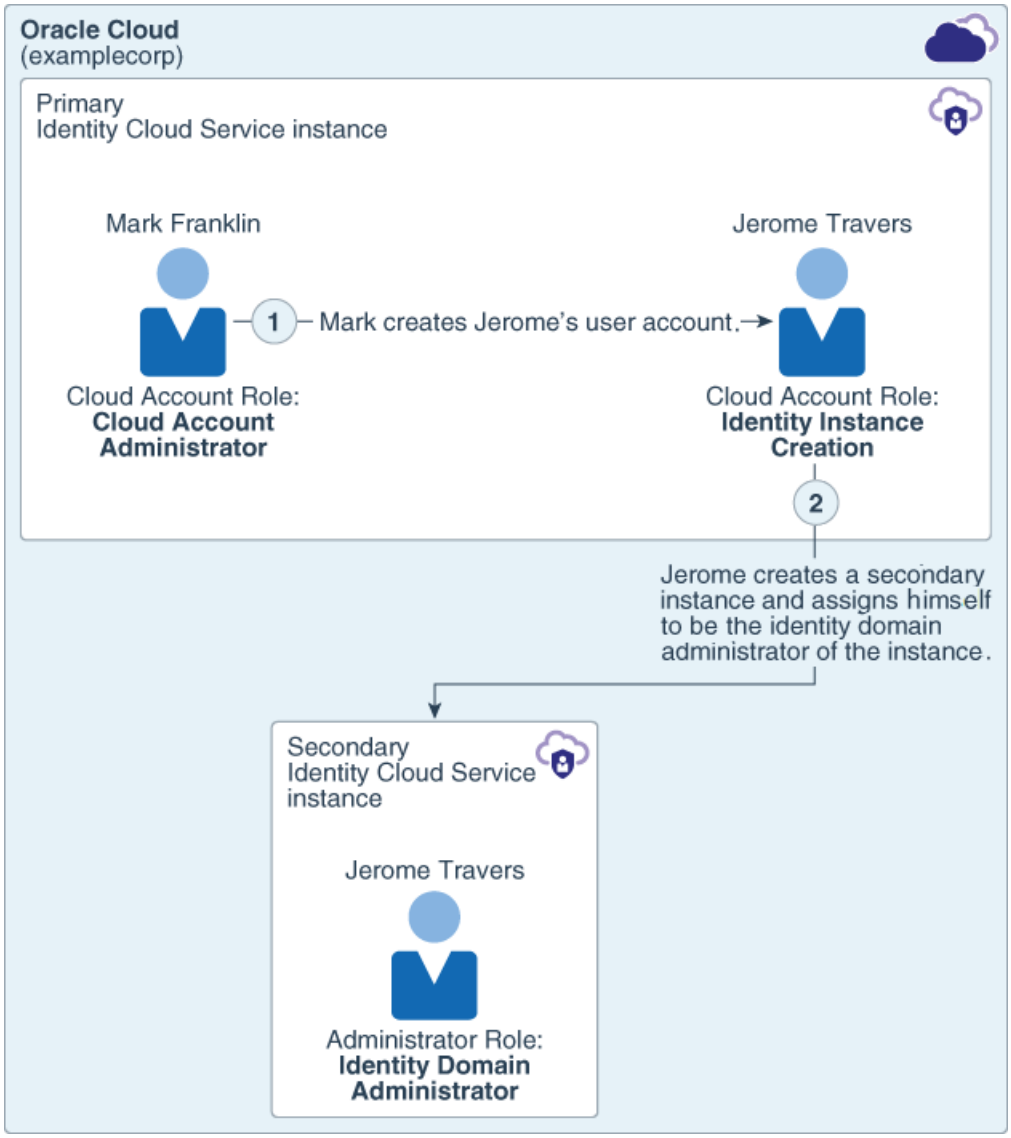


In Example Corp, Mark Franklin is the cloud account administrator of the **examplecorp** cloud account, and is the owner of the primary Oracle Identity Cloud Service instance. He has superuser privileges for this cloud account. Mark wants to have separate Oracle Identity Cloud Service environments for development and production purposes. He creates a user account for Jerome Travers, Example Corp's IT manager, and assigns the **Identity Instance Creation** cloud account role to him. This role gives Jerome the permissions to create and manage Oracle Identity Cloud Service secondary instances. By assigning the **Identity Instance Creation** role to Jerome, Mark delegates the creation of secondary instances to Jerome.

Jerome creates two secondary instances and assigns Angela Johnson, Example Corp's development manager, to be the identity domain administrator of the development instance, and Lyle Robert, Example Corp's production manager, to be the identity domain administrator of the production instance. Because they're identity domain administrators, Angela and Lyle have superuser privileges for their respective secondary Oracle Identity Cloud Service instances. They can manage users, groups, applications, and configuration data in their instances. All work that happens in one instance is isolated from work performed in the other instance so there's a complete separation of work in the development and production instances.

The diagram below shows the scenario in which Jerome Travers creates a secondary instance and assigns himself to be the identity domain administrator for that instance. Jerome now has access to two instances:

- The primary instance because Mark Franklin, the cloud account administrator, created an account for Jerome in that instance and assigned the **Identity Instance Creation** role to him.
- The secondary instance because Jerome is the identity domain administrator of that instance.



If Jerome signs in to Oracle Identity Cloud Service through the secondary instance, accesses the Oracle Cloud Infrastructure Classic Console, and clicks the **Identity Domain** menu in the upper-right corner, below the top menu bar, then two menu items appear: one for the primary instance and one for the secondary instance.

Jerome can use the **Identity Domain** menu to switch to the console associated with the primary Identity Cloud Service instance. He can perform actions associated with any roles assigned to him for either the primary or secondary instance. Because Mark assigned the **Identity Instance Creation** role to him for the primary instance, Jerome can create other secondary instances for the cloud account.

The following table lists the different steps that must be performed to set up secondary instances, the administrators involved for each instance, and what each administrator's tasks are for a particular step.

Table 1-1 Table 1-1 Example of Administrative Responsibilities During a Typical Workflow of Setting Up Multiple Instances

Step Detail	Cloud Account Administrator	Administrator Assigned to the Identity Instance Creation Role	Secondary Instance 1 Administrator (for example, Development)	Secondary Instance 2 Administrator (for example, Production)
Setup the Oracle Cloud account.	• Receives the cloud account administrator and identity domain administrator roles for the primary instance. • Signs in to Oracle Identity Cloud Service to reset their password. • Accesses the Oracle Cloud Infrastructure Classic Console from the Identity Cloud Service console.	No responsibilities for this administrator.	No responsibilities for this administrator.	No responsibilities for this administrator.
Create a user in the primary instance.	• Creates an account for the user who will create or manage secondary instances.	No responsibilities for this administrator.	No responsibilities for this administrator.	No responsibilities for this administrator.

Table 1-1 (Cont.) Table 1-1 Example of Administrative Responsibilities During a Typical Workflow of Setting Up Multiple Instances

Step Detail	Cloud Account Administrator	Administrator Assigned to the Identity Instance Creation Role	Secondary Instance 1 Administrator (for example, Development)	Secondary Instance 2 Administrator (for example, Production)
Delegate the ability to create or manage secondary instances.	• Assigns the **Identity Instance Creation** cloud account role to this user so that the user can create, modify, and remove secondary instances. See Before Creating a Secondary Instance to learn more about how to assign this cloud account role.	• Receives a notification that contains information about how to sign in to the primary Oracle Identity Cloud Service instance of the Oracle Cloud account. • Uses the **Access your Cloud Services** link in the notification to sign in with their user name and the temporary password that's generated by Oracle Identity Cloud Service. • Resets their password. • Clicks the **Dashboard** link on the **Guided Journey** page of the Oracle Cloud Infrastructure Classic Console to create or manage a secondary instance.	No responsibilities for this administrator.	No responsibilities for this administrator.

Table 1-1 (Cont.) Table 1-1 Example of Administrative Responsibilities During a Typical Workflow of Setting Up Multiple Instances

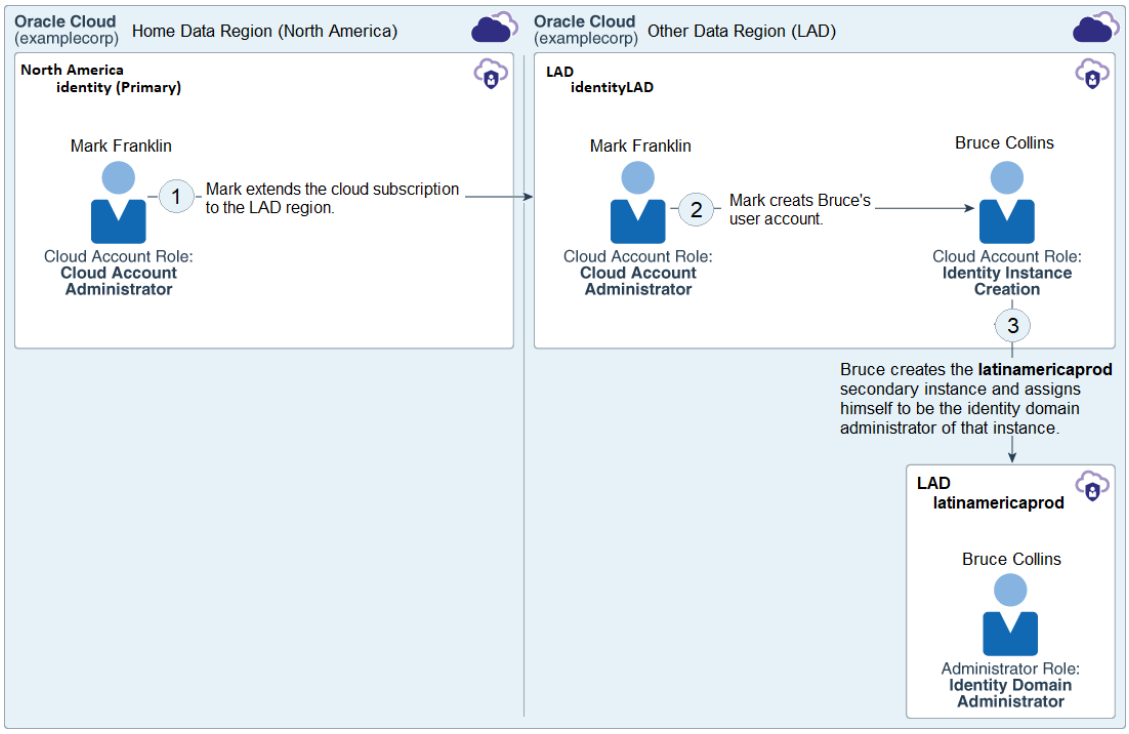
Step Detail	Cloud Account Administrator	Administrator Assigned to the Identity Instance Creation Role	Secondary Instance 1 Administrator (for example, Development)	Secondary Instance 2 Administrator (for example, Production)
Create a secondary instance (for example, Development).	No responsibilities for this administrator.	• Creates a secondary Development instance. • Assigns the Secondary Instance 1 administrator to be the owner of this instance.	• Becomes an identity domain administrator of the secondary instance. • Receives a notification email regarding this new administrator role as well as how to sign in to the secondary Oracle Identity Cloud Service instance of the Oracle Cloud account. • Clicks the link to the right of the **Admin Console URL** field to access the Identity Cloud Service console for this instance. • Resets their password.	No responsibilities for this administrator.

Table 1-1 (Cont.) Table 1-1 Example of Administrative Responsibilities During a Typical Workflow of Setting Up Multiple Instances

Step Detail	Cloud Account Administrator	Administrator Assigned to the Identity Instance Creation Role	Secondary Instance 1 Administrator (for example, Development)	Secondary Instance 2 Administrator (for example, Production)
Create a secondary instance (for example, Production).	No responsibilities for this administrator.	• Creates a secondary Development instance. • Assigns the Secondary Instance 2 administrator to be the owner of this instance.	No responsibilities for this administrator.	• Becomes an identity domain administrator of the secondary instance. • Receives a notification email regarding this new administrator role as well as how to sign in to the secondary Oracle Identity Cloud Service instance of the Oracle Cloud account. • Clicks the link to the right of the **Admin Console URL** field to access the Identity Cloud Service console for this instance. • Resets their password.
Administer the secondary instance.	No responsibilities for this administrator.	No responsibilities for this administrator.	• Uses the Identity Cloud Service console to create and manage more users, if needed, in this secondary instance, as well as to perform additional tasks as an identity domain administrator.	• Uses the Identity Cloud Service console to create and manage more users, if needed, in this secondary instance, as well as to perform additional tasks as an identity domain administrator.

Oracle Identity Cloud Service instances can also be created in data regions that are different from the data region that customers designate when they sign up for Oracle Cloud (the **home data region**). Before creating secondary instances in another data region, customers must extend their subscription to that region. They can then use the Oracle Cloud Infrastructure Classic Console to create secondary instances for the region. See Extending Your Subscription to Another Data Region.

The figure below shows an example of the relationship among various administrators of multiple instances.



In Example Corp (**examplecorp**), Mark Franklin, the cloud account administrator, extends the company's subscription to the Latin America (**LAD**) data region. Mark then creates a user account for Bruce Collins, Example Corp's IT manager for LAD, and assigns the **Identity Instance Creation** role to him. This role gives Bruce permissions to create and manage Oracle Identity Cloud Service secondary instances.

Bruce creates the **latinamericaprod** secondary instance and assigns himself to be the identity domain administrator of that instance. He can use the Identity Cloud Service console to manage users, groups, applications, and configuration data in the instance.

To learn more about the Identity Cloud Service console, see Access Service Consoles.

> **Note:**
>
> The instructions in the rest of this section describe how to create and manage multiple instances through the Identity Cloud Service console. To learn how to perform thes same tasks using the Instance Management feature, see Manage Oracle Identity Cloud Service Secondary Instances.

Before Creating a Secondary Instance

Before you create a secondary instance for Oracle Identity Cloud Service, ensure that:

- You've either set up an Oracle Cloud account or had an account created for you. See Create Users and Assign Roles in Getting Started with Oracle Cloud.
- You're either the cloud account administrator or you've been assigned to the **Identity Instance Creation** role so that you can create the secondary instance. See Learn About Cloud Acount Roles in Getting Started with Oracle Cloud.
- You're in the primary instance of the data region for which you want to create a secondary instance. See Identify and Switch Instances.
- You're familiar with the pricing model for your instance. This pricing model represents the billing metric for the instance you're creating. See Understanding the User Per Month Pricing Model for more information about this pricing model.

Create a Secondary Instance

From the Oracle Cloud Infrastructure Classic Console, you can create a secondary instance for Oracle Identity Cloud Service.

To create this secondary instance, use the **Identity Domain** menu to select the primary instance of the data region for which you want to create the secondary instance. See Identify and Switch Instances for more information about using the **Identity Domain** menu.

Only cloud account administrators or administrators who have been assigned to the **Identity Instance Creation** cloud account role can create a secondary instance.

Each Oracle Identity Cloud Service instance has an instance name and a URL. The instance name is assigned to your instance for Oracle Identity Cloud Service when it's created. The name must be unique within the identity domain.

If you're a user who's assigned to be the administrator of the secondary instance, then use the URL in the notification email that's sent to you to access the instance. If you're a cloud account administrator, then you can access the URL from the Oracle Cloud Infrastructure Classic Console.

If you exceed the maximum number of instances that you can create, then you'll get an error when you click **Create Instance** from the console.

1. Log in to the Identity Cloud Service console.
2. On the **Oracle Cloud** home page, click the **Oracle Cloud** page header.
3. If you aren't now on the **Oracle Cloud Infrastructure Classic** page:
    a. Click the avatar icon, and then select **Service User Console**.
    b. On the **Oracle Cloud My Home** page, click the **Oracle Cloud My Home** page header to go to the **Oracle Cloud Infrastructure Classic** page.
4. In the Oracle Cloud Infrastructure Classic Console, use the **Identity Domain** menu to select the primary instance of the data region for which you want to create a secondary instance, and then click **Create Instance**.
5. In the **Create Instance** dialog box, click the **All Services** tab.
6. In the **Identity Cloud** box, click **Create**. The **Create New Oracle Identity Cloud Service Instance** wizard opens. This wizard steps you through the process of creating an instance.
7. Complete the **Instance Details** page. Specify the following:
    a. **Name:** Specify a unique name for your instance. This name identifies your service within your identity domain. The instance name must start with a letter, and can have

up to 25 lowercase letters and numbers. You can't use spaces and special characters. The name that you provide will appear on Oracle Identity Cloud Service's **Sign In** page for that instance.

b. From the **Plan** list, select **Oracle Identity Cloud Service**.

c. **License Type**: Specify the User per Month pricing model for your instance.

d. In the **Initial Administrator Details** section, specify the administrator credentials for the instance that you're creating. Enter the email address, user name, first name, and last name, as required, in the respective fields.

   To have the administrator access the Oracle Cloud Infrastructure Classic Console with their email address, select the **Use email as user name** check box, and then in the **Email** field, enter the email address for the administrator account.

   To have the administrator access this console with their user name, don't select the **Use email address as user name** check box, and then, in the **User Name** field, enter the user name for the administrator.

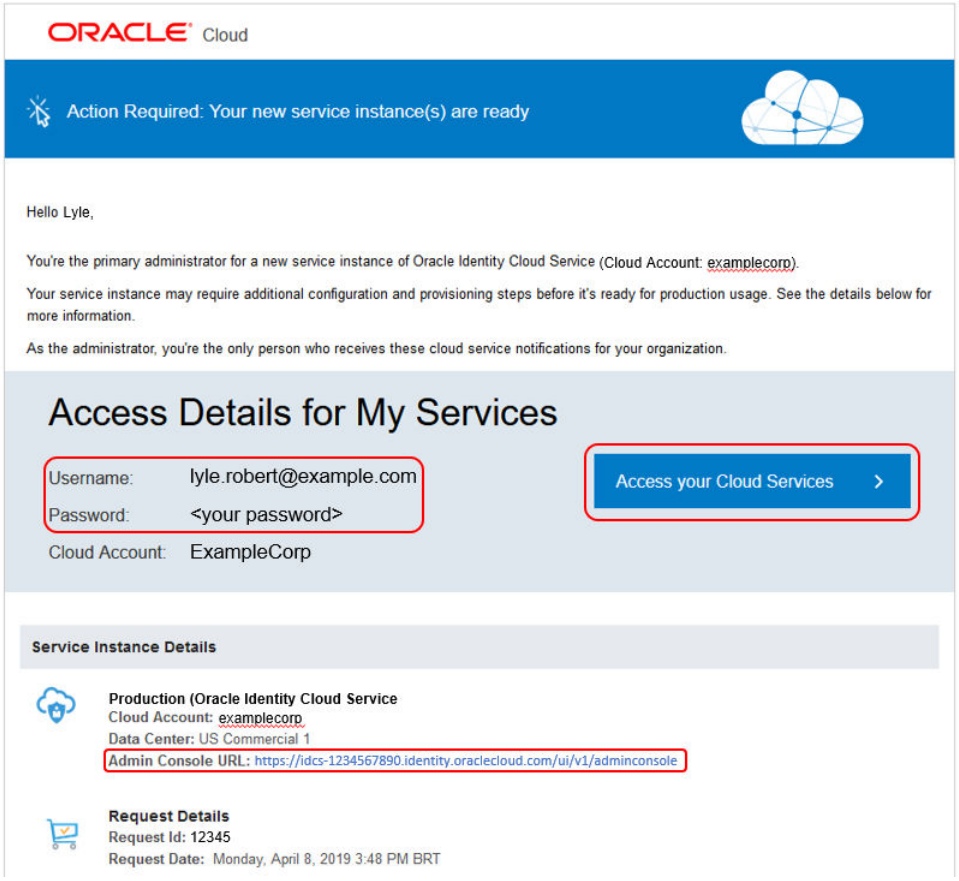
   If you're entering an existing administrator's login credentials, then ensure that the email address and user name are correct. Administrator details are populated automatically based on the logged-in user's details only if such information is available.

   If you're assigning a user to be the administrator of this instance, and this user is already the administrator of either the primary instance or another secondary instance, then the user can switch between the instances to work in each instance. See Identify and Switch Instances for more information about how to switch instances.

8. Click **Create**.

9. In the **Confirmation** window, click **Create**.

The instance is created and the status of the instance is set to **Initialized**. Oracle Cloud sends a **Your new Oracle Identity Cloud Service instance in Oracle Cloud <cloudaccountname> is ready** email notification to the administrator of the instance when the instance is active and ready to use. **<cloudaccountname>** is a placeholder for the name of the Oracle Cloud account that was used to create the secondary instance. For example, if the user name of the cloud account is examplecorp, then the name of the notification will appear as **Your new Oracle Identity Cloud Service instance in Cloud Account examplecorp is ready**.



The notification contains details about the user name and password for the administrator of the secondary instance as well as how this administrator can use this information to access both the Oracle Cloud Infrastructure Classic Console (**Access your Cloud Services**) and the Identity Cloud Service console (**Admin Console URL**).

Use the Oracle Cloud Infrastructure Classic Console to access the **Overview** tab of the **Service: Oracle Identity Cloud Service** page to verify that the instance you created appears. See Modify a Secondary Instance to learn how to access this tab.

You can click the instance name (for an active instance) or you can click the **Open Service Console** link to access the Identity Cloud Service console. For more information on managing the service instance, see Verify That Your Services Are Ready and Manage Your Oracle Cloud Service in Getting Started with Oracle Cloud.

Identify and Switch Instances

Important: If you're a subscriber to the Universal Credits pricing model, then this Oracle Identity Cloud Service feature is available.

After you create a secondary Oracle Identity Cloud Service instance, there are two instances: the primary instance and the secondary instance.

To ensure that you're accessing the secondary instance, and not the primary one, it's important that you learn how to distinguish when you're accessing the primary or secondary instance, and how to switch between them.

You can identify and switch instances from one of the following locations:

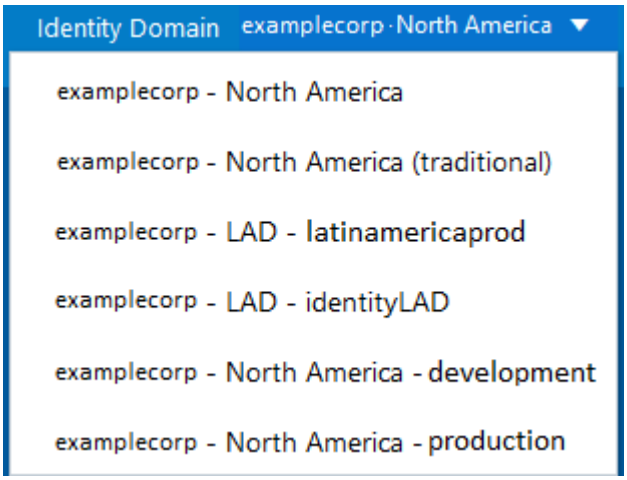
- **Sign In** page: If you're signing in to the secondary instance, then the name of the secondary instance appears in parenthesis after the name of the Oracle Cloud account.

For example, if the name of the Oracle Cloud account is **examplecorp** and the name of the secondary instance is **development**, then **examplecorp (development)** appears on the **Sign In** page. If you're signing in to the primary instance of your home data region, then only the Oracle Cloud account name appears on the **Sign In** page (for this example, **examplecorp**). If you're signing in to the primary instance of another data region, then the name of the instance appears in parenthesis after the name of the Oracle Cloud account. For example, if the name of the primary instance is **identityLAD**, then **examplecorp (identityLAD)** appears on the **Sign In** page.

- Oracle Cloud Infrastructure Classic Console: If you have been assigned to the **Identity Instance Creation** cloud account role in the primary instance or you have been designated to be the identity domain administrator of the secondary instance, then you can access this console. To use the Oracle Cloud Infrastructure Classic Console to switch between instances, a user must sign in to the secondary instance.

  If you have access to this console and you click the **Identity Domain** menu in the upper-right corner, below the top menu bar, then menu items appear. These menu items represent the primary and secondary instances that you have for all of your data regions. See Extending Your Subscription to Another Data Region.

  The top-most menu item is the primary instance of your home data region (for example, **examplecorp - North America**). The primary instance of the home data region is represented by the name of the cloud account and the name of the data region. All other primary and secondary instances contain the name of the cloud account, the name of the data region, and the name of the instance.



In this example, Example Corp (**examplecorp**) has signed up for Oracle Cloud and designated North America as its home data region. Then the subscription has been extended to the Latin America (LAD) data region. Because North America is the home data region, the primary instance appears as **examplecorp - North America**. **examplecorp - North America - development** and **examplecorp - North America - production** are secondary instances of this data region.

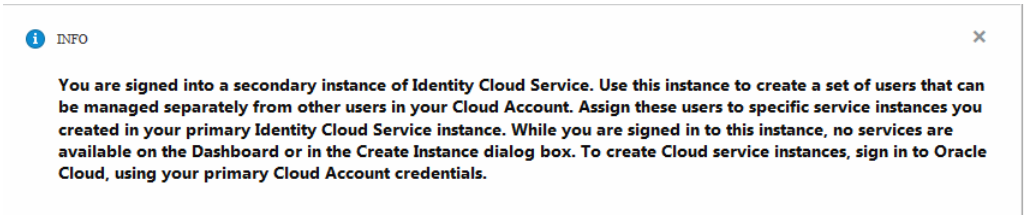
For the LAD data region, **examplecorp - LAD - identityLAD** is the primary instance and **examplecorp - LAD - latinamericaprod** is the secondary instance.

One menu item that appears in the **Identity Domain** menu is labeled **(traditional)**. For this example, this item is **examplecorp - North America (traditional)**. This is associated with a traditional cloud account which doesn't apply if you're using multiple instances.

If you're a user who has been assigned to be the identity domain administrator of secondary instances, then you'll see the primary instance and those instances to which you've been assigned.

If you have signed in using a secondary instance then an **Info** box appears, alerting you that you're in a secondary instance.

Figure 1-1 Secondary Instance Info Notification



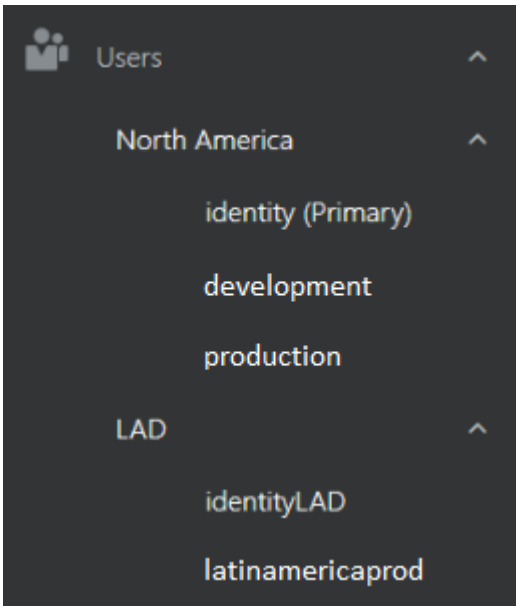
Although the **Info** box doesn't identify the name of the secondary instance, it's useful to confirm that you have signed in using a secondary instance of Identity Cloud Service.

- **Navigation Drawer**: In the Oracle Cloud Infrastructure Classic Console, expand the **Navigation Drawer**, and then expand **Users**. All of the data regions for which you have primary and secondary instances appear. See Extending Your Subscription to Another Data Region.

  By default, the first data region that appears is your home data region. All other data regions for which you have primary and secondary instances appear below the home data region.

  When you expand a data region, the first instance that appears is the primary instance. All secondary instances appear below the primary instance in the order that they were created.

  For the home data region, the primary instance appears as **identity (Primary)**. For all other data regions, the primary instance appears as **identity<data_region>**. For example, if you have a primary instance for the LAD data region, then it would appear as **identityLAD**.



For this example, Example Corp has subscribed to two data regions: one in North America and one in Latin America (LAD). Because the North America data region is the home data region, the primary instance appears as **identity (Primary)**. **development** and **production** are secondary instances of the North America data region.

For the LAD data region, **identityLAD** is the primary instance and **latinamericaprod** is the secondary instance.

If you click the name of an instance, the **User Management** page appears for that instance. In the **User Management** page, click **Identity Console** in the upper-right corner and the Identity Cloud Service console opens.

- **Identity Cloud Service console**: The names of both the primary or secondary instance and the Oracle Cloud account that was used to create this instance appear in this console. To access this information, click the user icon in the upper-right corner of the console, and then select **About** from the drop-down menu. The **Cloud Account Name** and **Instance Name** fields display the names of the Oracle Cloud account and the instance.

Important: By default, there's no single sign-on between Identity Cloud Service instances. If you switch between Identity Cloud Service instances, then you must sign in to each instance.

Modify a Secondary Instance

If you need to change information about a secondary instance, then you can modify it.

Prerequisite: You can modify a secondary instance only if you created the instance or are the administrator of that instance.

You may want to modify a secondary instance (for example, change the tier of its pricing model).

For the secondary instance, you may have selected the **Foundation** tier for the User per Month pricing model. However, you may want to use one of the **Standard** tiers so that you can integrate Oracle Identity Cloud Service with other Oracle Cloud services. These services include Oracle Platform-as-a-Service and Software-as-a-Service. Also, there are custom applications hosted on these two services and they leverage the identity management features and SSO for these services.

You can make change your license from Enterprise to any of the choices that say Monthly. No other license changes are permitted.

You can modify a secondary instance only if you created the instance or are the administrator of that instance.

1. Log in to the Identity Cloud Service console.
2. On the **Oracle Cloud** home page, click the **Oracle Cloud** page header.
3. If you aren't now on the **Oracle Cloud Infrastructure Classic** page:
   a. Click the avatar icon, and then select **Service User Console**.
   b. On the **Oracle Cloud My Home** page, click the **Oracle Cloud My Home** page header to go to the **Oracle Cloud Infrastructure Classic** page.
4. On the **Oracle Cloud Infrastructure Classic** page, in the **Active Services** section, locate the **Identity Cloud** tile.
5. Click the **Action** menu ⋮ in the tile, and then select **View Details**. The **Overview** tab of the **Service: Oracle Identity Cloud Service** page appears. In this tab, the **Service Instances** pane lists all available instances.
6. To filter the list, select from the following type of instances:
   - **Active**: Lists all active and available instances.

- **Inactive**: Lists all instances that don't have an **Active** status. For example, you might see instances with the following statuses: **Initialized**, **Initialization-in-progress**, **Canceled**, **Terminated**, or **Termination-in-progress**.
- **All**: Lists all instances.

7. Locate the secondary instance that you want to modify.
8. Click the **Action** menu to the right of the **Open Service Console** link, and then select **Modify** from the **Action** list. You can modify the pricing model for the secondary instance.
9. In the **License Type** menu, select the pricing model that you want to change for your instance, and then click **Modify**.
10. In the **Confirmation** window, click **Modify**.

Oracle Cloud sends a **Your service instance has been updated** email notification to the administrator. In the notification, details appear about the modification to the secondary instance (for this example, the change to the pricing model).

Remove a Secondary Instance

If you no longer need a secondary instance, then remove it.

Prerequisite: You can remove a secondary instance only if you created the instance or are the administrator of that instance.

1. Log in to the Identity Cloud Service console.
2. On the **Oracle Cloud** home page, click the **Oracle Cloud** page header.
3. If you aren't now on the **Oracle Cloud Infrastructure Classic** page:
   a. Click the avatar icon, and then select **Service User Console**.
   b. On the **Oracle Cloud My Home** page, click the **Oracle Cloud My Home** page header to go to the **Oracle Cloud Infrastructure Classic** page.
4. In the Oracle Cloud Infrastructure Classic Console, locate the **Identity Cloud** tile.
5. Click the **Action** menu in the tile, and then select **View Details**.
6. In the **Service Instances** pane of the **Overview** tab of the **Service: Oracle Identity Cloud Service** page, filter the list of instances. See Modify a Secondary Instance.
7. Locate the secondary instance that you want to remove.
8. Click the **Action** menu to the right of the **Open Service Console** link, and then select **Delete** from the **Action** list.
9. In the **Delete Service Instance** window, click **Delete**.

Oracle Cloud begins to remove the instance, and changes its the status to **Termination in progress**. After the instance is removed completely, Oracle Cloud updates the status of the service instance to **Purged**. Oracle Cloud sends a **Your service instance has been terminated** email notification to the administrator. In the notification, details appear about the instance, including the name of the instance that was removed and the Oracle Cloud account that was associated with it.

About Oracle Identity Cloud Service Concepts

Learn about the basic concepts behind the technologies used in Oracle Identity Cloud Service.

- Oracle Cloud Services

Oracle Cloud Services

Learn about Software as a Service (SaaS), Data as a Service (DaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS) services used in Oracle Cloud.

Oracle Cloud offers a host of cloud services.

Application services are classified into two categories:

- Software as a Service (SaaS): Provides a software licensing and delivery model in which software is licensed on a subscription basis and is centrally hosted.
- Data as a Service (DaaS): Provides data on demand to a user regardless of geographic or organizational separation of the provider and consumer.

Platform services are also classified into two categories:

- Platform as a Service (PaaS): Provides a platform allowing customers to develop, run, and manage applications without the complexity of building and maintaining the infrastructure typically associated with developing and deploying an application.
- Infrastructure as a Service (IaaS): Provides access to computing resources (that is, virtualized hardware and computing infrastructure) in Oracle Cloud across a public connection.

For a comprehensive list of the available Oracle Cloud SaaS, DaaS, PaaS, and IaaS services, go to https://www.oracle.com/cloud and from the **Oracle Cloud** menu, select that category of services that interests you. From the page that opens, you can find links to detailed information about each service.

Oracle Cloud securely integrates its different cloud services, customer applications, and cloud services from other vendors. For example; this integration let's you,

- Embed Oracle CX Sales within your own application running on Oracle Java Cloud Service - SaaS Extension.
- Extend Oracle Fusion Customer Relationship Management Cloud Service with a custom application.
- Tie together an Oracle Cloud service with functionality from other sites, such as Salesforce.
- Use an Oracle Cloud service as the infrastructure for building your own applications.

Identity Domain

Learn about the basic concepts behind an Identity Domain.

An **identity domain** is a construct for managing users and roles, integration standards, external identities, secure application integration through Oracle Single Sign-On (SSO) configuration and OAuth administration. OAuth is an authorization protocol (a set of rules) that allows a third-party website or application to access a user's data without the user sharing login credentials. An identity domain controls the authentication and authorization of the users signing in to a cloud service in Oracle Cloud, and what cloud service features they can access.

An Oracle Cloud service account is a unique customer account that can have multiple cloud services of different service types. For example, you could have three different cloud services, such as Oracle Java Cloud Service, Oracle Database Classic Cloud Service, and Oracle Cloud Infrastructure Compute Classic as part of a single Oracle Cloud service account.

Every Oracle Cloud service belongs to an identity domain. Multiple services can be associated with a single identity domain to share user definitions and authentication. Users in an identity domain can be granted different levels of access to each service associated with the domain to ensure a segregation of duties.

> **Note:**
>
> The term *tenant* is a synonym for *identity domain*. Oracle Cloud is a multitenant system, much like the tenants of a building. So, an identity domain represents one tenant of a multitenant system.

SAML, OAuth, and OpenID Connect

Learn about the basic concepts behind the SAML, OAuth, and OpenID Connect technologies used in Oracle Identity Cloud Service.

Security Assertion Markup Language (SAML) supports both authentication and authorization and is an open framework for sharing security information on the internet through XML documents. SAML includes three parts:

- SAML Assertion: How you define authentication and authorization information.
- SAML Protocol: How you ask (SAML Request) and get (SAML Response) the assertions you need.
- SAML Bindings and Profiles: How SAML assertions ride *on* (Bindings) and *in* (Profiles) industry-standard transport and messaging frameworks.

The OAuth 2.0 token service provided by the Oracle Cloud identity infrastructure provides secure access to the Representational State Transfer (REST) endpoints of cloud services by other cloud services and user applications.

OAuth 2.0 provides the following benefits:

- It increases security by eliminating the use of passwords in service-to-service REST interactions.
- It reduces the lifecycle costs by centralizing trust management between clients and servers. OAuth reduces the number of configuration steps to secure service-to-service communication.

Oracle Identity Cloud Service leverages the power of OpenID Connect and OAuth to deliver a highly-scalable, multi-tenant token service for securing programmatic access to custom applications by other custom applications, and for federated SSO and authorization integration with these applications:

- Use OAuth 2.0 to define authorization in Oracle Identity Cloud Service for your custom applications. OAuth 2.0 has an authorization framework, commonly used for third-party authorization requests with consent. Custom applications can implement both two-legged and three-legged OAuth flows.
- Use OpenID Connect to externalize authentication to Oracle Identity Cloud Service for your custom applications. OpenID Connect has an authentication protocol that provides

Federated SSO, leveraging the OAuth 2.0 authorization framework as a way to federate identities in the cloud. Custom applications participate in an OpenID Connect flow.

Using the OAuth 2.0 and OpenID Connect standards provides the following benefits:

- Federated SSO between the custom application and Oracle Identity Cloud Service. Resource owners (users accessing the custom application) need a single login to access Oracle Identity Cloud Service plus all applications integrated. Oracle Identity Cloud Service handles the authentication and credentials itself, insulating custom applications. This capability is provided by OpenID Connect with OAuth 2.0.
- Authorization to perform operations on third-party servers with consent. Resource owners can decide at runtime whether the custom applications should have authorization to access data or perform tasks for them. This capability is provided by OAuth 2.0.

SCIM

Learn about the basic concepts behind the SCIM technology used in Oracle Identity Cloud Service.

With Oracle Identity Cloud Service REST APIs, you can use a System for Cross-Domain Identity Management (SCIM) to securely manage your Oracle Identity Cloud Service resources, including identities and configuration data. These APIs provide an alternative to using the web-based user interface when you want to use Oracle Identity Cloud Service for your own UI or for clients.

You can manage users, groups, and applications, perform identity functions and administrative tasks, and manage your identity domain settings.

Oracle Identity Cloud Service provides SCIM templates to help you integrate your applications for provisoning and synchronization. See Use the SCIM Interface to Integrate Oracle Identity Cloud Service with Custom Applications.

Other Oracle Identity Cloud Service Key Concepts

Learn about the basic concepts behind the technologies used in Oracle Identity Cloud Service.

- **2-Step Verification:** An authentication method that requires users to use more than one way of verifying their identity, providing a second layer of security to their accounts.
- **Access request:** Allowing users to request group and application access from the Catalog, and view their access requests as well as the groups and applications to which they have access.
- **Access token:** A token that contains all the rights that a user has to access an application.
- **Account recovery:** This automated process is designed to help Oracle Identity Cloud Service users regain access to their accounts if they have trouble signing in, they're locked out, or they forget their passwords.
- **Adaptive Security:** This feature provides strong authentication capabilities for users, based on their behavior within Oracle Identity Cloud Service, and across multiple heterogeneous on-premises applications and cloud services. Adaptive Security is used to analyze a user's risk profile within Oracle Identity Cloud Service, based on their historical behavior, such as too many unsuccessful login attempts and too many unsuccessful MFA attempts, and real-time device context, such as impossible travel between locations, and logins from unknown devices, unfamiliar locations, and suspicious IP addresses. With this enriched context and risk information, Adaptive Security risk profiles each user, and arrives at its own risk score and an overall consolidated risk level (High, Medium, Low) that can be used with Oracle Identity Cloud Service policies to enforce a remediation action, such as

allowing or denying the user from accessing Oracle Identity Cloud Service and its protected applications and resources, requiring the user to provide a second factor to authenticate into Oracle Identity Cloud Service, and so on.

- **Administrator role:** A role that provides user accounts with administrative capabilities in Oracle Identity Cloud Service.
- **Application:** See *Custom application* and *Oracle application*.
- **App Catalog application:** An application that contains a preconfigured application template.
- **Application role:** An entitlement in an Oracle application.
- **Application template:** How a custom application is represented in Oracle Identity Cloud Service.
- **Bridge:** A link between a Microsoft Active Directory enterprise directory structure and Oracle Identity Cloud Service. Oracle Identity Cloud Service can synchronize with this directory structure so that any new, updated, or deleted user or group records are transferred into Oracle Identity Cloud Service. Because of this, the state of each record is synchronized between Microsoft Active Directory and Oracle Identity Cloud Service.
- **Bulk loading:** Loading a large amount of user, group, or application data into Oracle Identity Cloud Service automatically.
- **Bypass code:** A second verification method for Oracle Identity Cloud Service users when they forget their phones, don’t have service, or can’t access their computers. Users can generate bypass codes after they enroll in 2-Step Verification, and then store the codes in a safe place.
- **Confidential application:** A custom application that's accessed by multiple users, hosted in a secure and protected place (server), and uses OAuth 2.0.
- **Cross-Origin Resource Sharing (CORS):** Client applications that run on one identity domain can obtain data from another identity domain.
- **Custom application:** An application (such as a mobile application, a web page, a client application, or a server application) that you can integrate with Oracle Identity Cloud Service. By default, for security purposes, custom applications are trusted or confidential.
- **Default settings:** Oracle Identity Cloud Service settings that are applied to a customer's entire identity domain. These settings include the time zone, password recovery email, signing certificate settings, contact information, and language for the identity domain.
- **Delegated administration:** Providing user accounts with administrative capabilities in Oracle Identity Cloud Service.
- **Delegated authentication:** Enabling users to use their Microsoft Active Directory passwords to sign in to Oracle Identity Cloud Service to access resources and applications protected by Oracle Identity Cloud Service.
- **Digital certificate:** An electronic passport that allows a person, computer, or organization to exchange information securely over the Internet using the public key infrastructure (PKI). A digital certificate may be referred to as a public key certificate.
- **Federated SSO:** Provides a higher level of security and control for an identity provider because a security token is used to authenticate the user against both the identity provider and Oracle Identity Cloud Service.
- **Group:** The link between user accounts and applications in Oracle Identity Cloud Service. Groups are designed to ease the administration of privileges that you grant to user accounts.

- **Identity provider:** This type of provider, also known as an Identity Assertion provider, provides identifiers for users who want to interact with Oracle Identity Cloud Service using a website that's external to Oracle Identity Cloud Service.
- **Identity provider policy:** Criteria that Oracle Identity Cloud Service uses to display specific identity providers for users to sign in to Oracle Identity Cloud Service when they are accessing particular apps.
- **Job:** A batch execution of importing or exporting users, groups, or application roles in Oracle Identity Cloud Service.
- **Mobile application:** A custom application that's hosted directly on the resource owner's browser, machine, or mobile device.
- **Multi-Factor Authentication (MFA):** A method of authentication that requires the use of more than one factor to verify a user's identity.
- **Network perimeter:** A defined list of IP addresses that Oracle Identity Cloud Service can evaluate to determine whether users who use these IP addresses can sign in to Oracle Identity Cloud Service.
- **Oracle Mobile Authenticator (OMA) application:** A mobile device app that users can use as a second verification method.
- **Oracle application:** A complete and modular enterprise application, engineered from the ground up to be cloud-ready and to coexist seamlessly in mixed environments.
- **Password policy:** A set of password-related criteria that you set in Oracle Identity Cloud Service and assign to a group. The policy then applies to all users in the group.
- **Password recovery email address:** A user's email address to which Oracle Identity Cloud Service password recovery notifications are sent. By default, a user's primary email address is also the user's password recovery email address. However, a user has the option of specifying a password recovery email address that is different than the primary email address.
- **Passwordless authentication:** Allows access to the protected resource by entering the user name, and then completing an administrator-specified authentication method, instead of supplying a password.
- **Primary email address:** A user's email address to which all Oracle Identity Cloud Service notifications are sent.
- **Profile:** A collection of useful data about you in Oracle Identity Cloud Service. Your profile includes contact information, account information, and also settings that determine the time zone and language that displays for your account in the Identity Cloud Service console.
- **Provisioning:** Managing the lifecycle of user accounts in Software as a Service (SaaS) applications, such as creating and deleting accounts using Oracle Identity Cloud Service.
- **Provisioning Bridge:** A link between on-premises apps and Oracle Identity Cloud Service. The Provisioning Bridge can synchronize with these apps so that any new, updated, or deleted user or group records are transferred into Oracle Identity Cloud Service. As a result, the state of each record is synchronized between the apps and Oracle Identity Cloud Service.
- **Refresh token:** A secure mechanism to obtain a new access token when the current access token expires.
- **Resource server application:** A third-party custom application that provides services that a web application can consume on behalf of the user.
- **SAML application:** A custom application that's accessed by multiple users, hosted in a secure and protected place (server), and uses SAML 2.0.

- **Security Questions:** Questions presented to users as part of 2-Step Verification. See 2-Step Verification.
- **Self-registration profile:** A profile created by an administrator to manage different sets of users, approval policies, and applications in Oracle Identity Cloud Service.
- **Service provider:** A website such as Oracle Identity Cloud Service that hosts applications.
- **Sign-on policy:** Criteria that Oracle Identity Cloud Service uses to allow or deny access to apps that are assigned to users.
- **Social Login:** Accessing Oracle Identity Cloud Service using credentials from trusted public identity providers such as LinkedIn, Facebook, Twitter, Google, and Microsoft. Users can also log in to these providers to create an account in Oracle Identity Cloud Service if they don't have one.
- **Synchronization:** Controlling how operations such as creating and deleting accounts in SaaS applications are reflected in Oracle Identity Cloud Service.
- **Tag:** A key-value pair that is used to organize and identify an application.
- **Trusted partner:** Any application or organization, remote to Oracle Identity Cloud Service, that communicates with Oracle Identity Cloud Service.
- **User account:** How a user is represented in Oracle Identity Cloud Service. A user account enables the user to access the Oracle Cloud service to which they belong. In Oracle Identity Cloud Service, there is a one-to-one relationship between a user and a user account.
- **User life cycle:** The process flow of how a user account is created, managed, and deleted in Oracle Identity Cloud Service based on certain events or time factors.

About Oracle Identity Cloud Service Interfaces

The following summarizes the key interfaces to Oracle Identity Cloud Service:

- The service consoles: See Access Service Consoles.
- The client for the bridge: See Manage Microsoft Active Directory (AD) Bridges for Oracle Identity Cloud Service.
- Rest APIs: See REST API for Oracle Identity Cloud Service.

How to Begin with Oracle Identity Cloud Service Subscriptions

Here’s how to get started with Oracle Identity Cloud Service subscriptions:

1. Purchase a nonmetered subscription. See Buying a Nonmetered Subscription to an Oracle Cloud Service in Getting Started with Oracle Cloud.
2. Set up your account or activate your order. See Activating Your Order in Getting Started with Oracle Cloud.
3. Verify that Oracle Identity Cloud Service is ready to use. See Verifying That Metered Oracle Cloud Services Are Running or Verifying That a Service Is Running in Getting Started with Oracle Cloud.
4. Learn about user accounts, groups, applications, and application roles. See About Oracle Identity Cloud Service User Accounts and Groups and About Oracle Identity Cloud Service Applications and Application Roles.
5. Start using Oracle Identity Cloud Service. See Configure User Settings and Change Oracle Identity Cloud Service Default Settings.

6. Create accounts for your Oracle Identity Cloud Service users and groups. See Managing Oracle Identity Cloud Service Users and Managing Oracle Identity Cloud Service Groups.
7. Assign your users and groups to Oracle Identity Cloud Service applications and application roles. See Managing Oracle Identity Cloud Service Applications.

Supported Web Browsers

Oracle Identity Cloud Service supports the following web browsers:

OS	Chrome	Firefox	Internet Explorer **	Microsoft Edge	Safari
Android	Not Supported	Not Supported	N/A	Not Supported	N/A
iOS	Not Supported	Not Supported	N/A	Not Supported	Not Supported
Mac OSX	Supported	Supported	N/A	N/A	Supported
Windows	Supported	Supported	Supported (IE11 Only)	Supported	N/A

Note:

- Support for Microsoft Browsers will follow the same N-1 support policy that iOS provides. The most recent version plus one previous release. As of January 12th 2016, this means the most recent version of Microsoft Edge and IE11 only.
- Android/iOS mobile browsers are not supported.

How to Access Oracle Identity Cloud Service

Access Oracle Identity Cloud Service through a service web console or the REST API.

Depending on how you signed up for Oracle Cloud, you'll be directed to either the Oracle Cloud Infrastructure Console or the Oracle Cloud Infrastructure Classic Console.

Topics:

- Access Oracle Identity Cloud Service from the Oracle Cloud Infrastructure Console
- Access Oracle Identity Cloud Service from the Oracle Cloud Infrastructure Classic Console

Access Oracle Identity Cloud Service from the Oracle Cloud Infrastructure Console

On most Oracle Cloud accounts, you access the Oracle Identity Cloud Service console from the Oracle Cloud Infrastructure Console.

1. Sign in to Oracle Cloud.

   If you received a welcome email, use it to identify the URL, your user name, and your temporary password. After signing in, you will be prompted to change your password.

2. From the Oracle Cloud Infrastructure Console, click the navigation menu  in the top left corner, expand **Identity**, and then click **Federation**.
3. In the **Federation** page, click the **Oracle Identity Cloud Service Console** link.

   If multiple instances are listed, click the **Oracle Identity Cloud Service Console** link for the console instance you want to open.

Access Oracle Identity Cloud Service from the Oracle Cloud Infrastructure Classic Console

On some older Oracle Cloud accounts, you access the Oracle Identity Cloud Service console from the Oracle Cloud Infrastructure Classic Console.

1. Sign in to Oracle Cloud.

   If you received a welcome email, use it to identify the URL, your user name, and your temporary password. After signing in, you will be prompted to change your password.
2. From the Oracle Cloud Infrastructure Classic Console, click the navigation menu  in the top left corner, and then click **Users**.

   Alternatively, mouse-over **Users** and then click the name of one of the Oracle Identity Cloud Service instances on the sub menu that opens.
3. In the **User Management** page, click **Identity Console** in the upper right corner.

Access Service Consoles

This overview describes the ways that administrators and users can use the service consoles in conjunction with Oracle Identity Cloud Service.

Use the following sections to learn about key elements for each service console.

Topics:

Sign In Page

Learn how to sign in, set, and reset your password.

When your account has been added to Oracle Identity Cloud Service, you receive an activation email instructing you to activate your account. Click the activation link, and then set your password.

If you forget your own password and can't sign in to Oracle Identity Cloud Service, you can reset your password using your user name. See Recover Your Account.

There are various ways that you can sign in and authenticate including email, passwordless authentication, and social accounts, If your administrator has configured Passwordless Authentication, you can choose to use it to bypass the standard web-form-based authentication using email or a mobile device to sign in. For more details, see Understand Passwordless Authentication.

My Profile Console

Use this console to set up or modify your profile (for example, time zone and language preferences), manage your passwords, set your primary and recovery email addresses, and link your social login accounts if you are using social login.

To access the My Profile console, click the avatar icon in the top-right corner, and then select **My Profile**.

Element	Description
My Profile Details	Set up your profile information for the first time or modify your current profile information. See Set Up or Modify Your Profile.
Change My Password	Change your password to Oracle Identity Cloud Service. See Change Your Password.
Email Options	Change your primary email address. See Set Your Email Options.
Security	Set a recovery email address, provide a mobile number, or select and answer security questions to help you regain access to your account if you have trouble signing in, you're locked out, or you forget your password. See Set Your Account Recovery Options.
Social Accounts	Link your social account to your Oracle Identity Cloud Service user account so that you can use your social account's login credentials to access Oracle Identity Cloud Service.
My Access	View the groups and applications to which you have been granted access. See View Group and Application Access.
My Requests	View your requests for access to groups and applications. See View Group and Application Access Requests.

Identity Cloud Service Console

Depending on your administrator type, use this console to manage users, groups, applications, administrative settings and security settings, customize the service, and run reports.

Oracle Identity Cloud Service provides you with a **Navigation Drawer** to maximize the real estate of the Identity Cloud Service console.

To display the **Navigation Drawer**, click the **Action menu**  in the upper-left corner of the console. You'll see a listing of all folders and pages that compose the console.

Click a folder to see the pages associated with the folder. Then, click the menu item that represents the page that you want to display in the Identity Cloud Service console.

To hide the **Navigation Drawer**, click the **Action menu**  again.

The following table describes the key elements shown in the Identity Cloud Service console.

Page	Description
Dashboard	Use this introductory page for the Identity Cloud Service console to access the Oracle Identity Cloud Service documentation library, videos, tutorials, the dashboard, reports, and links that administrators use frequently.
Users	Create, manage, and remove user accounts. See Managing Oracle Identity Cloud Service Users.
Groups	Create, manage, and remove groups. See Managing Oracle Identity Cloud Service Groups.
Applications	Create, manage, and remove custom applications. See Managing Oracle Identity Cloud Service Applications.
Oracle Cloud Services	View Oracle applications. See Managing Oracle Identity Cloud Service Applications.
Jobs	Review the overall status of all jobs, the details for a specific job, and download a job file. See Viewing Jobs and Job Details.
Reports	Run user, application, and diagnostic data reports. See Running Oracle Identity Cloud Service Reports.
Settings	Set up and manage default settings, user settings, trusted partner certificates, notifications, password policies, branding, bridges, diagnostics, session settings, self-registration profiles, SDKs and custom applications, and schemas.
Default Settings	Specify whether users can set their own password recovery email address, the default tenant locale, and the default tenant contact information for an identity domain. See Change Default Settings.
User Settings	Specify whether the primary email address is required or optional to create a user account. See Manage User Settings in Oracle Identity Cloud Service.
Partner Settings	Add, manage, and use trusted partner certificates. See Manage Oracle Identity Cloud Service Trusted Partner Certificates.
Notifications	Customize and use notifications. See Customize Oracle Identity Cloud Service Notifications.
Password Policies	Set, test, modify, and evaluate password policies. See Managing Oracle Identity Cloud Service Password Policies.
Branding	Customize the **Sign In** page and brand the Identity Cloud Service console and notification templates by adding logos to them. See Customizing the Oracle Identity Cloud Service Interface.
Provisioning Bridges	Create, manage, and remove Provisioning Bridges. See Manage Provisioning Bridges for Oracle Identity Cloud Service.
Directory Integrations	Create, manage, and remove bridges. See Manage Microsoft Active Directory (AD) Bridges for Oracle Identity Cloud Service.
Diagnostics	Set the diagnostic type to capture operational logs. See Run Oracle Identity Cloud Service Reports.
Session Settings	Specify the session expiration and the logout URL for an identity domain. See Change Session Settings.

Page	Description
Self-Registration	Create self-registration profiles to manage different sets of users, approval policies, and applications. See Manage Self-Registration Profiles in Oracle Identity Cloud Service.
Downloads	Download software development kits (SDKs) to enable your mobile and Web applications to authenticate and integrate with Oracle Identity Cloud Service. Download applications, including the Oracle E-Business Suite (EBS) Asserter to integrate Oracle E-Business Suite with Oracle Identity Cloud Service, the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) to integrate your Linux environment with Oracle Identity Cloud Service to perform user authentication with first-factor and second-factor authentication, the Secure Form Fill Client to configure Secure Form Fill for your applications, the Identity Cloud Service Device Fingerprint Utility to enable the **Access for an unknown device** event of Adaptive Security for a custom sign-in page, and the Provisioning Bridge client to install, start, and stop the bridge. The Provisioning Bridge provides a link between your on-premises apps and Oracle Identity Cloud Service. See Download Oracle Identity Cloud Service SDKs and Applications.
Security	Set up and manage Terms of Use, delegated administration, Adaptive Security, identity providers, identity provider policies, sign-on policies, network perimeters, App Gateways, account recovery, Multi-Factor Authentication (MFA), MFA factors, OAuth, and delegated authentication.
Administrators	After you create or import user accounts, delegate administrative responsibilities for these accounts. See Add or Remove a User Account from an Administrator Role.
Adaptive Security	Activate Adaptive Security, and add, manage, and use risk providers. See Manage Adaptive Security in Oracle Identity Cloud Service.
Identity Providers	Add, manage, and use identity providers. See Manage Oracle Identity Cloud Service Identity Providers.
IDP Policies	Create, manage, and remove identity provider (IDP) policies. See Manage Oracle Identity Cloud Service Identity Provider Policies.
Sign-On Policies	Create, manage, and remove sign-on policies. See Manage Oracle Identity Cloud Service Sign-On Policies.
Network Perimeters	Define network perimeters. See Manage Oracle Identity Cloud Service Network Perimeters.
App Gateway	Use App Gateway to integrate web applications hosted either on a compute instance in a cloud infrastructure, or in an on-premises server with Oracle Identity Cloud Service for authentication purposes. See Manage Oracle Identity Cloud Service App Gateways.
Account Recovery	Configure factors that will help users regain access to their accounts if they have trouble signing in, they're locked out, or they forget their passwords. See Manage Account Recovery in Oracle Identity Cloud Service.

Page	Description
MFA	Enable and disable Multi-Factor Authentication (MFA), and configure MFA settings. See Manage Oracle Identity Cloud Service Multi-Factor Authentication Settings.
Factors	Configure authentication factors for MFA. See Manage Oracle Identity Cloud Service Multi-Factor Authentication Settings.
OAuth	Configure the default OAuth settings for your environment. See Configure OAuth Settings.
Delegated Authentication	Configure delegated authentication for bridges associated with Microsoft Active Directory domains. See Configure Delegated Authentication in Oracle Identity Cloud Service.

My Apps

On the **My Apps** page, you can access all apps assigned to you.

You can sort these apps by their names or by the dates when they were granted to you. For organizational purposes, you can designate preferred apps as favorites for future easy reference and access. See Access My Apps for more information about the **My Apps** page.

Catalog

Use this page to request access to groups of which you want to be a member and applications that you want to use.

See Request Group and Application Access.

2–Step Verification

Use this page to enroll in Multi–Factor Authentication (MFA) in Oracle Identity Cloud Service.

When you sign in to Oracle Identity Cloud Service, you're prompted for your user name and password, which is the first factor. You're then required to provide a second type of verification. This is called 2-Step Verification. The two factors work together to add an additional layer of security in Oracle Identity Cloud Service by using either additional information or a second device to verify your identity and complete the login process.

See Manage 2-Step Verification from the My Profile Console for more information about the **2–Step Verification** page.

About Oracle Identity Cloud Service User Accounts and Groups

This overview of user accounts and groups briefly explains what they are and how they are used.

A **user account** is an abstraction representing a way to be authenticated to access Oracle Identity Cloud Service. In Oracle Identity Cloud Service, the cardinality of relationship between user and account is one-to-one.

By default, all users can use their accounts to perform self-service capabilities in Oracle Identity Cloud Service. Users can update their profiles, reset their passwords, unlock their accounts, change their email preferences, and link social login accounts. As an Identity

Domain Administrator, you may want to provide a user account with administrative capabilities in Oracle Identity Cloud Service. For example, in order to off-load some responsibilities, you may want to assign a user the User administrator role so that they can manage users, groups, and group memberships. To provide a user account with administrative capabilities, you assign administrator roles to user accounts. See Understanding Administrator Roles for more information about administrator roles and privileges that you can assign to user accounts.

As an administrator, you have easy and controlled privilege management through groups. **Groups** are the links between user accounts and applications in Oracle Identity Cloud Service. Groups are designed to facilitate the administration of privileges that you grant to user accounts. See Managing Oracle Identity Cloud Service Groups.

About Oracle Identity Cloud Service Applications and Application Roles

This overview briefly describes applications and application roles.

Oracle Identity Cloud Service provides you with a secure and centralized cloud service to manage the relationships that your users and groups have with your:

- **Cloud-based Oracle applications:** A complete and modular set of web-based enterprise applications, engineered to be cloud-ready and coexist seamlessly in mixed environments. You can use Oracle Cloud applications by accessing the UI on your local web browser or through your mobile communications device connected to the Internet.
- **Custom applications:** Web applications that are written in a server-side language and can run on a server where the source code of the application isn't available to the public.

You can use Oracle Identity Cloud Service to grant users access to applications in two ways:

- Directly: Assigning users to applications.
- Indirectly: Assigning groups to applications. Users who are members of the groups are granted access to the applications.

In addition to granting users and groups access to Oracle applications, you can grant users and groups access to entitlements within applications. Each entitlement in an Oracle application is represented by an **Application Role**.

When a customer purchases or subscribes to any cloud services, the services are created in Oracle Identity Cloud Service as Applications. These services (Oracle Public Cloud Apps) have service consoles and the Application Roles control the authorization into these service consoles. Only PaaS services use Application Roles.

See Managing Oracle Identity Cloud Service Applications.

Typical Workflow for Using Oracle Identity Cloud Service

Oracle Identity Cloud Service has five administrator roles and one user role. To start using Oracle Identity Cloud Service as an administrator, click the following links. Each link provides you with a guide of how to start using Oracle Identity Cloud Service as that administrator or user.

- Identity Domain Administrator
- Security Administrator
- Application Administrator

- User Administrator
- User Manager
- Help Desk Administrator
- Audit Administrator

> **Note:**
>
> See Understanding Administrator Roles to learn more about the privileges for each administrator or user role.

Identity Domain Administrator

An identity domain administrator has superuser privileges for an identity domain in Oracle Identity Cloud Service. All other Oracle Identity Cloud Service administrators have a subset of these privileges.

To start using Oracle Identity Cloud Service as an identity domain administrator, use the typical workflow below.

Task	Description	Additional Information
Customize the interface.	Customize the **Sign In** page or brand the **Identity Cloud Service** console and notification templates by adding logos to them.	Customizing the Oracle Identity Cloud Service Interface
Customize the default settings.	Customize the default settings for both the identity domain and the session between the Oracle Identity Cloud Service client and the server.	Change Oracle Identity Cloud Service Default Settings
Manage user settings.	Specify whether the primary email address is required or optional to create a user account.	Change User Settings
Customize email notifications.	Customize email notifications for users and administrators.	Customize Oracle Identity Cloud Service Notifications
Customize the password policy.	Tailor the strength of the password policies.	Managing Oracle Identity Cloud Service Password Policies
Configure Multi-Factor Authentication (MFA)	Enable MFA when you want to require your administrators and users to provide a second type of verification when they log in: • Configure overall MFA policy settings such as which users are to use MFA and whether MFA is required. • Configure the type of factors that you want to allow and specific policies for those factors.	Configure Authentication Factors

Task	Description	Additional Information
Configure account recovery.	Configure factors that will help users regain access to their accounts if they have trouble signing in, they're locked out, or they forget their passwords.	Manage Account Recovery in Oracle Identity Cloud Service
Onboard users and groups.	Onboard users and groups by: • Installing, configuring, and running bridges • Importing users and groups • Creating users and groups	Manage Microsoft Active Directory (AD) Bridges for Oracle Identity Cloud Service Managing Oracle Identity Cloud Service Users Managing Oracle Identity Cloud Service Groups
Manage delegated authentication.	Configure delegated authentication for bridges associated with Microsoft Active Directory domains.	Configure Delegated Authentication in Oracle Identity Cloud Service
Create and manage custom applications.	Add and configure custom applications.	Managing Oracle Identity Cloud Service Applications
Assign users and groups to applications.	Assign users and groups to Oracle and custom applications.	
Perform delegated administration.	After you create or import user accounts, you can delegate administrative responsibilities for these accounts.	Managing Oracle Identity Cloud Service Users
Add and manage identity providers.	Add and manage identity providers to provide identifiers for users who want to interact with Oracle Identity Cloud Service using a website that's external to Oracle Identity Cloud Service.	Manage Oracle Identity Cloud Service Identity Providers
Manage identity provider policies.	Manage identity provider policies to restrict which identity providers appear on the **Sign In** page when users are accessing particular apps.	Manage Oracle Identity Cloud Service Identity Provider Policies
Define network perimeters.	Create network perimeters to restrict the IP addresses that users can use to log in to Oracle Identity Cloud Service.	Manage Oracle Identity Cloud Service Network Perimeters
Manage sign-on policies.	Manage sign-on policies to define criteria that Oracle Identity Cloud Service uses to allow or deny access to users for apps that are assigned to them.	Manage Oracle Identity Cloud Service Sign-On Policies
Manage Adaptive Security and risk providers.	Activate Adaptive Security, and add, manage, and use risk providers to evaluate risk-based activity for Oracle Identity Cloud Service users, and generate a risk score for these users, based on this activity. This risk score is a number that varies from risk provider to risk provider, reflecting user threat.	Manage Adaptive Security in Oracle Identity Cloud Service

Task	Description	Additional Information
Import trusted partner certificates.	Import certificates for trusted partners so that any application or organization, remote to Oracle Identity Cloud Service, can communicate with Oracle Identity Cloud Service.	Manage Oracle Identity Cloud Service Trusted Partner Certificates
Create Self-Registration Profiles	Add your customized header and footer logos, determine your allowed email domains, and add header, footer, success, and user consent text that will be used for self-registration.	Create Self-Registration Profiles
Run user and application reports.	Run user and application reports to, for example, review user login attempts or user access to applications.	Running Oracle Identity Cloud Service Reports
Download SDKs and applications.	Download software development kits (SDKs) to enable your mobile and Web applications to authenticate and integrate with Oracle Identity Cloud Service, the Oracle E-Business Suite (EBS) Asserter to integrate Oracle E-Business Suite with Oracle Identity Cloud Service, or the Secure Form Fill Client to configure Secure Form Fill for your applications.	Download Oracle Identity Cloud Service SDKs and Applications

Security Administrator

A security administrator can manage Oracle Identity Cloud Service security settings for an identity domain in Oracle Identity Cloud Service.

Security administrators can customize the interface, default settings, notifications, and the password policies, configure Multi-Factor Authentication (MFA), and manage bridges, identity providers, and trusted partner certificates. See Understanding Administrator Roles.

Task	Description	Additional Information
Customize the interface.	Customize the **Sign In** page or brand the **Identity Cloud Service** console and notification templates by adding logos to them.	Customizing the Oracle Identity Cloud Service Interface
Customize the default settings.	Customize the default settings for both the identity domain and the session between the Oracle Identity Cloud Service client and the server.	Change Oracle Identity Cloud Service Default Settings
Manage user settings.	Specify whether the primary email address is required or optional to create a user account.	Manage User Settings in Oracle Identity Cloud Service
Customize email notifications.	Customize email notifications for users and administrators.	Customize Oracle Identity Cloud Service Notifications

Task	Description	Additional Information
Customize the password policies.	Tailor the strength of the password policies.	Managing Oracle Identity Cloud Service Password Policies
Configure Multi-Factor Authentication (MFA)	Enable MFA when you want to require your administrators and users to provide a second type of verification when they log in: • Configure overall MFA policy settings such as which users are to use MFA and whether MFA is required. • Configure the type of factors that you want to allow and specific policies for those factors.	Configure Authentication Factors
Register App Gateway	Register App Gateway to protect access to enterprise applications.	Manage Oracle Identity Cloud Service App Gateways
Configure account recovery.	Configure factors that will help users regain access to their accounts if they have trouble signing in, they're locked out, or they forget their passwords.	Manage Account Recovery in Oracle Identity Cloud Service
Onboard users and groups.	Onboard users and groups by installing, configuring, and running bridges.	Manage Provisioning Bridges for Oracle Identity Cloud Service Manage Microsoft Active Directory (AD) Bridges for Oracle Identity Cloud Service
Manage delegated authentication.	Configure delegated authentication for bridges associated with Microsoft Active Directory domains.	Configure Delegated Authentication in Oracle Identity Cloud Service
Add and manage identity providers.	Add and manage identity providers to provide identifiers for users who want to interact with Oracle Identity Cloud Service using a website that's external to Oracle Identity Cloud Service.	Manage Oracle Identity Cloud Service Identity Providers
Manage identity provider policies.	Manage identity provider policies to restrict which identity providers appear on the **Sign In** page when users are accessing particular apps.	Manage Oracle Identity Cloud Service Identity Provider Policies
Define network perimeters.	Create network perimeters to restrict the IP addresses that users can use to log in to Oracle Identity Cloud Service.	Manage Oracle Identity Cloud Service Network Perimeters
Manage sign-on policies.	Manage sign-on policies to define criteria that Oracle Identity Cloud Service uses to allow or deny access to users for apps that are assigned to them.	Manage Oracle Identity Cloud Service Sign-On Policies

Task	Description	Additional Information
Manage Adaptive Security and risk providers.	Activate Adaptive Security, and add, manage, and use risk providers to evaluate risk-based activity for Oracle Identity Cloud Service users, and generate a risk score for these users, based on this activity. This risk score is a number that varies from risk provider to risk provider, reflecting user threat.	Manage Adaptive Security in Oracle Identity Cloud Service
Import trusted partner certificates.	Import certificates for trusted partners so that any application or organization, remote to Oracle Identity Cloud Service, can communicate with Oracle Identity Cloud Service.	Manage Oracle Identity Cloud Service Trusted Partner Certificates
Download SDKs and applications.	Download software development kits (SDKs) to enable your mobile and Web applications to authenticate and integrate with Oracle Identity Cloud Service. Download applications, including the Oracle E-Business Suite (EBS) Asserter to integrate Oracle E-Business Suite with Oracle Identity Cloud Service, the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) to integrate your Linux environment with Oracle Identity Cloud Service to perform user authentication with first-factor and second-factor authentication, Identity Cloud Service App Gateway to integrate your application with Oracle Identity Cloud Service for authentication purposes, the Secure Form Fill Client to configure Secure Form Fill for your applications, the Identity Cloud Service Device Fingerprint Utility to enable the **Access for an unknown device** event of Adaptive Security for a custom sign-in page, and the Provisioning Bridge client to install, start, and and stop the bridge. The Provisioning Bridge provides a link between your on-premises apps and Oracle Identity Cloud Service.	Download Oracle Identity Cloud Service SDKs and Applications

Application Administrator

An application administrator can manage Oracle Identity Cloud Service applications.

Application administrators can create, update, activate, deactivate, and delete applications. Application administrators can also grant and revoke access to applications for groups and users. See Understanding Administrator Roles.

Task	Description	Additional Information
Create and manage custom applications.	Add and configure custom applications.	Managing Oracle Identity Cloud Service Applications
Assign users and groups to applications.	Assign users and groups to Oracle and custom applications.	
Manage identity provider policies.	Manage identity provider policies to restrict which identity providers appear on the **Sign In** page when users are accessing particular apps.	Manage Oracle Identity Cloud Service Identity Provider Policies
Define network perimeters.	Create network perimeters to restrict the IP addresses that users can use to log in to Oracle Identity Cloud Service.	Manage Oracle Identity Cloud Service Network Perimeters
Manage sign-on policies.	Manage sign-on policies to define criteria that Oracle Identity Cloud Service uses to allow or deny access to users for apps that are assigned to them.	Manage Oracle Identity Cloud Service Sign-On Policies
Run application reports.	Run operational or historical reports that capture data about Oracle Identity Cloud Service applications.	Running Oracle Identity Cloud Service Reports

User Administrator

A user administrator can manage users, groups, and memberships for an identity domain in Oracle Identity Cloud Service.

A user administrator can onboard users and groups, assign users and groups to applications, and run user reports. See Understanding Administrator Roles.

Task	Description	Additional Information
Onboard users and groups.	Onboard users and groups by: • Configuring and running bridges • Importing users and groups • Creating users and groups	Manage Microsoft Active Directory (AD) Bridges for Oracle Identity Cloud Service Managing Oracle Identity Cloud Service Users Managing Oracle Identity Cloud Service Groups
Assign users and groups to applications.	Assign users and groups to Oracle and custom applications.	
Run user reports.	Run operational or historical reports that capture data about Oracle Identity Cloud Service user accounts.	Running Oracle Identity Cloud Service Reports

User Manager

A user manager can manage all users or users of selected groups in Oracle Identity Cloud Service.

User managers update, activate, deactivate, remove, and unlock user accounts. User managers can also reset passwords, reset authentication factors, and generate bypass codes for user accounts. See Understand Administrator Roles.

Task	Description	Additional Information
Update user accounts.	Modify user accounts using the **Users** page.	Edit Attribute Values for the User Account
Activate and deactivate user accounts.	Activate and deactivate user accounts using the **Users** page.	Activate User AccountsDeactivate User Accounts
Unlock a user account.	Unlock user accounts using the **Users** page.	Unlock User Accounts
Reset passwords for user accounts.	Reset passwords for user accounts using the **Users** page.	Reset Passwords for User Accounts
Reset authentication factors for user accounts.	Reset authentication factors for user accounts using the **Users** page.	Reset Authentication Factors for User Accounts
Generate bypass codes for user accounts.	Generate bypass codes for user accounts using the **Users** page.	Generate Bypass Codes for User Accounts
Remove user accounts.	Remove user accounts using the **Users** page.	Remove User Accounts

Help Desk Administrator

A help desk administrator can manage all users or users of selected groups in Oracle Identity Cloud Service.

Help desk administrators can view the details of a user and unlock a user account. Help desk administrators can also reset passwords, reset authentication factors, and generate bypass codes for user accounts. See Understand Administrator Roles.

Task	Description	Additional Information
Unlock a user account.	Unlock user accounts using the **Users** page.	Unlock User Accounts
Reset passwords for user accounts.	Reset passwords for user accounts using the **Users** page.	Reset Passwords for User Accounts
Reset authentication factors for user accounts.	Reset authentication factors for user accounts using the **Users** page.	Reset Authentication Factors for User Accounts
Generate bypass codes for user accounts.	Generate bypass codes for user accounts using the **Users** page.	Generate Bypass Codes for User Accounts

Audit Administrator

An audit administrator can run reports for an identity domain in Oracle Identity Cloud Service.

See Understanding Administrator Roles.

Task	Description	Additional Information
Run user and application reports.	Run operational or historical reports that capture data about Oracle Identity Cloud Service applications or user accounts.	Running Oracle Identity Cloud Service Reports

Deprecated Oracle Identity Cloud Service Software Appliances

Learn about deprecated software appliances in Oracle Identity Cloud Service.

Service Change Announcement

App Gateway Replaces App Gate

The software appliance **App Gate** has been replaced with **App Gateway**. As of **August 2019**, App Gate has been replaced with App Gateway. Both the App Gate and the App Gateway solutions are software appliances that you can use to provide Single Sign-On (SSO) and authorization for your on-premises applications. This enables you to use one appliance to provide SSO for multiple applications by allowing external users to access internal applications securely without needing a VPN client. There's no change in functionality between the old App Gate and the new App Gateway solution. However, as a customer you will need to replace App Gate with App Gateway and reconfigure your supported applications. Technical support for App Gate will end after **August 15, 2021**. See Manage Oracle Identity Cloud Service App Gateways and see Download and Extract the App Gateway Binary File to download the App Gateway and to ensure that you are using the latest version of App Gateway.

2

Understand Application Integration

In this chapter, you'll learn what application integration is, why you should integrate your applications with Oracle Identity Cloud Service, the types of application integrations, and how you can use the App Catalog, Microsoft Active Directory (AD) Bridge, Provisioning Bridge, and SCIM interface to integrate Oracle Identity Cloud Service with your Software-as-a-Service (SaaS), Microsoft Active Directory, enterprise LDAP, and custom applications.

Topics:

Why Should You Integrate Your Applications?

Application integration reduces the time to develop new applications because you offload the business logic to secure applications to Oracle Identity Cloud Service. This logic includes securing your users, protecting the resources within the applications, and enabling users to access your applications through single sign-on (SSO). Integrating your applications with Oracle Identity Cloud Service provides the user with a seamless experience. Because of SSO, the user doesn't have to remember different IDs and passwords for each application. When your applications are integrated with Oracle Identity Cloud Service, your administrative overhead is reduced greatly because you can manage the policies and users for your applications from one central place. From a compliance perspective, Oracle Identity Cloud Service provides you with a single location where you can manage the access that your users have to your applications.

As part of application integration, Oracle Identity Cloud Service is commonly used as either an identity provider or a service provider for applications. An **identity provider**, known as an Identity Assertion provider, provides identifiers for users who want to interact with Oracle Identity Cloud Service using a website that's external to Oracle Identity Cloud Service. A **service provider** is a website that hosts applications. You can enable an identity provider and define one or more service providers. Your users can then access the applications hosted by the service providers directly from the identity provider.

For example, a website can allow users to log in to Oracle Identity Cloud Service with their Google credentials. Google acts as the identity provider and Oracle Identity Cloud Service functions as the service provider. Google verifies that the user is an authorized user and returns information to Oracle Identity Cloud Service (for example, the user name and the email address of the user, if the email address differs from the user name).

Some applications may require a user account to exist in their local identity store before the user can sign in to access these applications.

When users aren't created in Oracle Identity Cloud Service or imported into Oracle Identity Cloud Service from a flat file, they need to be synchronized from an authoritative source, such as an HR application or a corporate LDAP directory. For this scenario, the authoritative source and the application have to be integrated with Oracle Identity Cloud Service for provisioning and synchronization purposes.

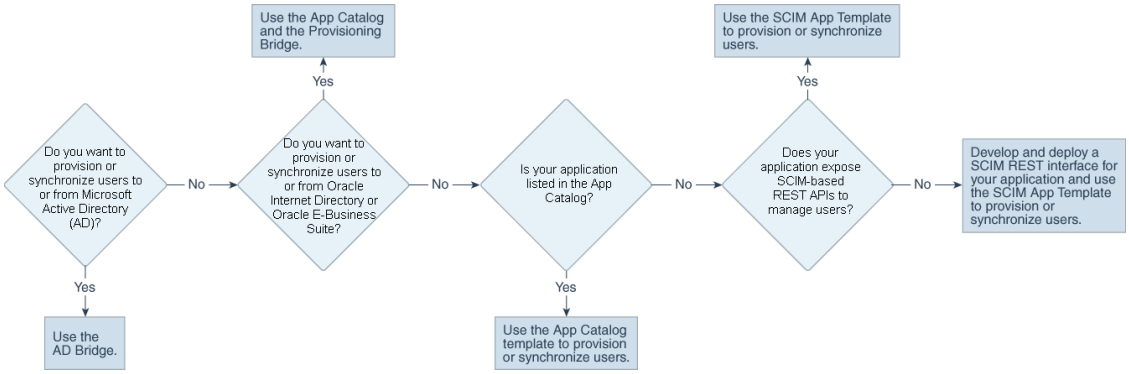
What Are the Types of Application Integrations?

This section provides you with the following information to help you understand the types of application integrations:

Topics:

Which Integration Method to Use?

Use the following flowchart to learn which method to use to integrate your application with Oracle Identity Cloud Service.



The following scenarios will help you understand this flowchart for synchronization and provisioning purposes:

Scenarios for User Synchronization

One of the following scenarios may apply when synchronizing users and groups from authoritative sources:

An HR Application as an Authoritative Source

When a company hires an employee, an HR representative adds that employee's information in the HR application directly. The HR application contains information about the user, such as the user's first name, last name, job role, and job location. This information is used to create an account for the user and assign applications to the user. For this scenario, you want to synchronize your user account into Oracle Identity Cloud Service from the HR application.

Oracle Identity Cloud Service supports integration with the HR application via the App Catalog. If your application isn't listed in the App Catalog, then you can build your own connector or use the Generic SCIM App Template. This template facilitates the configuration of your custom application when the SCIM APIs are exposed. If your application doesn't expose the SCIM APIs, then you can develop a custom SCIM gateway to act as an interface between Oracle Identity Cloud Service and your application.

A Corporate LDAP as an Authoritative Source

Some customers store users and groups into an LDAP, such as Microsoft Active Directory (AD) or Oracle Internet Directory. These users and groups can authenticate into Oracle Identity Cloud Service via SSO. For this to occur, first, the users and groups must be synchronized from the LDAP into Oracle Identity Cloud Service. To do this, use the Microsoft Active Directory Bridge (for AD) or the Provisioning Bridge (for Oracle Internet Directory).

Scenario for User Provisioning

Oracle Identity Cloud Service enables you to use app templates to provision users to applications. In the App Catalog, you'll find a list of app templates that support provisioning. These templates enable you to integrate these applications with Oracle Identity Cloud Service quickly. If your application isn't listed in the App Catalog, then use the Generic SCIM App Template.

Now that you know how to use the flowchart to select a method to integrate your application with Oracle Identity Cloud Service for provisioning and synchronization purposes, let's learn about each integration type in greater detail.

Integrate Oracle Identity Cloud Service with Applications from the App Catalog

This section provides answers to the following questions to help you understand how to use the App Catalog to integrate Oracle Identity Cloud Service with Software-as-a-Service (SaaS) applications:

Topics:

Why Integrate with SaaS Applications?

Over the past few years, customers are transitioning their access management system from an on-premises environment to a cloud-based one. This includes shifting their assets (such as their on-premises applications) into the cloud. Because of the proliferation of cloud-based SaaS applications in the market, Oracle Identity Cloud Service must be able to integrate with these applications. Oracle Identity Cloud Service has out-of-the-box integrations for thousands of SaaS applications. When a predefined integration isn't available for a SaaS application, Oracle Identity Cloud Service provides SAML and SCIM toolsets that will enable customers to integrate with it. By integrating your SaaS applications with Oracle Identity Cloud Service, you have one central place where you can not only manage your applications, but also the access that your users have to them.

What Is the App Catalog?

The App Catalog is a collection of partially configured application templates for thousands of SaaS applications, such as Amazon Web Services and Google Suite. Using the templates, you can define an application, configure SSO, and configure provisioning. Oracle creates and maintains the App Catalog for you, and provides step-by-step instructions that will help you to configure your applications.

What Are the Advantages of Using the App Catalog?

The App Catalog has out-of-the-box integrations for thousands of SaaS applications. When an application is available in the App Catalog, most of the metadata that Oracle Identity Cloud Service needs to integrate with the application already exists, so you don't have to define it. For most applications, it takes less than five minutes to configure them so that they can be integrated with Oracle Identity Cloud Service. All you have to do is go to the App Catalog, search for an application, create an instance of the application, and provide the connectivity details that Oracle Identity Cloud Service requires to communicate with it. When setting up applications, Oracle Identity Cloud Service features guided wizards that will help you configure them even further. This provides you with a consistent approach when using the App Catalog to integrate your applications with Oracle Identity Cloud Service.

Use Bridges to Integrate Oracle Identity Cloud Service with On-Premises Applications

This section provides answers to the following questions to help you understand how to use bridges to integrate Oracle Identity Cloud Service with on-premises applications, including Microsoft Active Directory (AD), an enterprise LDAP (such as Oracle Internet Directory), and a business application (such as Oracle E-Business Suite) that's used to manage and automate your business-related processes:

Topics:

Why Use Bridges to Integrate Oracle Identity Cloud Service with On-Premises Applications?

Most customers have Microsoft Active Directory (AD) as their central directory service. These customers also use AD as their network directory. This directory is where all of their workstations are connected to and where they manage their users.

In addition to AD, customers use

- An enterprise LDAP to centralize all of their user identities. So, a customer uses AD to manage their employees, but in the centralized LDAP, the customer manages their partners, consumers, and any other users with which the customer has relationships.
- Business applications to manage and automate processes across their enterprise. These processes include customer relationship management (CRM), enterprise resource planning (ERP), and supply chain management (SCM) processes.

For these reasons, it's imperative that Oracle Identity Cloud Service can integrate with AD, an enterprise LDAP (for example, Oracle Internet Directory), and an on-premises business application (such as Oracle E-Business Suite) to manage and automate the customer's CRM, ERP, SCM, and other business-related processes.

What Are the Types of On-Premises Application Integrations?

By using Oracle Identity Cloud Service, customers can control when they will migrate their directory-based applications to the cloud. In the interim, they can use one of the following:

- AD Bridge: This bridge provides a link between your AD enterprise directory structure and Oracle Identity Cloud Service. Oracle Identity Cloud Service can synchronize with this directory structure so that any new, updated, or deleted user or group records are transferred into Oracle Identity Cloud Service. Each minute, the bridge polls AD for any changes to these records and brings these changes into Oracle Identity Cloud Service. So, if a user is deleted in AD, then this change will be propagated into Oracle Identity Cloud Service. Because of this synchronization, the state of each record is synchronized between AD and Oracle Identity Cloud Service. After the user is synchronized from Microsoft Active Directory to Oracle Identity Cloud Service, if you activate or deactivate a user, modify the user's attribute values, or change the group memberships for the user in Oracle Identity Cloud Service, then these changes are propagated to Microsoft Active Directory through the AD Bridge. See Manage Microsoft Active Directory (AD) Bridges for Oracle Identity Cloud Service.
- Provisioning Bridge: This bridge provides a link between your enterprise LDAP or on-premises business application (such as Oracle Internet Directory or Oracle E-Business Suite) and Oracle Identity Cloud Service. Through synchronization, account data that's created and updated directly on the LDAP or business application is pulled into Oracle Identity Cloud Service and stored for the corresponding Oracle Identity Cloud Service users and groups. Any changes to these records will be transferred into Oracle Identity Cloud Service. Because of this, the state of each record is synchronized between the LDAP or business applicationand Oracle Identity Cloud Service.

  After users are synchronized from the on-premises business application to Oracle Identity Cloud Service, you can also use the Provisioning Bridge to provision users to the application. Provisioning allows you to use Oracle Identity Cloud Service to manage the lifecycle of users in the application. This includes creating, modifying, deactivating, activating, and removing users and their profiles across the application. Any changes that you make to users or their profiles in Oracle Identity Cloud Service are propagated to the business application through the Provisioning Bridge. See Manage Provisioning Bridges for Oracle Identity Cloud Service.

Use the SCIM Interface to Integrate Oracle Identity Cloud Service with Custom Applications

This section provides answers to the following questions to help you understand how to use the SCIM interface to integrate Oracle Identity Cloud Service with custom applications:

Topics:

Why Integrate with Custom Applications?

Let's say that you want to integrate your applications with Oracle Identity Cloud Service. Your applications are homegrown or aren't listed in the App Catalog, and an AD Bridge or Provisioning Bridge can't be used as a link between your applications and Oracle Identity Cloud Service.

A **custom application** is an application where the App Catalog or a bridge can't be used to integrate it with Oracle Identity Cloud Service. By integrating your custom applications with Oracle Identity Cloud Service, from one centralized cloud service, you can provide SSO capabilities for your applications, and provision and synchronize your users between the applications and Oracle Identity Cloud Service.

The App Catalog has thousands of applications that integrate with Oracle Identity Cloud Service. For most applications, Oracle Identity Cloud Service provides single sign-on (SSO) and user provisioning capabilities. User provisioning is not only giving users initial access to these applications, but also managing the complete lifecycle of the relationship that the users have with the applications.

What Is SCIM?

In the past, it was common that applications used to have their own user management APIs. Because the APIs for each application behave in a certain way, the developer had to understand the APIs specific to each application to build integrations for the applications.

To integrate your custom applications with Oracle Identity Cloud Service, Oracle recommends that you use the System for Cross-domain Identity Management (SCIM). SCIM provides developers with an abstraction layer. If APIs for the applications are exposed through SCIM, then developers don't have to learn the APIs associated with each application because the JSON format of the APIs is common across all applications.

In addition to SCIM being an open specification that standardizes user and group management across applications, it allows for the automation of user and group provisioning. You can provision and synchronize data for your users and groups across multiple applications.

With SCIM, you can define HTTP endpoints to create, read, update, and delete resources for entities such as users and groups. You can also use SCIM to extend the schemas for your company's users and groups. The SCIM specification defines a minimum set of attributes for the user schema, but this schema can be extended.

For example, suppose you need to provision the `Employee ID` custom attribute from the Oracle Identity Cloud Service user schema to your custom application. You can extend the default user schema, add this attribute, and map it between Oracle Identity Cloud Service and your application. The user schema in Oracle Identity Cloud Service can now adhere to the attributes associated with your custom application's identity store.

The SCIM specification also defines security for any request that you make using HTTP endpoints. Security is defined by using a secure (HTTPS) protocol to establish communication between the endpoints and the applications with which you're integrating, and requiring an authorization token that's used to access the request and perform the operations associated with it.

Use the following table to learn more about the SCIM specification:

Item	URL
Core schema	https://tools.ietf.org/html/rfc7643
Protocol	https://tools.ietf.org/html/rfc7644
Definitions, overview, concepts, and requirements	https://tools.ietf.org/html/rfc7642

Why Use SCIM?

If you look at how integrations used to be built, developers had to understand the APIs exposed for each application. There was no consistency regarding how to represent an identity in these applications.

By using SCIM, there's now a common standard of how you represent an identity in every application. Because all applications comply with the SCIM format, there's a harmonious flow in terms of how these identities are represented. This makes it easier for an identity management cloud service such as Oracle Identity Cloud Service to integrate with these applications.

Having a common standard for representing identities in applications improves developers' work efficiency and productivity because developers don't have to spend time to learn the APIs for each application. From a corporate standpoint, the time it takes to develop an integration from an identity system to the application will be reduced significantly. You can now run automations for the integration because there's a standard in terms of how you represent an identity and how you integrate with that identity.

By exposing your custom application's identity store with a SCIM-based interface, you avoid having to develop a custom connector between your application and Oracle Identity Cloud Service. This can be time-consuming, costly, and can lead to heavy maintenance in a future upgrade.

SCIM automates the user identity lifecycle management process and increases the security of data associated with your company’s users and groups.

As your company grows, your users and groups increase. Through the day-to-day operations of your company, you may experience situations such as employee turnover or the memberships that your users have with your company’s groups may change. Your company’s user accounts, groups, and group memberships increase significantly.

Because SCIM is a standard, your company’s user and group data is stored in a consistent way and can be communicated as such across different apps (including your custom apps). You can automate the provisioning and deprovisioning process and have Oracle Identity Cloud Service function as a single point to manage permissions and group memberships. By transferring your company’s user and group data automatically, you mitigate the risk of inadvertent errors.

By implementing SCIM, you improve your company’s security. Through SSO, your company's employees no longer have to sign on to each of their accounts individually. You can ensure security policy compliance for your users and their access to your company's applications.

When your employees are terminated or leave your company, you want your company's offboarding process to be consistent. This way, there's no chance that your company's administrators will forget to deprovision user accounts for applications that contain sensitive data. With SCIM, when users depart from your company, your administrators can terminate the accounts in Oracle Identity Cloud Service, and have peace of mind because these accounts will also be suspended or deleted in your SCIM-enabled apps.

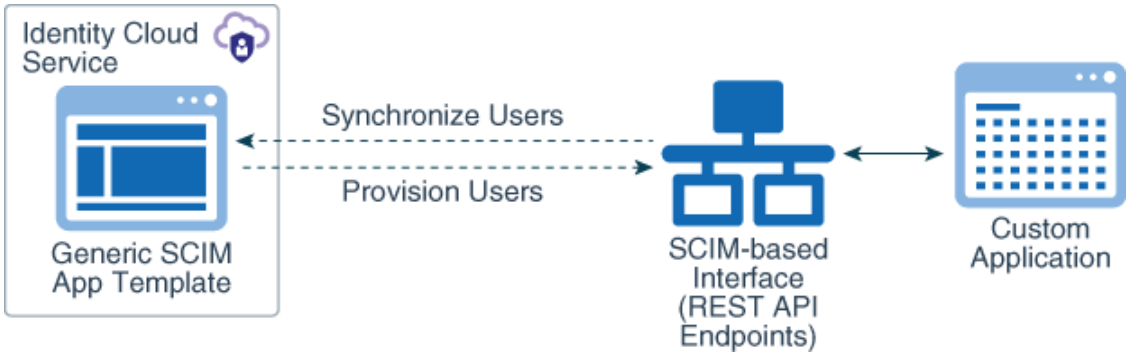
How Do You Use the Generic SCIM App Template?

In the App Catalog, there are thousands of applications that integrate with Oracle Identity Cloud Service. You may have your applications running on your premises or in the cloud, or you may be building your applications in different infrastructure systems such as Amazon Web Services or Oracle Cloud Infrastructure.

Oracle Identity Cloud Service has to provide not only integrations with the applications that are listed in the App Catalog, but also tools so that you can build integrations for your custom applications without developing code.

With the Generic SCIM App Template, you can configure your custom applications so that the SCIM APIs are exposed, and you don't have to develop a single line of code. All that's required is to go to the App Catalog and search for a SCIM-managed app template. To use this template, you only have to provide your endpoint URL and the details that Oracle Identity Cloud Service requires to connect to your application, and then map the attributes between your application and Oracle Identity Cloud Service.

With the Generic SCIM App Template, you can provision or synchronize users between your custom applications and Oracle Identity Cloud Service.



In this diagram, the Generic SCIM App Template has been configured to enable Oracle Identity Cloud Service to communicate with a custom application that has a SCIM-based interface. This interface uses REST API endpoints to provision and synchronize users between Oracle Identity Cloud Service and the custom application.

Before You Begin

Before you begin to use the Generic SCIM App Template:

- Get access to an instance of Oracle Identity Cloud Service.
- Make sure that you have the appropriate permissions to register applications and to manage security components in the Identity Cloud Service console.
- Ensure that there's HTTP(S) communication between Oracle Identity Cloud Service and the SCIM-based interface for your custom application.
- Make sure that you have a basic knowledge of the SCIM specification.

Assign Administrator Roles to Your User Account

Although you have access to an instance of Oracle Identity Cloud Service, you must be assigned to the following administrator roles in Oracle Identity Cloud Service to use the Generic SCIM App Template in the Identity Cloud Service console:

- Security administrator: To configure the security aspects of your Oracle Identity Cloud Service instance.
- Application administrator: To add a custom application and use the Generic SCIM App Template.

To assign these administrator roles to your user account:

1. Sign in to Oracle Identity Cloud Service with the credentials of your user account.

2. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Administrators**.
3. Expand the node for the **Security Administrator** role.
4. Click **Add**, select the check box for your user account, and then click **OK**.
5. Repeat steps 3 and 4 to assign the application administrator role to your user account.

Does Your Custom Application Have a SCIM-based Interface?

In this section, you learn what to do, based on whether your custom application has a SCIM-based interface.

If your custom application has this interface, then you can configure the Generic SCIM App Template to provision Oracle Identity Cloud Service users with your application. See Configure the Generic SCIM App Template.

If your custom application doesn't have this interface, then you can develop a custom SCIM gateway to act as the interface between Oracle Identity Cloud Service and your custom application. See How Do You Develop a Custom SCIM Gateway?

Configure the Generic SCIM App Template

In this section, you add an application using the Generic SCIM App Template, enable and configure connectivity for provisioning for your application, configure the application's attribute mappings for provisioning, select the provisioning operations for your application, enable and configure synchronization for your application, and test your application to verify that users are provisioned to it.

Add an Application Using the Generic SCIM App Template

In this section, you use the Generic SCIM App Template to add an application.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Applications**, click **Add**, and then select **App Catalog**.
2. In the **Type of Integration** section, click **Provisioning**, chose one of the following templates, and then click **Add**:
   - **GenericScim - Basic:** A Generic SCIM Template for SCIM interfaces that support basic authentication.
   - **GenericScim - Bearer Token:** A Generic SCIM Template for SCIM interfaces that support JWT tokens submitted as an authorization bearer.
   - **GenericScim - Client Credentials:** A Generic SCIM Template for SCIM interfaces that support client credentials for authentication.
   - **GenericScim - Resource Owner Password:** A Generic SCIM Template for SCIM interfaces that support the resource owner grant type.
3. In the **Details** pane of the corresponding template page, provide a name and description for your application, and then click **Next**.
4. In the **Provisioning** pane, click **Finish**. An instance of your application is created with five tabs: **Details**, **Provisioning**, **Import**, **Users**, and **Groups**.

   In the next section, you'll use the **Provisioning** tab to enable and configure connectivity for provisioning for your application.

Enable and Configure Connectivity for Provisioning for Your Application

In this section, you enable provisioning for your application and provide connectivity information for it. Oracle Identity Cloud Service uses this information to connect to your application's SCIM REST API endpoint for provisioning purposes.

1. Click the **Provisioning** tab.
2. Turn on the **Enable Provisioning** switch.
3. In the **Confirmation** window, click **OK**.
4. Use the following table to populate the fields of the **Configure Connectivity** section of the **Provisioning** tab.

Parameter	Description and Value Information	Additional Information
Host Name	The host name of your application's SCIM REST API endpoints. If the SCIM interface's URL is `https://api.example.com/scimgate/Users`, then the host name is `api.example.com`.	This parameter appears in the UI for all Generic SCIM App Templates.
Base URI	The base relative URL of your application's SCIM REST API. For example, if the SCIM interface's URL is `https://api.example.com/scimgate/Users`, then the Base URI is `/scimgate`.	This parameter appears in the UI for all Generic SCIM App Templates.
Administrator Username	The administrator's user name for your API authentication service. This value is sent as part of the body message of each request to your application's SCIM REST API. Format: Plain text.	This parameter appears in the UI for the **GenericScim - Basic** and **GenericScim - Resource Owner Password** templates.
Administrator Password	The administrator's password for your API authentication service. This value is sent as part of the body message of each request to your application's SCIM REST API. Format: Plain text.	This parameter appears in the UI for the **GenericScim - Basic** and **GenericScim - Resource Owner Password** templates.
HTTP Operation Types	By default, the template request uses the `PATCH HTTP` operation for any user's update operation. If your SCIM interface uses the `PUT HTTP` operation for user attribute updates, then use this field as per the example below. Example: `__ACCOUNT__.Update=PUT`	This parameter appears in the UI for all Generic SCIM App Templates.

Parameter	Description and Value Information	Additional Information
Access Token	The value of the access token to be used by the template when communicating with your application's SCIM REST API. Format: Plain text.	This parameter appears in the UI for the **GenericScim - Bearer Token** template.
Client Id	The client ID for your API authentication service. Format: Plain text.	This parameter appears in the UI for the **GenericScim - Client Credentials** and **GenericScim - Resource Owner Password** templates.
Client Secret	The client secret for your API authentication service. Format: Plain text.	This parameter appears in the UI for the **GenericScim - Client Credentials** and **GenericScim - Resource Owner Password** templates.
Scope	The scope for your application. Example: `https://www.example.com/auth/adm.direct.group https://www.example.com/auth/admin.direct.user`	This parameter appears in the UI for the **GenericScim - Client Credentials** and **GenericScim - Resource Owner Password** templates.
Authentication Server Url	The URL of your authentication service. Example: `https://api.example.com/oauth2/v1/token`	This parameter appears in the UI for the **GenericScim - Client Credentials** and **GenericScim - Resource Owner Password** templates.
Custom Authentication Headers	Used to send additional static header values to your API authentication service. Example: `Basic authorization base64encodedusernamepassword`	This parameter appears in the UI for the **GenericScim - Client Credentials** and **GenericScim - Resource Owner Password** templates.
Connection Timeout	How long (in milliseconds) Oracle Identity Cloud Service will wait to establish communication with your application's SCIM REST API. Format: Plain text.	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.
Socket Timeout	How long (in milliseconds) Oracle Identity Cloud Service will wait to receive data from your application's SCIM REST API. Format: Plain text.	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.

Parameter	Description and Value Information	Additional Information
Port Number	The port number of your application's SCIM REST API endpoints. The default port number for this parameter is `443`. For example, if the URL is `https//api.example.com:6355/scimgate/Users`, then set the port number value to `6355`.	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.
SSL Enabled	If your application's SCIM REST API endpoints don't require SSL, then set the value of this parameter to `false`. Default value: `true`	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.
JSON Resource Tag	The name of the attribute used in JSON messages when your application's SCIM REST API returns multiple resources. The default value is `Resources`. For example, if the response message of the user for the `GET` operation is: `{ "users": [ {user1}, {user2} ] }` then change the value to `users`.	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.
UID Attributes	The mapping between the `__UID__` (guid) internal attribute and your application's SCIM attribute for user and group object classes. Default value: `["Users=id", "Groups=id"]`	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.
Name Attributes	The mapping between the `__NAME__` internal attribute and your application's SCIM attribute for user and group object classes. Default value: `["Users=userName", "Groups=displayName"]`	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.
Status Attributes	The mapping between the `__ENABLE__` (status) internal attribute and your application's SCIM attribute for the user object class. Default value: `Users=active`	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.

Parameter	Description and Value Information	Additional Information
Password Attributes	Your application's SCIM REST API attribute that corresponds to the user's password. This is used for masking the password attribute in the log files. Default value: `Users=password`	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.
Date Attributes	The list of date attributes available for your application's SCIM REST API. Example: `Users=meta.lastModified,joiningDate`	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.
Date Format	The date-and-time format of the date attributes available for your application's SCIM REST API. Example: `MMM d, yyyy h:mm:ss a z`	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.
Content Type	The content-type header that your application's SCIM REST API expects Oracle Identity Cloud Service to send as a header HTTP request. Default value: `application/scim+json`	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.
Accept Type	The content-type header that is expected as an HTTP response from your application's SCIM REST API. Default value: `application/scim+json`	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.
Custom Headers	Used to send additional static header values to the SCIM REST API endpoints of your application. Format: `<headerName1>=<value>,<headerName2>=<value>`	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.
SCIM Version	The version of your application's SCIM REST API. Default value:`13`. The range for this attribute varies from `1` to `19`.	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.
OClass Mapping	Used to map an attribute of one object class to an attribute of another object class. For example, if the **groups** attribute of the `__ACCOUNT__` object class must be mapped to the `__GROUP__` object class, then enter `__ACCOUNT__.groups=__GROUP__`.	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.

Parameter	Description and Value Information	Additional Information
Default Batch Size	The default page or batch size for the `GET` operation. Default value: `200`	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.
Managed Object Classes	The classification type of the schemas that your application must manage. Default value: `["Users", "Groups"]`	You can't update the value for this parameter in the UI. To update this parameter value, use Oracle Identity Cloud Service's REST APIs.

5. After populating the fields accordingly, click **Test Connectivity** to verify whether Oracle Identity Cloud Service can communicate with your application's SCIM REST API endpoints.

   If a successful connection can be established, then a **Connection successful.** message appears.

   If you receive an error message, check the values that you provided, and then click **Test Connectivity** again. If the problem persists, then contact your system administrator.

   Before testing, you can save the application and use Oracle Identity Cloud Service REST APIs to update the parameter values that don't appear in the UI. After updating the parameter values, open the application again using the Identity Cloud Service console, and then click **Test Connectivity**.

Configure Attribute Mappings for Provisioning

You can change the default attributes mapped between Oracle Identity Cloud Service and your application's SCIM REST API.

1. In the **Provisioning** tab of your application page, scroll to the **Configure Attribute Mapping** section, and then click **Attribute Mapping**. The **Attribute Mapping** window appears, and shows the default attribute mappings.
2. In the **Attribute Mapping** window, add, modify, or remove attribute mappings, according to your application's SCIM user schema.
3. Click **OK**.

Select Provisioning Operations

You can select which provisioning operations are supported by your application's SCIM REST APIs.

1. In the **Provisioning** tab of your application page, scroll to the **Select Provisioning Operations** section.
2. Select the following operations:
   - Authoritative Sync: If you enable this operation, then your application will become an authoritative source for Oracle Identity Cloud Service. Users in the application will be synchronized into Oracle Identity Cloud Service. If you select this operation, then the other provisioning operations will be deactivated.
   - Create Account: If a user is assigned to your application in Oracle Identity Cloud Service, then an account will be created for the user in the application.

- Update Account: If an administrator edits the values of the provisioning form for a user who is assigned to your application, then these changes will be propagated to the application.
- De-activate Account: If an administrator activates or deactivates a user who is assigned to your application, then this change will be propagated to the application.
- Delete Account: If a user is removed from your application in Oracle Identity Cloud Service, then the user's account will be removed from the application.

Enable and Configure Synchronization for Your Application

By enabling and configuring synchronization for your application, Oracle Identity Cloud Service can synchronize user accounts from your application and match these accounts to corresponding Oracle Identity Cloud Service users.

1. Turn on the **Enable Synchronization** switch.
2. If you want Oracle Identity Cloud Service to communicate with your application's SCIM REST API to synchronize groups from the application into Oracle Identity Cloud Service as application roles, then click **Refresh Application Data**. Otherwise, go to the next step.
3. Use the following table to populate the fields of the **Configure Synchronization** section of the **Provisioning** tab.

Parameter	Description and Value Information
User Identifier	The Oracle Identity Cloud Service user attribute that matches user accounts synchronized from the application with users in Oracle Identity Cloud Service.
Application Identifier	The user attribute of your application's SCIM REST API that matches user accounts synchronized from the application with users in Oracle Identity Cloud Service.
When exact match is found	Use the values in this menu to control whether Oracle Identity Cloud Service or an application administrator confirms any user accounts that are synchronized from the application into Oracle Identity Cloud Service. Values are: • **Link and confirm**: Link the synchronized user accounts to corresponding Oracle Identity Cloud Service users automatically. • **Link but do no confirm**: Link the synchronized user accounts to corresponding Oracle Identity Cloud Service users, but application administrators need to confirm the linking operation in the **Import** tab manually.
Max. number of creates	The value that you enter in this field represents the maximum number of user accounts that are created in Oracle Identity Cloud Service from the application. If you don't want to limit how many user accounts are created, then leave this field blank.

Parameter	Description and Value Information
Max. number of deletes	The value that you enter in this field represents the maximum number of user accounts that are deleted in Oracle Identity Cloud Service after these accounts are deleted from the application. If you don't want to limit how many user accounts are deleted, then leave this field blank.
Synchronization schedule	Specify how often (in hours, days, or weeks) synchronization happens between the application and Oracle Identity Cloud Service automatically. If you want to synchronize the user accounts manually, then select **Never**.

4. Click **Finish**.

Test the Provisioning Operations You Selected

After you use the Generic SCIM App Template to configure your application, you must test the provisioning operations you selected to verify that they are operable.

1. In the **Applications** page, select and activate your application, and then click it to open the **Details** tab.
2. Click the **Users** tab, and then click **Assign**.
3. In the **Assign Users** window, choose a user, and then click **Assign**.
4. In the **Assign Application** window, populate any form fields needed to provision a user account to your application, and then click **Save**. Oracle Identity Cloud Service starts the provisioning operation to create a user account in your application.
5. Verify that the user account has been created in your application.
6. In the **Users** tab, deactivate the user, activate the user again, and remove the user from your application. Each change you make is reflected in the user account for your application.
7. Click the **Import** tab, and then click **Import**.

   Oracle Identity Cloud Service communicates with your application's SCIM REST API to get a list of all user accounts. Oracle Identity Cloud Service tries to match each user account with an existing user in Oracle Identity Cloud Service. If a user exists, then the user is assigned to your application. If the user doesn't exist, then you can perform one of the following actions manually:

   - **Assign Existing User:** Assign the user account to any user in Oracle Identity Cloud Service.
   - **Create New User and Link:** Add a new user to Oracle Identity Cloud Service, and then assign the user account to this newly created user.

How Do You Develop a Custom SCIM Gateway?

If your custom application doesn't provide a SCIM-based interface, then you can develop a custom SCIM gateway to act as the interface between Oracle Identity Cloud Service and your custom application. This gateway exposes your application's identity store as SCIM-based REST APIs, and then you can use the Generic SCIM App Template to integrate Oracle Identity Cloud Service with your application for provisioning or synchronization purposes.

Before developing your custom SCIM gateway, if you're a new developer who isn't familiar with the SCIM standard, then you must first understand the SCIM protocol. Then, see which identity

attributes are available for your custom application and model them as SCIM-based attributes. Next, utilize open-standard libraries to expose your custom application's APIs as SCIM APIs. Last, familiarize yourself with the create, read, update, and delete (CRUD) operations that you want your custom SCIM gateway to perform.

Supported Operations

`User` is a type of resource in the SCIM specification. To manage this resource, the SCIM gateway must expose REST API endpoints to enable operations such as creating, searching for, updating, and deleting users. The HTTP request for the operation that you want to perform and the HTTP response from that operation must be in a `JSON` format.

You can implement the following user operations:

User Operation	Description	HTTP Operation	HTTP Endpoint
Create a User	Create a user account in your custom application.	`POST`	`https://app.example.com/scimgate/Users`
Search Users	Obtain a list of all users with their attributes that are in your custom application.	`GET`	`https://app.example.com/scimgate/Users`
Search a User	Retrieve information about a specific user and their attributes in your custom application.	`GET`	`https://app.example.com/scimgate/Users/<id>`
Update a User Attribute	Update an attribute value of a user account in your custom application.	`PUT`	`https://app.example.com/scimgate/Users/<id>`
Delete a User	Remove a user account from your custom application.	`DELETE`	`https://app.example.com/scimgate/Users/<id>`

How Do You Secure the Custom SCIM Gateway?

Because you don't want unauthorized users or clients to access your custom SCIM gateway, you must secure it. To do this, use an authorization token to protect the HTTP(S) endpoints of your gateway. This token will validate the user or client to allow them to make appropriate HTTP calls to the gateway endpoints. If the token isn't present or is invalid, then the endpoints will return a `401 HTTP` response code because Oracle Identity Cloud Service isn't authorized to access the endpoints.

Oracle Identity Cloud Service uses the administrator's user name and password, which are configured when you register your custom application, to request an access token from the custom SCIM gateway. Oracle Identity Cloud Service can then use this token to access the gateway endpoints as an authorization bearer header.

Sample Implementation of a Custom SCIM Gateway

Oracle provides a sample `Node.js` application that conforms to SCIM specifications, and which you can use to develop a custom SCIM gateway to integrate it with your custom application.

This custom gateway exposes HTTP endpoints to enable operations such as creating, searching for, updating, and deleting users. The custom gateway stores information about the users locally in the `db.json` file. This file has the `JSON` format.

The sample application uses express and body-parser packages. The `server.js` file implements a route for users' endpoints:

```
"...
var express = require('express')
var app = express()
var bodyParser = require('body-parser');
app.use(bodyParser.json());
var config = require('./config.js');
..."
```

The `routes/users.js` file defines the SCIM REST API endpoints, and maps each endpoint to the corresponding JavaScript function:

```
"...
//Get operation for /Users endpoint
app.get('/scimgate/Users', users.findAll);

//Get operation for /Users/:id endpoint
app.get('/scimgate/Users/:id', users.findOne);

//Put operation for /Users endpoint
app.post('/scimgate/Users', users.create);

//Put operation for /Users endpoint
app.put('/scimgate/Users/:id', users.update);

//Delete operation for /Users endpoint
app.delete('/scimgate/Users/:id', users.delete);
..."
```

The `user.controller.js` file implements JavaScript functions to create, read, update, and delete users in the local user store, represented by the `userdb.json` file:

```
"...
exports.findAll = function(req, res){
console.log('Entering findAll function.');
...
};

exports.findOne = function(req, res) {
console.log('Entering findOne function.');
...
};
```

```
exports.create = function(req, res){ console.log('Entering create function.');
...
};

exports.update = function(req, res){
console.log('Entering update function.');
...
};

exports.delete = function(req, res){ console.log('Entering delete function.');
...
};

..."
```

The `userdb.json` file contains an array of users, and the structure of each user entry follows the SCIM specification standard, using a subset of the user attributes:

```
{
  "resources": [
    {
      "schemas": [
        "urn:ietf:params:scim:schemas:core:2.0:User"
      ],
      "id": "1",
      "externalId": "1",
      "userName": "user1@example.com",
      "name": {
        "formatted": "User 1 Name",
        "familyName": "Name",
        "givenName": "User 1"
      },
      "displayName": "User 1 DisplayName",
      "active": true,
      "password": "User1Password",
      "emails": [
        {
          "value": "user1@example.com",
          "type": "work",
          "primary": true
        }
      ]
    }
  ]
}
```

To authorize the client to make HTTP requests, the sample SCIM gateway application makes use of two environment variables that you must set before running the application: `ADMINUSER` and `ADMINPASS`. These variables represent the administrator's user name and password for your API authentication service. You provide values for these variables by setting up the `run.sh` shell script for Unix or Mac environments, or the `run.bat` batch script for Windows environments.

Oracle Identity Cloud Service sends these administrative credentials in the form of an authorization header for all requests to authenticate the administrator's credentials, and then accesses the custom SCIM gateway using the `basic` grant type.

You can modify the sample application's source code and implement other types of authentication methods to match your requirements.

You can also change the sample application's source code so that instead of contacting the local user store (represented by the `userdb.json` file), the new sample application contacts your application's identity store to create, read, update, and delete users.

Configure and Run the Custom SCIM Gateway Sample Application

In this section, you configure and run the custom SCIM gateway sample application to work with the **GenericScim - Basic** template.

1. Edit the `run` script file in the `root` folder of the sample SCIM gateway application, update the `ADMINUSER` and `ADMINPASS` values, and then save the file.

   If you're running the sample application in a Unix or Mac environment, then use the `run.sh` script. If you're using Windows, then use `run.bat`.

2. Open a command prompt or terminal, navigate to the `root` folder of the sample application, execute the `run` script by typing the name of the file, and then press **Enter** to start the sample application. You'll see log information that will help you to understand what the sample application is doing.

   Make sure the hostname of this sample application is reachable through the Internet so that Oracle Identity Cloud Service can contact the application.

Register the Custom SCIM Gateway Application

In this section, you register the custom SCIM gateway sample application with Oracle Identity Cloud Service.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Applications**, click **Add**, and then select **App Catalog**.
2. In the **Type of Integration** section, click **Provisioning**, locate the **GenericScim - Basic** template, and then click **Add**.
3. In the **Details** pane of the **GenericScim - Basic** page, enter `SCIM Gateway Application` for both the name and description of your application, and then click **Next**.
4. In the **Provisioning** pane, turn on the **Enable Provisioning** switch.
5. In the **Confirmation** window, click **OK**.
6. Use the following table to populate the fields of the **Configure Connectivity** section of the **Provisioning** tab.

Parameter	Value
Host Name	Enter the host name of your application.
Base URI	`/scimgate`
Administrator Username	`admin`
Administrator Password	Enter the administrator's password you have set in the run script of the sample application.
HTTP Operation Types	`__ACCOUNT__.Update=PUT`

For more information about the fields of the **Configure Connectivity** section, see the table in Enable and Configure Connectivity for Provisioning for Your Application.

7. Click **Finish** to save the application.

If you deploy and run the sample application in a non-HTTPS server or a server which doesn't contain a valid certificate, then you may need to use Oracle Identity Cloud Service's REST API to change the `SSLEnabled` parameter to `false`. If the server doesn't listen to the default HTTP(s) port numner, then change the `Port` parameter to the corresponding port number your application runs. After you update these parameters you can test connectivity between the application and Oracle Identity Cloud Service, and then activate the application.

Use REST APIs to Update the Custom SCIM Gateway Application

In this section, you use Oracle Identity Cloud Service REST APIs to update the `port`, and `sslEnabled` parameters of the custom SCIM gateway application.

1. Use a client credential application in Oracle Identity Cloud Service to acquire an access token. If a client credential application hasn't been created in your environment, then add one.
2. Use the access token as an authorization bearer to execute a `GET` request to the following endpoint: `https://yourtenant.identity.oraclecloud.com/admin/v1/Apps?filter=displayName co "SCIM Gateway Application"`

   The JSON response contains an ID value for this application.
3. Use the ID value and the access token from the previous steps to execute a `PATCH` request to the following endpoint: `https://yourtenant.identity.oraclecloud.com/admin/v1/Apps/"ID"`

   Replace the ID value with the ID value of your application, set the **Content-type** header to `application/json`, and provide the following content for the body:

```
{
  "schemas": [
    "urn:ietf:params:scim:api:messages:2.0:PatchOp"
  ],
  "Operations": [
     {
      "op": "replace",
      "path":
"urn:ietf:params:scim:schemas:oracle:idcs:extension:managedapp:App:bundleCo
nfigurationProperties[name eq \"sslEnabled"].value",
      "value": [ "false"]
    },
     {
      "op": "replace",
      "path":
"urn:ietf:params:scim:schemas:oracle:idcs:extension:managedapp:App:bundleCo
nfigurationProperties[name eq \"port\"].value",
      "value": [ "6355"]
    }
  ]
}
```

4. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Applications**, and then select **SCIM Gateway Application**.

5. In the **Provisioning** pane, click **Test Connectivity** to verify that a connection can be established between Oracle Identity Cloud Service and your custom SCIM gateway application.
6. Click **Finish**, and then click **Activate** to activate the application.

Test Your Custom SCIM Gateway Sample Application

Test your custom SCIM gateway sample application by provisioning Oracle Identity Cloud Service users with it.

1. In the **Applications** page, select your application, and then click it to open the **Users** tab.
2. In the **Users** tab, click **Assign**.
3. In the **Assign Users** window, choose a user, and then click **Assign**.
4. In the **Assign Application** window, populate the **Username**, **Full Name**, **Family Name**, **Given Name**, **Display Name**, and **Primary Email** form fields with values, and then click **Save**.
5. In the **Assign Users** window, click **OK**.

   Oracle Identity Cloud Service creates a user account in the `userdb.json` file of your application.
6. Open the `userdb.json` file and verify that a user account has been created. Then, close the file.
7. In the **Users** tab, click the **Action** menu  to the right of the user, and then select **Deactivate**.
8. After one minute, open the `userdb.json` file and verify that the corresponding user account has a `false` value for the `active` attribute. Then, close the file.
9. In the **Users** tab, click the **Action** menu to the right of the user, and then select **Activate**.
10. After one minute, open the `userdb.json` file and verify that the corresponding user account has a `true` value for the `active` attribute. Then, close the file.
11. In the **Users** tab, select the user, and then click **Revoke**.
12. In the **Confirmation** window, click **OK**.
13. After the **Confirmation** window closes, open the `userdb.json` file and verify that the corresponding user account has been removed. Then, close the file.

Part II
Perform Identity Administration

Learn how to perform important administrative functions that you must do right away, and others that you will return to later.

Chapters

3

Manage Oracle Identity Cloud Service Users

This section describes how to manage Oracle Identity Cloud Service users. This includes but not limited to creating user accounts, assigning groups to user accounts, importing user accounts, and multiple factor authentication for user accounts.

Topics:

Typical Workflow for Managing Oracle Identity Cloud Service Users

With the user management feature in Oracle Identity Cloud Service, you can perform tasks such as creating, managing, and removing user accounts.

Task	Description	Additional Information
Understand the user life cycle.	You can learn about the process flow of how a user account is created, managed, and deleted in Oracle Identity Cloud Service.	Understand the User Life Cycle
Understand administrator roles.	You can learn about administrator roles that you can assign to Oracle Identity Cloud Service users.	Understand Administrator Roles
Create user accounts.	You can create user accounts using the **Users** page.	Create User Accounts
View details about user accounts.	You can view details about user accounts using the **Users** page.	View Details About User Accounts
Modify user accounts.	You can modify user accounts using the **Users** page.	Edit Attribute Values for the User Account Assign Groups to the User Account Remove Groups from the User Account Assign Applications to the User Account Remove Applications from the User Account
Activate and deactivate user accounts.	You can activate and deactivate user accounts using the **Users** page.	Activate User Accounts Deactivate User Accounts
Import and export user accounts.	You can import and export user accounts using the **Users** page.	Import User Accounts Export User Accounts
Generate bypass codes for user accounts.	You can generate bypass codes for user accounts using the **Users** page.	Generate Bypass Codes for User Accounts
Reset authentication factors for user accounts.	You can reset authentication factors for user accounts using the **Users** page.	Reset Authentication Factors for User Accounts
Unlock user accounts.	You can unlock user accounts using the **Users** page.	Unlock User Accounts
Delegate administrative responsibilities for user accounts.	You can delegate administrative responsibilities to user accounts using the **Administrators** page.	Add or Remove a User Account from an Administrator Role
Generate personal access tokens.	Generate tokens that client applications can use to access an API or a resource application within a limited period of time.	Generate Personal Access Tokens
Send invitations to users to activate their accounts.	You can send invitations to users to activate their user accounts using the **Users** page.	Send Invitations to Users to Activate Their Accounts
Reset passwords for user accounts.	You can reset passwords for user accounts using the **Users** page.	Reset Passwords for User Accounts
Remove user accounts.	You can remove user accounts using the **Users** page.	Remove User Accounts

You can create, manage, and remove user accounts by:

- The Identity Cloud Service console

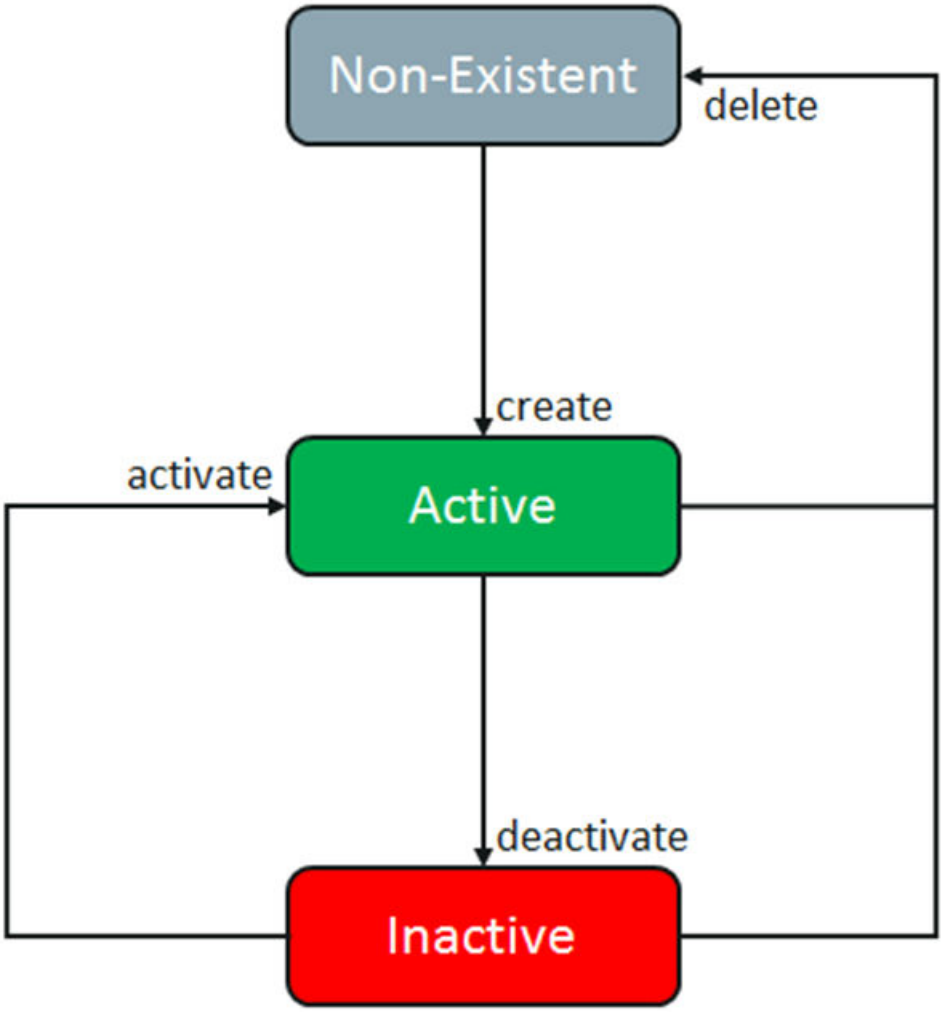
- SCIM-based APIs

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Understand the User Life Cycle

User life cycle is a term to describe the process flow of how a user account is created, managed, and deleted in Oracle Identity Cloud Service based on certain events or time factors.

A user account goes through various stages in the life cycle. The stages are non-existent, deactivated, activated, and deleted.



You can define business requirements for each transition of the user life cycle. Use the sample scenarios listed in the following table to establish the link between user life cycle transitions and business objectives.

Current State	Operation	Sample Scenario	Process Description
Non-existent	Create	Human resources (HR) enters user profile information for a new hire.	If the new hire's start date isn't a future date, then the user account is introduced into Oracle Identity Cloud Service with an **Activated** status. If the new hire's start date is a future date, then the user account is created in Oracle Identity Cloud Service, and is then deactivated.
Deactivated	Activate	The user's start date is in effect.	The user account is activated in Oracle Identity Cloud Service, and the user can now log in and use this Oracle Cloud service. The user can access all groups, applications, and administration role privileges assigned to the user account.
Activated	Modify	The user is promoted to a new position. HR changes the job title of the user.	New groups, applications, and administration roles are assigned to the user account. Old irrelevant groups, applications, and administration roles are removed from the user account.
Activated	Deactivate	The user takes a one-year sabbatical from the company. HR manually deactivates the user account on the last working day of the user. The user rejoins the company after some period. HR activates the user account.	The user account is deactivated in Oracle Identity Cloud Service, and the user can no longer log in and use this Oracle Cloud service. The user account can be activated again.

Current State	Operation	Sample Scenario	Process Description
Activated	Delete	The user retires from the company. HR manually deletes the user account on the last working day of the user.	The user account is removed from Oracle Identity Cloud Service. All groups, applications, and administration role privileges assigned to the user account are revoked as part of the workflow. If you remove (delete) the user, the audit data of the user remains in the system. To manually (and immediately) purge the audit data of the deleted user, see Purge Audit Data for the Deleted User.

The following concepts are integral to user lifecycle management:

- User Account: A user account represents a user in Oracle Identity Cloud Service, and enables the user to access the Oracle Cloud service to which they belong. In Oracle Identity Cloud Service, there is a one-to-one relationship between a user and a user account. By default, all users can use their accounts to perform self-service capabilities in Oracle Identity Cloud Service. Users can update their profiles, reset their passwords, unlock their accounts, and change their email preferences.
- Administrator Role: You may want to provide a user account with administrative capabilities in Oracle Identity Cloud Service. To do this, you assign administrator roles to user accounts. See Understand Administrator Roles.
- Group: Oracle Identity Cloud Service provides easy and controlled privilege management through groups. Groups are the links between user accounts and applications in Oracle Identity Cloud Service. Groups are designed to ease the administration of privileges that you grant to user accounts or other groups. See Manage Oracle Identity Cloud Service Groups.
- Application: Oracle applications are a complete and modular set of enterprise applications, engineered from the ground up to be cloud-ready and to coexist seamlessly in mixed environments.

  You can use Oracle Identity Cloud Service to grant access to Oracle applications in two ways:

  – Directly: Assigning users to the applications
  – Indirectly: Assigning groups to the applications. Any users who are members of the groups are granted access to the applications.

  In addition to granting users and groups access to Oracle applications, you can grant users and groups access to entitlements within applications. For example, you use Oracle Identity Cloud Service to grant John Doe and Jane Doe access to Oracle Java Cloud Service. You want John Doe to have administrator privileges for Oracle Java Cloud Service, but Jane Doe to have user privileges only.

  Each entitlement in an Oracle application is represented by an **application role**. So by assigning John Doe to the application administrator role of Oracle Java Cloud Service, he

can not only access this Oracle Cloud service, but he can also function as an administrator within it.

See Manage Oracle Identity Cloud Service Applications for more information about how you can use Oracle Identity Cloud Service to grant and revoke access rights for users and groups to applications and application roles.

Understand Administrator Roles

In the following topic, you learn about Oracle Identity Cloud Service administrator roles and the privileges associated with each role.

In your organization, you might want administrators to have different rights of access to various tasks and resources in Oracle Identity Cloud Service. For example, the identity domain administrator has superuser privileges for an Oracle Identity Cloud Service identity domain. This administrator may want to delegate some of their responsibilities to other users to carry out the tasks associated with these responsibilities, such as managing system configuration and security settings, applications, users, groups, group memberships, and so on. To do this, the administrator assigns these users to other Oracle Identity Cloud Service administrator roles. Users who are assigned to these roles will be able to perform specific tasks that are associated with the roles.

The following table lists the Oracle Identity Cloud Service administrator roles that you can assign to users and describes the privileges for each administrator role. See Add or Remove a User Account from an Administrator Role.

Administrator Role	Privileges
Identity domain administrator	Has superuser privileges for an identity domain in Oracle Identity Cloud Service Identity domain administrators can: • Manage users, groups, applications, system configuration, and security settings • Perform delegated administration by assigning users to different administrative roles • Enable and disable Multi-Factor Authentication (MFA), configure MFA settings, and configure authentication factors • Create self-registration profiles to manage different sets of users, approval policies, and applications
Security administrator	Manage Oracle Identity Cloud Service system configuration and security settings for an identity domain in Oracle Identity Cloud Service. Security administrators can customize the interface, default settings, notifications, and the password policies, configure Multi-Factor Authentication (MFA), and manage the Microsoft Active Directory (AD) Bridge, Provisioning Bridge, identity providers, and trusted partner certificates.
Application administrator	Manage Oracle Identity Cloud Service applications. Application administrators can create, update, activate, deactivate, and delete applications. Application administrators can also grant and revoke access to applications for groups and users.
User administrator	Manage users, groups, and group memberships for an identity domain in Oracle Identity Cloud Service.
User manager	Manage all users or users of selected groups in Oracle Identity Cloud Service. User managers can update, activate, deactivate, remove, and unlock user accounts. User managers can also reset passwords, reset authentication factors, and generate bypass codes for user accounts.

Administrator Role	Privileges
Help desk administrator	Manage all users or users of selected groups in Oracle Identity Cloud Service. Help desk administrators can view the details of a user and unlock a user account. Help desk administrators can also reset passwords, reset authentication factors, and generate bypass codes for user accounts.
Audit administrator	Run reports for an identity domain in Oracle Identity Cloud Service.
User	Perform self-service capabilities in Oracle Identity Cloud Service. Users can update their profiles, reset their passwords, change their email preferences, link their social accounts to Oracle Identity Cloud Service, request access to groups and applications, view their access requests, access groups and applications assigned to them, and enroll in Multi-Factor Authentication (MFA). **Note:** By default, all Oracle Identity Cloud Service users are granted the User role. You can assign a user to the additional administrator roles that appear in this table.

> **Note:**
>
> See Typical Workflow for Using Oracle Identity Cloud Service to learn more about the tasks that users who belong to each administrator and user role can perform in Oracle Identity Cloud Service.

Create User Accounts

You can create user accounts only if you are granted access to the identity domain administrator or user administrator role in the **Administrators** page of the Identity Cloud Service console.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Users**, and then click **Add**.
2. In the **First Name** and **Last Name** fields of the **Add User** window, enter the user's first and last name.
3. To have the user log in to Oracle Identity Cloud Service with their email address:
   a. Leave the **Use the email address as the user name** check box selected.
   b. In the **User Name / Email** field, enter the email address for the user account.

   OR
4. To have the user log in to Oracle Identity Cloud Service with their user name:
   a. Clear the **Use the email address as the user name** check box.
   b. In the **User Name** field, enter the user name that the user is to use to log in to the Identity Cloud Service console.

> **Note:**
>
> The value that you enter into the **User Name** field can be either a valid email address or a non-email string. If it's a non-email string, then the following characters are allowed:
>
> - a-z
> - A-Z
> - 0-9
> - Special characters !@#$%^&*()_+=-{}[]|\:"';<>?/.,
> - White space

c. In the **Email** field, enter the email address for the user account.

> **Note:**
>
> If you turned off the **Allow primary email address as optional** switch in the **User Settings** page, then you must provide an email address in the **Email** field to create the user account.
>
> If you turned this switch on, then you can create the account without entering an email address in the **Email** field.

5. To assign the user account to a group, click **Next**. Otherwise, click **Finish**.
6. In the **Add User** window, select the check box for each group that you want to assign to the user account. Click **Finish**.

View Details About User Accounts

With the **Users** page, you can see profile information for a user account, any groups or apps to which the account is assigned, and risk data collected for the account.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Users**.
2. Click the user account about which you want to view additional information.

> **Tip:**
>
> To search for users, in the search field, enter all or part of the beginning of the user name, first name, or last name that you want to locate, and then press **Enter**. To fine-tune your search, click the search field again, and then select a status (Active, Inactive, or Locked).

3. Click **Details** to see additional profile information that you can edit, such as:

Profile Information	Description
The user's primary email address	This email address is the user's email address to which Oracle Identity Cloud Service will send notifications. See Understanding the Types of Notifications.
The user's password recovery email address	If the user forgets their password, then Oracle Identity Cloud Service will send notifications to this email address.
If the user account is a federated single sign-on (SSO) account	With a federated account, a user can interact with Oracle Identity Cloud Service through an identity provider, using a website that's external to Oracle Identity Cloud Service. See Adding an Identity Provider.
The user's country, time zone, and preferred language	If these attribute values for the user are different from attribute values that you set for the identity domain, then you can modify them. See Set Up or Modify Your Profile.
The custom and enterprise attributes assigned to a user	If you can't find an attribute that you need from the base Oracle Identity Cloud Service schema attributes, you can add your own custom attributes. See Add a Custom Schema Attribute.

4. Click **Groups** to see a list of any groups assigned to the user account.

   You can assign groups to the user account or remove groups from the user account.

5. Click **Access** to see a list of any applications assigned to the user account.

   You can assign applications to the user account or remove applications from the user account.

> **Note:**
>
> The Active icon for each application on the Access tab represents the active status of the user account and not the application status. The status remains active as long as the user account is active, regardless of whether the application is active or inactive.

6. Click **Security** to see risk data collected for the user, including whether the user is enrolled in Multi-Factor Authentication (MFA).

> **Note:**
>
> If you don't see the **Security** tab, then activate Adaptive Security or at least one risk provider. See Activate Adaptive Security and Activate a Risk Provider. Also, see Understand Risk Providers to learn more about risk ranges, risk providers, and risk scores associated with user accounts because you must be familiar with these concepts to understand the panes of the **Security** tab.

7. In the **User Risk Scores** pane, click the default risk provider to view the risk incidents and details associated with this risk provider for the user account.

> **Note:**
>
> If you don’t see the default risk provider, then activate it. See Activate a Risk Provider.

Two panes appear below the default risk provider: **Risk Incidents** and **Details**.

- The **Risk Incidents** pane displays a graph that illustrates user-threat risk scores and risk scores after remediation for a selected time interval. The risk scores are displayed as per the risk score ranges.
- The **Details** pane displays incidents associated with actions that a user is performing in Oracle Identity Cloud Service.

  There are three incidents (or events) that Oracle Identity Cloud Service uses to lower the risk score of the user:

  – **Time-based risk-score re-evaluation:** The user's risk score has been lowered because Oracle Identity Cloud Service detected that the user hasn't committed risky activity over a period of time. The score is reduced periodically as long as there are no threat events.

  – **Successful user password reset:** The user reset their Oracle Identity Cloud Service password.

  – **Successful user login:** The user signed in to Oracle Identity Cloud Service.

  > **Note:**
  >
  > If the default risk provider is deactivated, then the user’s risk score won’t be lowered.

> **Tip:**
>
> Expand the **Access from suspicious IP addresses** event and click the **Information** icon to the right of the IP address to see why the integrated IP reputation provider blacklisted it. Reasons include:
>
> - Spam Sources: The IP address is tunnelling spam messages through proxy, anomalous SMTP activities, or forum spam activities.
> - Windows Exploits: The IP address is offering or distributing malware, shell code, rootkits, worms, or viruses.
> - Web Attacks: The IP address is involved in attacks such as cross-site scripting, iFrame injection, SQL injection, cross-domain injection, or domain password brute force.
> - Botnets: The IP address is seen in Botnet C&C channels and infected zombie machines are controlled by the bot master.
> - Scanners: The IP address is seen in reconnaissance such as probes, host scans, domain scans, and password brute force.
> - Denial of Service: The IP address is noticed in DOS, DDOS, anomalous SYN flood, and anomalous traffic detection.
> - Phishing: The IP address is hosting phishing sites and other kinds of fraud activities such as Ad Click Fraud or Gaming Fraud.
> - Proxy: The IP address is providing proxy and anonymization services. This also includes TOR anonymizer IP addresses.
> - Mobile Threats: The IP address is associated with malicious and unwanted mobile applications.
> - Package: This IP address is associated with information about all other reasons.
> - TOR Proxy: The IP address acts as an exit node for the TOR Network. The exit node is at the last point along the proxy chain and makes a direct connection to the originator's intended destination.
> - Reputation: This IP address is associated with other IP addresses (for example, through common ownership, having the same subnet, and so on). This IP address is classified as high risk because of documented threat activity.

> **Note:**
>
> If the default risk provider is deactivated, then the user's risk score won't be increased.

8. In the **Risk Incidents** pane, filter the data that appears in this graph by completing one of the following options:
   a. To view risk score ranges that represent user-threat risk scores and risk scores after remediation for the current day, week, or month, or since the user signed in to Oracle Identity Cloud Service for the first time, from the drop-down menu, select **1 Day**, **1 Week**, **1 Month**, or **All**.

b. To specify a custom date-and-time range to view risk-related user activity for the user account, click the left **Calendar** icon to specify the start date and time, and the right **Calendar** icon to set the end date and time.

9. In the **Details** pane, click an incident (either a threat or an event) to learn more about it.

Edit Attribute Values for the User Account

After viewing details about a user account, you can modify the account by editing attribute values for the user account.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Users**.
2. Click the user account that you want to modify.
3. Click **Details**.
4. Modify an attribute value for the user account by:
   a. Entering a value in the attribute field (for example, the **City** field).
   b. Turning on or off a switch (for example, the **Federated** switch).
   c. Selecting a value from the drop-down list (for example, a Microsoft Active Directory (AD) domain from the **Authenticated By** list).

> **Note:**
>
> You can't edit attribute values for your user account. To do this, access the **My Profile Details** tab of the My Profile console. S

> **Note:**
>
> A new feature of Oracle Identity Cloud Service is delegated authentication. Delegated authentication allows identity domain administrators and security administrators to specify whether users can use their Oracle Identity Cloud Service or AD passwords to sign in to Oracle Identity Cloud Service to access resources protected by Oracle Identity Cloud Service, such as the My Profile console, Identity Cloud Service console, and apps assigned to the user.
>
> For example, suppose you configured delegated authentication for an AD Bridge in Oracle Identity Cloud Service so that a user can use their AD password to authenticate into Oracle Identity Cloud Service. Or, perhaps the user's account can be synchronized between AD and Oracle Identity Cloud Service from more than one bridge.
>
> From the **Authenticated By** list, you can select which AD domain contains the user's credentials to sign in to Oracle Identity Cloud Service. Or, select **Oracle Identity Cloud Service** to have the user sign in with their Oracle Identity Cloud Service password.
>
> > **Note:**
> >
> > If you don’t see a value in the **Authenticated By** list, click **View Authentication Source**. If you configured delegated authentication for this user account, then the AD Bridge associated with the user’s AD domain appears. Otherwise, **Oracle Identity Cloud Service** is displayed.

5. After editing attribute values for the user account, click **Update User**.

Assign Groups to the User Account

After viewing details about a user account, you can modify the account by assigning groups to the user account.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Users**.
2. Click the user account that you want to modify.
3. Click **Groups**.
4. Click **Assign**. To search for groups to assign to the user account, in the search field, enter all or part of the beginning of the group names or descriptions that you want to locate, and then press **Enter**.
5. In the **Assign Groups** window, select the check box for each group that you want to assign to the user account.
6. Click **OK**.

Remove Groups from the User Account

After viewing details about a user account, you can modify the account by removing groups from the user account.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Users**.
2. Click the user account that you want to modify.
3. Click **Groups**.

> **Tip:**
>
> To search for groups to remove from the user account, in the search field, enter all or part of the beginning of the group names or descriptions that you want to locate, and then press **Enter**.

4. Select the check box for each group that you want to remove from the user account.
5. Click **Revoke**.
6. In the **Confirmation** window, click **OK**.

Assign Applications to the User Account

After viewing details about a user account, you can modify the account by assigning applications to it.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Users**.
2. Click the user account that you want to modify.
3. Click **Access**.
4. Click **Assign**.
5. In the **Assign Applications** window, click **Assign** for each application that you want to assign to the user account.
6. If you're assigning a managed application to the user account, then an **Assign Application** window appears, containing a form for the application. To populate this form:
   a. Enter the required values for the form.
   b. If the form contains multi-valued attributes, then an **Add** button appears to the right of each attribute. Click **Add**, and then in the **Allowed Values** window, select the values for the attribute, and click **OK**.

      > **Tip:**
      >
      > To remove an existing value from the attribute, click the **X** button to the right of the value.

   c. Click **Save**.

> **Note:**
>
> See Architecture Diagram Defining Oracle Identity Cloud Service and Provisioning Integration for more information about managed applications and application forms.
>
> The **Active** icon for each application in the **Access** tab represents the active status of the user account and not the application status. The status remains active as long as the user account is active, regardless of whether the application is active or inactive.

7. Click **OK**.

> **Note:**
>
> If you assigned a managed application to the user account, then you can modify the values of the application form. To do this, click the **Action** menu, select **Edit**, change the appropriate values, and then click **Save**.
>
> Also, if you have enabled and configured synchronization for an App Catalog app, and assigned the app to a user account, then you can activate or deactivate the user's account with the app. To do so:
>
> a. Click the **Action** menu to the right of the App Catalog app that you assigned to the user.
>
> b. Click **Activate** or **Deactivate**.
>
> c. In the **Confirmation** window, click **OK**.

Remove Applications from the User Account

After viewing details about a user account, you can modify the account by removing applications from it.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Users**.
2. Click the user account that you want to modify.
3. Click **Access**.
4. Select the check box for each application that you want to remove from the user account.
5. Click **Revoke**.
6. In the **Confirmation** window, click **OK**.

Activate User Accounts

Activating a user account reinstates the access rights of the user account for Oracle Identity Cloud Service.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Users**.

2. Select the check box for each deactivated user account that you want to activate, or to activate all deactivated user accounts, search for accounts with a status of *Inactive*. Then, select the **Select All** check box.

> **Tip:**
>
> A deactivated account is designated by a red circle with a white line through the circle.

3. Click **Activate**.
4. In the **Confirmation** window, click **OK**.

Deactivate User Accounts

Deactivating a user account temporarily disables the access rights that the user account has to Oracle Identity Cloud Service.

Deactivated users are not be able to login until you reactivate the user account. Group memberships and application roles remain intact and are available once the user account is reactivated.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Users**.
2. Select the check box for each activated user account that you want to deactivate.

> **Tip:**
>
> To deactivate all activated user accounts, search for accounts with a status of *Active*. Then, select the **Select All** check box.

3. Click **Deactivate**.
4. In the **Confirmation** window, click **OK**.

Import User Accounts

If you are an identity domain administrator or a user administrator, you can batch import user accounts using a comma-separated values (CSV) file.

Before you can import user accounts, first create a CSV file that is properly formatted for the import process. To create and prepare a file for import, follow these steps.

1. Use these sample files as a starting point.
2. Extract the compressed file and then open the `Users.csv` file.
3. Review and then delete any demo data in the `Users.csv` file.

> **Tip:**
>
> To familiarize yourself with the import process, consider importing just the demo data. You can then delete the unwanted demo data from Oracle Identity Cloud Service before you begin importing production data.

4. Create an import file using the `Users.csv` file. The `Users.csv` file is a simple text file in a tabular format (rows and columns). The first row in the file defines the columns (fields) in your table.

> **Note:**
>
> - The maximum number of rows in the user import file must not exceed 100,000 and the import file size must not exceed 52 MB.
> - At a minimum, the file must have these exact column headings and the fields in these columns must be unique.
>   - User ID
>   - Last Name
>   - First Name
>   - Work Email
>   - Primary Email
>   - Primary Email Type
> - For each account, you create a new row (line) and enter data into each column (field). Each row equals one record.
> - The IDs of the users that you want to import into Oracle Identity Cloud Service must contain at least three characters. The names of the groups that you want to import into Oracle Identity Cloud Service must contain at least five characters.
> - The telephone numbers of the users that you want to import must meet the requirements of the RFC 3966 specification.
> - When importing users, the attribute `Recovery` cannot be specified as one of valid values for **Primary Email Type**. The valid values for Primary Email Type are `home`, `work`, or `other`.
> - If you want users to use their federated accounts to sign in to Oracle Identity Cloud Service, then you must set the **Federated** column to `TRUE` for those users. When the federated flag is set, Oracle Identity Cloud Service no longer manages the federated user's password. This prevents Oracle Identity Cloud Service from forcing a password change for these imported user accounts.
> - If you don't want users to be notified that Oracle Identity Cloud Service created accounts for them, then you must set the **ByPass Notification** column to `TRUE` for those users. The ByPass Notification flag determines whether an email notification is sent after creating or updating a user.
> - To create a CSV file, you can use a standard spreadsheet application, such as Microsoft Excel or Google Sheets, or you can use a text editor, such as Notepad or TextPad.

5. Save your file in a CSV format.
   a. Open the CSV file with a text editor, such as Notepad.
   b. Save the file with UTF-8 for encoding.

> **Note:**
>
> If you do not save the file in a CSV format with UTF-8 encoding, the import fails. Saving the file in UTF-8 format ensures that non-English characters display properly.

To import user accounts:

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Users**.
2. Click **Import**.
3. In the **Import Users** dialog box, click **Browse** to locate and select the CSV file that contains the user accounts to import.

   > **Note:**
   >
   > Click **Download sample file** in the dialog box to download a sample file.

4. Verify that the path and name of the CSV file that you selected appear in the **Select a file to import** field.
5. Click **Import**.

   If a user account is missing a required value, such as the user's first name, last name, or user name, then Oracle Identity Cloud Service can't import it. If Oracle Identity Cloud Service can't import a user account, then it evaluates the next account in the CSV file.
6. After Oracle Identity Cloud Service evaluates all user accounts, review the job results.
   - If the job *can* be processed immediately, then a dialog box appears with the **Job ID** link for your import job. Click the link and review the details that appear on the **Jobs** page.
   - If the job *cannot* be processed immediately, then a message appears with a **Schedule ID** in it. Copy that **Schedule ID**, and use it to search for the job on the **Jobs** page. The job will appear when processing completes. Go to Step 7.

   > **Tip:**
   >
   > Oracle Identity Cloud Service assigns a job ID to each file that's imported or exported, for auditing purposes.

7. On the **Jobs** page, locate the job that you want to view, and then click **View Details**.

   A table displays the first names, last names, email addresses, user names, and statuses of the user accounts that you imported into Oracle Identity Cloud Service.

> **Note:**
>
> If a user account can be imported into Oracle Identity Cloud Service, then a **Creation Succeeded** or **Update Succeeded** link appears for the status, depending on whether you imported a new account or modification to an existing account. To see granular details about the account, click the link.
>
> If a user account can't be imported, then a **Creation Failed** or **Update Failed** link appears for the status. To see information about why the account or modification can't be imported into Oracle Identity Cloud Service, click the link.

8. Review the details that appear on the **Jobs** page.

   This page shows how many accounts you imported, how many accounts imported successfully, and how many accounts can't be imported because of a system error.

Export User Accounts

Using the Oracle Identity Cloud Service admin console, you can export the user accounts for the following attributes only: *User Name*, *Work Email*, *Home Email*, *Primary Email Type*, *Honorific Prefix*, *First Name*, *Middle Name*, *Last Name*, *Honorific Suffix*, and *Password*.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Users**.
2. To export all user accounts, click **Export**, and then select **Export All**.

   OR

   To export only some user accounts, select the check box for each user account that you want to export. Click **Export**, and then select **Export Selected**.

   > **Tip:**
   >
   > The number that appears within parentheses to the right of **Export All** is how many user accounts are created in Oracle Identity Cloud Service. The number within parentheses to the right of **Export Selected** is how many user accounts you selected to export.

3. In the **Confirmation** window, click **OK**.
4. After Oracle Identity Cloud Service creates the export file, you need to review the results.
   - If the job *can* be processed immediately, then a dialog box appears with the **Job ID** link for your import job. Click the link and review the details that appear on the **Jobs** page.
   - If the job *cannot* be processed immediately, then a message appears with a **Schedule ID** in it. Copy that **Schedule ID**, and use it to search for the job on the **Jobs** page. The job will appear when processing completes.
5. On the **Jobs** page, locate the job that you want to view, and then click **View Details**.

   A page shows how many user accounts you exported, how many accounts Oracle Identity Cloud Service exported successfully, and how many accounts can't be exported because of a system error.
6. Click **Download**.

7. Save your file in a UTF-8 format. Saving the file in UTF-8 format ensures that non-English characters display properly.
8. (Optional) In addition to saving the file in UTF-8 format, if you are using Microsoft Excel to open and save the file, perform the additional steps to ensure that non-English characters display properly.
    a. In Microsoft Excel, open a new workbook, click the **Data** tab, and then choose **From Text**.
    b. On the **Import Text File** window, choose your CSV file, and then click **Import**.
    c. For **Original data type**, select **Delimited**, for **File origin**, select **65001: Unicode (UTF-8)**, and then click **Next**.
    d. For **Delimiters**, select **Comma**, deselect all other options, and then click **Next**.
    e. For **Column data format**, select **General**, and then click **Finish**.
    f. Click **OK**.
    g. Save the file.

Generate Bypass Codes for User Accounts

You can increase security for user accounts by using Multi-Factor Authentication (MFA) capabilities provided by Oracle Identity Cloud Service. MFA adds an extra layer of identity verification to the login process by requiring a user to provide a second verification method, such as a one-time passcode (OTP) for the device associated with the user's account, notification, short message service (SMS), also known as a text message, or security questions.

The ability to generate a bypass code is available to the user after the user enrolls in 2-Step Verification. The user can generate a bypass code and store it for later use or request that an administrator generate a bypass code for the user. For example, when a user has forgotten their phone, doesn't have cell service, or can't access their computer, at the **2-Step Verification** page, the user can contact the help desk to have an administrator generate a bypass code.

As a result, the user can use this bypass code as a one-time 2-Step Verification method to log in to Oracle Identity Cloud Service.

In addition, the administrator can set when the bypass code expires, and how often the bypass code can be used for the user account.

> **Note:**
>
> The user must already be enrolled in MFA to use a bypass code or request that one be generated for the user.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Users**.
2. Click the user account for which you want to generate a bypass code.
3. Click the **Action** menu, and then select **Generate Bypass Code**.
4. In the **Bypass Code expires after** region of the **Generate Bypass Code** window, set when the bypass code expires.

a. Set the time (in days, hours, and minutes) that the bypass code will expire. After this time elapses, the user can't use the bypass code.

b. If you don't want the bypass code to expire, then click **Never Expires**.

5. In the **Bypass Code can be used** region of the **Generate Bypass Code** window, specify how often the bypass code can be used.

   a. If the bypass code can be used only one time, then click **Once**.

   b. If the bypass code can be used for a finite number of times, then click the button to the left of the text box. Enter a number in the text box that represents how many times the bypass code can be used.

   c. If the bypass code can be used for an unlimited number of times, then click **Unlimited**.

6. Click **OK**.

7. In the **Bypass Code** window, click **Email**. A notification is sent to the user. This notification contains the bypass code that the user uses as a one-time 2-Step Verification method to log in to Oracle Identity Cloud Service.

Reset Authentication Factors for User Accounts

Reset all verification factors for users enrolled in Multi-Factor Authentication (MFA) if a user's device can't be used to provide a second factor for authentication. Resetting all verification factors removes any existing factors in which the user is enrolled.

Resetting all verification factors removes any existing factors in which the user is enrolled. The next time the user logs in, the user is prompted to enroll in 2–Step Verification and account recovery.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Users**.

2. Click the user account for which you want to reset authentication factors.

3. Click the **Action** menu  , and then select **Reset Factors**.

4. In the **Confirmation** window, click **OK**.

Unlock User Accounts

After a consecutive number of unsuccessful login attempts to Oracle Identity Cloud Service, a user account is locked. The user receives a notification that contains a link that the user can click to reset their password and unlock their account. An administrator can unlock accounts without requiring a password reset.

If a user's account is locked, and the user or an administrator doesn't unlock the account, then Oracle Identity Cloud Service will unlock it automatically. An administrator can set this time period ranging between 5 minutes and 24 hours. See Modify the Custom Password Policy.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Users**.

2. Click the user account that you want to unlock.

> **Tip:**
>
> To display all user accounts that are locked, click the search field and select the **Locked** status.

3. Click the **Action** menu , and then select **Unlock User**.
4. In the **Confirmation** window, click **OK**.

Add or Remove a User Account from an Administrator Role

After you create or import user accounts in Oracle Identity Cloud Service, you can delegate administrative responsibilities for these accounts.

By default, all users can perform self-service capabilities in Oracle Identity Cloud Service, such as updating their profiles, resetting their passwords, and changing their email preferences. You may want to provide a user account with administrative capabilities. For example, you may want a user to manage applications in Oracle Identity Cloud Service. So, you would assign the user account to the application administrator role.

A user account can be assigned to more than one administrator role. The user account inherits the privileges for each administrator role assigned to the account. If a user account is assigned to both the application administrator role and the user administrator role, then the user can manage applications, users, groups, and group memberships in Oracle Identity Cloud Service.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Administrators**.
2. Expand the node for the administrator role for which you want to add or remove a user account, and then perform one of the following:
   - To add a user account to an administrator role, click **Add**, select the check box for each user account that you want to add, and then click **OK**.

     If you're adding users to the user manager role, then after selecting the check box for each user that you're adding to this role, you must also select one of the following options:

     – **Manage all users**: These users can manage all users in the Oracle Identity Cloud Service identity domain.

     – **Manage selected groups of users**: These users can manage only those users who belong to the groups that you select. After selecting this option, enter or select the groups to be managed by these users.

     After making this selection, click **OK**. If you want to modify either the users who are assigned to the user manager role or the groups that these users can manage, then click the **Action** menu , and select **Edit** from the drop-down menu that appears.
   - To remove a user account from an administrator role, select the user account that you want to remove, click **Remove**, and then in the **Confirmation** window, click **OK**.

Generate Personal Access Tokens

An access token is an authorization that's used by a client application to access an API or a resource application within a limited period of time.

The time-bound access tokens inform the resource application that the client is authorized to access the application and perform specific actions specified by the scope that's granted.

You can download access tokens only if an identity domain administrator assigns administrator roles or resource applications to your user account.

To generate personal access tokens:

1. Access Oracle Identity Cloud Service console, click the avatar icon on the top-right corner, and then click **My Access Tokens**.
2. You can download an access token in the following ways:
   - Select **Invokes Identity Cloud Service APIs** to specify the available administrator roles that are assigned to you. The APIs from the specified administrator roles will be included in the token.
   - Select **Invokes other APIs** to select confidential applications that are assigned to the user account.
     a. Click **Select an Application** to add a configured confidential resource application. On the **Select an Application** window, the list of assigned confidential applications displays.
     b. Click applications to select them, and then click **Add**. The **My Access Tokens** page lists the added applications.
3. In the **Token Expires in (Mins)** field, select or enter how long (in minutes) the access token you're generating can be used before it expires. You can choose to keep the default number or specify between **1** and **527,040**.
4. Click **Download Token**. The access token is generated and downloaded to your local machine as a **tokens.tok** file.

Send Invitations to Users to Activate Their Accounts

After a user account is created in Oracle Identity Cloud Service, a Welcome invitation is sent to the user, requesting that the user activate the account. The new user account must be activated before it can be used.

If the user account isn't activated after a designated amount of time, then the Oracle Identity Cloud Service administrator can send another invitation to the user to activate the account.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Users**.
2. Select the check box for each user account to which you want to send an invitation.

> **Tip:**
>
> To send invitations to all user accounts, select the **Select All** check box.

3. Click **More**, and then select **Resend Invitation**.

4. In the **Confirmation** window, click **OK**.

Reset Passwords for User Accounts

You can use Oracle Identity Cloud Service to reset the password for a user account. When you request a password change, Oracle Identity Cloud Service sends a notification to the user so that the user can provide a new password for the account.

You can reset a password for a single account, for multiple accounts, or for all accounts in the identity domain.

You can't reset the passwords for deactivated user accounts. To activate all deactivated user accounts, search for accounts with a status of *Inactive*. Then select the **Select All** check box.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Users**.
2. Select the check box for each user account for which you want to reset the password.

> **Tip:**
>
> To reset the passwords for all user accounts, do not select any check boxes, and go to Step 3.

3. Click **More**, and then perform one of the following choices.
   - If you selected either a single or multiple user accounts:
     a. Select **Reset Password**.
     b. In the **Confirmation** window, click **OK**.
   - If you didn't select any user accounts (because you want to reset the passwords for all accounts):
     a. Select **Reset All Passwords**.
     b. In the **Confirmation** window, click **OK**.

Remove User Accounts

You can remove user accounts who no longer need access to the service. You can remove either a single user account or multiple accounts.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Users**.
2. Select the check box for each user account that you want to remove.
3. Click **More**, and then click **Remove**.
4. In the **Confirmation** window, click **OK**.

> **Note:**
>
> If you remove (delete) a user, the audit data of the user remains in the system.

4

Manage Oracle Identity Cloud Service Groups

Learn how to manage Oracle Identity Cloud Service groups.

Topics

Typical Workflow for Managing Oracle Identity Cloud Service Groups

With the group management feature in Oracle Identity Cloud Service, you can perform tasks such as creating, managing, and removing groups.

Task	Description	Additional Information
Understand groups.	Learn about groups, including how groups are used to link user accounts to applications in Oracle Identity Cloud Service.	Understand Groups
Create groups.	Create new groups using the **Groups** page.	Create Groups
View details about groups.	View details about groups using the **Groups** page.	View Details About Groups

Task	Description	Additional Information
Modify groups.	Modify groups using the **Groups** page.	Edit Attribute Values for the Group Assign User Accounts to the Group Remove User Accounts from the Group Assign Applications to the Group Remove Applications from the Group
Import and export groups.	Import and export groups using the **Groups** page.	Import Groups Export Groups
Remove groups.	Remove groups using the **Groups** page.	Remove Groups

You can create, manage, and remove groups by using:

- The Identity Cloud Service console
- SCIM-based APIs

In this section, you learn how to create, manage, and remove groups by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Understand Groups

As an identity domain administrator or user administrator, you use Oracle Identity Cloud Service groups to manage the accounts of users to whom you want to grant access to Oracle applications or application roles.

In Oracle Identity Cloud Service, groups are the links between user accounts and applications. They contain privileges that you grant to users. Groups ease the administration of user privileges.

Using groups, you can:

- Designate the applications and application roles that users can access through the Identity Cloud Service console
- Assign users to the groups
- Designate other Oracle Identity Cloud Service administrators to perform actions on groups:
  - Assigning or removing members to or from the current group
  - Modifying other characteristics of the group, such as the group description

> **Note:**
>
> The **All Tenant Users** group is a group that's created by Oracle Identity Cloud Service. All Oracle Identity Cloud Service users are assigned to this group, by default. If you assign this group to any of your applications, then all users are assigned to these applications indirectly.
>
> For a user, the **All Tenant Users** group doesn't appear in the **Groups** tab because this group is assigned automatically when a new user is created. Also, because this group is created by Oracle Identity Cloud Service, and not by an administrator, you can't delete this group.

Create Groups

You can create groups in Oracle Identity Cloud Service.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Groups**.
2. Click **Add**.
3. In the **Name** and **Description** fields of the **Add Group** window, enter the name and descriptive information about the group.
4. To allow users to request access to this group, click **User can request access**.
5. To assign user accounts to the group, go to step 6. Otherwise, click **Finish**.
6. Click **Next**.
7. Select the check box for each user account that you want to assign to the group, and then click **Finish**.

> **Tip:**
>
> To search for user accounts to assign to the group, in the search field, enter all or part of the beginning of the user names, first names, or last names of the user accounts that you want to locate, and then press **Enter**.

View Details About Groups

By default, in the **Groups** page, you can see the name and description for each group.

You can also see other information about a group, such as any user accounts assigned to the group.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Groups**.
2. Click the group about which you want to view additional information.

> **Tip:**
>
> To search for groups, enter all or part of the beginning of the group name that you want to locate in the **Search Groups** field, and then press **Enter**.

3. Click **Details**.

   In this tab, you see information about the group, including the name and description of the group, and whether users can request access to this group. You can edit attribute values for the group.

4. Click **Users**.

   In this tab, you see a list of user accounts assigned to the group. You can assign user accounts to the group or remove user accounts from the group.

5. Click **Access**.

   In this tab, you see a list of any applications assigned to the group. You can assign applications to the group or remove applications from the group.

Edit Attribute Values for the Group

You can modify a group by editing attribute values for the group.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Groups**.
2. Click the group that you want to modify.
3. Click **Details**.
4. Enter or select the modification in the attribute field (for example, modify the group name in the **Name** field or select the **User can request access** check box).
5. Click **Update**.

Assign User Accounts to the Group

You can modify a group by assigning user accounts to the group.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Groups**.
2. Click the group that you want to modify.
3. Click **Users**.
4. Click **Assign**.

> **Tip:**
>
> To search for user accounts to assign to the group, in the search field, enter all or part of the beginning of the user names, first names, or last names of the user accounts that you want to locate, and then press **Enter**.

5. In the **Assign Users** window, select the check boxes for the user accounts that you want to assign to the group.
6. Click **OK**.

Remove User Accounts from the Group

You can modify a group by removing user accounts from the group.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Groups**.
2. Click the group that you want to modify.
3. Click **Users**.
4. Select the check box for each user account that you want to remove from the group.

> **Tip:**
>
> To search for user accounts to remove from the group, in the search field, enter all or part of the beginning of the user names, first names, or last names of the user accounts that you want to locate, and then press **Enter**.

5. Click **Revoke**.
6. In the **Confirmation** window, click **OK**.

Assign Applications to the Group

You can modify a group by assigning applications to the group.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Groups**.
2. Click the group that you want to modify.
3. Click **Access**.
4. Click **Assign**.
5. In the **Assign Applications** window, click **Assign** for each application that you want to assign to the group.
6. If you're assigning a managed application to the group, then an **Assign Application** window appears, containing a form for the application. To populate this form:
   a. Enter the required values for the form.
   b. If the form contains multi-valued attributes, then an **Add** button appears to the right of each attribute. Click **Add**, and then in the **Allowed Values** window, select the values for the attribute, and click **OK**.

   > **Tip:**
   >
   > To remove an existing value from the attribute, click the **X** button to the right of the value.

   c. Click **Save**.

> **Note:**
>
> See Architecture Diagram Defining Oracle Identity Cloud Service and Provisioning Integration for more information about managed applications and application forms.
>
> The **Active** icon for each application in the **Access** tab represents the active status of the group and not the application status. The status remains active as long as the group is active, regardless of whether the application is active or inactive.

7. Click **OK**.

> **Note:**
>
> If you assigned a managed application to the group, then you can modify the values of the application form. To do this, click the **Action** menu  , select **Edit**, change the appropriate values, and then click **Save**.

8. (Optional) To assign an application to all users, search for and click the **All Tenant Users** group, and then click **Access**. Click **Assign**, search for the application, and assign it to the group.

Remove Applications from the Group

You can modify a group by removing applications from the group.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Groups**.
2. Click the group that you want to modify.
3. Click **Access**.
4. Select the check box for each application that you want to remove from the group.
5. Click **Revoke**.
6. In the **Confirmation** window, click **OK**.

Import Groups

If you are an identity domain administrator or a user administrator, you can batch import groups using a comma-separated values (CSV) file.

Before you can import groups, first create a CSV file that is properly formatted for the import process. To create and prepare a file for import, follow these steps.

1. Use these sample files as a starting point.
2. Extract the compressed file and then open the `Groups.csv` file.
3. Review and then delete any demo data in the `Groups.csv` file.

> **Tip:**
>
> To familiarize yourself with the import process, consider importing just the demo data. You can then delete the unwanted demo data from Oracle Identity Cloud Service before you begin importing live data.

4. Create an import file using the `Groups.csv` file. The `Groups.csv` file is a simple text file in a tabular format (rows and columns). The first row in the file defines the columns (fields) in your table. At a minimum, the file must have these exact column headings.
   - Display Name
   - Description
   - User Members

   > **Tip:**
   >
   > Ensure that the fields in these columns are unique. Also, verify that the user names that appear in the **User Members** column already exist in Oracle Identity Cloud Service.

   For each account, you create a new row (line) and enter data into each column (field). Each row equals one record.

   > **Important:**
   >
   > The IDs of the users that you want to import into Oracle Identity Cloud Service must contain at least three characters. The names of the groups that you want to import into Oracle Identity Cloud Service must contain at least five characters.
   >
   > The telephone numbers of the users that you want to import must meet the requirements of the RFC 3966 specification.
   >
   > The maximum number of rows in group import file must not exceed 100,000 and the import file size must not exceed 52 MB.

   To create a CSV file, you can use a standard spreadsheet application, such as Microsoft Excel or Google Sheets, or you can use a text editor, such as Notepad or TextPad.

5. Save your file in a CSV format.
   a. Open the CSV file with a text editor, such as Notepad.
   b. Save the file with UTF-8 for encoding. Saving the file in UTF-8 format ensures that non-English characters display properly.

   > **Note:**
   >
   > If you do not save the file in a CSV format with UTF-8 encoding, the import fails.

To import groups:

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Groups**.
2. Click **Import**.
3. In the **Import Groups** window, click **Browse** to locate and select the CSV file that contains the groups to import.

> **Note:**
>
> Click **Download sample file** in the dialog box to download a sample file.

4. Verify that the path and name of the CSV file that you selected appear in the **Select a file to import** field.
5. Click **Import**.

   The reasons for Oracle Identity Cloud Service not being able to import a group are that the group already exists or the group is missing a required value, such as the group name. If Oracle Identity Cloud Service can't import a group, then it evaluates the next group in the CSV file.
6. After Oracle Identity Cloud Service evaluates all groups, review the job results.
   - If the job *can* be processed immediately, then a dialog box appears with the **Job ID** link for your import job. Click the link and review the details that appear on the **Jobs** page.
   - If the job *cannot* be processed immediately, then a message appears with a **Schedule ID** in it. Copy that **Schedule ID**, and use it to search for the job on the **Jobs** page. The job will appear when processing completes. Go to Step 7.

> **Tip:**
>
> Oracle Identity Cloud Service assigns a job ID to each file that's imported or exported, for auditing purposes.

7. On the **Jobs** page, locate the job that you want to view, and then click **View Details**.

   The **Job Details** page shows how many groups you imported, how many groups imported successfully, and how many groups can't be imported because of a system error. For each group that you imported successfully, this page also shows how many user accounts are assigned to the group.
8. To see more information about a group, click **View Details**.

> **Note:**
>
> If the group can be imported into Oracle Identity Cloud Service or the user accounts can be assigned to the group, then a **Creation Succeeded** or **Update Succeeded** link appears for the status, depending on whether you imported a new group or modification to an existing group or group membership. To see granular details about the group, click the link.
>
> If a group can’t be imported, then a **Creation Failed** or **Update Failed** link appears for the status. To see information about why the group or modification can’t be imported into Oracle Identity Cloud Service, click the link.

Export Groups

You can export groups from Oracle Identity Cloud Service in order to import groups into another identity domain.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Groups**.
2. Click **Export**, and then select **Export All** to export all groups.

   OR

   To export only some groups, select the check box for each group that you want to export. Click **Export**, and then select **Export Selected**.

   > **Tip:**
   >
   > The number that appears within parentheses to the right of the **Export All** menu item is the total number of groups in Oracle Identity Cloud Service. The number within parentheses to the right of the **Export Selected** menu item is how many groups you selected to export.

3. In the **Confirmation** window, click **OK**.
4. After Oracle Identity Cloud Service creates the export file, a **Job ID** link appears. Click the link.
5. In the **Jobs** page, review the job details such as how many groups you exported, how many groups Oracle Identity Cloud Service exported successfully, and how many groups can't be exported because of a system error.
6. Click **View Details**, and then click **Download**.
7. Save your file in a UTF-8 format. Saving the file in UTF-8 format ensures that non-English characters display properly.
8. (Optional) In addition to saving the file in UTF-8 format, if you're using Microsoft Excel to open and save the file, then perform the additional steps to ensure that non-English characters display properly.
   a. In Microsoft Excel, open a new workbook, click the **Data** tab, and then choose **From Text**.
   b. On the **Import Text File** window, choose your CSV file, and then click **Import**.
   c. For **Original data type**, select **Delimited**, for **File origin**, select **65001: Unicode (UTF-8)**, and then click **Next**.
   d. For **Delimiters**, select **Comma**, deselect all other options, and then click **Next**.
   e. For **Column data format**, select **General**, and then click **Finish**.
   f. Click **OK**.
   g. Save the file.

Remove Groups

You can remove unused groups from Oracle Identity Cloud Service.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Groups**.
2. Select the check box for each group that you want to remove, or to remove all groups, select the **Select All** check box.
3. Click **Remove**.
4. In the **Confirmation** window, click **OK**.

5

Manage Oracle Identity Cloud Service Applications

Learn how to manage applications for Oracle Identity Cloud Service.

Topics:

Typical Workflow for Managing Oracle Identity Cloud Service Applications

With the application management feature in Oracle Identity Cloud Service, you can perform tasks such as creating, managing, and removing applications.

Task	Description	Additional Information
Understand cloud applications.	You can learn about cloud applications, including the two types of applications you can manage in Oracle Identity Cloud Service: Oracle applications and custom applications.	About Oracle and Custom Applications
Use case for adding applications.	You can learn about a use case that describes the process flow for adding two custom applications in Oracle Identity Cloud Service: a trusted application and a resource server application.	Use Case: Adding Applications
Add applications.	You can add applications using the **Applications** page.	Add Applications
View details about applications.	You can view details about applications using the **Applications** page.	View Details About Applications

Task	Description	Additional Information
Modify applications.	You can modify applications using the **Applications** page.	About Modifying Applications
Activate and deactivate applications.	You can activate and deactivate applications using the **Applications** page.	Activate Applications Deactivate Applications
Remove applications.	You can remove applications using the **Applications** page.	Remove Applications

You can create, manage, and remove applications by using:

- The Identity Cloud Service console
- SCIM-based APIs

In the following sections, you learn how to manage applications by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Understand Cloud Applications

Oracle Identity Cloud Service provides you with a secure and centralized cloud service to manage your applications.

Topics:

About Cloud Applications

Cloud applications are web-based applications that function in the cloud. These applications can be accessed from anywhere, and at any time, over the web. Examples of cloud applications are Google, Salesforce, and Dropbox.

About Oracle and Custom Applications

Oracle applications are a complete and modular set of enterprise applications, engineered to be cloud-ready. In Oracle Cloud, you'll find a broad range of software as a service (SaaS), platform as a service (PaaS), and infrastructure as a service (IaaS) applications. You can use these applications as part of a subscription-based service; there's no software license or hardware to buy and manage. Oracle handles all the supporting underlying technologies.

You can extend Oracle applications or even build your own custom cloud applications in Oracle Cloud. **Custom applications** are applications (such as a mobile application, a web page, a client application, or a server application) that you can integrate with Oracle Identity Cloud Service. By default, for security purposes, custom applications are trusted or confidential.

Oracle Identity Cloud Service leverages the power of OpenID Connect and OAuth to deliver a highly-scalable, multi-tenant token service for securing programmatic access to custom

applications by other custom applications, and for federated SSO and authorization integration with these applications:

- Use OAuth 2.0 to define authorization in Oracle Identity Cloud Service for your custom applications. OAuth 2.0 has an authorization framework, commonly used for third-party authorization requests with consent. Custom applications can implement both two-legged and three-legged OAuth flows.
- Use OpenID Connect to externalize authentication to Oracle Identity Cloud Service for your custom applications. OpenID Connect has an authentication protocol that provides Federated SSO, leveraging the OAuth 2.0 authorization framework as a way to federate identities in the cloud. Custom applications participate in an OpenID Connect flow.

Using the OAuth 2.0 and OpenID Connect standards provides the following benefits:

- Federated SSO between the custom application and Oracle Identity Cloud Service. Resource owners (users accessing the custom application) need a single login to access Oracle Identity Cloud Service plus all applications integrated. Oracle Identity Cloud Service handles the authentication and credentials itself, insulating custom applications. This capability is provided by OpenID Connect with OAuth 2.0.
- Authorization to perform operations on third-party servers with consent. Resource owners can decide at runtime whether the custom applications should have authorization to access data or perform tasks for them. This capability is provided by OAuth 2.0.

About Enterprise Applications

Enterprise applications are web applications that require App Gateway to integrate with Oracle Identity Cloud Service for authentication and authorization purposes.

Enterprise applications work similarly to confidential applications if you configure the **Client Configuration** and **Resource Server Configurations** section under **OAuth Configuration** tab.

To configure an enterprise application to work with App Gateway for authentication and authorization purposes you need to know the following information about your web application:

- The web application's base URL. For example, if a known URL of your application is `http://myapp.internal.example.com:3266/myapp/private/home`, then the base URL is `http://myapp.internal.example.com:3266`.
- The list of resources of your web application. For example, if your web application exposes the following URLs: functionalities A to Z in the following format `/myapp/private/funcA` to `/myapp/private/funcZ`, a home page `/myapp/private/home`, a logout URL `/myapp/logout`, an about page `myapp/public/about`, and an index page `/myapp/index`, then the list of all resources of your web application is:
  - URLs from `/myapp/private/funcA` to `/myapp/private/funcZ`
  - `/myapp/private/home`
  - `/myapp/logout`
  - `/myapp/public/about`
  - `/myapp/index`
- For each resource, define which resources require the user to be authenticated, which don't require user authentication, and which resource represents the log out action. Below are examples of authenticated and non-authenticated resources:

  - Resources from `/myapp/private/funcA` to `/myapp/private/funcZ`, and `/myapp/private/home` require the user to be authenticated.
  - `/myapp/logout` logs the user out.
  - Both `myapp/public/about` and `/myapp/index` are public resources and don't require the user to be authenticated.
- For each resource, define who can access which resource and which HTTP Method will be allowed or denied access. For example, you can define that all members of group **Employees** are allowed access to make `GET` and `POST` HTTP requests to resource `/myapp/private/home`, only members of group **MyGroupA** can access `/myapp/private/funcA`, and only users accessing from within network perimeter **IntranetIPs** can access resources from `/myapp/private/funcB` to `/myapp/private/funcZ`.
- Identify URL patterns that apply to your list of resources. In the previous example, the URL pattern `/myapp/private/.*` matches all the application's functionality URLs and the home page URL. All these URLs may require the same kind of authentication.

About the Relationship Between Oracle Identity Cloud Service and Applications

In Oracle Identity Cloud Service, each custom application is represented by an **application template**. This configuration template is used to define the identity, access, and configuration information that Oracle Identity Cloud Service requires to communicate with the application.

When you purchase an Oracle application, an instance of the application is created in your identity domain and appears in the **Oracle Cloud Services** page automatically.

For a custom application, you must configure Oracle Identity Cloud Service so that it can communicate with the application. You use an application wizard to create a custom application. By doing so, in your identity domain, you add the information that Oracle Identity Cloud Service uses to communicate with the application. See Add Applications. Custom applications are shown in the **Applications** page.

You can use Oracle Identity Cloud Service to grant users access to applications in two ways:

- Directly: Assigning users to the applications
- Indirectly: Assigning groups to the applications. Any users who are members of the groups are granted access to the applications.

In addition to granting users and groups access to Oracle applications, you can grant users and groups access to entitlements within applications. For example, you use Oracle Identity Cloud Service to grant John Doe and Jane Doe access to Oracle Java Cloud Service. You want John Doe to have administrator privileges for Oracle Java Cloud Service, but Jane Doe to have only user privileges.

Each entitlement in an Oracle application is represented by an **application role**. So by assigning John Doe to the application administrator role of Oracle Java Cloud Service, he can access this Oracle Cloud service and he can function as an administrator within it.

Architecture: SAML and Provisioning Integration with Oracle Identity Cloud Service

Oracle Identity Cloud Service is enabled to integrate with the provisioning and SAML integration making it simple and convenient to use.

Topics:

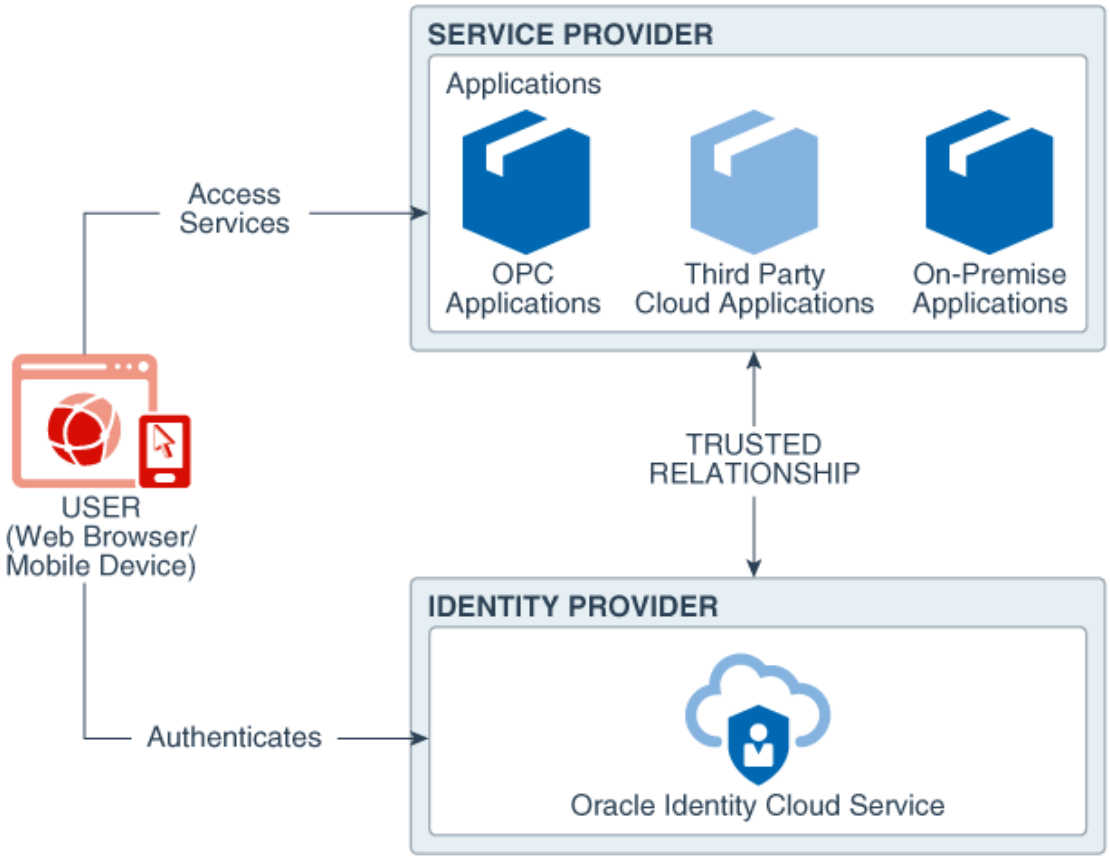
Architecture Diagram Defining Oracle Identity Cloud Service and SAML Integration

Security Assertion Markup Language (SAML) is an XML-based system for authentication and authorization between a Service Provider (SP) and an Identity Provider (IdP). It is a standard single sign-on (SSO) format where authentication information is exchanged through digitally signed XML documents.

In this process, the SP trusts the IdP to authenticate users and in return, the IdP generates an authentication assertion suggesting that a particular user has been authenticated.

The following architecture diagram illustrates the integration between Oracle Identity Cloud Service and SAML.

Figure 5-1 Architecture Diagram: Oracle Identity Cloud Service and SAML Integration



SAML Authentication includes three important roles:

- Oracle Identity Cloud Service as the Identity Provider
- Pre-integrated Cloud Services as the Service Provider
- User (Web Browser/ Mobile Device)

Oracle Identity Cloud Service SAML integration currently supports the following features:

- SP initiated Web SSO
- IdP initiated Web SSO
- SP initiated Single Logout
- IDP initiated Single Logout

Oracle Identity Cloud Service provides a generic SAML template to connect to all custom SAML applications.

All applications listed in the Oracle Identity Cloud Service Application Catalog are partially configured templates. These applications are created and maintained by Oracle and contain pre-built integrations with major cloud services making them simple and convenient. Using these applications, you can configure SSO and configure other functionalities in a standard format.

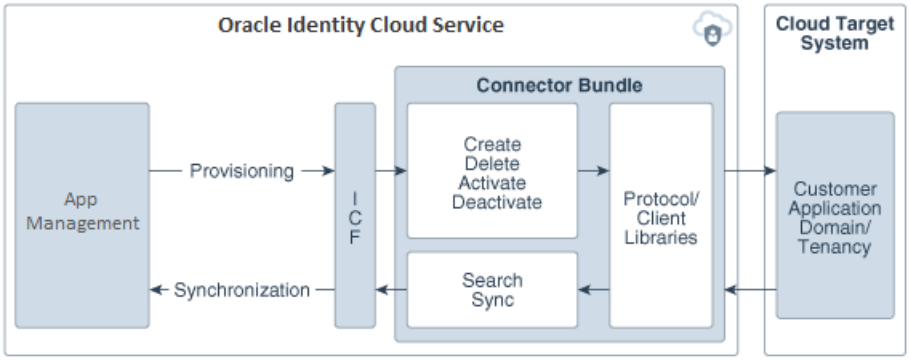
Architecture Diagram Defining Oracle Identity Cloud Service and Provisioning Integration

The customer application is configured as a managed application of Oracle Identity Cloud Service. Through provisioning operations performed on Oracle Identity Cloud Service, accounts are created and updated on the target system for Oracle Identity Cloud Service Users. Through synchronization, account data that is created and updated directly on the target system is pulled into Oracle Identity Cloud Service and stored for the corresponding Oracle Identity Cloud Service Users.

The Identity Connector Framework (ICF) is a component that is required to use identity connectors. ICF is distributed with Oracle Identity Cloud Service and doesn't require configuration or modifications.

The following architecture diagram illustrates the integration between Oracle Identity Cloud Service and Provisioning.

Figure 5-2 Architecture Diagram: Oracle Identity Cloud Service and Provisioning Integration



During provisioning:

1. App Management calls ICF.
2. ICF sends a CREATE request to the Connector Bundle.
3. The Connector Bundle calls the target API for provisioning.
4. The target API accepts provisioning data from the Connector Bundle.
5. The target API carries out the required operation on the target system.
6. The target API then sends the response from the target system to the Connector Bundle.

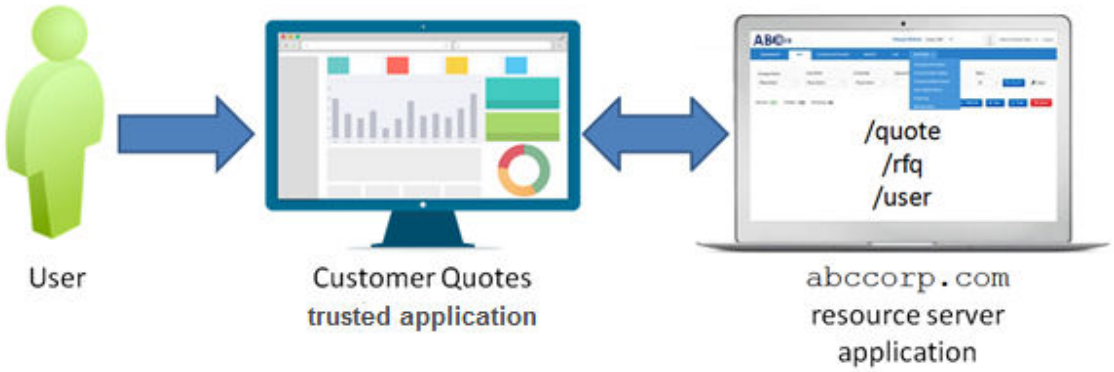
During synchronization:

1. A synchronization job calls ICF.
2. ICF sends a SEARCH request to the Connector Bundle.
3. The Connector Bundle calls the target API for the synchronization operation.
4. The API extracts user records that match the synchronization criteria.
5. The API sends these records through the Connector Bundle and ICF to the synchronization job, which syncs the records with Oracle Identity Cloud Service.

Each record fetched from the target system is compared with the user accounts that are already provisioned to Oracle Identity Cloud Service Users. If a match is found, then the update made to the account from the target system is copied to the user account in Oracle Identity Cloud Service. If a match isn't found, then the user ID of the record is compared with the user ID of each Oracle Identity Cloud Service User. If the user ID matches, then data in the target system record is used to provision the user account to the Oracle Identity Cloud Service User.

Use Case: Adding Applications

To understand how to add custom applications in Oracle Identity Cloud Service, read this use case.



For this use case, a user accesses the Customer Quotes trusted application. This trusted application is a client application that makes REST API calls to the `abccorp.com` resource server application. A **resource server application** is a third-party application that provides services that a trusted application can consume on behalf of the user.

For this example, the `abccorp.com` resource server application is a financial application that contains REST APIs that can be used to make a quote (`/quote`), request for a quote (`/rfq`), or get information about the user (`/user`).

When the user accesses the Customer Quotes trusted application, the application makes REST API calls to the `abccorp.com` resource server application on behalf of the user. In this example, the user doesn't communicate directly with the `abccorp.com` application.

Because the Customer Quotes application performs actions on behalf of the user, the application needs access to the `/quote`, `/rfq`, and `/user` REST APIs available with the `abccorp.com` application. To make these REST API calls, the Customer Quotes application might ask for the user's consent. This consent can come at any time that the Customer Quotes application calls for these REST APIs in the `abccorp.com` application.

The user logs in to Oracle Identity Cloud Service and accesses the Custom Quotes application, through single sign-on, by using OAuth 2.0 and OpenID Connect, as this is a way of federating identities in the cloud. Because the Customer Quotes application is authorized on behalf of the user to make the `/quote`, `/rfq`, and `/user` REST API calls to the `abccorp.com` application, the user can use the Customer Quotes application to make a quote, request for a quote, and get information about the user. Any additional actions that the user wishes to perform through the Customer Quotes application won't be allowed.

To build this workflow, you create and activate two custom applications in Oracle Identity Cloud Service:

- The `abccorp.com` resource server application. This application has REST APIs (resources) that other applications, such as the Customer Quotes application, can access. In this

example, the user doesn't access the resource server application directly, but indirectly through the Customer Quotes application.

You register resources of the `abccorp.com` resource server application. **Application resources** are API calls that are authorized by Oracle Identity Cloud Service. For this example, the application resources are the `/quote`, `/rfq`, and `/user` REST APIs. For security and auditing purposes, you can specify whether the user must give consent to access these resources.

- The Customer Quotes trusted application. The user uses this application to access the REST APIs of the `abccorp.com` application.

  When you create this custom application, you want to generate an authorization code for the user when the user logs in to Oracle Identity Cloud Service. The **authorization code** is then sent to the Customer Quotes application to retrieve an access token. The **access token** contains all the rights that the user has to access the resource server application. For this example, these rights include making a quote, requesting a quote, and retrieving information about the user.

  Because the access token's lifetime is short, you may want to generate a refresh token. A **refresh token** is a secure mechanism to obtain a new access token when the current access token expires. This way, the Customer Quotes application can access the APIs of the `abccorp.com` application without asking for user consent again.

See Add Applications, Activate Applications, and Deactivate Applications for more information about creating and activating custom applications in Oracle Identity Cloud Service.

About Adding Applications

Learn about the various applications available and how to add them in Oracle Identity Cloud Service.

Topics:

Add Applications

You can add Oracle Applications or Custom Applications in Oracle Identity Cloud Service, if you are assigned to either the identity domain administrator role or the application administrator role.

See Add or Remove a User Account from an Administrator Role for information about assigning users to administrator roles.

You can add the following types of custom applications in Oracle Identity Cloud Service:

- App Catalog application: Add an application from the Application Catalog, which contains pre-configured application templates.
- SAML application: Accessed by multiple users and hosted in a secure and protected place (server). Create a Security Assertion Markup Language (SAML) application that supports SAML for single sign on. This allows users to single sign-on (SSO) into your software as a service (SaaS) applications that support SAML for SSO.
- Mobile application: Hosted directly on the resource owner's browser, machine, or mobile device. An example of this type of application is an Android or iPhone application. A mobile application can run in multiple environments outside of your control. Since these environments are not trusted, this type of application has reduced integration options.
- Confidential application: Accessed by multiple users and hosted in a secure and protected place (server). The application uses OAuth 2.0. Applications that can protect their OAuth client id and client secret are called confidential applications
- Enterprise application: Web applications that require App Gateway to integrate with Oracle Identity Cloud Service for authentication purposes. Oracle App Gateway passes HTTP headers to the application after authenticating and authorizing user's access.

Tip:

- You can access the Onboarding Applications infographic to see how to add custom applications in Oracle Identity Cloud Service.
- You can access the Integrating a Custom Client Application tutorial to see how to integrate a custom client application with Oracle Identity Cloud Service.
- You can access the Integrating a Custom Resource Server Application tutorial to see how to integrate a custom resource server application with Oracle Identity Cloud Service.
- See Manage Oracle Identity Cloud Service App Gateways for information on integrating an enterprise application with Oracle Identity Cloud Service.

Add a Confidential Application

You can use Oracle Identity Cloud Service to add a confidential application. Confidential applications run on a protected server.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications.**
2. Click **Add.**

3. In the **Add Application** page, click **Confidential Application**.
4. In the **Add Confidential Application** wizard's **Details** page, in the **App Details Section**, use the following table to configure application details and the display settings.

Option	Description
Name	Enter a name for the confidential application. You can enter up to 125 characters. For applications with lengthy names, the application name appears truncated in the **My Apps** page. Consider keeping your application names as short as possible.
Description	Enter a description for the confidential application. You can enter up to 250 characters.
Application Icon	Click **Upload** to add an icon that represents the application. This icon appears next to the name of the application on the My Apps page and the Applications page.
Application URL	Enter the URL (HTTP or HTTPS) where the user is redirected after a successful login. This value is also known as the SAML RelayState parameter. HTTPS format is suggested. HTTP should only be used for testing purposes.
Custom Login URL	In the **Custom Login URL** field, specify a custom login URL. However, if you are using a default login page provided by Oracle Identity Cloud Service, then leave this field blank.
Custom Logout URL	In the **Custom Logout URL** field, specify a custom logout URL. However, if you are using a default login page provided by Oracle Identity Cloud Service, then leave this field blank.
Custom Error URL	This is an optional field. Enter the error page URL to which a user has to be redirected, in case of a failure. If not specified, the tenant specific Error page URL will be used. If both the error URLs are not configured, then the error will be redirected to the Oracle Identity Cloud Service Error Page (/ui/v1/error). When a user tries to use social authentication (ex: Google, Facebook, and so on) for logging into Oracle Identity Cloud Service, the callback URL must be configured in the Custom Error URL field. Social providers need this callback URL to call Oracle Identity Cloud Service and send the response back after social authentication. The provided callback URL is used to verify whether the user exists or not (in the case of first time social login), and display an error if the social authentication has failed. This is the URL where the callback is sent with social registration user details, if a successful logged-in social user account does not exist in Oracle Identity Cloud Service.

Option	Description
Linking callback URL	This is an optional field. Enter the URL that Oracle Identity Cloud Service can redirect to after linking of a user between social providers and Oracle Identity Cloud Service is complete. When you create a custom app using Oracle Identity Cloud Service custom SDK and integrate with Oracle Identity Cloud Service Social Login, the custom app needs to have the Linking callback URL which can be redirected after linking of the user between social provider and Oracle Identity Cloud Service is complete.
Tags	Click **Add Tag** to add tags to your confidential applications to organize and identify them. See Adding Tags to an Application.
Display in My Apps	Select the check box if you want the confidential application to be listed for users on their My Apps pages. In this case you need to configure the application as a resource server. When you select the **Display in My Apps** check box in applications, the app is then visible in the **My Apps** page, but selecting this check box doesn't enable or disable SSO to the app. The flag to enable or disable SSO comes from the app template. Use the Oracle Identity Cloud Service REST APIs to update this flag. You cannot set the SSO flag from the UI. See REST API for Oracle Identity Cloud Service.
User can request access	Select the check box if you want end users to be able to request access to the app from their **My Apps** page by clicking **Add Access.** If self service is not enabled, users won't see the **Add Access** button.

5. At the top of the **Add Confidential Application** wizard's **Details** page, click **Next**.

   A confirmation message indicates that the application has been added in a deactivated state.

6. On the **Add Confidential Application** wizard's **Client** page:
   - To skip configuring authorization for your application at this time:
     a. Click **Skip for later**.
     b. Continue with step 7 below.
   - To configure authorization information for your application now:
     a. Click **Configure this application as a client now**.
     b. In the **Authorization** and **Token Issuance Policy** sections that open, use the following table to fill in the information.

Option	Description
Resource Owner	Use when the resource owner has a trust relationship with the confidential application, such as a computer operating system or a highly privileged application, because the confidential application must discard the password after using it to obtain the access token.
Client Credentials	Use when the authorization scope is limited to the protected resources under the control of the client or to the protected resources registered with the authorization server. The client presents its own credentials to obtain an access token. This access token is either associated with the client's own resources, and not a particular resource owner, or is associated with a resource owner for whom the client is otherwise authorized to act
JWT Assertion	Use when you want to use an existing trust relationship expressed as an assertion and without a direct user approval step at the authorization server. The client requests an access token by providing a user JSON web token (JWT) assertion or a third-party user JWT assertion and client credentials. A **JWT assertion** is a package of information that facilitates the sharing of identity and security information across security domains.
SAML2 Assertion	Use when you want to use an existing trust relationship expressed as a SAML2 assertion and without a direct user approval step at the authorization server. The client requests an access token by providing a user SAML2 assertion or a third-party user SAML2 assertion and client credentials. A **SAML2 assertion** is a package of information that facilitates the sharing of identity and security information across security domains.
Refresh Token	Select this grant type when you want a refresh token supplied by the authorization server, and then use it to obtain a new access token. Refresh tokens are used when the current access token becomes invalid or expires and don't requiring the resource owner to reauthenticate.
Authorization Code	Select this grant type when you want to obtain an authorization code by using an authorization server as an intermediary between the client application and resource owner. An **authorization code** is returned to the client through a browser redirect after the resource owner gives consent to the authorization server. The client then exchanges the authorization code for an access (and often a refresh) token. Resource owner credentials are never exposed to the client.

Option	Description
Implicit	If the application can't keep client credentials confidential for use in authenticating with the authorization server, then select this check box. For example, your application is implemented in a web browser using a scripting language such as JavaScript. An access token is returned to the client through a browser redirect in response to the resource owner authorization request (rather than an intermediate authorization).
Device Code	Select the **Device Code** grant type if the client doesn't have the capability to receive requests from the OAuth Authorization Server, for example, it cannot act as an HTTP server such as game consoles, streaming media players, digital picture frames, and others. In this flow, the client obtains the user code, device code, and verification URL. The user then accesses the verification URL in a separate browser to approve the access request. Only then can the client obtain the access token using the device code.
TLS Client Authentication	Select the **TLS Client Authentication** grant type to use the client certificate to authenticate with the client. If a token request comes with an X.509 client certificate and the requested client is configured with the **TLS Client Authentication** grant type, the OAuth service uses the Client_ID in the request to identify the client and validate the client certificate with the certificate in the client configuration. The client is successfully authenticated only if the two values match. **Optional** For added security, before enabling the **TLS Client Authentication** grant type, enable and configure OCSP validation and import a trusted partner certificate. See Enable X.509 Certificate Authentication and Import a Trusted Partner Certificate.
Allow non-HTTPS URLs	Select this check box if you want to use HTTP URLs for the **Redirect URL, Logout URL,** or **Post Logout Redirect URL** fields. For example, if you are sending requests internally, want a non-encrypted communication, or want to be backward-compatible with OAuth 1.0, then you can use an HTTP URL. Also, select this check box when you are developing or testing your application and you may not have configured SSL. This option is provided as a convenience and is not recommended for production deployments.

Option	Description
Redirect URL	Enter the application URL where the user is redirected after authentication. **Note:** Provide an absolute URL. Relative URLs are not supported.
Logout URL	Enter the URL where the user is redirected after logging out of the confidential application.
Post Logout Redirect URL	Enter the URL where you want to redirect the user after logging out of the application.
Client Type	Select the client type. The available client types are **Trusted** and **Confidential**. Choose **Trusted** if the client can generate self signed user assertions. Then, to import your signing certificate that the client uses to sign its self-signed assertion, click **Import.**
Allowed Operations	• Select the **Introspect** check box if you want to allow access to a token introspection end point for your application. If the confidential application can't keep client credentials confidential for use in authenticating with the authorization server, then select this check box. For example, your confidential application is implemented in a web browser using a scripting language such as JavaScript. An access token is returned to the client through a browser redirect in response to the resource owner authorization request (rather than an intermediate authorization code). • Select the **On behalf Of** check box if you want to ensure that access privileges can be generated from the user's privileges alone. This allows the client application to access endpoints to which the user has access, even if the client application by itself would not normally have access.
ID Token Encryption Algorithm	Choose one of the available content encryption algorithms so that id tokens passed through third parties, such as browsers, are encrypted. The default is `none`. • You can also set the parameter from `/App` endpoint by passing `idTokenEncAlgo` with the appropriate algorithm. • You can set this parameter for confidential and enterprise applications. (You cannot set it for mobile applications.)

Option	Description
Bypass Consent	If enabled, this attribute overwrites the **Require Consent** attribute for all the scopes configured for the application, and then no scope will require consent.
Allowed Client IP Address	Prerequisites: • Enabling the Allowed Client IP Address feature. This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service. • To use this feature you, you must first have a network perimeter configured. See Add a Network Perimeter. Set one of the following options for the allowed client IP address. • **Anywhere** - The token request is allowed from anywhere. There is no perimeter. • **In one or more of these network perimeters** - Select the network perimeters so that a token request is only allowed from them.
Authorized Resources	Select one of the following options to allow a client application to access authorized resources: • **All** – Access any resource within a domain (All). See Accessing All Resources. • **Tagged** – Access any resource with matching tags (Tagged). See Accessing Resources With Matching Tags. • **Specific** – Access only those resources where an explicit association between the client and the resource (Specific) exists. See Accessing Resources With Specific Scopes. **Note:** The option to define an authorized resource is available to only confidential applications. Mobile applications don't have the option to define a trust scope. See Account Trust Scope for additional scope information as well as request and response examples for use with the Oracle Identity Cloud Service REST APIs.

Option	Description
Tags	**Note:** Tags are available only when you select the **Tagged** option. It remains hidden for the other two Authorized Resource options. Click **Tagged** to enable your confidential application to access tags from other applications. See Adding Tags to an Application.
Resources	If you want your application to access APIs from other applications, then click **Add** in the **Token Issuance Policy** section of the **Add Confidential Application** page. Then, in the **Add Scope** window, select the applications that your application references. **Note:** You can delete scopes by clicking the **x** icon next to the scope. However, you can't delete scopes that are protected.
Grant the client access to Identity Cloud Service Admin APIs	Click **Add** to enable your confidential application to access Oracle Identity Cloud Service APIs. In the **Add App Role** window, select the application roles that you want to assign to this application. This enables your application to access the REST APIs that each of the assigned application roles can access. For example, select **Identity Domain Administrator** from the list. All REST API tasks available to the identity domain administrator will be accessible to your application. You can delete the application roles by clicking the **x** icon for the row of the required application role. **Note:** You can't delete protected application roles. See Apps/App Roles endpoint for a complete list of which endpoints each application role can access.

7. Click **Next**.
8. On the **Add Confidential Application** wizard's **Resources** page:

- To specify that no resources of your confidential application will be protected by OAuth 2.0, or to protect them at a later time:
  a. Click **Skip for later**.
  b. Continue with step 9 below.
- To protect resources for your application now, and to make the application visible on the **My Apps** page:
  a. Click **Configure this application as a resource server now**.
  b. Use the following table to fill in the information in the **Configure application APIs that need to be OAuth protected** section that opens.

Option	Description
Access Token Expiration	Define how long (in seconds) the access token associated with your confidential application remains valid.
Is Refresh Token Allowed	Select this check box if you want to use the refresh token that you obtain when using the Resource Owner, Authorization Code, or Assertion grant types.
Refresh Token Expiration	Define how long (in seconds) the refresh token, which is returned with your access token and is associated with your confidential application, remains valid.
Primary Audience	Enter the primary recipient where the access token of your confidential application is processed.
Secondary Audiences	Enter the secondary recipients where the access token of your confidential application is processed, and click **Add**. The secondary recipient appears in a tabular column, and the **Protected Column** allows you to know whether the secondary audience is protected or not..
Add (Allowed Scopes)	To specify which parts of other applications that you want your application to access, click this button to add those scopes to your confidential application. Applications must interact securely with external partner or confidential applications. Also, applications from one Oracle Cloud service must interact securely with applications in another Oracle Cloud service. Each application has application scopes that determine which of its resources are available to other applications.

9. Click **Next**.
10. On the **Add Confidential Application** wizard's **Web Tier Policy** page, click **Next**.
11. On the **Add Confidential Application** wizard's **Authorization** page, if you want Oracle Identity Cloud Service to control access to the application based on grants to users and groups, select the **Enforce Grants as Authorization** check box.
    - **Selected**: Users can access the application only if you assign or grant access.
    - **Unselected**: Any authenticated user has access to the application.
12. Click **Finish.**

The application has been added in a deactivated state.

13. Record the **Client ID** and **Client Secret** that appear in the **Application Added** dialog box.

    To integrate with your confidential application, use this ID and secret as part of your connection settings. The **Client ID** and **Client Secret** are equivalent to a credential (for example, an ID and password) that your application uses to communicate with Oracle Identity Cloud Service.

14. Click **Close.**

    The new application's details page is displayed.

15. At the top of the page, to the right of the application name, click **Activate**.

16. In the **Activate Application?** dialog box, click **Activate Application**.

Add Enterprise Applications

Enterprise Applications use App Gateway as a reserve proxy protecting web applications by restricting unauthorized network access to them.

Topics:

Add an Enterprise Application

An enterprise application enables you to secure web applications that are protected by the Oracle App Gateway.

To add an enterprise application in Oracle Identity Cloud Service, you need to configure the list of application resources (web application's URLs or URL patterns), create an authentication policy for each resource, and create an authorization policy for each resource. For each authentication policy, you define an authentication method, and header variables for App Gateway to include in the request before forwarding the request to the application.

1. Sign into the Identity Cloud Service console as an application administrator.
2. Expand the **Navigation Drawer**, click **Applications**, and then click **Add**.
3. In the **Add Application** page, click **Enterprise Application**.
4. In the **Details** pane of the **Add Enterprise Application** page, provide a name for the application, enter the application URL, complete all other fields as necessary, and then click the **Next >** icon.

> **Note:**
>
> The application URL is the URL that you want users to use to access your enterprise application. Use the host name and port number of the App Gateway. If you have multiple instances of App Gateway, then use the host name and port number of the load balancer.

5. In the **OAuth Configuration** pane, click the **Next >** icon.

   Use the **OAuth Configuration** pane to configure the enterprise application to act as a confidential application by providing client and resource server configurations.

6. In the **SSO Configuration** pane, click **Finish**.

   You configure the **Resources**, **Authentication Policy** and **Authorization Policy** sections under the **SSO Configuration** pane later.

7. Click **Activate**, and then click **OK** in the **Confirmation** widow to activate the application.

Configure Resources

You can create resources individually by adding one resource for each of your application's URLs, or use regular expression to create a resource which represents a collection of URLs for your application.

A resource represents a URL or URL Pattern for which you want to restrict access or intend to gve anyone to access. You need the list of resources of your application. See About Enterprise Applications.
Policy mapping is hierarchical in App Gateway. So, order of the resources defined is very important. See the following example:

If the user is accessing a resource `/myapp/logout.html`, and we have authentication policy in below order:

1. `/.*` (public)
2. `/.*/logout.html` (Form+logout)

Then policy match stops at point #1 (`/.*`) and the same policy shall be applied which is "public" in this case.

Similarly, if user is accessing a resource `/myapp/logout.html` and we have authentication policy in below order.

1. `/.*/logout.html` (Form+logout)
2. `/.*` (public)

In this case. policy match stops at point 1 (`/.*/logout.html`) and same policy shall be applied which is "Form+logout".

Something else to be aware of is that applications which do their own login integrations can run into problems when their integrations accessed static resources during login, but the resources were not made public. This causes the login process to fail. To avoid this happening, you should use the `public` authentication method for your public static resources such as CSS, JavaScript, image files as follows:

- Group all public static resources together, for example under `/myapp/public/resources`.
- State that these directories should use the `public` authentication method using a regex such as `/myapp/public/.*`.

To configure resources:

1. In the **Application Details** page, click the **SSO Configuration** tab of your enterprise application page, expand the **Resources** section, and then click **Add** to add a resource.
2. In the **Add Resource** dialog, provide a name for the resource and the resource URL. If you want to use a regular expression as the resource URL value, then select **Regex**, so that App Gateway evaluates the **Resource URL** value as a pattern.

   For example, if you want to protect the application endpoint `http://myapp.internal.example.com:3266/private/home`, you can enter `/private/home` as the value for **Resource URL**. If you want to protect any page under the `/private` context, then enter `/private/.*` as value for **Resource URL**, and select **Regex**.

   See Use Regular Expressions.

Configure an Authentication Policy

Create an authentication policy for each resource you created for your enterprise application.

An authentication policy defines which authentication method to use to protect your enterprise application's resources, and whether App Gateway will add header variables to the request it forwards to the application.

1. In the **SSO Configuration** tab of your enterprise application page, expand the **Authentication Policy** section, and then click **Add** under **Managed Resources**.
2. In the **Add Resource** window, select the resource for which you want to configure an authentication policy from the list of resources that you created in the **Resources** section.
3. Use the following table to define the **Authentication Method** for the resource you have selected:

Table 5-1 Authentication Methods

Authentication Method	Description
Basic Auth	The **Basic Auth** method performs HTTP Basic authentication. If the request doesn't contain an `Authentication Basic` header, then user's browser will prompt for credentials. The credentials sent in the `Authentication Basic` header is validated in Oracle Identity Cloud Service.
Basic Auth+Logout	This method is used to protect the application's resource (URL) that represents the application's log out process. When App Gateway intercepts a request to this resource, the HTTP logout process is initiated. This process deletes any HTTP session cookie created by the **Basic Auth+Session** authentication method. After the logout process finishes, App Gateway forwards the user browser to the requested application's resource. Note that the HTTP logout process doesn't clear any credentials cached by the browser in the current browser session and then the user may not be prompted again for later requests.
Basic Auth+Session	Works the same as **Basic Auth**. After the credential is validated, it creates an HTTP session cookie (`ORA_OCIS_CG_BA_SESSION`).

Table 5-1 (Cont.) Authentication Methods

Authentication Method	Description
Form or Access Token	In this authentication method, App Gateway delegates credentials collection and validation to Oracle Identity Cloud Service. If an `Authorization Bearer` header is present in the request, then the authentication is similar to a resource server flow. If a `user-agent` header is present, then a user browser flow takes place. The user browser flow redirects the user browser to Oracle Identity Cloud Service for credentials collection and validation, and then creates an OAuth session cookie (`ORA_OCIS_CG_SESSION_*`). If an `Authorization session` header is present in the request and the OAuth session cookie is missing or invalid, then the usual OAuth login flow is suppressed and a `401` HTTP error code will be returned along with a `WWW-Authenticate: Bearer error="invalid_session"` header. This is used by applications that may trigger an unwanted login when their requests contain a `user-agent` header, but not an `Authorization Bearer` header, allowing them to handle re-authentication themselves.
Form+Logout	This method is used to protect the application's resource (URL) that represents the application's log out process. This resource's URL doesn't need to be exposed by the application, as App Gateway redirects the user browser to Oracle Identity Cloud Service's OAuth logout endpoint (`/oauth2/v1/userlogout`), instead of forwarding the request to the application URL. In the **Add Resource** window, the **Post-Logout URL** is the URL which App Gateway redirects the user browser after signing the user out. You can also provide a **Post-Logout State** parameter value to be used by the post-logout URL page of the application.
Multitoken	Performs authentication based on the contents of the `Authorization` header of the request: • If the request contains an `Authorization Basic` header, then App Gateway handles this authentication as **Basic Auth**. • If the request contains an `Authorization Bearer` or `Authorization Session` header, App Gateway handles this authentication as **Form or Access Token**. • If the Authorization header is missing or has any other value, then a `401 Unauthoized` HTTP error is returned.
Multitoken+Fallthrough	Same as **Multitoken**, but if the `Authorization` header is not `Basic`, `Bearer`, or `session`, then instead of presenting the `401 Unauthoized` HTTP error, request App Gateway acts as authentication method was **Basic Auth**.

Table 5-1 (Cont.) Authentication Methods

Authentication Method	Description
Anonymous	• If a valid OAuth session cookie is present, then the headers configured in the authentication policy are added to the request and the request is forwarded to the application. • If the OAuth session cookie is missing or expired, works the same as the **Public** authentication method. In this case, a `REMOTE_USER` header with value `anonymous` is added to the request. For both options, the headers configured in the authentication policy are added to the request, but authentication is not performed.
Public	No authentication is performed. The request is forwarded to the application as is.
Unsupported	This method always returns `500 Not Supported` HTTP error code. For example, you can use this method to disable access to a protected URL that is available in the application but you don't want users to access it.

4. The authentication method you selected in the previous step is valid for all HTTP Methods (`GET`, `HEAD`, `DELETE`, `PUT`, `OPTIONS`, `CONNECT`, `POST`, or `PATCH`). If you want to specify different authentication methods for HTTP methods (for example, the **Form + Access Token** authentication method for the `GET` HTTP method and the **Multitoken** authentication method for the `POST` HTTP method), then you can do so by using the **Authentication Method Overrides** menu. Select the **HTTP Method**, and then the **Authentication Method** you want. If you need to override more than one HTTP method, then repeat this step multiple times.
5. If you want to add an header variable to the request so that App Gateway forwards it to the application, click the plus **+** icon for **Headers**, provide the name, and then either select the value for the header variable from the list of user attributes, enter a fixed value, or provide an expression. To add more than one header variable, click the **+** icon for **Headers** multiple times.

   For example, let's suppose the application requires a header variable named `USERLOGGEDIN` to be present in every request so that the applications knows the ID of user signed in to Oracle Identity Cloud Service. You need to add one header variable, enter `USERLOGGEDIN` for the **Name** field, and then either select **User Name** from the drop down list or enter `$subject.user.userName` for **Value**.

   > **Note:**
   >
   > You can select a user attribute from the drop down menu or provide an expression using any attribute from Oracle Identity Cloud Service's SCIM user schema as header variable value. See Supported Header Value Expressions for Authentication Policies.

   In the **Configure Authentication Policy for this application** section, if App Gateway is configured in SSL mode (HTTPs), then confirm that **Require Secure Cookies** is selected. This flag sets the secure header to avoid cookies being used in non-secure HTTP

communication. If App Gateway is configured in non-SSL mode (HTTP), then deselect **Require Secure Cookies**.

For security reasons, make sure the **Disable Audience Validation** check box isn't selected. The audience validation check box is used to ensure the token has been issued by App Gateway's known issuer, in this case Oracle Identity Cloud Service. If you disable audience validation, App Gateway won't validate the audience of the token, which makes the application vulnerable to attacks.

Configure an Authorization Policy

Create an authorization policy for each resource in your enterprise application and define the conditions in which users are allowed or denied access to the resource.

Prerequisite

Enable Authorization Policy. This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service.

> **Note:**
>
> Although the **Authorization Policy** section appears during enterprise application configuration, the ability for App Gateway and Oracle Identity Cloud Service to validate authorization must be turned on for you. If you don't file a Service Request, your App Gateway won't perform authorization verification despite you having configured the **Authorization Policy** section.

> **Note:**
>
> Authorization policies only work for resources that you protect with **Form or Access Token** authentication method in an authentication policy. If your resource is protected with any other authentication method, App Gateway doesn't perform authorization check when users try to access the resource using a web browser.

Authorization policies define under what conditions users are allowed or denied access to application resources. When App Gateway intercepts an HTTP request to a resource endpoint, App Gateway verifies whether the enterprise application in Oracle Identity Cloud Service contains authorization policies for the resource. If so, then App Gateway verifies whether the HTTP request matches one of the rules configured to allow or deny access.

For example, you can configure an allow rule to allow all members of the **Employees** group to access the `/myapp/private/home` resource, and configure a deny rule to deny access to this resource for users authenticated by the **My External SAML IDP** identity provider.

1. In the **SSO Configuration** tab of your enterprise application page, expand the **Authorization Policy** section.
2. In the **Allow Rules** section, click **Add**, specify a **Rule Name**, and then complete the following fields.

Table 5-2 Add Allow Rule Options

Conditions	Description
If the resource is	Select one of the resources configured in the enterprise application.
And the HTTP Method is	Select the HTTP Methods associated with this rule. The rule will be valid only for the selected HTTP Methods.
And if the user is authenticated by	Select the identity providers that are active in Oracle Identity Cloud Service. If the user is signed in using one of these identity providers, then App Gateway allows access to the resource. **Local IDP** refers to users authenticated by Oracle Identity Cloud Service.
And is a member of these groups	Select Oracle Identity Cloud Service's groups. If the signed in user is a member of one of the selected groups, then App Gateway allows access to the resource.
And is not one of these users	Select Oracle Identity Cloud Service users. If the signed in user is not one of the selected users, then App Gateway allows access to the resource.
And the user's client IP address is	Select the IP address range the HTTP request are made from. • **Anywhere**: App Gateway doesn't validate the IP address from where the HTTP request was made. • **In one or more of these network perimeters**: Select this option, and then select the network perimeters associated with this rule. If the IP address from where the HTTP request was made is specified in one of the network perimeters, then Access Gateway allows access to the resource.
And access is	Select a time of the day (**From** and **To**), select which days of the week, and then the timezone in which the rule is valid. App Gateway allows access to the resource only if the HTTP Request is made within the period configured.

All the conditions configured for an allow rule must be met so that App Gateway can perform the action configured for the rule.

3. In the **Actions** section of the **Add Allow Rule** window, click **Add** for **Headers**, enter name for the HTTP header and then select a user attribute as value. Repeat this step for all headers you want to configure for this rule.

   If the user is allowed access to the resource, App Gateway adds these header variables with the corresponding values to the HTTP request before forwarding the request to the application.

4. Click **Add** to add the allow rule.

5. In the **Deny Rules** section, click **Add Deny Rule**, specify a **Rule Name**, and then complete the following fields.

Table 5-3 Add Deny Rule Options

Conditions	Description
If the resource is	Select one of the resources configured for the enterprise application.
And the HTTP Method is	Select the HTTP Methods to associate with this rule.

Table 5-3 (Cont.) Add Deny Rule Options

Conditions	Description
And if the user is authenticated by	Select identity providers that are active in Oracle Identity Cloud Service. If the user is signed in using one of these identity providers, then App Gateway denies access to the resource. **Local IDP** refers to users authenticated by Oracle Identity Cloud Service.
And is a member of these groups	Select Oracle Identity Cloud Service groups. If the signed in user is member of one of the selected groups, then App Gateway denies access to the resource.
And is not one of these users	Select the Oracle Identity Cloud Service users. If the signed in user is not one of the selected users, then App Gateway denies access to the resource.
And the user's client IP address is	Select the IP address range the HTTP request are made from. • **Anywhere**: App Gateway doesn't validate the IP address from where the HTTP request was made. • **In one or more of these network perimeters**: Select this option, and then select the network perimeters to associate with this rule. If the IP address from where the HTTP request was made is specified as one of the network perimeters, then Access Gateway denies access to the resource.
And access is	Select a time of the day (**From** and **To**), select which days of the week, and then the timezone in which the rule is valid. App Gateway denies access to the resource if the HTTP Request is made within the period configured.

All the conditions configured for a deny rule must be met so that App Gateway can perform the action configured for the rule.

6. In the **Actions** section of the **Add Deny Rule** window, select the action App Gateway must perform when a deny rule condition matches the resource's HTTP request.
   - **None**: App Gateway redirects the user browser to the URL you've set in the **Custom Error URL** parameter of the enterprise application. If this parameter has no value, then App Gateway redirects the user browser to the URL set in the **Error URL** parameter of the **Session Settings**.
   - **Logout**: Logs the user out from Oracle Identity Cloud Service.
7. Click **Add** to add the deny rule.
8. In the **Settings** section, select **Time to live** in minutes to define for how long App Gateway caches any authorization policy evaluation that has been performed.

   By caching these policy evaluation, App Gateway doesn't need to communicate with Oracle Identity Cloud Service in subsequent HTTP request made by the user for the same resource.

Use Regular Expressions

Use regular expressions (regex) to define a URL pattern which represents more than one URL of your enterprise application and for which you can apply the same authentication policy and the same authorization policy.

Create a list of all URLs for your application, and then to define URL patterns that map similar URLs, in which you want to define common authentication and authorization policies.

The authorization engine of App Gateway supports all tokens available to create regular expressions, such as Character Classes, Anchors, Escaped Characters, Group & References, Lookaround, Quantifiers & Alternation, and Substitution.

Below is a list of common operators supported by App Gateway's authorization engine:

Table 5-4 Common Regex Operators Supported by App Gateway Authorization Engine

Operator	Description	Example
Match-any-character Operator (`.`)	The period character represents this operator.	`a.b` matches any three-character string beginning with `a` and ending with `b`
Match-zero-or-more Operator (*)	This operator repeats the smallest possible preceding regular expression as many times as necessary (including zero) to match the pattern	`a*` matches any string made up of zero or more `a`'s. In another example, `fo*` has a repeating `o`, not a repeating `fo`. Hence, `fo*` matches `f`, `fo`, `foo`, and so on.
Match-one-or-more Operator (+)	This operator is similar to the match-zero-or-more operator except that it repeats the preceding regular expression at least once.	`ca+r` matches `car` and `caaaar`, but not `cr`
Match-zero-or-one Operator (?)	This operator is similar to the match-zero-or-more operator except that it repeats the preceding regular expression once or not at all.	`ca?r` matches both `car` and `cr`, but nothing else.
Negate (^)	Negate an expression.	`^a` matches any character except `a`
Grouping Operators (`(...)`)	Regex treats expressions inside the parenthesis just as mathematics and programming languages treat a parenthesized expression as a unit. The expressions are processed before the expression outside the parenthesis.	`f(a|b)a` matches `faa` and `fba`, which means the operation `a|b` is processed before the rest.
Alternation Operator (`|`)	Alternatives match one of a choice of regular expressions: if you put the character(s) representing the alternation operator between any two regular expressions `a` and `b`, the result matches the union of the strings that `a` and `b` match.	`foo|bar|quux` would match any of `foo`, `bar` or `quux` As another example, `(` and `)` are the open and close-group operators, then `fo(o|b)ar` would match either `fooar` or `fobar`. On the other hand, `foo|bar` would match `foo` or `bar`

Table 5-4 (Cont.) Common Regex Operators Supported by App Gateway Authorization Engine

Operator	Description	Example
List Operators ([...] and [^ ...])	A matching list matches a single character represented by one of the list items. An item is a character, a character class expression, or a range expression. Non matching lists are similar to matching lists except that they match a single character not represented by one of the list items.	`[ab]` matches either `a` or `b`. `[ad]*` matches the empty string and any string composed of just `a`'s and `d`'s in any order. As a non matching example, `[^ab]` matches any character except `a` or `b`
Range Operator (-)	Represents those characters that fall between two elements in the current collating sequence.	`[a-f]` represents all the characters from `a` through `f` inclusively.
Digit (\d)	Matches any digit character (0-9).	Same as `[0-9]`
Not Digit (\D)	Matches any character that is not a digit character (0-9).	Same as `[^0-9]`
Escape (\)	Makes the next character in the expression means the character itself but not an operator.	`\.` means period, not the Match-any-character operator.

Example 5-1 Use of Regular Expression

For example, if you want to allow only authenticated users access for any page of the application that starts with `my` and are under the path `/mybank`, then you can use the regular expression `/mybank/my.*`

The dot (.) and the star (*) together represents any sequence of zero or more consecutive characters after the prefix `my`.

In this example, the URLs `/mybank/myCredits` and `/mybank/myDebits` match the `/mybank/my.*` pattern, but `/mybank/about` doesn't.

Supported Header Value Expressions for Authentication Policies

When you configure enterprise application's authentication policies, you can add header variables to requests forwarded to the application, by selecting a user attribute from a list of pre defined user attributes, or by entering an expression.

In the header **Value** field for Authentication Policies, you can provide a simple literal string or an attribute identifier instead of selecting the user attribute from the drop down list. If you use an attribute identifier, App Gateway attempts to replace the attribute identifier by the value of the attribute after authentication happens.

The following types of attribute identifiers are supported by authentication policies:

- **Application**: This attribute identifier accesses the information of the enterprise application registered in Oracle Identity Cloud Service.
  Format: `$subject.client.<attr>`
- **User**: This attribute identifier accesses information of the user signed in to Oracle Identity Cloud Service.

Format: `$subject.user.<attr>`

- **Request**: This attribute identifier accesses request information.
  Format: `$request.<attr>`

For user attribute scope, App Gateway supports any simple top-level attribute in the JSON Response from `/admin/v1/Users` such as `string`, `boolean`, or `int` values.

App Gateway also supports user extension attributes as header value expressions for authentication policies, using the following format `$subject.user.urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:<attributeName>`, and custom attributes using the following format `$subject.user.urn:ietf:params:scim:schemas:idcs:extension:custom:User:<customAttributeName>`

Table 5-5 Example of User Attribute Scope Names and Return Values

Attribute Name	Header Value Expression	Description
Full Name	`$subject.user.name`	The user's full name.
User Name	`$subject.user.userName`	The user's login username.
Emails	`$subject.user.emails` Other types of emails also supported: `$subject.user.emails.recovery`, `$subject.user.emails.other`, `$subject.user.emails.home`, and `$subject.user.emails.work`.	The user's primary email address.
Phone Numbers	`$subject.user.phoneNumbers` Other types of phone numbers supported: `$subject.user.phoneNumbers.mobile`, `$subject.user.phoneNumbers.home`, and `$subject.user.phoneNumbers.work`.	The user's phone number.
Addresses	`$subject.user.addresses`	The user's mailing address.
Groups	`$subject.user.groups`	A list of comma-separated group names to which the user is assigned to through direct or indirect membership.
idcsCreatedBy	`$subject.user.idcsCreatedBy`	The display name of the user or application who created this resource.
idcsLastModifiedBy	`$subject.user.idcsLastModifiedBy`	The display name of the user or application who modified this resource.
Department	`$subject.user.urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:department`	The user's department.
Employee Number	`$subject.user.urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employeeNumber`	The user's employee number.

Example of supported values for request attribute scope:

Table 5-6 Example of Request Attribute scope names and supported values

Attribute Name	Header Value Expression	Description
policy_appname	`$request.policy_appname`	Returns the name of the enterprise application registered in Oracle Identity Cloud Service.
policy_name	`$request.policy_name`	Returns the policy name of the specific policy matched for the request.
policy_res	`$request.policy_res`	Returns the resource URL pattern matched for the request. The format is: "<type>:<pattern>" Example: `text:/my/resource` or `regex:/my/resource/.*`
policy_action	`$request.policy_action`	Returns the HTTP Method (`GET`, `POST`, etc) used to access the requested resource.
res_host	`$request.res_host`	Returns the host name from the original Request.
res_port	`$request.res_port`	Returns the port number from the original Request.
res_type	`$request.res_type`	Returns the protocol (HTTP or HTTPS) of the original Request.
res_url	`$request.res_url`	Returns the full requested URL.

Default Headers and Cookies App Gateway Adds to the Request

By Default App Gateway adds header variables and cookies to any request forwarded to a protected enterprise applications. The following is a list of these headers and cookies and their respective values.

Headers

Header Name	Description	Authentication Method Usage
idcs_service_url	The value of this header is your Oracle Identity Cloud Service's base URL. For example, `https://idcs-tenant.identity.oraclecloud.com`	Used by all authentication method.
idcs_cloudgate_id	The **Client ID** value for the App Gateway registered in Oracle Identity Cloud Service.	Used by all authentication method.
idcs_client_id	The **Client ID** value for the App Gateway registered in Oracle Identity Cloud Service.	App Gateway adds this header to the request forwarded to the enterprise application if the resource is protected by **Anonymous** or **Public** authentication methods.

Header Name	Description	Authentication Method Usage
idcs_authn_method	The authentication method configured in the enterprise application's authentication policy. Value depending on the authentication method used: • If resource is protected by **Anonymous** authentication method, then value is `anonymous`. • If resource is protected by **Form or Access Token** authentication method, then value is `oauth`. • If resource is protected by **Basic Auth** or **Basic Auth+ Session** authentication methods, then value is `http`. • If resource is protected by **Multitoken** authentication method, then value is `multitoken`. • If resource is protected by **Multitoken+Fallthrough** authentication method, then value is `multitoken` or `fallthrough` depending if the authorization header is a known value or not.	App Gateway adds this header to the request forwarded to the enterprise application if the resource is protected by any authentication method except **Public**.
idcs_authn_strength	Identifies if the user authentication has happened in 1 or 2 steps. If the user has signed in with Oracle Identity Cloud Service using their credentials only, then the authentication strenght is `1`. If the user has signed in using multi-factor authentication, then authentication level is `2`.	App Gateway adds this header to the request forwarded to the enterprise application if the resource is protected by any authentication method except **Public** and **Anonymous**.
remote_user	Username of the user signed in to Oracle Identity Cloud Service. If the resource is protected by **Anonymous** authentication method, then the value of this header is `anonymous`.	App Gateway adds this header to the request forwarded to the enterprise application if the resource is protected by any authentication method except **Public**.
idcs_remote_user	Username of the user signed in to Oracle Identity Cloud Service. If the resource is protected by **Anonymous** authentication method, then the value of this header is `anonymous`.	App Gateway adds this header to the request forwarded to the enterprise application if the resource is protected by any authentication method except **Public**.

Header Name	Description	Authentication Method Usage
idcs_remote_user_mappingattr	The Oracle Identity Cloud Service user schema attribute used to identify the signed in user. For example, `userName`.	App Gateway adds this header to the request forwarded to the enterprise application if the resource is protected by any authentication method except **Public** and **Anonymous**.
idcs_session_id	The session ID value Oracle Identity Cloud Service creates after user signs in.	App Gateway adds this header to the request forwarded to the enterprise application if the resource is protected by **Form or Access Token** or **Basic Auth+ Session** authentication method.
idcs_user_assertion	Value of the identity token issued by Oracle Identity Cloud Service.	App Gateway adds this header to the request forwarded to the enterprise application if the resource is protected by **Form or Access Token** authentication method.
idcs_user_display_name	Value of the `displayname` attribute of the user signed in with Oracle Identity Cloud Service.	App Gateway adds this header to the request forwarded to the enterprise application if the resource is protected by any authentication method except **Public** and **Anonymous**.
idcs_user_id	Value of the unique identifier attribute of the user signed in with Oracle Identity Cloud Service.	App Gateway adds this header to the request forwarded to the enterprise application if the resource is protected by any authentication method except **Public** and **Anonymous**.
idcs_user_tenant_name	Oracle Identity Cloud Service tenant name.	App Gateway adds this header to the request forwarded to the enterprise application if the resource is protected by any authentication method except **Public** and **Anonymous**.

Cookies

Cookie Name	Description	Authentication Method Usage
ORA_OCIS_CG_SESSION_<idcs-tenant>_<aapgateway_host>	After the user authenticates with Oracle Identity Cloud Service, App Gateway sets this cookie to the request forwared to the application. The cookie name is composed by `ORA_OCIS_CG_SESSION` prefix, concatenated with Oracle Identity Cloud Service's tenant name, and sufixed with the App Gateway's **Host** value.	App Gateway adds this header to the request forwarded to the enterprise application if the resource is protected by **Form or Access Token** authentication method.

Configure Authorized Resources

Authorized resources define the way a client can access the resources in a confidential application.

Topics:

Accessing All Resources

The **All** authorized resource option enables the client to access any resource within a domain.

Select **All** to allow your application to request an access token for trusted or confidential client using the scope `urn:opc:resource:consumer::all`. This option provides a wide scope. The access token in the response contains the audience `urn:opc:resource:scope:account` and the scope `urn:opc:resource:consumer::all`, which gives access to any of the services that are in the same domain without requiring explicit association with target services.

Use only the `urn:opc:resource:consumer::all` scope in the request. An invalid scope error is returned if you attempt to include both the `urn:opc:resource:consumer::all` scope and another scope in the same request, such as `urn:opc:idm:__myscopes__`.

In the account mode, clients can get token for any specific resource provided either `urn:opc:resource:consumer::all` or the specific resource is added in the allowed scopes

Apart from the scope defined above, you can also specify fine-grained scope as follows:

- `urn:opc:resource:consumer:paas::read`
- `urn:opc:resource:consumer:paas:stack::all`
- `urn:opc:resource:consumer:paas:analytics::read`

> **Note:**
>
> The requested scope should always exist and match, either directly or hierarchically, the client's defined allowed scopes to allow the client access to the resource.

For example, a client uses the `urn:opc:resource:consumer:paas:analytics::read` scope in its request for access to a resource. If the scope directly matches an allowed scope defined, then in the returned access token the audience is `urn:opc:resource:scope:account` and the scope is `urn:opc:resource:consumer:paas:analytics::read`.

If the allowed scope defined by the client is `urn:opc:resource:consumer:paas::read`, then the client is allowed to access the resource hierarchically if the client requests one of the following scopes:

- `urn:opc:resource:consumer:paas::read`
- `urn:opc:resource:consumer:paas:analytics::read`

However, if the requested scope is `urn:opc:resource:consumer:paas:analytics::write` with a different qualifier, then the client isn't allowed access to the resource.

> **Note:**
>
> The **All** option doesn't provide access to the Oracle Identity Cloud Service admin APIs. You must continue to use the scope `urn:opc:idm:__myscopes__` to access the admin APIs.

To generate a refresh token in addition to the access token, use the scope `urn:opc:resource:consumer::all offline_access` in the request.

Access Resources With Matching Tags

The **Tagged** authorized resource option enables the client to access any resource with matching tags.

Request an access token using the trusted or confidential client and request the scope `urn:opc:resource:consumer::all`. The access token in the response contains the audience `urn:opc:resource:scope:tag=<base64 encoded JSON>` and the scope `urn:opc:resource:consumer::all`, which gives access to Resource Apps that have tags that match the allowed tags specified in the Client App.

In the tags mode, clients can get token for any specific resource provided either the client has matching tags with the resource and `urn:opc:resource:consumer::all` or the specific resource is added in the allowed scopes.

Select **Tagged** to enable your confidential application to access tags from other applications.

When you select **Tagged**, you can choose scopes from an OPC application that aren't specific, such as `urn:opc:resource:consumer`.

To select scopes:

1. Select **Tagged**.
2. Select **Add Scope** under **Resources**.
3. Select `urn:opc:resource:consumer` on the **Select Scope** page and click **>**.
4. Select the OPC scopes that you want to add and provide a named qualifier, such as read and write to each of the scopes. You can edit these qualifiers dynamically.
5. Click **Add**.

   The scopes appear under **Resources**.

In addition to using the `urn:opc:resource:consumer::all` scope, you can also specify the following fine-grained scopes:

- `urn:opc:resource:consumer:paas::read`
- `urn:opc:resource:consumer:paas:stack::all`
- `urn:opc:resource:consumer:paas:analytics::read`

> **Note:**
>
> The requested scope should always exist and match, either directly or hierarchically, the client's defined allowed scopes to allow the client access to the resource.

For example, a client uses the `urn:opc:resource:consumer:paas:analytics::read` scope in its request for access to a resource. If the scope directly matches an allowed scope defined, then in the returned access token the audience is `urn:opc:resource:scope:tag=<base64 encoded JSON>` and the scope is `urn:opc:resource:consumer:paas:analytics::read`.

For client allowed tags `color:green` and `color:blue`, the sample JSON is as follows:

```
{"tags":[{ "key":"color","value":"green"},{"key":"color","value":"blue"} ]}
```

If the allowed scope defined by the client is `urn:opc:resource:consumer:paas::read`, then the client is allowed to access the resource hierarchically if the client requests one of the following scopes:

- `urn:opc:resource:consumer:paas::read`
- `urn:opc:resource:consumer:paas:analytics::read`

However, if the requested scope is `urn:opc:resource:consumer:paas:analytics::write`, then the client isn't allowed access to the resource, since that isn't one of the allowed scopes defined by the client.

Access Resources With Specific Scopes

The **Specific** authorized resource option enables the client to access only those resources where an explicit association between the client and the resource exists.

Leave **Specific** selected (the default) to allow your application to acquire an access token with permissions based on an explicit association between the client and target services. Then, use the **Add** button to select the applications that your application references.

The **Specific** option is assigned by default to confidential applications created prior to Oracle Identity Cloud Service version 17.4.2. To use the **All** option, you must open the application from the Oracle Identity Cloud Service administration console, and then select **All**.

Add a Mobile Application

You can use Oracle Identity Cloud Service to add a mobile application. Mobile applications use OAuth 2.0 and they cannot maintain the confidentiality of their client secrets.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. Click **Add**.
3. In the **Add Application** page, click **Mobile Application**.
4. In the **App Details** section of the Add Mobile Application page, use the following table to configure the application details.

Option	Description
Name	Enter a name for the mobile application. You can enter up to 125 characters. For applications with lengthy names, the application name appears truncated in the **My Apps** page. Consider keeping your application names as short as possible.
Description	Enter a description of the mobile application. You can enter up to 250 characters.
Application Icon	Click **Upload** to add an icon that represents the application. This icon appears next to the name of the application on the My Apps page and the Applications page.
Custom Login URL	In the **Custom Login URL** field, specify a custom login URL. However, if you are using a default login page provided by Oracle Identity Cloud Service, then leave this field blank.
Custom Logout URL	In the **Custom Logout URL** field, specify a custom logout URL. However, if you are using a default login page provided by Oracle Identity Cloud Service, then leave this field blank.
Custom Error URL	This is an optional field. Enter the error page URL to which a user has to be redirected, in case of a failure. If not specified, the tenant specific Error page URL will be used. If both the error URLs are not configured, then the error will be redirected to the Oracle Identity Cloud Service Error Page (/ui/v1/error). When a user tries to use social authentication (ex: Google, Facebook, and so on) for logging into Oracle Identity Cloud Service, the callback URL must be configured in the Custom Error URL field. Social providers need this callback URL to call Oracle Identity Cloud Service and send the response back after social authentication. The provided callback URL is used to verify whether the user exists or not (in the case of first time social login), and display an error if the social authentication has failed.
Linking callback URL	This is an optional field. Enter the URL that Oracle Identity Cloud Service can redirect to after linking of a user between social providers and Oracle Identity Cloud Service is complete. When you create a custom app using Oracle Identity Cloud Service custom SDK and integrate with Oracle Identity Cloud Service Social Login, the custom app needs to have the Linking callback URL which can be redirected after linking of the user between social provider and Oracle Identity Cloud Service is complete.
Tags	Click **Add Tag** to add tags to your mobile applications to organize and identify them. See Adding Tags to an Application.

Option	Description
Display in My Apps	Select the check box if you want the mobile application to be listed for users on their My Apps pages. In this case you need to configure the application as a resource server. When you select the **Display in My Apps** check box in applications, the app is then visible in the **My Apps** page, but selecting this check box doesn't enable or disable SSO to the app. The flag to enable or disable SSO comes from the app template. Use the Oracle Identity Cloud Service REST APIs to update this flag. You cannot set the SSO flag from the UI. See REST API for Oracle Identity Cloud Service.
User can request access	Select the check box if you want end users to be able to request access to the app from their **My Apps** page by clicking **Add Access.** If self service is not enabled, users won't see the **Add Access** button.

5. Click **Next**. A message confirms that the application has been added in deactivated state.
6. In the **Authorization** and **Accessing APIs from Other Application** sections of the Add Mobile Application page, use the following table to configure application details.

Option	Description
Allowed Grant Types	Select the check box for the grant types that this application is allowed to use when requesting validation. • Select the **Refresh Token** grant type when you want a refresh token supplied by the authorization server, and then use it to obtain a new access token. Refresh tokens are used when the current access token becomes invalid or expires and don't requiring the resource owner to reauthenticate. • Select the **Authorization Code** check box when you want to obtain an authorization code by using an authorization server as an intermediary between the client application and resource owner. An **authorization code** is returned to the client through a browser redirect after the resource owner gives consent to the authorization server. The client then exchanges the authorization code for an access (and often a refresh) token. Resource owner credentials are never exposed to the client. • Select the **Implicit** check box if the application can't keep client credentials confidential for use in authenticating with the authorization server. An access token is returned to the client through a browser redirect in response to the resource owner authorization request (rather than an intermediate authorization code). • Select the **Device Code** grant type if the client doesn't have the capability to receive requests from the OAuth Authorization Server, for example, it cannot act as an HTTP server such as game consoles, streaming media players, digital picture frames, and others. In this flow, the client obtains the user code, device code, and verification url. The user then accesses the verification url in a separate browser to approve the access request. Only then can the client obtain the access token using the device code.
Allow non-HTTPS URLs	Select this check box if you want to use HTTP URLs for the **Redirect URL**, **Logout URL**, or **Post Logout Redirect URL** fields. For example, if you are sending requests internally, want a non-encrypted communication, or want to be backward-compatible with OAuth 1.0, then you can use an HTTP URL. Also, select this check box when you are developing or testing your application and you may not have configured SSL. This option is provided as a convenience and is not recommended for production deployments.

Option	Description
Redirect URL	Enter the application URL where the user is redirected after authentication.
Logout URL	Enter the URL where the user is redirected after logging out of the application.
Post Logout Redirect URL	Enter the URL where you want to redirect the user after logging out of the application.
Allowed Operations	• Select the **Introspect** check box, if you want to allow access to a token introspection end point for your application. • Select the **On behalf Of** check box, if you want to ensure that access privileges can be generated from the user's privileges alone, so that a client application can access endpoints to which the user has access, even if the client application by itself would not normally have access.
Bypass Consent	If enabled, this attribute overwrites the **Require Consent** attribute for all the scopes configured for the application, and then no scope will require consent.
Resources	If you want your application to access APIs from other applications, then click **Add Scope** in the **Token Issuance Policy** section of the **Add Mobile Application** page. Then, in the **Add Scope** window, select the applications that your application will reference.
Grant the client access to Identity Cloud Service Admin APIs	Click **Add** to enable your mobile application to access Oracle Identity Cloud Service APIs. In the **Add App Role** window, select the application roles that you want to assign to this application. This enables your application to access the REST APIs that each of the assigned application roles can access. For example, select **Identity Domain Administrator** from the list. All REST API tasks available to the identity domain administrator will be accessible to your application. You can delete the application roles by clicking the **x** icon for the row of the required application role. **Note:** You can't delete protected application roles. See Apps/App Roles endpoint for a complete list of which endpoints each application role can access.

7. Click **Next**.
8. If you want Oracle Identity Cloud Service to control access to the application based on grants to users and groups, select the **Enforce Grants as Authorization** check box.

Select this check box if you want users to access only the application that you assigned or granted access to. If the check box is not selected, any authenticated user has access to the application regardless of the assignment status.

9. Click **Finish**. A message confirms that the application has been added in deactivated state. To activate your application see Activating Applications.
10. Note the Client ID that appears in the **Application Added** window. This information also appears on the Configuration tab in the Details section for the application. To integrate with your application, use this ID as part of your connection settings. Because a mobile application runs on a mobile device, Oracle Identity Cloud Service does not generate a Client Secret for this type of application.
11. Click **Close**.

Add a SAML Application

Create a Security Assertion Markup Language (SAML) application and grant it to users so that your users can single sign-on (SSO) into your SaaS applications that support SAML for SSO.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. Click **Add**.
3. In the Add Application window, click **SAML Application**.
4. In the **App Details** section of the Add SAML Application page, provide values for the following fields:
   - In the **Name** field, enter a name for the application.

     For applications with lengthy names, the application name appears truncated in the **My Apps** page. Consider keeping your application names as short as possible.
   - In the **Description** field, enter 250 or fewer characters to provide a description of the application.
   - Click **Upload** to add an icon for your application.
   - In the **Application URL / Relay State** field, enter a value which will be sent to the SAML SP as the SAML RelayState parameter.
   - In the **Custom Login URL** field, specify a custom login URL. However, if you are using a default login page provided by Oracle Identity Cloud Service, then leave this field blank.
   - In the **Custom Logout URL** field, specify a custom logout URL. However, if you are using a default login page provided by Oracle Identity Cloud Service, then leave this field blank.
   - In the **Custom Error URL** field, enter the error page URL to which a user has to be redirected, in case of a failure. This is an optional field. However, if not specified, the tenant specific Error page URL will be used. If both the error URLs are not configured, then the error will be redirected to the Oracle Identity Cloud Service Error Page (/ui/v1/error).

     When a user tries to use social authentication (ex: Google, Facebook, and so on) for logging into Oracle Identity Cloud Service, the callback URL must be configured in the Custom Error URL field. Social providers need this callback URL to call Oracle Identity Cloud Service and send the response back after social authentication. The provided callback URL is used to verify whether the user exists or not (in the case of first time social login), and display an error if the social authentication has failed.

- In the **Linking callback URL** field, enter the URL that Oracle Identity Cloud Service can redirect to after linking of a user between social providers and Oracle Identity Cloud Service is complete. This is an optional field.

  When you create a custom app using Oracle Identity Cloud Service custom SDK and integrate with Oracle Identity Cloud Service Social Login, the custom app needs to have the Linking callback URL which can be redirected after linking of the user between social provider and Oracle Identity Cloud Service is complete.

- Click **Add** to add **App Links** that are associated with the application. The **Link** window appears. App Links are services such as Mail or Calendar that are offered by applications such as Google or Office 365.

  In the **Link** window:

  a. In the **Name** field, enter the **App Link** name.

  b. In the **Link** field, enter the URL used to access the application.

  c. Click **Upload** to upload an icon.

  d. Select **Visible** check box if you want your application to appear automatically on each user's **My Apps** page.

  > **Note:**
  >
  > Selecting this check box does not enable or disable SSO into the application.

  e. Click **Add**.

  The App Link information appears in the **App Details** section of the application page.

  To remove an **App Link**, select the row, and then click **Remove**.

  > **Note:**
  >
  > There is a delay (a few seconds) between clicking **Remove** and the App no longer appearing on the My Apps page. App Link deletion (and grants related to those App Links) is asynchronous. Wait a few seconds for the asynchronous task to remove the App and its grants before trying **My Apps** again.

5. In the **Tags** section of the Add SAML Application page, click **Add Tag** to add tags to your SAML application to organize and identify it. See Adding Tags to an Application.

6. In the **Display Settings** sections of the Add SAML Application page, make the following selections:

   - Select **Display in My Apps** check box to specify whether you want the SAML App to be listed on the My Apps page.

     When you select the **Display in My Apps** check box in applications, the app is then visible in the **My Apps** page, but selecting this check box doesn't enable or disable SSO to the app.

     The flag to enable or disable SSO comes from the app template. Use the Oracle Identity Cloud Service REST APIs to update this flag. You cannot set the SSO flag from the UI. See REST API for Oracle Identity Cloud Service.

- Select the **User can request access** check box if you want the app to be listed in the Catalog. This option allows end users to request access to the app from their **My Apps** page by clicking **Add** and then selecting the app from the Catalog.

> **Note:**
>
> Don't forget to activate the application so that users can request access.

7. Click **Next** to configure SSO details for the SAML application.
8. In the **General** section of the SSO Configuration page, define the following:
   - **Entity ID**: Enter a globally unique name for a SAML entity. It usually takes a URL of an identity provider or a service provider as a value.
   - **Assertion Consumer URL**: Enter the URL to which the SAML identity provider will send the SAML assertion. This URL must begin with either the HTTP or HTTPS protocol.
   - **NameID Format**: Select the type of format to use for the NameID. The service provider and the identity provider use this format to easily identify a subject during their communication.

     > **Note:**
     >
     > When you integrate Oracle Identity Cloud Service with MS SharePoint app based on WS Fed 1.1 protocol, the following options are not available in the NameID format: **Persistent**, **Kerberos**, and **Transient**.

   - **NameID Value**: Select the NameID Value to identify the user that is logged in. The available options are **User Name** , the user's **Primary Email** address and **Expression**. When you select the **Expression** option, enter a path expression as a value in the text box. There is no character limit for the value, however, there are validation rules that are performed on the value for any invalid characters that cannot be mapped.

     Some examples of path expressions are listed below:

     – To send "home email" as the value of the assertion attribute, use `$(user.emails[type eq "home"].value)`.

     – To send users first name concatenated with last name as the assertion attribute, use `#concat($(user.name.givenName), $(user.name.familyName))`.

     – To send an account attribute called `SALARY` as the value of the assertion attribute, use `$(account.SALARY)`.

     – To include an attribute `department` from custom schema extension, use `$(user.urn:ietf:params:scim:schemas:idcs:extension:custom:User:department)`.

   - **Signing Certificate**: Upload the signing certificate that is used to encrypt the SAML assertion.

> **Note:**
>
> Some browsers show file paths prepended with `c:\fakepath\`. This behavior is a security feature of the browser and does not disrupt the upload process.

9. Expand **Advanced Settings** on the SSO Configuration page, and then use the following table to define a more fine-grained SAML configuration.

Option	Description
Signed SSO	Select **Assertion** to indicate that you want the SAML assertion signed. Select **Response** when you want the SAML authentication response signed.
Include Signing Certificate in Signature	Select the check box to include the signing certificate in the signature, for example, when the application requires that the signing certificate is sent along with the assertion.
Signature Hashing Algorithm	Select the type of signing algorithm that you want to use to sign the assertion or the response, either **SHA-256** or **SHA-1**. SHA-256 generates a fixed 256-bit hash. SHA-1 generates a 160-bit hash value known as a message digest. **Note:** In a FIPS enabled environment, set the **Signature Hashing Algorithm** to SHA-256, the only supported hashing algorithm, to avoid errors during SSO.
Enable Single Logout	Select to configure SAML single logout. Single logout enables a user to lot out of all participating sites in a federated session almost simultaneously. This check box is selected by default. Clear it if you do not want to enable single logout.
Logout Binding	Select whether the log out request is sent as a REDIRECT (transported using HTTP 302 status-code response messages) or a POST (transported in HTML form-control content, which uses a base-64 format). This list box appears only if you select the **Enable Single Logout** check box.
Single Logout URL	Enter the location (HTTP or HTTPS) where the log out request is sent. This field appears only if you select the **Enable Single Logout** check box.
Logout Response URL	Enter the location (HTTP or HTTPS) where the log out response is sent. This field appears only if you select the **Enable Single Logout** check box.
Encrypt Assertion	Select if you want to encrypt the assertion, and then define the encryption algorithm that you want to use and upload the encryption certificate.
Encryption Certificate	Click **Upload** to upload the encryption certificate that's used to encrypt the SAML assertion. This button appears only if you select the **Encrypt Assertion** check box.
Encryption Algorithm	Select which encryption algorithm you want to use to encrypt the SAML assertion. This list box appears only if you select the **Encrypt Assertion** check box.

Option	Description
Key Encryption Algorithm	Select which key encryption algorithm you want to use to encrypt the SAML assertion. This list box appears only if you select the **Encrypt Assertion** check box.

10. Expand **Attribute Configuration** on the SSO Configuration page to add user-specific and group-specific attributes to the SAML assertion. This is useful if your application uses user-specific or group-specific attributes, and you want to send that information as part of the SAML assertion.
11. Click the plus sign next to **Attributes**, and then use the following table to specify the user attribute that you want to include. User information in the attribute statement contains a list of attributes. Each attribute includes a name and a list of values (in the case of multiple attribute values). Each value includes a value and the format of the value.

Option	Description
Name	Enter the name of the SAML assertion attribute.
Format	Select the format of this SAML assertion attribute: **Basic**, **URI Reference**, or **Unspecified**. **Note:** When you integrate Oracle Identity Cloud Service with MS SharePoint app based on WS Fed 1.1 protocol, **Format** drop-down is replaced with **Namespace**.
Type	Select one of the options below to specify the value of the assertion attribute.: • **User Attribute** Select this option to choose one of the predefined list of user attributes or group attributes in the **Value** drop-down as the value of the assertion attribute. In order to specify group attributes, select **User Attribute** and in the **Value** field, select **Group Membership**. • **Expression/Literal** Select this option when you cannot use any of the predefined values in the **Value** drop-down. You can provide an expression in the **Value** text box to specify the value of the SAML assertion attribute. In order to specify group attributes, select **Expression/Literal** and specify an expression to fetch the groups. Example: The following expression specifies that the value of the SAML attribute should be the names of all the groups to which the user belongs: `$(user.groups[*].display)`.

Option	Description
Value	Select or enter the value to send as part of the assertion based on the **Type** that you have selected. When the type is **User Attribute**, you can select one of the predefined list of user attributes as the value of the assertion attribute. Select the **Group Membership** option in the drop-down if you want to send the users group membership as the value of the assertion attribute. The **Condition** and **Value** columns appear when you choose **Group Membership**. When the type is **Expression/Literal**, the value field is a text box and you can enter any path expression to specify what should be the value of the assertion attribute. Some examples of path expressions are listed below: • To send a list of literal values as the value of the assertion attribute, use `["value1", "value2", "value3"]`. • To send "home email" as the value of the assertion attribute, use `$(user.emails[type eq "home"].value)`. • To send users first name concatenated with last name as the assertion attribute, use `#concat($(user.name.givenName), $(user.name.familyName))`. • To send an account attribute called `SALARY` as the value of the assertion attribute, use `$(account.SALARY)`. • To include an attribute `department` from custom schema extension, use `$(user.urn:ietf:params:scim:schemas:idcs:extension:custom:User:department)`. • To send a literal value as the value of assertion, use `aLiteralValue`.
Condition	Select a condition from the drop-down to filter the group memberships. This field is enabled only when you select **User Attribute** as **Type** and **Group Membership** as **Value**. The available values are: **Equals**, **Starts with**, and **All Groups**.
Value	Enter the filter value to use when filtering the group memberships.

12. When you are creating SAML app from scratch rather than creating a preconfigured SAML app created from the App Catalog, the **Authentication and Authorization** section appears. The **Enforce Grants as Authorization** check box is selected by default. This check box enables users to access only the application that you assigned or granted access to. If the check box is selected, Oracle Identity Cloud Service can control access to the SAML application based on grants to users and groups. If the check box is not selected, any authenticated user has access to the application regardless of the assignment status.

13. To import the Identity Cloud Service signing certificate into your application, click **Download Signing Certificate** to first download the certificate file in PEM format. This certificate is used by the SAML application to verify that the SAML assertion is valid.
14. To import the Identity Cloud Service Identity Provider metadata into your application, click **Download Identity Provider Metadata** to first download the metadata file in XML format.

    The SAML application needs this information so that it can trust and process the SAML assertion that is generated by Identity Cloud Service as part of the federation process. This information includes, for example, profile and binding support, connection endpoints, and certificate information.

    To get the issuing Oracle Identity Cloud Service root certificate, see Obtaining the Root CA Certificate from Oracle Identity Cloud Service.

    To learn about the other options that can be used to access SAML metadata, see Access SAML Metadata.
15. Click **Finish**. The application is added in a deactivated state. To activate your application, see Activating Applications.

Upgrade a SAML Application

You can upgrade your SAML application if there is any upgradable change to your application.

If your SAML application has an update, you will see the **Upgrade** button visible in the UI. Click and upgrade the application.
After upgrading to a provisioning application, you need to configure the provisioning parameters and run a Full Sync similar to adding a new provisioning application.

See Adding an App Catalog Application for information on configuring the provisioning parameters.

About the App Catalog Application

Learn how to create an App Catalog application, enable provisioning and synchronization, import, and synchronize user accounts in this section. The App Catalog is a collection of partially configured application templates for popular Software as a Service (SaaS) applications, such as Amazon Web Services and Google Suite. Using the templates, you can define the application, configure SSO, and configure provisioning.

Topics:

- Add an App Catalog Application
- Enable Provisioning for an App Catalog Application
- Enable Synchronization for an App Catalog Application
- Import User Accounts from a Software as a Service Application
- Synchronize User Accounts
- Work with the Synchronization Failure Report

Add an App Catalog Application

Oracle creates and maintains the App Catalog, which is a collection of application templates, for you and provides step-by-step instructions on how to configure most of the popular Software as a Service (SaaS) applications, such as Amazon Web Services and Google Suite.

See Oracle Identity Cloud Service - Application Catalog to find the runbook for your application.

To add an App Catalog application:

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. Click **Add**.
3. In the **Add Application** window, click **App Catalog**.
4. Locate an application by choosing a **Category** (predefined by Oracle), searching for the application by entering a string that begins with the application name, or to see all applications, click **All**.
5. Click **Add**.
6. Complete the **App Details** by entering a **Name**, **Description**, and **Application URL**.

   > **Note:**
   >
   > For applications with lengthy names, the application name appears truncated in the **My Apps** page. Consider keeping your application names as short as possible.

7. In the **Custom Error URL** field, enter the error page URL to which a user has to be redirected, in case of a failure. This is an optional field. However, if not specified, the tenant specific Error page URL will be used. If both the error URLs are not configured, then the error will be redirected to the Oracle Identity Cloud Service Error Page (/ui/v1/error).

   When a user tries to use social authentication (ex: Google, Facebook, and so on) for logging into Oracle Identity Cloud Service, the callback URL must be configured in the Custom Error URL field. Social providers need this callback URL to call Oracle Identity Cloud Service and send the response back after social authentication. The provided callback URL is used to verify whether the user exists or not (in the case of first time social login), and display an error if the social authentication has failed.
8. In the **Linking callback URL** field, enter the URL that Oracle Identity Cloud Service can redirect to after linking of a user between social providers and Oracle Identity Cloud Service is complete. This is an optional field.

   When you create a custom app using Oracle Identity Cloud Service custom SDK and integrate with Oracle Identity Cloud Service Social Login, the custom app needs to have the Linking callback URL which can be redirected after linking of the user between social provider and Oracle Identity Cloud Service is complete.
9. In the **Tags** section, click **Add Tag** to add tags to your App Catalog application to organize and identify it. See Adding Tags to an Application.
10. In the **Display Settings** section, select **Display in My Apps**.

> **Important:**
>
> If you do not select **Display in My Apps**, the application does not display in the **My Apps** page.

When you select the **Display in My Apps** check box in applications, the app is then visible in the **My Apps** page, but selecting this check box doesn't enable or disable SSO to the app.

The flag to enable or disable SSO comes from the app template. Use the Oracle Identity Cloud Service REST APIs to update this flag. You cannot set the SSO flag from the UI. See REST API for Oracle Identity Cloud Service.

11. Select the **User can request access** check box, if you want the app to be listed in the **Catalog**. This option allows end users to request access to applications from their **My Apps** page by clicking **Add** and then selecting the app from the **Catalog**.

    Don't forget to activate the application so that users can request access.

12. Click **Next** and complete the **SSO Configuration**.

    - Click **Download Signing Certificate** to import the Oracle Identity Cloud Service signing certificate into your application. This certificate is used by the SAML application to verify that the SAML assertion is valid.
    - Click **Download Identity Provider Metadata** to import the Oracle Identity Cloud Service Identity Provider metadata into your application. The SAML application needs this information so that it can trust and process the SAML assertion that is generated by Oracle Identity Cloud Service as part of the federation process. This information includes, for example, profile and binding support, connection endpoints, and certificate information.

    To get the issuing Oracle Identity Cloud Service root certificate, see Obtaining the Root CA Certificate from Oracle Identity Cloud Service.

    To learn about the other options that can be used to access SAML metadata, see Access SAML Metadata.

13. Click **Finish**. The application has been added in deactivate state. To activate your application see Activating Applications.
14. Click **Applications**, locate the application that you just added, and activate it.
15. Select the application.
16. To assign users to the application, click **Users**.

The applications you assign to the user display on the **My Apps** page. Newly assigned applications and applications that a user has not yet accessed appear first in the application list and have an asterisk icon in the application tile. The icon appears on the tile until the user accesses the application.

Enable Provisioning for an App Catalog Application

User provisioning and synchronization are an important aspect of application management. Provisioning allows you to manage the lifecycle of accounts in applications like creating and deleting accounts using Oracle Identity Cloud Service. For example, when you grant the user access to an application such as Google Suite, then this user account is automatically created in Google Suite. This allows you to quickly add new users to multiple applications and de-

provision users from those applications instantly when they change roles or leave your organization.

You can enable and configure provisioning for App Catalog applications either when adding the app or later when modifying it. When you enable provisioning by selecting the option, the following steps appear:

1. Configure Connectivity

   Configure your app connectivity by providing values for the respective fields and by testing connectivity.

2. Configure Attribute Mapping

   Using **Attribute Mapping** you can map Oracle Identity Cloud Service attributes to the attributes in your application account. You can verify the existing default mapping and, if necessary, change mappings by selecting appropriate values from the drop-down list for the required user attribute. You can add rows to map missed attributes and delete rows to exclude duplicate attribute mapping. To add a new attribute for provisioning, click **Add Row**, specify the attributes in the **User** and your application account columns, and then click **OK**. For example, if you want to add the **External ID** field, enter `$(user.externalId)` in the **User** column, and then select the corresponding field from the drop-down list in the applications account column.

   > **Note:**
   >
   > As a best practice, don't share allowed values between app templates. There must be a one-to-one mapping between an app template and an allowed value, since an associated allowed value is deleted when an app template is deleted.

3. Select Provisioning Operations

   Any app that supports provisioning and synchronization can be an authoritative app. If authoritative sync is configured, using Oracle Identity Cloud Service, you can automatically create, modify, delete, and activate or disable users based only on the corresponding data from the authoritative application. However, the regular provisioning operations are not allowed while authorization sync is enabled.

   When authoritative sync is enabled, the following actions happen automatically:

   - If a user is not present in Oracle Identity Cloud Service, then the user is automatically created.
   - If an authoritative synced user is deleted from the application, then the user is also deleted from Oracle Identity Cloud Service.
   - If attributes of an authoritative synced user are modified, then the attributes for the user are also modified in Oracle Identity Cloud Service.

   When **Authoritative Sync** is enabled, then the provisioning operations aren't permitted from Oracle Identity Cloud Service to the target application. To manage user accounts in the application using provisioning, clear the **Authoritative Sync** check box. The following provisioning operations appear:

   - **Create Account**: Select to create an account when the app is granted to the user.
   - **De-activate Account**: Select to disable this account. To activate the account, clear the check box.
   - **Delete Account**: Select to delete the account in the app when the Oracle Identity Cloud Service user is deleted.

> **Important:**
>
> When you configure the connection between your app and Oracle Identity Cloud Service, check and verify any pre-filled user name and password field entries as these may not be the credentials to access your application.

To configure provisioning and synchronization for your application, follow the specific runbook for the application. See Oracle Identity Cloud Service - Application Catalog.
After you have enabled Provisioning, you can perform the following actions:

- Assign users or groups to your App Catalog application to start the user provisioning process for your application. See Assigning Users to Custom Applications and Assigning Groups to Custom Applications.
- Enable and configure synchronization. To enable and configure synchronization, see Enable Synchronization for an App Catalog Application.

Enable Synchronization for an App Catalog Application

User provisioning and synchronization is an important aspect of application management. After enabling provisioning, synchronization allows you to control how operations like creating and deleting accounts in Software as a Service (SaaS) applications are reflected in Oracle Identity Cloud Service.

You can enable and configure synchronization for App Catalog applications either when adding the app or later when modifying it. Be aware that you can only enable synchronization after enabling provisioning. To enable provisioning, see Enabling Provisioning for an App Catalog Application. Follow the runbook for your specific SaaS app to enable and configure synchronization.

To enable and configure synchronization:

1. If not already there, click **Applications** and then the name of the SaaS app that you want to configure.
2. Click the **Provisioning** tab.
3. Click the **Enable Synchronization** switch.
4. In the **Configure Synchronization** section, modify the attributes following the runbook for your specific SaaS application.

Synchronization has to be enabled in order to import user accounts from your SaaS app.

> **Note:**
>
> If the number of created objects (user accounts) and deleted recorded objects (synced user accounts) exceeds the maximum number allowed, the sync job aborts. The maximum number of objects created or recorded objects deleted is an approximate maximum limit, not a precise limit due to the parallel processing of synced objects.

See Importing User Accounts from a Software as a Service Application.

Import User Accounts from a Software as a Service Application

After enabling provisioning and synchronization for your App Catalog app, you may want to import the existing user accounts from your Software as a Service (SaaS) applications and link them to Oracle Identity Cloud Service users.

To import your SaaS user accounts you need to verify that:

- The app is activated. To activate your app, see Activating Applications.
- Provisioning is enabled. See Enabling Provisioning for an App Catalog Application.
- Synchronization is enabled. See Enabling Synchronization for an App Catalog Application.

1. If not already there, click **Applications** and then the name of the app that you want to configure.

   The **Details** page is displayed. Verify that the app is activated.

2. Click the **Import** tab.

   The page lists the result of the last import if any and the actions you need to perform. See Synchronizing User Accounts.

3. If you want to invoke an on-demand synchronization, click the **Import** icon. If the icon is grayed out, click the **Provisioning** tab and verify that Provisioning and Synchronization are enabled, and the app is activated.

4. A message confirms that the job for importing user accounts is running successfully.

After the import finishes the page lists the imported user accounts.

Synchronize User Accounts

After synchronizing your SaaS app with Oracle Identity Cloud Service, you will see the result of the import including the number of users created, deleted, and updated. You can do a general search based on account name, user e-mail or user name. You can also filter and search the results based on Situation and Synchronization Status. Select values from the respective drop-down lists to view user accounts matching the search criteria. These are helpful when you have to find a set of user accounts based on their situation or status from a huge number of results.

The **Import** page provides you with the overall status information, whether the **Last Import** succeeded, failed, or is still running. If the import succeeded, then the result is listed as follows:

- **Start Date** is the date and time you started the import job.
- **End Date** determines date and time the import job finished.
- **Accounts Created** shows the number of Oracle Identity Cloud Service accounts that got created during the import based on your synchronization settings.
- **Accounts Deleted** lists the number of Oracle Identity Cloud Service accounts that got deleted during the import based on your synchronization settings.
- **Accounts Updated** notes the number of Oracle Identity Cloud Service accounts that got changed during the import.

1. This table summarizes the result of successfully running an import. For each SaaS app account it shows whether there exists a matching Oracle Identity Cloud Service user and the action that you need to perform to link the SaaS user account to an Oracle Identity Cloud Service user.

Column	Description
Account	Shows the name of the SaaS app user account.
Situation	Lists whether a matching Oracle Identity Cloud Service user exists or does not exist based on your synchronization configuration: • **No match is found** indicates that you need to manually select which action to take. • **Exact match is found** indicates that an Oracle Identity Cloud Service user exists that matches the synchronization criteria that you configured. • **Multiple Matches are found** indicates that there are multiple matches found for a user. You need to manually select one of the available actions. • **Manually linked** is the result of any action that you performed to link this SaaS app account to an Oracle Identity Cloud Service user.
User	Shows the email address and user name of the Oracle Identity Cloud Service user.
Action	If there is no matching Oracle Identity Cloud Service user you need to select the appropriate action from the drop-down list: • **Assign Existing User**: The **Assign User** page lists all existing Oracle Identity Cloud Service users and allows you to select the one that you want to link with this SaaS app account. • **Create New User and Link**: The **Add User** page allows you to create a new Oracle Identity Cloud Service user.
Status	Lists the status or whether you need to confirm linking the SaaS app account to the Oracle Identity Cloud Service user.

2. Take the appropriate action to link your SaaS user accounts with Oracle Identity Cloud Service accounts.

Work with the Synchronization Failure Report

You can view the synchronization failure report of a provisioning application from the **Import** tab. The report contains the sync failures for the selected application. This report is useful in finding out the reason behind sync failures that occurred during account and object sync of an application.

1. Select the **Import** tab of the application, and click **Synchronization Failure**.
2. In the **Synchronization Failure Report** page, you can use the filters to narrow down the result based on the following criteria:

   a. **Dates Range**

      Choose the number of days for which you need the failure report. The possible values are: **30 Days**, **60 Days**, **90 Days** and **Custom Dates**. Select **Custom Dates** to run the filter for a customized date range. Enter the **Start Date** and the **End Date** in the text box or select them from the calendar.

   b. **Application**

      The Application filter is case sensitive.

   c. **Object Type**

   To narrow down the result based on **Application** or **Object Type**, choose a value from the drop-down menu and enter a corresponding value in the text box. The available drop-down values are: **Equals**, **Contains**, **Begins With** and **Ends With**.

> **Example:**
>
> If you select **Equals** as the value for drop-down menu and enter Google App in the text box, the filter will display entries only for Google App. If you select **Begins With** as the value for drop-down menu and enter G in the text box, the filter will display entries for applications starting with the letter G.

3. Once you have set the filter, click **Run** to display the search result.

   The following table describes the various columns in the search filter:

Filter Columns	Description
Application	Displays the name of the Application.
Object Type	Displays the type of the Object, for example, Account, Group, Organization, Printer, and so on.
Object Identifier	Displays the unique ID of the object from which sync is performed.
Name	Displays the name of the object on which sync is performed.
Date	Displays the time when the sync was performed on the object.
ECID	Displays the value of the Event Correlation Identifier (ECID).
Failure Reason	Displays the reason behind the synchronization failure.

4. Click **Download** to export the search result in the tabular column to a CSV file.

Add Tags to an Application

You're an identity domain administrator or application administrator who wants to create custom attributes for your applications that can be used to search for the applications more effectively. To do this, you add tags to your applications. Tags are key-value pairs that are used to organize and identify applications.

For example, suppose you're creating three versions of an application: one for development purposes, one for testing purposes, and one that will be used in production. You can create the following tags for these versions: **Version: Development**; **Version: Testing**; and **Version: Production**.

There are two kinds of tags that you can add to your application:

- **Tags**

  You can create new tags for your Confidential, Mobile, SAML, and App Catalog applications using the **Tags** section in the **Details** pane. You can use these tags to identify and organize your applications.

  To add new tags to your application:

  1. In the **Tags** area of the **Details** pane, click **Add Tag**.
  2. In the **Tag Key** and **Tag Value** fields of the **Add Tags** window, enter or select the key-value pair for the tag you're creating.

  To create more tags, click **Add Tag**, and repeat the process. You can add up to 100 tags.

- **Tagged**

  You can add existing tags from other applications to Confidential Applications only using the **Tagged** of **Token Issuance Policy** section. Based on the tags selected, your client application can access resource applications that have similar tags.

  To add existing tags from other applications:

  1. In the **Tagged** area of the **Token Issuance Policy** pane, click **Add Tag**.
  2. In the **Add Tags** window, search for the key-value pair of the tag that you're adding from another application by entering the search criteria in the **Tag Key** and **Tag Value** fields.

> **Note:**
>
> You can delete tags by clicking the **X** icon next to the tag. However, some tags are protected and cannot be deleted.

Assign Applications to Oracle Identity Cloud User Using Account Form

Account Form will be visible in the UI with account attributes for a provisioning application if the underlying App Template supports it. With Account Form, when you grant a provisioning application to a user, you can provide account values and when you edit a provisioned account, you can update existing account values.

To provide specific values using account form:

1. In the Identity Cloud Service console, expand the Navigation Drawer, and then click **Applications**.
2. Select **User** tab and click **Assign**.
3. In the Assign Users window, select a user who needs access to the application.
4. Click **Assign** for each user you want to assign to the application.
5. In the Assign Application window, the attribute values are populated based on the mappings already provided in the Attribute Mapping section in the Provisioning configuration tab. You can choose to keep the populated values or update any attribute value.
6. Click **Save** to create the account for the user chosen in step 3.
7. Select the next user you want to provide access to this application. Repeat steps 3 through 5 for the next user.
8. Click **OK** after you have assigned the application to the users. The user account is assigned.

When you want to update account attribute, activate or deactivate the account or revoke the account in Oracle Identity Cloud Service, it would automatically update the respective changes in the application.

Create a Custom Secure Form Fill App

Create and edit custom secure form fill applications for Oracle Identity Cloud Service.

Topics:

Typical Workflow for Creating a Custom Secure Form Fill App

Create, manage, and update custom secure form fill apps in Oracle Identity Cloud Service.

The following table summarizes the suggested tasks that you perform when creating a custom secure form fill app.

Task	Description	Additional Information
Install the Secure Form Fill Admin Client (Oracle Enterprise Single Sign-On (ESSO) Administrative Console).	The ESSO Administrative Console is part of the Secure Form Fill Admin Client. Use the ESSO Administrative Console to create secure form fill configuration files for your custom secure form fill apps in Oracle Identity Cloud Service.	Install the Secure Form Fill Admin Client
Create a secure form fill configuration file.	The ESSO Administrative Console is part of the Secure Form Fill Admin Client. Using the ESSO Administrative Console, create a secure form fill configuration file to be used when creating a custom secure form fill app in Oracle Identity Cloud Service.	Create a Secure Form Fill Configuration File Update a Custom Secure Form Fill App This documentation explains, at a high level, how to use the ESSO Administrative Console only as it pertains to custom secure form fill apps and Oracle Identity Cloud Service. For additional instructions, see the help in the ESSO Administrative Console.

Task	Description	Additional Information
Export the secure form fill configuration file.	The ESSO Administrative Console is part of the Secure Form Fill Admin Client. Using the ESSO Administrative Console, export the secure form fill configuration file that you will import into Oracle Identity Cloud Service when creating or updating a custom secure form fill app in Oracle Identity Cloud Service.	Create a Secure Form Fill Configuration File
Create a custom secure form fill app.	Using the Oracle Identity Cloud Service Admin Console, create a custom secure form fill app.	Create a Secure Form Fill App in Oracle Identity Cloud Service
Assign users and groups.	Using the Oracle Identity Cloud Service Admin Console, assign users and groups to the custom secure form fill app.	About Modifying Applications
Activate the app.	Using the Oracle Identity Cloud Service Admin Console, activate the app so that users can access it.	Activate Applications
Install the secure form fill plugin.	Install the Oracle Secure Form Fill Plugin in order to launch secure form fill apps.	Install the Google Chrome Plugin
Test the custom secure form fill app.	Test the custom secure form fill app that you created before releasing the app.	Test a Custom Secure Form Fill App
Update the custom secure form fill app.	Update the custom secure form fill app that you created as needed.	Update a Custom Secure Form Fill App

Understand Custom Secure Form Fill Apps

Custom secure form fill apps give you the flexibility to define tenant-level form fill apps that are not in the global Oracle App Catalog.

Secure Form Fill is the Oracle Identity Cloud Service alternative for single sign-on into apps that require auto-form fill but don't support OAuth, SAML, or federated sign-on methods.

Users enter their application credentials for form-fill-enabled apps in Oracle Identity Cloud Service once. Oracle Identity Cloud Service stores and encrypts the information, and automatically fills in the login form so that users can sign in without having to re-enter the information each time.

Oracle Identity Cloud Service stores the user's credentials in an encrypted format using strong encryption combined with a customer-specific private key. When a user launches the secure form fill application, which in turn prompts the login page, Oracle Identity Cloud Service detects and securely fills the user's credentials, submits the credentials to the app login page, and then the user is automatically signed in.

Prerequisites for Creating a Custom Secure Form Fill App

Learn the prerequisites for creating a custom secure form fill app.

Ensure that you have the following prerequisites in place before creating and testing a secure form fill app.

- A Windows operating system version 7, 8 or 10 with:
  - Local admin rights enabled
  - 32-bit Java Runtime Environment (JRE) in order to access local help content for the Secure Form Fill Admin Client.
- At least one of the following supported desktop browsers:
  - Mozilla Firefox
  - Google Chrome

> **Note:**
>
> The mobile browsers are not certified. To view the list of certified browsers, see Supported Web Browsers

- Secure Form Fill Admin Client. See Install the Secure Form Fill Admin Client.

> **Note:**
>
> The ESSO Administrative Console is part of the Secure Form Fill Admin Client.

- Oracle Identity Cloud Service tenant (17.2.2 or greater)

  .
- Administrator privileges for Oracle Identity Cloud Service. See Understand Administrator Roles.

Install the Secure Form Fill Admin Client

You use the Secure Form Fill Admin Client (Oracle Enterprise Single Sign-On (ESSO) Administrative Console) to create and update secure form fill configuration files for your custom secure form fill apps in Oracle Identity Cloud Service. Use these instructions to install the Secure Form Fill Admin Client.

1. On the Download page, locate the Secure Form Fill Admin Client and download it. See Downloading Oracle Identity Cloud Service SDKs and Applications.

> **Note:**
>
> The ESSO Administrative Console is part of the Secure Form Fill Admin Client.

2. In the download location, unzip the file.
3. Double-click the installer to launch the install wizard, and then click **Next**.
4. Choose the **Complete** installation option, click **Next,** and then click **Install**.
5. When the installation completes, click **Finish**.

> **Note:**
>
> This documentation explains, at a high level, how to use the ESSO Administrative Console only as it pertains to custom secure form fill apps and Oracle Identity Cloud Service. For additional instructions, see the help in the ESSO Administrative Console.

Create a Secure Form Fill Configuration File

You use the Secure Form Fill Admin Client (Oracle Enterprise Single Sign-On (ESSO) Administrative Console) to create secure form fill configuration files for your custom secure form fill apps in Oracle Identity Cloud Service. Use these instructions to create secure form fill configuration files and then import those files into Oracle Identity Cloud Service.

Prerequisite: An installation of Secure Form Fill Admin Client. See Install the Secure Form Fill Admin Client.

> **Note:**
>
> This documentation explains, at a high level, how to use the Secure Form Fill Admin Client only as it pertains to custom secure form fill apps and Oracle Identity Cloud Service. For additional instructions, see the help in the ESSO Administrative Console.

1. Launch the Secure Form Fill Admin Client.
2. Right-click **Applications**, and then choose **New Web App**. Alternatively, use any of the following options:
   - Click **Applications**, **Add**, and then choose **Application Type: Web** in the first pane of the wizard.
   - Choose **Insert**, **Application**, and then choose **Application Type: Web** in the first pane of the wizard.
3. On the Add Application dialog, enter the name of the application.

> **Important:**
>
> The application name must be the same name that you use when you create the secure form fill application in Oracle Identity Cloud Service.

4. Leave all other default options selected, and then click **Finish**.
5. Choose the **Logon** form type. This form is used to set up your Web app template.

> **Note:**
>
> No other form type is supported at this time.

6. In the **Address** field, enter the URL for the Web app, click **Go**, and then navigate to the login page.

A list of all fields on the login page appear on the bottom of the screen. Select any field in the list on the bottom of the screen to highlight the field on the page.

7. Using the fields on the bottom of the screen, complete the following steps:

> **Note:**
>
> Do no use the **Third Field** or **Fourth Field** options.
>
> Do not use the **Allow multiple field designation** option.
>
> Only use ordinals if field names or attribute level matching is not possible. Using ordinals is less amenable to page changes or differences across browsers, and are not recommended in most scenarios.

   a. Select the user name field, right-click, and then choose **Username/ID**.
   b. Select the password field, right-click, and then choose **Password**.
   c. Select the submit button, right-click, and then choose **Submit**.
8. Click **OK**.
9. On the Web dialog, make any of the following changes as necessary.

Option	Description
Identification Tab	Click **Edit** to make changes. This is the URL or the URLs of the form to configure. For applications that have varying text in their URLs, you can use substrings or regular expressions to specify how to match the variable text. Your Match Type change options are: • **Exact Match.** Exact match matches a URL exactly as specified. This is generally an edge case and rarely used. • **Wildcards.** – ? matches any single character. – * matches zero or more occurrences of any character. If wild cards are used, to avoid a potential security issue, do not perform mid-string wildcard matches. Always exact match the start of the URL, for example, `https://server?.somesite.com/*`. • **Regular Expressions.** This is the recommended option. Use the set of regular expressions to specify a string pattern that the form-fill agent should recognize as a match, for example, `URL1=.*?https://www\.expedia\.co\.jp/user/login.*`

Option	Description
Fields Tab	Click a field, and then using the **Edit** button, adjust the primary form fields used for detection and injection of the user name and password fields as well as the submit button. **Note:** Do not use the **SendKeys** or the **SendKeys using journal hook** options. Check each field type to ensure that it is the appropriate type (such as Password for password input fields, and so on). Your Field options are: • **Field Identification.** Allows you to fine tune how the input fields for the form are located. Field identification can be adjusted for any form field. Click the ellipsis to display the Field identification options. Identify fields by: – **Name.** The default and recommended option. Beware that not every input field has a name or sometimes the name is not consistent every time the page is loaded. If so, then it is recommended to use the Matching option. – **Ordinal.** This option identifies fields based on sequence. This option is not the recommended alternative since it is easily impacted by minor changes to the page. Also, the fields and the field ordinals can be inconsistent across browsers. – **Matching.** Identifies fields based on tag types, attribute values, HTML, and so on. This option is the recommended option if Name is not possible. Often, matching is used to match the “id” attribute of the input field or a regex on the name attribute. Matching can be a regex, substring match, or whole string match. • **Events.** Pre Inject and Post Inject events allow secure form fill to trigger a specific event on the field before and after injecting the credentials into that field. This is useful as some fields will not recognize that injection has occurred unless a specific event is triggered in that field. Event values are: blur, change, click, focus, focusin, focusout, input, keydown, keypress, keyup, and mouseover.

Option	Description
Matching Tab	Create or modify granular page matching criteria for the selected web form. Secure Form Fill in Oracle Identity Cloud Service uses the matching criteria you supply here to distinguish among similar forms. Matching can also be used to refine the detection match criteria, that is, the set of HTML tags and values you use to identify a specific field to perform more specific matching beyond just the form fields themselves.
Proxy Tab	Do not change these settings.

10. Click **OK**.
11. Use the following tabs to make any necessary changes:

> **Note:**
>
> Do not change any other options on any other sub tabs other than those listed in the table below.

Option	Description
General Tab	Enter a description for the Web app. Add, edit, or delete forms. This option allows you to set all the forms relevant for this Web app. Use this option if you have multiple login forms for your Web app. All other settings are not required for secure form fill and should not be changed.
Error Loop Tab	Secure form fill supports the detection of an error loop condition. Error loop conditions generally occur if secure form fill has the wrong credentials for the Web app and attempts to submit these credentials repeatedly to the Web app. • **Logon timeout (sec.).** The maximum time in seconds between successive logon attempts before a logon error is triggered. • **Max. retries.** The maximum number of retries (after first try) allowed before a logon error is triggered.
Miscellaneous Tab	• **Logon Loop Grace Period.** Allows control over the response during login grace period (for example, controls reinjection). • **Auto-submit.** Use this option to turn auto-submit on or off for all forms used by the app.

12. Click **File**, and then **Save**.

> **Tip:**
>
> Name the Web app file, the *.ini file, and the name of the Oracle Identity Cloud Service app with the same name.

13. To export the file, click **File**, **Export**, select the application to export, and then click **OK**.
14. Name the file.

> **Important:**
>
> The *.ini file name must match the name of the application created in Oracle Identity Cloud Service.

Post requisite: Create an Oracle Identity Cloud Service secure form fill app.

Create a Secure Form Fill App in Oracle Identity Cloud Service

After you create a configuration file in the Oracle Enterprise Single Sign-On (ESSO) Administrative Console, the next step is to create a secure form fill app in Oracle Identity Cloud Service.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Applications**, and then click **Add**.
2. In the **Add Application** window, click **App Catalog**.
3. In the Type of Integration section, click **Form Fill**, locate **Generic Secure FormFill App Template**, and then click **Add**.
4. Complete the **App Details** by entering a **Name**, **Description**, and **Application URL**.

> **Important:**
>
> The application name must match the file name of the .ini file created in the ESSO Administrative Console.

5. Optional. In the **Tags** section, click **Add Tag** to add tags to your App Catalog application to organize and identify it.
6. In the **Display Settings** section, select **Display in My Apps**.

> **Important:**
>
> If you do not select **Display in My Apps**, the application does not display in the **My Apps** page for users.

When you select the **Display in My Apps** check box in applications, the app is then visible in the **My Apps** page, but selecting this check box doesn't enable or disable SSO to the app.

7. Select the **User can request access** check box, if you want the app to be listed in the **Catalog**. This option allows end users to request access to applications from their **My Apps** page by clicking **Add** and then selecting the app from the **Catalog**.
8. Click **Add**.
9. Click **Import** to import the secure form fill configuration file that you created in the ESSO Administrative Console.

   The application has been added in deactivate state. To activate your application, click **Activate** next to the app name.
10. To assign users to the application, click **Users**.

> **Tip:**
>
> Assign the application to yourself or a test user. This will save you time when testing the secure form fill app.

11. To assign groups to the application, click **Groups**.

The applications you assign to the user or group displays on the **My Apps** page. Newly assigned applications and applications that a user has not yet accessed appear first in the application list and have an asterisk icon in the application tile. The icon appears on the tile until the user accesses the application.

Install and Use the Secure Form Fill Plugin

You must install the Oracle Secure Form Fill Plugin in order to launch secure form fill apps. Once installed, you are able to access My Apps from your browser toolbar.

Topics:

- Install the Google Chrome Plugin
- Install the Mozilla Firefox Plugin
- Access My Apps from the Browser Using the Plugin

Install the Google Chrome Plugin

If you are using Google Chrome and you need to install the plugin, you are prompted to go to the Extensions on Google Chrome and install the Oracle Secure Form Fill Plugin from the Oracle Identity Cloud Service user interface. You will be prompted to download the plug-in from the My Apps page the first time that you access to a secure form fill app.

Install the Mozilla Firefox Plugin

If you are using Mozilla Firefox and you need to install the plugin, instead of downloading the Secure Form Fill Mozilla Firefox plug-in from the Mozilla Store, install the Secure Form Fill Mozilla Firefox plug-in from the My Apps page. You will be prompted to download the plug-in from the My Apps page the first time that you access to a secure form fill app.

Access My Apps from the Browser Using the Plugin

After you install the Oracle Secure Form Fill Plugin, when you are logged in to Oracle Identity Cloud Service you can access My Apps from your browser toolbar.

Search for apps and see favorite apps by clicking the My Apps icon  .

Test a Custom Secure Form Fill App

After you create a custom secure form fill app in Oracle Identity Cloud Service, you should test the app before deploying it to your organization.

Prerequisites:

- A custom secure form fill app created in Oracle Identity Cloud Service.
- The custom secure form fill app is set to display on the My App page.
- The custom secure form fill app is assigned to you as a user or as a group.

1. Log in to the Identity Cloud Service console to access the My Apps page.
2. Install the secure form fill plug in, if you have not already installed it, and then refresh your browser.
3. Launch the app, enter the credentials for the application, and then click **Login**.

A successful result is Oracle Identity Cloud Service injecting the user name and password, and then clicking the submit button.

If you are having issues, check the settings for your Web app in the Oracle Enterprise Single Sign-On (ESSO) Administrative Console, export the *.ini file if necessary, check the settings for your app in Oracle Identity Cloud Service, and try again.

Update a Custom Secure Form Fill App

To update a custom secure form fill app, you first update the Web app using the Secure Form Fill App, export the configuration file in (*.ini), and then update the custom secure form fill app in Oracle Identity Cloud Service.

Prerequisite:

- A Web app created in the Secure Form Fill Admin Client. See Installing the Secure Form Fill Admin Client.
- A custom secure form fill app created in Oracle Identity Cloud Service.

1. If you need to update the Web app and configuration file created in Secure Form Fill Admin Client, update the Web app first, and then save and export the file. See Create a Secure Form Fill Configuration File.
2. If you need to make changes to the custom secure form fill app in Oracle Identity Cloud Service, access the application as an Identity domain administrator, make any necessary changes, import the new configuration file (if necessary), and then save the app. See Create a Secure Form Fill App in Oracle Identity Cloud Service.

You should now test your new configuration. See Test a Custom Secure Form Fill App.

Import and Synchronize User Accounts Using a Flat File in Oracle Identity Cloud Service UI

You can import and synchronize user accounts of third party cloud applications using a flat file and manage them in Oracle Identity Cloud Service.

Topics:

Import User Accounts Using Oracle Identity Cloud Service UI

You can perform a full sync import of the user accounts of third party cloud applications using a flat file in the Oracle Identity Cloud Service UI. In a full sync import, the imported user accounts in the CSV file replaces any existing user who is already assigned into the application.

To import user accounts:

1. Create a CSV file for import in the following format or download the CSV file along with user data from the target system apps:

   ```
   ID, NAME, ACTIVE
   hercule.poirot@sampleapp.com,hercule.poirot@sampleapp.com,true
   ```

   This table provides a description of the attributes in the CSV format file:

Attribute Name	Description	Sample Value
ID	The unique identifier of the account in the target.	hercule.poirot@sampleapp.com
NAME	The name of the account. NAME is the primary input that is matched with the username of a particular user in Oracle Identity Cloud Service.	hercule.poirot@sampleapp.com
ACTIVE	The status of the account on the target. The possible values are `true` and `false`. If the value is `true`, the user account is imported and activated. If the value is `false`, the user is imported in a deactivated state.	true

2. In the Admin Console, expand the Navigation Drawer, and select **Applications**.
3. Select the required application in which you want to import user accounts.
4. Navigate to the **Import** tab.

5. **Note:**

   Import from CSV file is enabled for applications that support flat file synchronization.

   Select **Import**, browse for the CSV file, and import it.

6. Refresh the page to view the import result. If the import succeeds, then the user accounts present in the CSV file displays.
7. Select **Users** tab to view the imported users.

   Note:

   You need to refresh the **Users** tab to view the imported users.

8. Observe that the users with a `true` value for **ACTIVE** attribute are activated. However, if a user account has `false` value for **ACTIVE** attribute, the user account is imported in a deactivated state.

   Note:

   The **Users** tab displays only the matched and confirmed users.

9. Synchronize the imported user accounts.

See Synchronizing Imported User Accounts to synchronize the imported user accounts with the users in Oracle Identity Cloud Service.

Synchronize Imported User Accounts

After you import the user accounts, if a matching user account doesn't exist in Oracle Identity Cloud Service, you can either assign the user account to an existing user or create a new user for the user account.

If the imported user account exists, an exact match is found and no further action is required. The synchronization status of the user account is set as confirmed.

Topics:

- Assign an Existing User
- Create and Link a New User with the User Account
- Manage Synchronized User Accounts

Assign an Existing User

You can assign an existing user to the imported user account in the **Import** tab if the user is present in Oracle Identity Cloud Service and the import failed to match the user.

To assign an existing user:

1. Select **Assign Existing User** option from the **Select an Action** drop-down list.

2. In the **Assign User** window, search for an existing user, select the required user and click **OK**.

The user account is manually linked to the existing user and the synchronization status is confirmed.

Create and Link a New User with the User Account

You can create a new user and link the user with the imported user account in the **Import** tab if there are no existing users to assign.

To create and link a new user:

1. Select **Create New User and Link** option from the **Select an Action** drop-down list.
2. In the **Add User** window, enter the following user details: **First Name**, **Last Name**, **User Name** or **Email**.
3. Select **Use the email address as the user name** option if you want to use the email address as your user name and click **OK**.

The user account is manually linked to the new user and the synchronization status is confirmed.

Manage Synchronized User Accounts

You can activate, deactivate, assign and revoke imported user accounts from the **Users** tab.

You can perform the following actions on the synchronized user accounts:

- To activate an imported user account, select the **Action** menu, and click **Activate**.
- To deactivate an imported user account, select the **Action** menu, and click **Deactivate**.
- To remove any imported user account, select the user, and click **Revoke**.
- To assign any other user to the application apart from the synchronized users from the flat file, click **Assign**. Choose a user from the list of existing users and click **OK**. The assigned user account is displayed and is in an activated state.

View Details About Applications

By default, you can see the name and description for each application in Oracle Identity Cloud Service.

By clicking an application name, you can view high-level and configuration information about the application. For Oracle applications, you can also see the roles associated with the application, and the Oracle Identity Cloud Service groups and users assigned to the application.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. In the **Applications** page, click the application name for which you want additional information.

> **Tip:**
>
> To search for applications, enter all or part of the beginning of the application name that you want to locate in the search field, and then press **Enter**. To fine-tune your search, click the search field again, and then select a status.

3. To view high-level information about the application, such as the application type, name, description, icon, URL, links, and whether the application will appear on the **My Apps** page, click **Details**.
4. To view configuration information about the application, click **Configuration**. For custom SAML applications, this tab is labeled **SSO Configuration** because, by granting SAML applications to users, they can single sign-on (SSO) into SaaS applications that support SAML for SSO. See Add a Confidential Application, Add a Mobile Application, and Add a SAML Application.
5. For Oracle applications, to view roles associated with the application, click **Application Roles**. You can assign users and groups to an application role or remove users and groups from the application role. See About Modifying Applications.
6. For Oracle applications, to view the names and descriptions of any groups assigned to the application, click **Groups**.
7. For Oracle applications, to view the names, email addresses, and phone numbers of any users assigned to the application, click **Users**. You can filter and sort this list of users.
   - To display only those users who are assigned to a particular application role, click **Show**, and then select the application role.
   - To display users who are assigned to any application role, click **Show**, and then select **All Role Members**.
   - To sort the users in ascending order by their names or email addresses, click **Sort By**, and then select **Name** or **Email**.

About Modifying Applications

Learn about assigning users and groups to applications; and importing and exporting users and groups for Oracle and custom applications.

Topics:

Modify Applications

After configuring, you can modify Oracle and custom applications to assign users and groups, edit high-level information, import users and groups into the applications, export users and groups from applications, and perform specific configuration tasks for custom applications.

To modify applications:

1. In the Identity Cloud Service admin console, expand the **Navigation Drawer**, and then click **Applications**.
2. Click the application you want to modify. The **Applications** page expands to open a sub page that displays high-level information about the application.

   You can perform the following tasks in Oracle and custom applications:

   - **Oracle Applications**:
     - Assign users and groups.
     - Remove users and groups.

       The **Groups** and **Users** tabs are used to display groups and users assigned to application roles of an Oracle application. Although you can filter and sort this list of users and groups, you can't modify the list. You can't edit values that appear in these tabs.
     - Import Users and Groups for Oracle Application Roles.
     - Export Users and Groups for Oracle Application Roles.

       > **Note:**
       >
       > If you assign user accounts to Oracle application roles and then deactivate the accounts, Oracle Identity Cloud Service prevents the users from accessing the roles. To enable the users to access the Oracle application roles to which they are assigned, activate the user accounts. See Activate User Accounts and Deactivate User Accounts.

     - View High-Level Information

     See Modify Oracle Applications.
   - **Custom Applications:**
     - Assign users and groups.
     - Remove users and groups.
     - Edit high-level information and configuration information.
     - Edit Web Tier Policies for Trusted Applications.
     - Regenerating a Client Secret and generating tokens for Trusted Applications
     - Edit single sign-on (SSO) configuration for SAML Applications.

     See Modify Custom Applications.

Modify Oracle Applications

You can assign and remove users and groups to Oracle Applications, and import and export users and groups for Oracle Application Roles. You can just view the high level information and cannot edit any of the values in Oracle Applications.

Topics:

- Assign Users to Oracle Applications
- Remove Users from Oracle Applications
- Assign Groups to Oracle Applications

- Remove Groups from Oracle Applications
- Edit High-Level Information for Oracle Applications

Assign Users to Oracle Applications

To assign users to Oracle applications, use the **Application Roles** tab. You can assign users to Oracle applications only after you activate the applications.

1. Click **Application Roles**.
2. Select the check box for the application role of the Oracle application to which you want to assign users.
3. Click **More**, and then select **Assign Users**.
4. In the **Assign Users** window, select the check box for each user that you want to assign to the application role.
5. Click **Assign**.

   The application role displays a user icon and a **Users Assigned** link. The link displays the number of users that you assigned to the application role.
6. Click **Users Assigned**.
7. In the **Users Assignments** window, verify that you see the users that you assigned to the application role.
8. Click **Close**.

Remove Users from Oracle Applications

To remove users from Oracle applications, use the **Application Roles** tab. You can remove users from Oracle applications only after you activate the applications.

1. Click **Application Roles**.
2. Select the check box for the application role of the Oracle application from which you want to remove users.

   > **Tip:**
   >
   > You can see which application roles have users assigned to them by the user icon and the **Users Assigned** link that appears in the application role.
3. Click **More**, and then select **Revoke Users**.
4. In the **Revoke Users** window, select the check box for each user that you want to remove from the application role.
5. Click **Revoke**.

Assign Groups to Oracle Applications

Once you activate the applications, you can assign groups to Oracle applications by using the **Application Roles** tab.

1. Click **Application Roles**.
2. Select the check box for the application role of the Oracle application to which you want to assign groups.

3. Click **More**, and then select **Assign Groups**.
4. In the **Assign Groups** window, select the check box for each group that you want to assign to the application role.
5. Click **Assign**.

> **Note:**
>
> The **All Tenant Users** group is a default group that's created by Oracle Identity Cloud Service. All Oracle Identity Cloud Service users are assigned to this group, by default. If you assign this group to any of your applications, then all users are assigned to these applications indirectly.

The application role displays a group icon and a **Groups Assigned** link. The link displays the number of groups that you assigned to the application role.

6. Click **Groups Assigned**.
7. In the **Groups Assignments** window, verify that you see the groups that you assigned to the application role.
8. Click **Close**.

Remove Groups from Oracle Applications

You can remove groups from Oracle applications from the **Application Roles** tab. You can remove groups from Oracle applications only after you activate the applications.

1. Click **Application Roles**.
2. Select the check box for the application role of the Oracle application from which you want to remove groups.

> **Tip:**
>
> You can see which application roles have groups assigned to them by the group icon and the **Groups Assigned** link that appears in the application role.

3. Click **More**, and then select **Revoke Groups**.
4. In the **Revoke Groups** window, select the check box for each group that you want to remove from the application role.

> **Note:**
>
> The **All Tenant Users** group is a default group that's created by Oracle Identity Cloud Service. All Oracle Identity Cloud Service users are assigned to this group, by default. If you remove the **All Tenant Users** group from your applications, then access rights to these applications are revoked for every Oracle Identity Cloud Service user.

5. Click **Revoke**.

Edit High-Level Information for Oracle Applications

When you create an instance of an Oracle application in your identity domain, the application instance appears in the **Applications** page. As a Service Administrator, you can edit some of the high-level information for Oracle Applications. However, you can't edit attributes that are protected. Even in an editable attribute, you can't update certain values that were seeded by the system.

To view and edit high-level information about Oracle application, such as the application type, name, description, icon, URL, links, and whether the application will appear on the **My Apps** page, click **Details**.

As of 18.2.6 release, the tabular column lists the editable UI field names, respective attributes, whether the seeded values can be updated and whether new values can be added to the editable field names:

UI Elements	Attributes	Update Seeded Values	Add New Values
Description	description	Yes	N/A
Tags	tags	No	Yes
Allowed Scopes	allowedScopes	No	Yes
Allowed Tags	allowedTags	No	Yes
Redirect URL	redirectUris	Yes	N/A
Access Token Expiration	accessTokenExpiry	Yes	N/A
Refresh Token Expiration	refreshTokenExpiry	Yes	N/A
Scope	scopes	No	Yes
Secondary Audiences	protectableSecondaryAudiences	No	Yes
Is Refresh Token Allowed	allowOffline	Yes	N/A
Enforce Grants as Authorization	allowAccessControl	N/A	N/A
Trust Scope	trustScope	N/A	N/A
Activate	active	N/A	N/A

Not all attributes correspond to UI fields:

- WebTier policy tab and all the UI fields within the tab are controlled and edited by one attribute `urn:ietf:params:scim:schemas:oracle:idcs:extension:webTierPolicy:App:webTierPolicyJson`.
- `grantedAppRoles` attribute records each App Role defined by another application that has been granted to the client.
- `signonPolicy` editable attribute indicates that you can assign Oracle Applications to Sign-On Policy.

> **Note:**
>
> You cannot change any of the fields other than the ones listed above for Oracle Public Cloud applications. You will encounter an error if you click **Save** after you try editing any of these values.

Apart from editing certain attributes, you can perform the following with Oracle Public Cloud Applications:

- Edit only single scope. Bulk removal of scopes is not supported.
- Grant client access to the Oracle Identity Cloud Service APIs

  In order to enable your application to access Oracle Identity Cloud Service APIs, click **Add**.

  In the **Add App Role** window, select the application roles that you want to assign to this application. This enables your application to access the REST APIs that each of the assigned application roles can access.

  For example, select **Identity Domain Administrator** from the list. All REST API tasks available to the identity domain administrator will be accessible to your application.

  You can't remove the following:

  – The assigned application roles from the application by clicking the **x** icon for the row of the required application role

  – The App Roles that were granted when an Oracle Public Cloud application was created because those seeded values are protected

  See Apps/App Roles endpoint for a complete list of which endpoints each application role can access.

About Importing Users and Groups for Oracle Application Roles

You can use Oracle Identity Cloud Service to import users and groups to assign them to Oracle application roles of Oracle applications automatically.

See Add or Remove a User Account from an Administrator Role for information about assigning users to administrator roles.

Topics:

- Create and Prepare a Comma-Separated Value File
- Import Users and Groups for Oracle Application Roles

Create and Prepare a Comma-Separated Value File

Learn how to create and prepare a comma-separated value (CSV) file to import either a single user or group, or multiple users or groups.

> **Note:**
>
> Importing application roles imports application roles memberships only. The application roles must already exist in Oracle Identity Cloud Service. If the application roles don't exist you will receive an error for the membership import for that application role.

1. Use these sample files as a starting point.
2. Extract the compressed file and then open the `AppRoleMembership.csv` file.
3. Review and then delete any demo data in the `AppRoleMembership.csv` file.

   To familiarize yourself with the import process, consider importing just the demo data. You can then delete the unwanted demo data from Oracle Identity Cloud Service before you begin importing live data.
4. Create an import file using the `AppRoleMembership.csv` file. The `AppRoleMembership.csv` file is a simple text file in a tabular format (rows and columns). The first row in the file defines the columns (fields) in your table. At a minimum, the file must have these exact column headings.
   - Entitlement Value
   - Grantee Name
   - Grantee Type
5. As a best practice, ensure that the fields in these columns are unique.
6. For each account, create a new row (line) and enter data into each column (field). Each row equals one record. The maximum number of membership roles that can be imported in a single job must not exceed 10,000.
7. To create a CSV file, use a standard spreadsheet application, such as Microsoft Excel or Google Sheets, or use a text editor, such as Notepad or TextPad.
8. Save your file in a CSV format. If you do not save the file in a CSV format with UTF-8 encoding, the import fails.

> **Note:**
>
> If you exported application role memberships prior to version 17.2.2 of Oracle Identity Cloud Service, and you want to import them back into Oracle Identity Cloud Service, you need to change the column headings in your CSV file to **Entitlement Value**, **Grantee Name**, and **Grantee Type** before doing so.

Import Users and Groups for Oracle Application Roles

You can use Oracle Identity Cloud Service to import users and groups using a comma-separated value (CSV) file to assign them to Oracle application roles.

> **Note:**
>
> To import or export users and groups for application roles, you must be assigned to either the identity domain administrator role or the application administrator role.

To import users and groups for Oracle application roles:

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. In the **Applications** page, click the Oracle application that has roles to which you want to assign users and groups.

   > **Note:**
   >
   > Importing application roles imports application roles memberships only. The application roles must already exist in Oracle Identity Cloud Service. If the application roles don't exist you will receive an error for the membership import for that application role.

3. Click **Application Roles**.
4. Click **Import**.
5. In the **Import Application Roles** window, click **Browse** to locate and select the CSV file that contains the users and groups to import.

   > **Note:**
   >
   > Click **Download sample file** in the dialog box to download a sample file.

6. Verify that the path and name of the CSV file that you selected appear in the **Select a file to import** field.
7. Click **Import**.

   If a user or a group is missing a required value, such as the user name or the group name, then Oracle Identity Cloud Service can't import the user or group. If Oracle Identity Cloud Service can't import the user or group, then it evaluates the next user or group in the CSV file.
8. After Oracle Identity Cloud Service evaluates all users and groups, review the job results.
   - If the job *can* be processed immediately, then a dialog box appears with the **Job ID** link for your import job. Click the link. Review the details that appear on the **Jobs** page.
   - If the job *cannot* be processed immediately, then a message appears with a **Schedule ID** in it. Copy that ID and use it to search for the job on the **Jobs** page. The job will appear when processing completes. Go to Step 9.

Oracle Identity Cloud Service assigns a job ID to each file that's imported or exported, for auditing purposes.

9. On the **Jobs** page, locate the job that you want to view, and then click **View Details**.

   A table appears that displays the user names or group names, classification types (User or Group), and status of the users and groups that you imported and assigned to Oracle application roles in Oracle Identity Cloud Service.

See Export Job Errors to download a CSV file of any errors to your local machine.

Export Users and Groups for Oracle Application Roles

You can use Oracle Identity Cloud Service to export users and groups assigned to Oracle application roles of Oracle applications.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. In the **Applications** page, click the Oracle application that has application roles with users and groups assigned to them.
3. Click **Application Roles**.
4. To export all application roles that have users and groups assigned to them, click **Export**, and then select **Export All**.

   To export only some application roles, select the check box for each application role that you want to export. Click **Export**, and then select **Export Selected**.

   > **Tip:**
   >
   > The number that appears within parentheses to the right of **Export All** is how many application roles are created for the Oracle application. The number within parentheses to the right of **Export Selected** is how many application roles that you selected to export.

5. In the **Export Application Roles** window, click **Export Application Roles**.
6. After Oracle Identity Cloud Service creates the export file, a **Job ID** link appears. Click the link.
7. Review the details that appear in the **Jobs** page. This page shows how many application roles that you attempted to export, how many application roles Oracle Identity Cloud Service exported successfully, and how many application roles can't be exported because of a system error.
8. Click **Download**.

See Export Job Errors to download a CSV file of any errors to your local machine.

Modify Custom Applications

You can assign and remove users and edit high-level information in custom applications.

Topics:

- Assign Users to Custom Applications
- Remove Users from Custom Applications

- Assign Groups to Custom Applications
- Remove Groups from Custom Applications
- Edit High-Level Information for Custom Applications
- Edit Configuration Information for Custom Applications
- Edit SSO Configuration Information for SAML Applications
- Import User Accounts from a Flat File Using REST APIs
- Regenerate a Client Secret for Confidential Applications
- Generate Tokens for Trusted Applications

Assign Users to Custom Applications

Custom applications are non Oracle Public Cloud (OPC) services. You can modify custom applications by assigning users to them. Users can access the **My Apps** page to view these applications.

Prerequisite:

- The application must be activated.
- The application must be assigned to the current user who is accessing the **My Apps** page
- The **Display in My Apps** check box must be selected in the **Details** tab in the applications.

You can directly assign users to an application as follows.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. Click the application that you want to modify.
3. Click **Users**.
4. Click **Assign**.
5. In the **Assign Users** window, do one of the following:
   a. Select the check box for each user that you want to assign to the application.
   b. For a provisioned application, select **Assign** next to the user that you want to assign to the application. Enter the required values for the form, and then click **Save**.

> **Note:**
>
> If the form contains multi-valued attributes, then an **Add** button appears to the right of each attribute. Click **Add**, and then in the **Allowed Values** window, select the values for the attribute, and click **OK**.

6. Click **OK**.

> **Note:**
>
> If you assigned a provisioned application to the user, then you can modify the values of the application form. To do this, click the **Action** menu, select **Edit**, change the appropriate values, and then click **Save**.
>
> You can activate or deactivate an user's account assigned to a synchronized app that's created from the App Catalog. To do so:
>
> 1. Click the **Action** menu to the right of the user account that you assigned to the application.
> 2. Click **Activate** or **Deactivate**.
> 3. In the **Activate Account?** or **Deactivate Account?** window, click **OK**.
>
> See Enabling Provisioning for an App Catalog Application for more information about configuring provisioning for an application to manage the lifecycle of user accounts in the application.

Remove Users from Custom Applications

You can modify custom applications by removing users from them. Users can no longer view these applications through the **My Apps** page.

Prerequisite: The application must be activated.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. Click the application that you want to modify.
3. Click **Users**.
4. Select the check box for each user that you want to remove from the application.
5. Click **Revoke**.
6. Complete one of the following choices:
   - To remove one user from the custom application, in the **Revoke User?** dialog box, click **Revoke User**.
   - To remove multiple users from the custom application, in the **Revoke Users?** dialog box, click **Revoke Users**.

Assign Groups to Custom Applications

You can modify custom applications by assigning groups to them. Users who are members of these groups can access the **My Apps** page to view these applications.

Prerequisite: The application must be activated.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. Click the application that you want to modify.
3. Click **Groups**.

4. Click **Assign**.
5. In the **Assign Groups** window, do one of the following.
   a. Select the check box for each group that you want to assign to the application.
   b. For a provisioned application, select **Assign** next to the group that you want to assign to the application. Enter the required values for the form, and then click **Save**.

> **Note:**
>
> If the form contains multi-valued attributes, then an **Add** button appears to the right of each attribute. Click **Add**, and then in the **Allowed Values** window, select the values for the attribute, and click **OK**.
>
> The **All Tenant Users** group is a default group that's created by Oracle Identity Cloud Service. All Oracle Identity Cloud Service users are assigned to this group, by default. If you assign this group to any of your applications, then all users are assigned to these applications indirectly.

6. Click **OK**.

> **Note:**
>
> If you assigned a provisioned application to the group, then you can modify the values of the application form. To do this, click the **Action** menu  , select **Edit**, change the appropriate values, and then click **Save**.

Remove Groups from Custom Applications

You can modify custom applications by removing groups from them. Users who are members of these groups can no longer view these applications through the **My Apps** page

Prerequisite: The application must be activated.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. Click the application that you want to modify.
3. Click **Groups**.
4. Select the check box for each group that you want to remove from the application.

   The **All Tenant Users** group is a default group that's created by Oracle Identity Cloud Service. All Oracle Identity Cloud Service users are assigned to this group, by default. If you remove the **All Tenant Users** group from your applications, then access rights to these applications are revoked for every Oracle Identity Cloud Service user.
5. Click **Revoke**.

Edit High-Level Information for Custom Applications

You can edit high-level information for custom applications.

1. Click **Details**.

2. To modify an attribute value, enter the modification in the attribute field (for example, modify the application name in the **Name** field).
3. Click **Save**.

Edit Configuration Information for Custom Applications

You can edit configuration information for custom applications.

1. Click **Configuration**.
2. Expand the **Client Configuration** node.
3. Modify a configuration value for the custom application by:
   - Entering the value in the attribute field (for example, in the **Redirect URL** field, entering the application URL where the user is redirected after authentication)
   - Clicking a button (for example, adding a resource to the custom application by clicking **Add** or removing a scope for a trusted application by clicking **Remove**)
   - Selecting or clearing the check box (for example, allowing the resource owner to be a grant type for the custom application by selecting **Resource Owner**)
   - Selecting the value from the menu (for example, selecting **User Administrator** from the **Grant the client access to Identity Cloud Service Admin APIs.** list to enable the custom application to access user administrator-related APIs)
4. If your custom application is a confidential or a mobile application, then you can switch **Bypass Consent** on or off.
5. If your custom application is a confidential application, then expand the **Resources** node.

   If your custom application is a mobile application, then the **Resources** node doesn't appear in the **Configuration** tab. This is because confidential applications run on a protected server, and mobile applications run on an unauthenticated web browser or a mobile device.
6. Modify a configuration value for the protected resources of your confidential application. See step 3 for more information about how to edit configuration values.
7. Click **Save**.

See Add a Confidential Application, Add a Mobile Application, and Add a SAML Application for more information about the configuration settings for client applications.

Edit Consent Information for Custom Applications

Application administrators can customize the information that appears in the OAuth consent page for applications.

If your application's resources is configured to require consent, then Oracle Identity Cloud Service provides a consent page in which users must allow to access the application's resources. By default, this consent page is branded with Oracle Identity Cloud Service information, but you can customize the information that appears in the page.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. Click the application that you want to modify, and then click the **Configuration** tab.
3. Expand the **Consent Information** node.
4. Provide values for the fields that you want to customize, and then click **Save**.

Edit SSO Configuration Information for SAML Applications

You can edit SSO configuration information for SAML applications.

See Add a SAML Application for more information about the SSO configuration settings for SAML applications.

To edit SSO configuration:

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. Select the SAML application from the list of applications listed.
3. Click **SSO Configuration**.
4. In the **General** section, modify required SAML assertion attributes for the SAML application by:
   - Entering a value in the attribute field (for example, in the **Assertion Consumer URL** field, entering the endpoint at the service provider to which the SAML assertion will be sent by the SAML identity provider)
   - Selecting a value from a list (for example, selecting the type of format to use for the NameID from the **NameID Format** list)
   - Clicking a button (for example, uploading the signing certificate that is used to encrypt the SAML assertion by clicking **Upload**)
5. Expand the **Advanced Settings** section to modify optional advanced SAML assertion settings for the SAML application (for example, selecting **Assertion** from the **Signed SSO** list to indicate that you want the SAML assertion signed).
6. Expand the **Attribute Configuration** section to modify user-specific and group-specific attributes for the SAML application (for example, selecting the type of user attribute from the **User Attribute** list or selecting the condition by which you want to filter the group memberships from the **Condition** list).
7. Click **Save**.

Import User Accounts from a Flat File Using REST APIs

Some target applications do not support synchronization with Oracle Identity Cloud Service. You can onboard the user accounts from these applications by importing the accounts from a flat file using REST APIs.

To import user accounts from a flat file:

1. Create a CSV file for import in the following format or download the CSV file along with User data from the target system apps:

   ```
   ID, NAME, ACTIVE
   hercule.poirot@sampleapp.com,hercule.poirot@sampleapp.com,true
   ```

   This table provides a description of the attributes in the CSV format file:

Attribute Name	Description	Sample Value
ID	The unique identifier of the account in the target. The ID should match the target attribute that is defined for this application.	hercule.poirot@sampleapp.com
NAME	The name of the account.	hercule.poirot@sampleapp.com
ACTIVE	The status of the account on the target. The possible values are true and false.	true

2. Upload the flat file to the storage server. You can use the below curl command to upload a file to storage using the tenant admin access token.

```
curl -k
-X POST
-H "Authorization: Bearer <Tenant Admin Access Token Value>"
-F "contentType=text/csv"
-F "isPublic=false"
-F file=@"/scratch/$USER/flatfile.csv" "https://<tenant base url>/
storage/v1/Files"
```

   Make note of the `fileName` attribute from the response.

3. To get the value of the application id, use the following request:

```
curl -k
-X GET
-H "Authorization: Bearer <Tenant Admin Access Token Value>"
-H "Content-Type:application/scim+json"
"https://<tenant base url>.identity.oraclecloud.com/admin/v1/Apps?
filter=displayName co \"<Your application name>\""
```

   Make note of the value of the `id` attribute from the response.

4. To get the value for the `resourceType`, use the following request:.

```
curl -k
-X GET
-H "Authorization: Bearer <Tenant Admin Access Token Value>"
-H "Content-Type:application/scim+json"
"https://<tenant base url>.identity.oraclecloud.com/admin/v1/Apps/<appID>?
attributes=urn:ietf:params:scim:schemas:oracle:idcs:extension:managedapp:Ap
p:objectClasses"
```

   Copy the value of the `objectClasses.resourceType` from the response. The resourceType value has a prefix of "ManagedApp" followed by a GUID.

5. Run the `ManagedObjectSync` reconciliation job using a json file with information of the resourceType and the csv file you uploaded.

   Create and save a JSON file with the following content:

```
{
  "schemas": [
```

```
        "urn:ietf:params:scim:schemas:oracle:idcs:JobSchedule"
    ],
    "jobType":"ManagedObjectSync" ,
    "runNow":true,
     "parameters": [
    {
      "name": "resourceType",
      "value": "<Dynamic ResourceType ID from app>"
    },
    {
      "name": "isIncremental",
      "value": "false"
    },
    {
      "name": "isFileBased",
      "value": "true"
    },
    {
      "name": "fileURI",
      "value": "<fileName of the file in the storage. Format: files/
201702110205/testFileName-1486778745812-5318.csv>"
    }]
}
```

POST request:

```
curl -k
-X POST
-H "Content-Type:application/scim+json"
-H "Authorization: Bearer <Tenant Admin Access Token Value>"
-d @"/scratch/$USER/runjob.json"
https://<tenant base url>/job/v1/JobSchedules
```

After you run the command, verify that the users in the csv file have been assigned to the application.

6. Optionally, you can check the status of the scheduled job using the JobHistories API. See REST API for Oracle Identity Cloud Service.

Regenerate a Client Secret for Confidential Applications

When you create a confidential application, you use a Client ID and a Client Secret as part of your connection settings. You can regenerate your Client Secret at any time for a confidential application using the Identity Cloud Service console.

Prerequisite: An existing confidential application in Oracle Identity Cloud Service

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. Click the confidential application for which you want to regenerate a Client Secret.
3. Click **Configuration**.
4. Expand the **General Information** node.
5. Click **Regenerate**. The new Client Secret appears in the **Client Secret** dialog box.

6. Click **Close**.

Generate Tokens for Confidential Applications

When you create a confidential application and you configure the client to use the **JWT Assertion** grant type, you can generate access tokens at any time using the Identity Cloud Service console.

Prerequisite: An existing trusted application in Oracle Identity Cloud Service with the client configured to use the **JWT Assertion** grant type and activated.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. Click the confidential application for which you want to generate an access token.
3. Click **Generate Access Token**.
4. In the **Generate Token** pop-up window, use the following table to configure which scopes should be included in the access token:

Option	Description
Available Scopes	Click **Available Scopes** to get the access token to access any resources configured for the application. If the scopes are defined from multiples resource servers, the token cannot be generated. Use the **Customized Scopes** option and make sure that the selected scopes are from the same resource server.
Customized Scopes using **Invokes Identity Cloud Service APIs**	a. Click **Customized Scopes** and **Invokes Identity Cloud Service APIs**. b. From the list of all the roles that are assigned to the client application you can select those roles that you want to include or remove to limit the scopes to be populated in the resulting token.
Customized Scopes using **Invokes Other APIs**	a. Click **Customized Scopes** and **Invokes Other APIs**. b. The UI displays a list of all the scopes assigned to the application. You can select any desired scopes as long as those scopes are from the same resource server.
Include Refresh Token	If the Refresh Token grant type is configured for your client application and the resource server which the scopes belong to allows the refresh token to be generated, the **Include Refresh Token** check box is enabled to be used. The refresh token is used to obtain a new access token without requiring the user to reauthenticate.

5. Click **Download Token**.

> **Note:**
>
> The downloaded token gets saved as a `tokens<n>.tok` file in the download folder of your browser.

Activate Applications

Activating applications reinstates the access rights to applications for users and groups. You can use Oracle Identity Cloud Service to activate multiple applications simultaneously.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. Select the check box for each deactivated application that you want to activate, or to activate all deactivated applications, search for accounts with a status of **Inactive**. Then, select the **Select All** check box.

> **Tip:**
>
> A red circle with a white line through the circle indicates a deactivated application.

3. Complete one of the following choices:
   - To activate one application, click **Activate**, and then click **OK** in the **Confirmation** window.
   - To activate more than one application, click **Activate**, and then click **OK** in the **Confirmation** window.

> **Note:**
>
> You cannot activate an Oracle Public Cloud application from the UI. You will encounter an error if you try activating any Oracle Public Cloud application.

Deactivate Applications

Deactivating an application temporarily disable the access rights to applications that users or groups have. You can use Oracle Identity Cloud Service to deactivate multiple applications simultaneously.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. Select the check box for each activated application that you want to deactivate, or to deactivate all applications, search for accounts with a status of **Active**. Then, select the **Select All** check box.

> **Tip:**
>
> A green check mark indicates an activated application.

3. Complete one of the following choices:
   - To deactivate one application, click **Deactivate**, and then click **OK** in the **Confirmation** window.
   - To deactivate more than one application, click **Deactivate**, and then click **OK** in the **Confirmation** window.

> **Note:**
>
> You cannot deactivate an Oracle Public Cloud application from the UI. You will encounter an error if you try deactivating any Oracle Public Cloud application.

Remove Applications

You can use Oracle Identity Cloud Service to remove multiple applications simultaneously.

1. In the Identity Cloud Service admin console, expand the **Navigation Drawer**, and then click **Applications**.
2. To remove an application, deactivate it first.
3. Select the check box for each application that you want to remove, or to remove all applications, select the **Select All** check box.
4. Complete one of the following choices:
   - To remove one application, click **Remove**, and then click **OK** in the **Confirmation** window.
   - To remove more than one application, click **Remove**, and then click **OK** in the **Confirmation** window.

> **Note:**
>
> You cannot remove an Oracle Public Cloud application from the UI. You will encounter an error if you try removing any Oracle Public Cloud application.

6

Manage Oracle Identity Cloud Service Jobs

Learn how to bulk load data into Oracle Identity Cloud Service.

Topics:

Understand Bulk Loading Data

Oracle Identity Cloud Service may be one among many repositories in your organization. When you start using Oracle Identity Cloud Service, you might want to load data from the other repositories into Oracle Identity Cloud Service. Bulk loading offers a solution to this requirement.

Bulk loading is aimed at automating the process of loading a large amount of data into Oracle Identity Cloud Service. You can bulk load data after you subscribe to Oracle Identity Cloud Service or at any time during the production lifetime. You can bulk load users, groups, and application roles. Onboarding Users and Groups are administrative tasks.

You can access the Bulk Loading Users and Groups Using CSV Files tutorial to see how to import user accounts into Oracle Identity Cloud Service.

You can bulk load data using the following methods:

- The Identity Cloud Service console
- SCIM-based APIs

In this section, you learn how to bulk load data by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Typical Workflow for Bulk Loading Data

To start bulk loading data, refer to the typical workflow described in this section.

After each import step, analyze the data recorded during the bulk load operation.

If the job *can* be processed immediately, a dialog box appears with the **Job ID** link for your import job, click the link. Review the details that appear on the **Jobs** page.

If the job *can't* be processed immediately, a message appears with a **Schedule ID** in it. Copy that **Schedule ID**, and use it to search for the job on the **Jobs** page. The job will appear when processing completes.

This page shows how many accounts you imported, how many accounts imported successfully, and how many accounts can't be imported because of a system error. Common issues that prevent the system from importing the account include:

- Invalid email address format
- Invalid field formats
- Missing required fields
- Invalid CSV file

If there are many invalid accounts, correct the errors in the import file and then import the file again. See View Jobs and Job Details.

Task	Description	Additional Information
Step 1: Import users.	Use this task to create users only.	Import User Accounts
Step 2: Import groups.	Use this task to create groups and user memberships.	Import Groups
Step 3: Import application role memberships.	Use this task to create application role memberships for users and groups.	Import Users and Groups for Oracle Application Roles
Step 4: (Optional) Gather diagnostic data from the bulk load operation.	If you encounter errors during a bulk load operation and you cannot fix them by modifying the entries in the import file, you can set a diagnostics level to capture operational logs during the bulk load operation. You can then view those logs to help you to determine the cause of the problem.	See Run Oracle Identity Cloud Service Reports.
Step 5: (Optional) Resolving errors after a bulk load operation.	If you encounter errors during a bulk load operation, resolve the errors and then try the bulk load operation again. One of the reasons that Oracle Identity Cloud Service cannot import a user account is, for example, because the account is missing a required value, such as the user's first name, last name, or user name. If Oracle Identity Cloud Service can't import a user account, then it evaluates the next account in the CSV file. View the details of the import job. If the job contains errors, you can export those errors to see the cause. If you cannot resolve the errors, use the diagnostic data report to capture operational logs to see if you can determine the cause of the problem.	View Jobs and Job Details See Run Oracle Identity Cloud Service Reports. Export Job Errors

Use Best Practices for Bulk Loading Data

Implementing these best practices when bulk loading data reduces the possibility of errors occurring during the bulk load process. Read and understand this section before you start bulk loading data.

Use the following sections to learn about key elements for bulk loading data.

Topics:

Bulk Loading File Specifications

Learn about the bulk loading files specification to reduce the possibility of errors.

Regardless of which data that you are bulk loading, the bulk loading file itself must meet the following specifications:

- Use a comma as the delimiter between the values
- Save the file in a CSV format (*.csv)
- Limit file size to 52 MB

> **Tip:**
>
> Although the system upload limit is 52 MB, as a best practice, segment your bulk load files into small, manageable sets of data. For example, import just one *user* to familiarize yourself with the process. You can then import a larger set of *users*, for example, 100 *users*. If you do not experience any import errors, increase the import file size according to your level of comfort.

The bulk load file is a simple text file in a tabular format (rows and columns). The first row in the file defines the columns (fields) in your table. At a minimum, the import file must have these exact column headings.

Bulk Load File	Required Column Headings
Users	User ID
	Last Name
	First Name
	Work Email
Groups	Display Name
	Description
	User Members
	Requestable

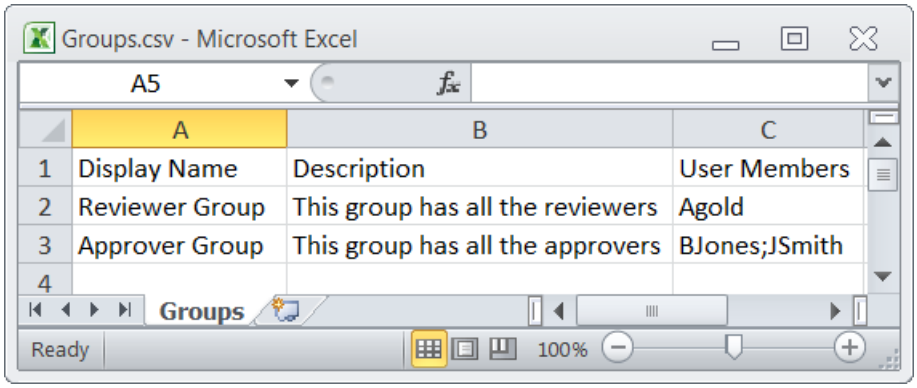
Bulk Load File	Required Column Headings
Application Role Membership	Entitlement Value Grantee Name Grantee Type App Name

For each account, you create a new row (line) and enter data into each column (field). Each row equals one record.

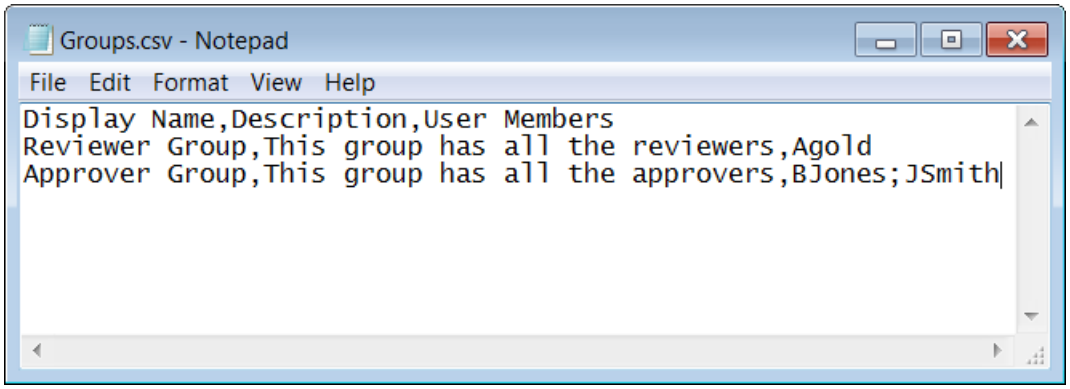
To create an import file, you can use a standard spreadsheet application, such as Microsoft Excel or Google Sheets, or you can use a text editor, such as Notepad or TextPad.

> **Important:**
>
> Whichever application you use to create the file, ensure that you save the file in a valid CSV format.

Spreadsheet applications make it easy to create, edit, and save import files. You can use standard features to add and delete rows of data, edit individual fields, search for records, or sort the list. The following illustration shows an example of *group* account data defined in a Microsoft Excel file. The layout lets you easily review the data.



When you save your spreadsheet as type CSV (*.csv), a comma separates the values in each row. For example, the following illustration shows the group data from the Microsoft Excel spreadsheet, saved as CSV file, and opened in Notepad.



List of User Attributes for CSV Column Headers

Oracle Identity Cloud Service provides a list of user attributes that you can use as column headers while importing or exporting user accounts using a comma-separated values (CSV) file.

The following list of user attributes are supported:

User Name	Created Date
Active	Formatted Name
Work Address Street	Honorific Prefix
Work Address Locality	First Name
Work Address Region	Middle Name
Work Address Postal Code	Last Name
Work Address Country	Honorific Suffix
Work Address Formatted	Nick Name
Home Address Street	Work Phone
Home Address Locality	Mobile No
Home Address Region	Home Phone
Home Address Country	Fax
Home Address Postal Code	Pager
Home Address Formatted	Other Phone
Other Address Street	Recovery Phone
Other Address Locality	Primary Phone Type
Other Address Region	Preferred Language
Other Address Country	Profile Url
Other Address Postal Code	Time Zone
Other Address Formatted	Title
Primary Address Type	User Type
Display Name	Cost Center
Work Email	Department
Home Email	Division
Primary Email Type	Employee Number
Other Email	Manager
Recovery Email	Organization Name
Work Email Verified	ByPass Notification

Home Email Verified	Federated
Other Email Verified	Locked
Recovery Email Verified	Locked Reason
External Id	Locked Date
Locale	Password

Sample Files

To assist you to bulk load data, Oracle provides sample files for you to use. You can download the compressed sample files in the Identity Cloud Service console or from a link provided by Oracle. Whether you download the sample files from the Identity Cloud Service console or from a link provided by Oracle, the sample files are the same.

To download the sample files from the Identity Cloud Service console, click the Download sample file link.

To create an import file, you can use a standard spreadsheet application, such as Microsoft Excel or Google Sheets, or you can use a text editor, such as Notepad or TextPad.

Important:

If you're using the sample file to import application role memberships, then make sure the column headings are **Entitlement Value**, **Grantee Name**, and **Grantee Type** (instead of **Display Name**, **Member**, and **Member Type**). If the column headers aren't correct, then change them accordingly.

Also, if you exported application role memberships before version 17.2.2 of Oracle Identity Cloud Service, and you want to import them back into Oracle Identity Cloud Service, then you must change the column headers before doing so.

Tip:

First import the appropriate sample file with the sample data to familiarize yourself with the process. When you are comfortable with the process, delete the sample data, and then import live data.

Workflow

Before you start bulk loading data, make sure that you understand the typical bulk loading data workflow.

Workflow is described in Typical Workflow for Bulk Loading Data.

Deactivate Notifications

While you are testing, deactivate notifications so that users don't receive unnecessary notifications.

You can deactivate all notifications or you can choose which notifications are enabled and which notifications are not enabled. See Deactivate Notifications.

Test Bulk Loading Data

Test bulk loading data with a small sample set to ensure that the import file is successfully configured.

After successful testing, you can then import live data.

View Jobs and Job Details

Review the overall status of all jobs, the details for a specific job, and download a job file on the **Jobs** page.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Jobs**.
2. Use the following options to search for a specific job or to filter the job results that currently display:
   - To search for a specific job, enter your search criteria in the search field.
   - To specify a custom date range, click **Date Range**. To activate a date picker tool to select this range, click the **Calendar** icon in the **Start Date** and **End Date** fields.
   - To display jobs with only a specific status, choose a status from the **Filter by Status** drop-down list.
3. (Optional) To view the details of any job, in any job row, click **View Details**.
4. (Optional) To download a job file, in the **Job Details** view, click **Download**.

Export Job Errors

To help you to resolve errors, export a list of the job errors.

To make it easier to review and correct errors, you can export those errors to a CSV file on your local machine.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Jobs**.
2. Locate the specific job for which you want to export errors.

   You can only export errors for a job with a **Completed with Errors** status. Jobs with a **Failed** status do not have errors to export.
3. Click **View Details**.
4. Click **Export Errors**.

   A comma-separated value (CSV) file downloads to your local machine. The CSV file contains a record for each error that includes the error type and the error description.

7

Run Oracle Identity Cloud Service Reports

Learn about the types of reports available and how to view Oracle Identity Cloud Service reporting data.

Topics:

Typical Workflow for Running Oracle Identity Cloud Service Reports

With the reporting feature in Oracle Identity Cloud Service, you can run user, application, and diagnostic data reports.

Task	Description	Additional Information
Understand the types of reports.	Learn about the types of reports that you can run. Understand the data reported in each report type, and discover what you can filter the data in a report to focus on the data you want.	Understand the Types of Reports
Run reports.	From the **Reports** page, you can run reports.	Run Reports
Organize the report data.	To improve efficiency, filter and sort the data for each report type.	Organize Report Data
Filter report data.	Apply filters to the report data.	Filter Report Data
Export report data	Export data as a CSV file.	Export Report Data

You can run user, application, and diagnostic data reports by using:

- The Identity Cloud Service console
- SCIM-based APIs

In the following sections, you learn how to run reports by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Understand the Types of Reports

As an audit administrator, identity domain administrator, or application administrator, you can run operational or historical reports that capture data about Oracle Identity Cloud Service users, applications, and diagnostic log levels.

The following reports are available.

- Audit Log: Capture system activity such as successful and failed logins, user creation, update and deletion, etc.
- Notification Delivery Status: View the email notification delivery status for events such as new users, self-initiated password changes, etc.
- Successful Login Attempts: View users who have logged in to Oracle Identity Cloud Service successfully.
- Unsuccessful Login Attempts: View users who have not logged in to Oracle Identity Cloud Service successfully.
- Dormant Users: View users who have not logged into Oracle Identity Cloud Service since a specified date.
- Application Access Report: View how many times users logged in to both Oracle Identity Cloud Service, and Oracle and custom applications in your identity domain.
- Application Role Privileges Reports: View application role grants and revocations for users and groups for applications that are configured in Oracle Identity Cloud Service.
- Diagnostic Data: View logging data captured in Oracle Identity Cloud Service.

You can access the Auditing Users, Groups, and Applications and Performing Self-Service Diagnostics infographics to see how to run user, application, and diagnostic data reports in Oracle Identity Cloud Service.

Administrators

Check which administrators can access different report types.

You must have the correct administator role for the type of report you want to view, run, or download.

Identity domain administrator

This is a super-user account, and the identity domain administrator can access all reports.

Security administrator

The security administrator can access the dormant users report.

Application administrator

The application administrator can access:

- The application access report
- The application role privileges report

Audit administrator

The audit administrator can access:

- Successful login attempts report
- Unsuccessful login attempts report
- Dorman users report
- The Application Access report
- The Application Role Privileges report

User administrator

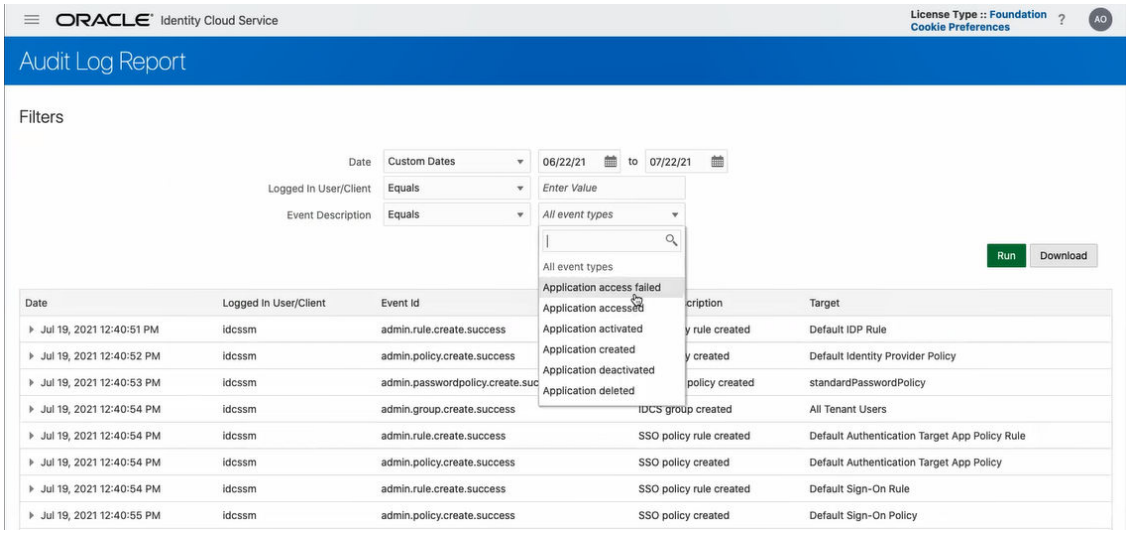
The user administrator can access:

- Successful login attempts report
- Unsuccessful login attempts report

Audit Log Report

The audit log captures system activity such as successful and failed logins, user creation, update and deletion, and so on. A number of different event types are captured, and you can search for specific types of event, or by date.

Example of an Audit Log



Data

The audit log report shows:

- The date and time of an event.
- The logged in user or client who caused the event.
- The event id.
- A description of the event.
- The target of the event.

Additional details

For each row in the report, you can click on > to expand details for that entry. The additional information for each row is:

- The Execution Context Id
- Client IP
- SSO Comments
- SSO Browser
- Matched Sign-On Policy Rule
- Authentication Level
- User's device information, that is, the device fingerprint
- Protected resource
- SSO Policy Obligation

Filtering the results

You can filter the audit log report to show:

- Results from a specific date range. Audit log events are only kept for 90 days, so you cannot search from earlier than 90 days ago.
- The logged in user or client. This is case sensitive and you must enter the user name exactly as it appears on the system.
- The description of the event. Start typing the name of the description, or choose from the list.

Audit Log Events

The following events are reported in the Audit Log:

- Application access failed
- Application accessed
- Application activated
- Application created
- Application deactivated
- Application deleted
- Application granted
- Application revoked
- Application updated
- Bypasscode created
- Group deleted
- IDCS group created
- MFA factor enrolled
- Notification delivered
- Notification not delivered
- Password changed
- Password policy created
- Password policy updated

- Password reset
- Password reset by Admin
- SSO policy created
- SSO policy rule cfreated
- SSO policy rule updated
- SSO policy updated
- User activated
- User added to group
- User created
- User deactivated
- User deleted
- User login
- User login failed
- User logout
- User removed from group
- User updated

Notification Delivery Status Report

Capture system activity such as successful and failed logins, user creation, update and deletion, and so on.

Data

The notification delivery status report shows:

- The email address of the recipient.
- The channel, for example, email.
- The notification delivery status, for example, Delivered.
- The date and time it was delivered.
- The description associated with the notification.

Filtering the results

You can filter the report to show:

- Results from a specific channel.
- The email address of the recipient.
- The notification delivery status.

Successful Login Attempts Report

You can use the successful login attempts report to view users who have logged in to Oracle Identity Cloud Service successfully.

Data

The successful login attempts report shows:

- The user name or client.

> **Note:**
>
> This will just show users who have logged into Oracle Identity Cloud Service using their Oracle Identity Cloud Service credentials (user name and password, or user name and second factor). User names of federated users logging in via an identity provider are not displayed.

- The date and time of the successful login.
- The provider.

Filtering the results

You can filter the report to show:

- Results from the last 30 days, the last 60 days, or the last 90 days.
- Results from a specific date range.

Unsuccessful Login Attempts Report

You can use the unsuccessful login attempts report to view users who have not logged in to Oracle Identity Cloud Service successfully.

Data

The unsuccessful login attempts report shows:

- The overall number of successful and unsuccessful logins
- The user name or client.

> **Note:**
>
> This will just show users who have logged into Oracle Identity Cloud Service using their Oracle Identity Cloud Service credentials (user name and password, or user name and second factor). User names of federated users logging in via an identity provider are not displayed.

- The date and time of the unsuccessful login.
- Any comments about the unsucessful login.

Filtering the results

You can filter the report to show:

- Results from the last 30 days, the last 60 days, or the last 90 days.
- Results from a specific date range.

Dormant Users Report

View users who have not logged into Oracle Identity Cloud Service since a specified date.

Data

The dormant users report shows:

- The user name or client.

> **Note:**
>
> This will just show users who have logged into Oracle Identity Cloud Service using their Oracle Identity Cloud Service credentials (user name and password, or user name and second factor). User names of federated users logging in via an identity provider are not displayed.

- The last successful login date.
- The full name associated with the user name or client.
- The primary email address for the account.

Filtering the results

You can filter the report to show:

- Results from a specific date range.
- The user name or client. This is case sensitive and you must enter it exactly as it appears on the system.

Application Access Report

You can use the application access report to view how many times users logged in to both Oracle Identity Cloud Service, and Oracle and custom applications in your identity domain.

Data

The application access report shows:

- The name of the user.
- The email address used in the login.
- Whether the action was a success or failure.
- The name of the application.
- The date and time of access or attempted access.

Filtering the results

You can filter the report to show:

- The name of the user.
- The login email.
- The name of the application.
- Results from a specific date range.

Application Role Privileges Report

You can use the application role privileges report to view application role grants and revokes for users and groups for applications that are configured in Oracle Identity Cloud Service.

Data

The application role privileges report shows:

- The name of the admin who approved the application role privilege.
- Name of the application where application role privilege has been granted or revoked.
- The name of the application role.
- Whether it is for a single user, or for a group.
- The date and time of when the privilege was granted or revoked.

Filtering the results

You can filter the report by:

- Approver.
- Application name.
- The user or group.
- The application role name.
- Results from a specific date range.

Run the Diagnostic Data Report

Use the Diagnostic Data report to view logging data captured in Oracle Identity Cloud Service for diagnostic purposes.

Data and filtering the results

The information reported in the Diagnostic Data report is:

- Correlation ID: The correlation identifire for the request.
- Type: The diagnostic level of the record.
- Message: The diagnostic message that has been recorded.
- Component: The name of the micro-service which raised the message.
- The timestamp when the diagnostic message has been recorded.

You can filter by any of these values, and also by the user name or client.

There are two steps to perform to get diagnostic data:

- Set the logging level at which you capture operational logs. You do this in the **Settings** menu.
- Then go to the Diagnostic Data report in **Reports** where the data is displayed.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Diagnostics**.
2. Click **Diagnostics Type** to set the Oracle Identity Cloud Service log level.
   - To capture high-level logging information only, select **Activity View**.
   - To capture both mid-level and high-level logging information, select **Data View**.
   - To capture detailed logging information, select **Service View**.
3. Toggle **Identify item in search results** on to identify the resources returned in the diagnostic log.
4. Click **Save** to activate data logging in Oracle Identity Cloud Service. You can view logging data captured over the next 15 minutes for diagnostic purposes.

> **Note:**
>
> After 15 minutes, the Oracle Identity Cloud Service log level reverts to **None** automatically.

5. In the Identity Cloud Service console, go to the **Reports** page.
6. In the **Reports** page, expand the **Diagnostics** node.
7. Click the **Diagnostic Data** report. Detailed report information appears.
8. Filter the data that appears in the Diagnostic Data report.
9. To download a comma-separated values (CSV) version of the report, click **Download Report**.

Organize Report Data

With Oracle Identity Cloud Service, you can organize the report data to increase your efficiency by:

- Filtering the report data: After you run a report, Oracle Identity Cloud Service displays the report data in tabular form, which can sometimes contain a large amount of data. Instead of scrolling through many report pages for the information that you need, refine the data by filtering it. For example, view all the report data that Oracle Identity Cloud Service recorded over a designated time interval. Or, customize a date or time range to see this data.
- Sorting the report data: Sort the report data in the table in ascending or descending order. Place the mouse pointer in a column heading to see an up-arrow button. Click the up-arrow button once to sort the data in ascending order, and click the button again to sort the data in descending order.

Filter Report Data

You can filter the report results to focus on a particular date, or a specific user, or the type of even recorded. The filters available depend on the type of report.

1. With the report open, user the filter fields to specify the results you want. You can see the filters for each report type in the description of that report.
2. Click **Run**.

The filtered report is displayed on the screen. You can sort the columns by clicking on the column headings.

Export Report Data

You can download report data for:

- Audit log report
- Successful and unsuccessful login reports
- Application access and application role privileges reports
- Diagnostic data report

Oracle Identity Cloud Service supports CSV report generation.

1. With the report open, apply any filters and click **Run**.
2. Click **Download**.
3. Choose a location for the download file, or have it open in Excel.

The report is created.

Run Reports

To run Oracle Identity Cloud Service reports, you must be assigned to the identity domain administrator role, the audit administrator role, or the application administrator role.

See Add or Remove a User Account from an Administrator Role for more information about assigning administrator roles to users.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Reports**.
2. Click on the report you want.

The report is displayed on the screen. You can use filters to search for specific information, and you can download the results.

8

Manage Oracle Identity Cloud Service Secondary Instances

Learn how to use the Identity Cloud Service console to manage Oracle Identity Cloud Service instances.

Topics:

Typical Workflow for Managing Oracle Identity Cloud Service Secondary Instances

With the instance management feature in Oracle Identity Cloud Service, you can create, update, and remove secondary Oracle Identity Cloud Service instances for your Oracle Cloud account.

Task	Description	Additional Information
About primary and secondary instances	Understand the concept of primary and secondary instances, including why you want to use multiple instances for your cloud services and applications.	About Primary and Secondary Service Instances
Create a secondary instance	Learn how to create secondary instances, using the **Instance Management** page.	Create a Service Instance
Update secondary instances details	Learn how to update secondary instance details, such as display name and the license type, using the **Instance Management** page	Update Secondary Instance Details
Remove secondary instances	Remove a secondary instance using the **Instance Management** page.	Remove a Service Instance

About Primary and Secondary Service Instances

Customers want to have separate Oracle Identity Cloud Service service instances to use with their cloud services and applications.

Multiple service instances can be used to match development and production environment segregation requirements or because you want to isolate your employees access from your customers.

Each Oracle Identity Cloud Service instance is completely different and isolated from other service instances. They have different users, groups, applications, and can have different identity and security requirements. Using separate service instances can help you maintain the isolation of administrative controls over each of them.

When multiple instances are utilized, you have a **primary** service instance and one or more **secondary** service instances. For example, a primary instance comes with your Oracle Cloud account and from within this instance console, the cloud account administrator can create one or more additional (**secondary**) service instances.

To create secondary service instances, you need to sign in as the cloud account administrator of a primary service instance or as the user specified during a primary service instance creation. Only this administrator can create secondary service instances and specify the identity domain administrators for them.

During the creation of a secondary service instance, you provide administrator credentials. This administrator becomes the identity domain administrator of the secondary service instance and has superuser privileges within the instance. Although the identity domain administrator of a secondary instance may have the same user name as a user in the primary instance, they are different users who might have different privileges in each instance, and will have separate passwords.

Regarding secondary instances, there are no new administrator or user processes to learn. The process to perform any administrative or user task in a secondary instance is identical to the process for performing it in the primary instance.

Important: The identity domain administrator of a secondary instance can't create secondary instances of Oracle Identity Cloud Service from their instance. The Instance Management feature is only available for the primary Oracle Identity Cloud Service instance within a cloud region.

Create a Service Instance

Create secondary Oracle Identity Cloud Service instances using the Instance Management feature.

Creating a secondary service instance includes assigning an identity domain administrator to the instance and selecting the license type for the instance.

1. Sign in to the Identity Cloud Service console of your primary instance as the cloud account administrator.
2. Expand the **Navigation Drawer**, and then click **Instance Management.**
3. In the **Instance Management** page, click **Add**.
4. In the **Add Instance** window, provide the following instance information, and then click **Save**:

Table 8-1 Add Instance information.

Parameter	Description
Display Name	The display name of the instance.

Table 8-1 (Cont.) Add Instance information.

Parameter	Description
Administrator's First Name	Identity domain administrator's first name for this instance.
Administrator's Last Name	Identity domain administrator's last name for this instance.
Administrator's User Name	Identity domain administrator's user name for this instance.
Administrator's Email	Identity domain administrator's email for this instance.
License Type	Select one of the license types available for Oracle Identity Cloud Service. See About Oracle Identity Cloud Service Pricing Models

Your new instance appear in the list with status of **Processing**. After the instance is created, the status changes to **Active**.

> **Note:**
>
> If the status changes to **Failed**, click the **Failed** icon to access the log files to learn why the instance couldn't be created.

Update Secondary Instance Details

You can modify details about a secondary service instance, including its **Display Name** and **License Type**.

1. In the **Instance Management** page, click the **Action** menu to the right of the instance you want to modify, and then click **Edit**.
2. In the **Edit Instance** window you can:
   - Change the value in the **Display Name** field.
   - Use the **License Type** menu to select a different license type for the secondary instance.
3. After you modified the values, click **Save**.

Remove a Service Instance

If you no longer need a service instance, then you can remove it.

1. In the **Instance Management** page, click the **Action** menu to the right of the instance you want to remove, and then click **Remove**.
2. In the **Confirmation** window, click **OK**.

Part III
Configure Administrator Settings

Learn how to configure important administrative settings.

Chapters

9

Change Oracle Identity Cloud Service Default Settings

Learn how to manage your default identity domain settings for Oracle Identity Cloud Service.

To manage default identity domain settings, you must be assigned to the identity domain administrator or security administrator role. See Add or Remove a User Account from an Administrator Role.

Topics:

Change Default Settings

Default settings are applied to your entire identity domain in the Cloud. You can specify settings such as the time zone, password recovery email, and language.

To open this page, you must be assigned the identity domain administrator role or the security administrator role.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Default Settings**.
2. Set the locale. Users can override the default time zone and language settings in the **My Profile Details** tab in the **My Profile** console.
   - To specify a default time zone, from the **Timezone Setting** drop-down list, select a time zone.
   - To specify a default language, from the **Language Setting** drop-down list, select a language.

   > **Important:**
   >
   > Choose the language of the target audience. Do not choose a country-specific language unless you are targeting a specific country. For example, choose *French* to display the text to all *French* users. Choose *French (Canada)* to display the text to all *Canadian French* users but not other *French* speaking users.
3. Set the **Access Signing Certificate** option.
   - Turn on this option to allow clients to access the tenant signing certificate and the SAML metadata without logging in to Oracle Identity Cloud Service.

- Turn off this option to prevent clients from accessing the tenant signing certificate and the SAML metadata until they authenticate by logging in to Oracle Identity Cloud Service.

4. In the **Email Addresses** field, provide the default contact email addresses.

   These email addresses appear in notifications sent to users. Enter the email addresses that you want users to contact if they need help. To separate multiple email addresses, use a comma.

5. Select **Audit Retention Interval** as either 30, 60 or 90 days. The tenant will purge the audit data for all the users, based on the interval set here. As an administrator, when you delete a user, you can manually purge the audit data of that user by entering the GUID. The entire audit data of that user will be deleted permanently from the Tenant.

6. Click **Save**.

Purge Audit Data for the Deleted User

When you delete a user, the audit data of the user remains in the system. Using Purge option, you can manually and immediately purge the audit data of that deleted user.

To purge the audit data of the deleted user, perform the following procedure:

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Default Settings**.
2. Under the **Audit** section, in the **Purge audit data for the deleted user** text box, enter the GUID of the deleted user and click **Purge**.

Access SAML Metadata

When setting up SSO with a SAML Identity Provider or a SAML Application, you need to provide your Oracle Identity Cloud Service domain's SAML configuration details to the partner provider. This information is typically contained in an XML document called SAML metadata.

Oracle Identity Cloud Service offers two ways to download SAML metadata: a button in the Identity Cloud Service console, or directly accessing an endpoint URL. In most scenarios, the simplest method is to click the button in the Identity Cloud Service console. However, if you need options not supported by the Identity Cloud Service console, such as the `adfsmode="true"` query parameter, you need to directly access the metadata endpoint URL.

Method	Instructions	References
Method One Download the metadata from the Identity Cloud Service console. This is the simplest way to obtain the SAML metadata for your Oracle Identity Cloud Service domain. Use this method, whenever possible.	For a SAML Application, click the **Download Identity Provider Metadata** button for the partner SAML Application. For a SAML Identity Provider, click the download button for **Service Provider Metadata**.	See • Add a SAML Application • Add a SAML Identity Provider • Add an App Catalog Application

Method	Instructions	References
Method Two Make the metadata URL publicly accessible. Use this method, for example, if the administrator of your partner Identity Provider or SAML Application is not an Oracle Identity Cloud Service Identity domain administrator. You can also use this method if your partner Identity Provider or SAML Application is able to automatically retrieve your Oracle Identity Cloud Service domain's metadata, by using a configured URL.	Turn on the **Access Signing Certificate** option under **Default Settings** in the Identity Cloud Service console. Once you turn the option on, `https://<IDCS-Service-Instance>.identity.oraclecloud.com/fed/v1/metadata` will be accessible in the browser without authentication.	See Change Default Settings.
Method Three Generate an OAuth access token and use an authenticated GET request to the `/fed/v1/metadata` endpoint, using cURL or another REST client.	If the identity domain administrator doesn't want to make its metadata URL publicly accessible, they can access the metadata by passing a valid Oracle Identity Cloud Service `access_token` in the HTTP Authorization header, by using a tool such as cURL or Postman.	See Generate Access Token and Other OAuth Runtime Tokens for more information regarding how to get and use an access token. See Using the Postman Collection and Using cURL for more information regarding how to invoke Oracle Identity Cloud Service REST APIs.
Method Four Download the SAML metadata for Active Directory Federation Services (ADFS) using a URL.	1. Make the metadata URL publicly accessible using **Method Two** above. 2. Navigate to the metadata URL `https://<IDCS-Service-Instance>.identity.oraclecloud.com/fed/v1/metadata?adfsmode=true` using your browser, replacing `<IDCS-Service-Instance>` with your Identity Cloud Service tenant ID. 3. Save the file locally on your computer. Do not copy from the browser window and paste the contents into a file. 4. Optionally, revert the public accessibility of your metadata URL.	No references.

Obtain the Root CA Certificate from Oracle Identity Cloud Service

When you setup Service Providers and Identity Providers for Federated SSO, you need to download the metadata file and the signing and encryption certificates. However, these

certificates are not self-signed and are issued by a root certificate. Hence, for a proper setup and function, you need to get the root certificate and install it at the Federation partner. Follow the procedure below to obtain the root certificate.

To obtain the root CA certificate:

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Default Settings**.
2. Turn on the switch under Access Signing Certificate to enable clients to access the tenant signing certificate without logging in to Oracle Identity Cloud Service.
3. Click **Save** to save the default settings.
4. Refer to the REST API document for the detailed installation of cURL.
5. Use this URL `https://tenant-base-url/admin/v1/SigningCert/jwk` as the endpoint.
6. Execute the following cURL command to save the root certification file:

   ```
   curl -k -i -H "Accept: application/scim+json,application/json" --request GET
   "https://tenant-base-url/admin/v1/SigningCert/jwk"
   ```

   After you execute the command, the following code is returned:

7. Open a Notepad and paste the key in the following manner:

   ```
   -----BEGIN CERTIFICATE-----
   [Paste the highlighted key here]
   -----END CERTIFICATE-----
   ```

   For example (abbreviated):

   -----BEGIN CERTIFICATE-----

   "MIIDdDCCAlygAwIBAgIGAVw4Ns68MA0GCS...../VaWgoMQ6J9t9CLarai"

   -----END CERTIFICATE-----

8. Save this file as your root certification file.

10

Manage User Settings in Oracle Identity Cloud Service

This section describes how to manage user settings in Oracle Identity Cloud Service.

Topics:

Typical Workflow for Managing User Settings in Oracle Identity Cloud Service

With the user settings feature in Oracle Identity Cloud Service, you can perform tasks such as changing user settings. For example, you can make the primary email address for a user account a required or optional attribute.

Task	Description	Additional Information
Change user settings.	Change settings for Oracle Identity Cloud Service user accounts using the **User Settings** page.	Change User Settings

You can manage user settings by:

- The Identity Cloud Service console
- SCIM-based APIs

The following sections describe how to manage user settings by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Change User Settings

You can change settings associated with user accounts. For example, you can make the primary email address for a user account a required or optional attribute.

By making the primary email address optional, if Oracle Identity Cloud Service integrates with another cloud service or on-premises application, then a user's email address can be propagated from that service or application back into Oracle Identity Cloud Service, and designated to be the user's primary email address in Oracle Identity Cloud Service.

To change user settings, you must be assigned to the identity domain administrator role or the security administrator role.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **User Settings**.
2. If you don't want the primary email address to be a required attribute for user accounts, then turn on the **Allow primary email address as optional** switch.

   OR

   If you want the primary email address to be a required attribute, then turn off this switch.
3. Click **Save**.
4. In the **Confirmation** window, click **OK**.

> **Note:**
>
> You must sign out and sign back in to the Identity Cloud Service console for the change you made to the **Allow primary email address as optional** switch to take effect.

11

Manage Oracle Identity Cloud Service Trusted Partner Certificates

Learn how to manage trusted partner certificates for Oracle Identity Cloud Service.

Topics

Typical Workflow for Managing Oracle Identity Cloud Service Trusted Partner Certificates

With the trusted partner certificate feature in Oracle Identity Cloud Service, you can perform tasks such as importing, viewing, and deleting trusted partner certificates.

Task	Description	Additional Information
Understand trusted partner certificates.	Learn about trusted partners and trusted partner certificates.	Understand Trusted Partner Certificates
Import a trusted partner certificate.	You can import a trusted partner certificate using the **Trusted Partner Certificates** page.	Import a Trusted Partner Certificate
View details about a trusted partner certificate.	View details about a trusted partner certificate using the **Trusted Partner Certificates** page.	View Details About a Trusted Partner Certificate
Delete a trusted partner certificate.	Delete a trusted partner certificate using the **Trusted Partner Certificates** page	Delete a Trusted Partner Certificate

You can import, view, and delete trusted partner certificates by using:

- The Identity Cloud Service console
- SCIM-based APIs

In the following sections, you learn how to manage trusted partner certificates by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Understand Trusted Partner Certificates

In this topic, you learn about trusted partners and trusted partner certificates.

A **trusted partner** is any application or organization, remote to Oracle Identity Cloud Service, that communicates with Oracle Identity Cloud Service.

Oracle Identity Cloud Service uses identity propagation to communicate with a trusted partner. During identity propagation, a front-end Oracle Identity Management product, such as Oracle Access Manager, challenges a user and authenticates the user's credentials.

After the user's identity is validated, a token is generated. This token is used in place of a password to prove that the user is who he or she claims to be. The asserted identity is then passed into Oracle Identity Cloud Service. Because the identity has already been established, Oracle Identity Cloud Service trusts that it is a valid user identity, and can use it, as required.

For example, Oracle Identity Cloud Service receives a user assertion from Oracle Access Manager. As a result, a user can use Oracle Access Manager to log in to a portal associated with a trusted partner. This portal takes the user to the Home page of an order management system. The Home page displays the orders the user made from the order management system.

The first step in establishing a trusted partner is to determine the partner's role in the trust relationship. A trusted partner can be a source site (one that generates an SSO assertion) or a destination site (one that consumes an SSO assertion).

Currently, trusted partners generate SSO assertions that Oracle Identity Cloud Service consumes.

To ensure that the assertions are transmitted to Oracle Identity Cloud Service securely, the information contained in the assertions is encrypted in X.509 digital certificates. These certificates are known as **trusted partner certificates**.

Oracle Identity Cloud Service uses trusted partner certificates that have Distinguished Encoding Rules (DER) file extensions.

Enable X.509 Certificate Authentication

Prerequisites

- Enable **X.509 Certificate Authentication** or the **OAuth2 TLS** grant type. This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service.
- Import a trusted partner certificate. See Import a Trusted Partner Certificate.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Partner Settings**.
2. Turn on **OCSP Validation** and complete the following fields.
   - **OCSP Responder URL**: Enter the OCSP Responder URL.
   - **Allow Access if OCSP response is UNKNOWN**: Select this checkbox to allow access for unknown certificates.
   - **Signing Certificate Alias**: Select a partner certificate alias.
3. Click **Save**.

Import a Trusted Partner Certificate

You can use Oracle Identity Cloud Service to import a trusted partner certificate. To import the certificate, use a Distinguished Encoding Rules (DER) file.

See Understand Trusted Partner Certificates for more information about trusted partner certificates.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Partner Settings**.
2. In the **Trusted Partner Certificates** page, click **Import**.
3. In the **Import** window:
   a. In the **Alias** field, enter an alias for the trusted partner certificate (for example, `TPcert1`).

      The certificate that you import is an authorization certificate for the trusted partner. It contains a *keystore*. The keystore is used to authenticate and encrypt the data for the trusted partner for security purposes. A keystore entry is identified by an *alias*.

   b. To locate and select the DER file that contains the trusted partner certificate to import, click **Browse**.
   c. Verify that the path and name of the DER file you selected appear in the **Certificate** field.
   d. Click **Import**.

View Details About a Trusted Partner Certificate

After importing a trusted partner certificate into Oracle Identity Cloud Service, you can view details about it.

By default, you can see the alias, SHA-1 and SHA-256 thumbprints, start date, and end date for each certificate that you import into Oracle Identity Cloud Service. You can see either the abbreviated version of a certificate (the *thumbprint*) or the entire certificate.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Partner Settings**.
2. In the **Trusted Partner Certificates** page, verify that you see the following information about the imported trusted partner certificate:

Task	Description
Alias	The alias for the trusted partner certificate. See Import a Trusted Partner Certificate for more information about the certificate alias.
SHA-1 Thumbprint	A hash value computed over the complete certificate, which contains all its fields, including the signature. If SHA-1 is used as the algorithm to encrypt the certificate, then the encrypted value appears in this column. Otherwise, the column is empty.
SHA-256 Thumbprint	If SHA-256 is used as the algorithm to encrypt the certificate, then the encrypted value appears in this column. Otherwise, the column is empty.

Task	Description
Certificate Start Date	The date and time after which Oracle Identity Cloud Service can use the certificate to authenticate the trusted partner.
Certificate End Date	The date and time after which Oracle Identity Cloud Service can no longer use the certificate to authenticate the trusted partner.

3. You can view the entire trusted partner certificate, as opposed to the certificate's thumbprint. To do so, select the certificate and click **View**. After viewing the certificate, click **OK**.

Delete a Trusted Partner Certificate

You can use Oracle Identity Cloud Service to remove a trusted partner certificate.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Partner Settings**.
2. In the **Trusted Partner Certificates** page, click the certificate that you want to remove, and click **Delete**.
3. In the **Confirmation** window, click **OK**.

12

Customize Oracle Identity Cloud Service Notifications

Learn how to customize notifications for Oracle Identity Cloud Service users and administrators.

Topics:

Typical Workflow for Customizing Oracle Identity Cloud Service Notifications

With the notification feature in Oracle Identity Cloud Service, you can customize and use notifications.

Task	Description	Additional Information
Understand the types of notifications.	You can learn about the types of Oracle Identity Cloud Service notifications that you can customize for users and administrators.	Understand the Types of Notifications
Understand how to customize notifications.	You can examine a workflow that illustrates how to customize notifications in Oracle Identity Cloud Service.	Understand How to Customize Notifications
Activate notifications.	You can activate notifications using the **Notifications** page.	Activate Notifications
Select notifications.	You can select notifications using the **Notifications** page.	Select Notifications
Specify recipients for notifications.	You can specify recipients for notifications using the **Notifications** page.	Specify Recipients for Notifications

Task	Description	Additional Information
Modify notification templates.	You can modify notification templates using the **Notifications** page.	Modify Notification Templates
Verify notifications.	You can verify the customizations that you make to Oracle Identity Cloud Service notification templates.	Verify Notifications
Deactivate notifications.	You can deactivate notifications using the **Notifications** page.	Deactivate Notifications

You can customize and use notifications by using:

- The Identity Cloud Service console
- SCIM-based APIs

In this section, you learn how to customize notifications by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Understand the Types of Notifications

You can customize email notifications in Oracle Identity Cloud Service for users and administrators.

> **Tip:**
>
> In addition to customizing notifications for other Oracle Identity Cloud Services users and administrators, you can also view and act on email notifications that require your attention. To access these notifications, click **Notifications** in the upper-right corner of the Console. See Get Started with Oracle Identity Cloud Service.

Topics:

- About User Notifications
- About Administrator Notifications

About User Notifications

Learn about the various user notifications available in Oracle Identity Cloud Service.

The following user notifications are available in Oracle Identity Cloud Service:

Name	Description
Welcome	A user is notified that an administrator created an account for the user. The notification contains a link that the user clicks to activate the account.

Name	Description
Self-Registration Email Verification	After a user creates an account successfully through the self-registration process, this notification is sent to the user to verify the user's email address.
Welcome Federated SSO User	A federated SSO user is notified that an administrator created an account for the user. The notification contains a link that the user clicks to activate the account.
Welcome Delegated Authentication User	A user whose authentication is delegated is notified that an administrator created an account for the user. The notification contains a link that the user clicks to activate the account.
Resend Welcome	If a user doesn't activate the account using the link provided in the Welcome notification, then the administrator can send this notification. The user is notified again that the administrator created the account for the user. The notification contains a link that the user clicks to activate the account.
Resend Welcome To Delegated Authentication User	If a user whose authentication is delegated doesn't activate the account using the link provided in the Welcome Delegated Authentication User notification, then the administrator can send this notification. The user is notified again that the administrator created the account for the user. The notification contains a link that the user clicks to activate the account.
Password Recovery Request	This notification is sent to a user if the user requests a password reset. This notification contains a URL that the user clicks to be redirected to the **Password Reset** page. The user provides a password as part of the password recovery process. After the activation process is complete, the user is logged in automatically.
Recovery Email Verification	After a user changes their password recovery email address, this notification is sent to the user to verify the address.
Primary Email Verification	After a user changes their primary email address, this notification is sent to the user to verify the address.
Secondary Email Verification	After a user changes their secondary email address, this notification is sent to the user to verify the address.
Password Change	This notification is sent to the user to inform the user that the password was changed successfully. This event is initiated by the user.
Password Reset	This notification is sent to the user to inform the user that the password was reset successfully. This event is initiated by the user.
Password Has Been Changed by an Administrator to a Known Value	This notification is sent to users when the administrator changes the passwords for users to a known value. This notification is used for testing purposes only. Both the administrator and the users know the common password.

Name	Description
Admin Requesting a Password Reset on Behalf of a User	This notification is sent to a user if the administrator initiates changing the password for the user. The system creates a randomly generated value for the password. If the administrator initiates resetting the password for the user, then a notification is sent to the user along with a URL where the user can reset the password.
User Activation	A user is notified that an administrator activated the user's account. The notification contains a link that the user clicks to log in to the account.
User De-activation	A user is notified that an administrator deactivated the user's account.
User Account Locked	This notification is sent to a user if the user account is locked because the user was unsuccessful in logging in after a consecutive number of attempts. This notification contains a link that the user can click to unlock the account.
Exceeded Maximum Number of Account Recovery Attempts	After a user exceeds the maximum number of attempts to reset their password to recover their account, this notification is sent to the user’s primary email address.
User Account Unlocked	This notification is sent to a user after the user's account is unlocked. This occurs after the user accesses the link in the User Locked notification to unlock the account.
User Profile Updated by Administrator	An administrator can update a user's profile by changing attribute values associated with the user's account. A notification is sent to the user. A user can modify their profile and receive the same notification. The user accesses the **My Profile** page to see the modifications made to the profile. The changes appear in a different foreground or background color.
User Profile Replaced by Administrator	An administrator can replace attribute values of a user's profile. A notification is sent to the user. A user can replace attribute values of their profile and receive the same notification. The user accesses the **My Profile** page to see the attribute value replacements made to the profile. The changes appear in a different foreground or background color.
Device Enrollment Request to Enable 2-Step Verification	This notification contains instructions and links about how to download the Oracle Mobile Authenticator app. It also has an enrollment URL. After the user downloads the app, the user taps the enrollment URL to configure the user account in the app.
2-Step Verification User Account Locked	This notification is sent to a user if the user account is locked because of unusual activity detected on the account as part of the two-step verification process.
2-Step Verification Federated SSO User Account Locked	This notification is sent to a federated SSO user if the user account is locked because of unusual activity detected on the account as part of the two-step verification process.

Name	Description
2-Step Bypass Code Verification	This notification contains a bypass code that is generated by the administrator or user. The user can use this bypass code to complete the two-step verification process.
Enable Kerberos Authentication Request	This notification is sent to a user who's assigned to a Kerberos application for the first time. By clicking the link in the notification, the user logs into Oracle Identity Cloud Service, which enables generation of long-term keys. This is a prerequisite for Kerberos authentication. The user can then use the principal name provided in the notification and the Oracle Identity Cloud Service password to access the Kerberos application to perform authentication to applications that support it.
New Access Request Submitted	This notification is sent to a user after they submit an access request.
Access Request Fulfilled	This notification is sent to a user after their access request has been fulfilled.
2–Step Email One-Time Passcode Verification	This notification contains a one-time passcode (OTP) that’s sent to a user. The user uses this OTP to complete 2–Step Verification.
New Device Login Detected with Your Account	If an attempt is made to log in to a user's account from a device, IP address, or web browser, and Oracle Identity Cloud Service doesn't recognize that the device, address, or browser is associated with the account, then this notification is sent to the user. The notification contains a link that the user can click to reset their SSO password in case the user doesn't recognize the login attempt.

About Administrator Notifications

Learn about the various administrator notifications available in Oracle Identity Cloud Service.

The following administrator notifications are available in Oracle Identity Cloud Service.

> **Note:**
>
> All Email addresses specified in the Contact field (under Settings, Default Settings) receive these notifications. Also note that this list of email addresses is also used for other notifications sent out from Oracle Identity Cloud Service.

Name	Description
Job Has Been Started	An administrator is notified that a job for importing or exporting Oracle Identity Cloud Service groups, users, or application roles, or for resetting passwords for all Oracle Identity Cloud Service users, has been started.

Name	Description
Job Has Been Canceled	An administrator is notified that a job for importing or exporting Oracle Identity Cloud Service groups, users, or application roles, or for resetting passwords for all Oracle Identity Cloud Service users, has been canceled.
Job Is Complete	An administrator is notified that a job for importing or exporting Oracle Identity Cloud Service groups, users, or application roles, or for resetting passwords for all Oracle Identity Cloud Service users, is complete.
Job Has Failed	An administrator is notified that a job for importing or exporting Oracle Identity Cloud Service groups, users, or application roles, or for resetting passwords for all Oracle Identity Cloud Service users, has failed.
Quota Limit Exceeded	This notification is sent to an administrator when the administrator has exceeded the allowed resource quota for the Oracle Identity Cloud Service instance. To increase the quota limit, upgrade to Oracle Identity Cloud Service Basic or Oracle Identity Cloud Service Standard.
From Email Domain Validation Initiated	An administrator is notified that validation of the email domain that's entered in the email address in the **From Email Address** field on the **Notifications** page has been initiated, and a validation email will be sent to the postmaster account of this domain.
Email Address Validation Initiated for From Email Address	An administrator is notified that validation of the email address that's entered in the **From Email Address** field on the **Notifications** page has been initiated, and a validation email will be sent to this email address.
Synchronization Job Summary	After synchronizing users, groups, application accounts, and entitlements from an application into Oracle Identity Cloud Service, an administrator receives an email notification. The notification contains a summary of the synchronization and a link. Clicking the link takes the administrator to the **Import Results** page. In this page, the administrator can view the status of the synchronization job.
Notify an administrator when connectivity between AD-AD bridge-IDCS Server is broken	An administrator is notified that connectivity between AD, the AD Bridge, and Oracle Identity Cloud Service is broken.
Notify an administrator when connectivity between AD-ADBridge-IDCS Server is restored	An administrator is notified that connectivity between AD, the AD Bridge, and Oracle Identity Cloud Service is restored.
Notify an administrator when an update for AD Bridge is available	An administrator is notified that an update for AD Bridge is available for download.
Notify an administrator when sync between AD-ADBridge-IDCS Server is successful	An administrator is notified that an AD Bridge sync has successfully completed. The email contains detail information such as the number of users and groups imported, the number users and groups that failed to import, and the number of users delinked as well as the number that failed to delink.

Name	Description
Notify an administrator when sync between AD-ADBridge-IDCS Server has failed	An administrator is notified that an AD Bridge sync has failed.

> **Note:**
>
> The Job Has Been Started, Job Has Been Canceled, Job is Complete, and Job Has Failed administrator notifications contain a link. Clicking the link for each notification takes the administrator to the **Jobs Status** page of the console where the administrator can view details about the job.

Understand How to Customize Notifications

Oracle Identity Cloud Service provides you with email templates for user and administrator notifications.

See About User Notifications for a listing of these notification templates.

You can tailor the recipients and content of these templates to meet the business and security requirements for your enterprise applications.

The following workflow illustrates how to customize notifications in Oracle Identity Cloud Service:

1. Activate Notifications. By activating notifications, you enable Oracle Identity Cloud Service to send notifications to users and administrators. See Activate Notifications.
2. Select Notifications. After activating notifications in Oracle Identity Cloud Service, you can select notifications to customize. See Select Notifications.
3. Specify Recipients for Notifications. After activating and selecting notifications, you can configure Oracle Identity Cloud Service to send the notifications either to all Oracle Identity Cloud Service users or to a limited number of recipients (for testing purposes). See Specify Recipients for Notifications.
4. Modify Notification Templates. After activating and selecting notifications, and specifying their recipients, you can modify the notification templates that you selected. See Modify Notification Templates.
5. Verify Notifications. You can verify the customizations that you made to the Oracle Identity Cloud Service notification templates you selected. See Verify Notifications.
6. Deactivate Notifications. By deactivating notifications, you prevent Oracle Identity Cloud Service from sending notifications to users and administrators. See Deactivate Notifications.

You can access the Customizing the Service infographic to see how to customize notifications.

Activate Notifications

By activating notifications, you enable Oracle Identity Cloud Service to send notifications to users and administrators.

To activate notifications, you must be assigned to either the identity domain administrator role or the security administrator role. See Add or Remove a User Account from an Administrator Role.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Notifications**.
2. Turn on the **Status** switch.
3. To send a validation email to the postmaster account of the email's domain, click **Domain**. After the domain is verified, any email address from the domain is valid.

   OR

   To send a validation email to the email address that you enter in the **From Email Address** field, click **Email**. Unlike the **Domain** option, a validation is initiated for every email address even though they're from the same domain.
4. In the **From Email Address** field, enter the email address that will appear in the **From Email** field for all notifications.
5. Click **Save**.

   If you selected **Domain** in step 2, then an email notification will be sent to the postmaster to verify the email address and validate the domain associated with the address. Otherwise, if you selected **Email**, then a validation email will be sent to the email address that you entered in the **From Email Address** field.
6. In the **Confirmation** window, click **OK**.
7. If you see a **Pending Domain Verification** or **Pending Email Verification** status, then click **Check Status**.

   Oracle Identity Cloud Service checks whether verification is done to the email address through the email sent to the postmaster or email account. If it's verified, then the status changes from **Pending Domain Verification** to **Domain Verified** or from **Pending Email Verification** to **Email Verified**. If it's not verified, then the status remains as **Pending Domain Verification** or **Pending Email Verification**.
8. If the email address isn't verified, then access the notification that's sent to the email address you provided, click the verification link in the notification, and click **Check Status** again. The status will change to **Email Verified**.

   OR
9. If the domain isn't verified, then contact the postmaster of your company so that the postmaster can verify the domain associated with the email address.

Select Notifications

After activating notifications in Oracle Identity Cloud Service, you can select notifications to customize.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Notifications**.
2. Click **Configure**.

On this tab, you see a "master list" of user and administrator notifications that you can select.

3. Select the check box for each notification that you want to customize.
4. Click **Save**.
5. In the **Confirmation** window, click **OK**.

Specify Recipients for Notifications

After activating and selecting notifications, you can configure Oracle Identity Cloud Service to send the notifications either to all Oracle Identity Cloud Service users or to a limited number of recipients (for testing purposes).

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Notifications**.
2. Click **Recipients**.
3. Set the **Limited Recipients List** switch.
   - To send the notifications you selected to all Oracle Identity Cloud Service users, turn the switch **Off**.
   - To send the notifications to a limited number of recipients, turn the switch **On**. In the **Testing Email Addresses** text area, enter the email addresses of the users who will receive the notifications. Use commas to separate email addresses.
4. Click **Save**.
5. In the **Confirmation** window, click **OK**.

Modify Notification Templates

After activating and selecting notifications, and specifying their recipients, you can modify the notification templates that you selected. To meet the business and security requirements for your enterprise applications, tailor the content of these notifications.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Notifications**.
2. Click **Email Templates**.

> **Tip:**
>
> The notification templates that appear in the **Email Templates** tab reflect the selections you made in the **Configure** tab. See Select Notifications.

3. Expand a notification template by clicking the right-arrow button that's associated with the template.
4. To modify a notification template, use the following table:

Important:

When adding or changing a variable in an email template, ensure that you use the correct syntax. For example, to use the variable *user.displayName*, the correct syntax is *${user.displayName}*.

Field	Description
Language	Select the language for the notification. To see a preview of the notification in the language you select, click **View** and access the **Email Body** text area.
Encoding	Verify that UTF-8 appears as the character encoding for the notification (for security, encryption, and backward-compatibility purposes). This character encoding can encode all possible characters of the notification, or code points, in Unicode.
From Email	Verify that the email address for this field matches the email address you entered in the **From Email Address** field. This field is the email address that will appear in the **From Email** field for all notifications. If you haven't verified the domain or the email address, then this value will be the previously validated email address or email address from the previously validated domain. As soon as the domain or the email address is validated (the status changes from **Pending Domain Validation** or **Pending Email Verification** to **Domain Verified** or **Email Verified**), then the verified email address will appear in the **From Email Address** field for all notifications.
Subject	Enter or provide variables for content that appears in the **Subject** field of the email notification.

Field	Description
Email Body	The content of the notification template. In this text area, you can customize the content to meet your needs. In addition to a customization toolbar, Oracle Identity Cloud Service provides you with variables to use. These variables are replaced with values specific to your business at runtime. They are: • *${account.emailID}*: The email address of the user's account from which an attempt is made to log in using a device, IP address, or web browser that Oracle Identity Cloud Service doesn't recognize. • *${actorDisplayName}*: The identity domain administrator's email address • *${admin.resource.name}*: The name of the Kerberos application • *${app.displayName}*: The display name of the application that contains the users, groups, application accounts, and entitlements that are synchronized into Oracle Identity Cloud Service • *${app.id}*: The ID of this application • *${authentication.targetApp}*: The name of the Microsoft Active Directory domain that contains the account of the user who's authenticating into Oracle Identity Cloud Service • *${bypasscode.expiry}*: The time (in minutes) before a bypass code expires • *${bypasscode.usage}*: How many times a bypass code can be used • *${bypasscode.value}*: The bypass code that the user or administrator generates for use as part of the 2-Step Verification process • *${companyName}*: The name of the company that will appear in the notification **Note:** When you use the *${companyName}* variable, be sure to add your company name to the **Company Name** field in the **Branding** page. If you don't, then your company's details won't appear in email notifications, SMS notifications, or in the Oracle Mobile Authenticator (OMA) app when a user completes MFA enrollment. See Customize the Sign In Page for more information about populating the **Company Name** field.

Field	Description
	• *${contactEmails}*: The system administrator's email address • *${date}*: The date associated with the action of the notification (for example, resetting a password) • *${device.agent}*: User agent information. • *${device.enrollmentURL}*: The configuration URL containing parameters used to configure the Oracle Mobile Authenticator app • *${device.ipAddress}*: The IP address from which an attempt is made to log in to a user's account, but which Oracle Identity Cloud Service doesn't recognize. • *${device.location}*: Device location. • *${domain}*: The realm (or domain) that contains the Kerberos application • *${email}*: The email address that appears in the **From Email Address** field • *${emailId}*: The user's email address • *${end.dateTime}*: The date and time at which the job to synchronize users, groups, application accounts, and entitlements from an application into Oracle Identity Cloud Service finished • *${footerImage}*: The image that will appear in the footer region of the notification • *${headerImage}*: The image that will appear in the header region of the notification • *${homePageRedirectUrl}*: The redirect URL for the notification that can be used if the link in the notification doesn't work. This URL redirects users to the **Home** page of Oracle Identity Cloud Service. • *${job.displayName}*: The display name of the job that's started, canceled, completed, or failed • *${job.historyId}*: The ID number of the job that's started, canceled, completed, or failed • *${kerberos.principalName}*: The Kerberos principal name that the user uses to access the Kerberos application to perform authentication to applications that support it • *${linkExpirationTime}*: A date-and-time stamp, after which the link in the notification will be expired • *${masked_UID}*: The account of the user who requests a one-time passcode (OTP) to enroll in 2–Step Verification. • *${OTP}*: The one-time passcode (OTP) that’s sent to a user for the user to complete 2–Step Verification. • *${quota.limit}*: The allowable quota limit for the resource type. If an administrator can create 500,000 user accounts, then 500,000 represents the quota limit. • *${quota.resourceType}*: The classification type of the Oracle Identity Cloud Service

Field	Description
	entity (or resource) for which there is a quota limit (for example, users) • *${quota.usage}*: Records of the resource type that were created. If an administrator created 600,000 accounts, then 600,000 represents the quota usage. • *${redirectUrl}*: The redirect URL for the notification that can be used if the link in the notification doesn't work • *${request.createdOn}*: The date and time that the request was created • *${request.requestedItem}*: The groups or applications to which a user is requesting access • *${request.requesteeDisplayName}*: The display name of the user who submitted a request for access to groups or applications • *${start.dateTime}*: The date and time at which the job began to synchronize users, groups, application accounts, and entitlements from an application into Oracle Identity Cloud Service • *${sync.status}*: The status of the job that's used to synchronize users, groups, application accounts, and entitlements from an application into Oracle Identity Cloud Service • *${sync.summary}*: A summary of this synchronization job • *${tenantName}*: The name of the identity domain (or tenant) • *${time}*: The time associated with the action of the notification • *${user.displayName}*: The user's first name and last name (or display name) • *${user.userName}*: The user's user name • *${userToken}*: A token that Oracle Identity Cloud Service uses to identify the user • *${validity}*: The amount of time (in minutes), after which the OTP will no longer be valid. As a result, the user can't use it to enroll in 2–Step Verification.

Tip:

To undo the changes that you make to a notification template, click **Cancel**. If you click **Cancel**, then all your changes are lost.

5. Click **Save**.
6. In the **Confirmation** window, click **OK**.

Verify Notifications

You can verify the customizations that you made to the Oracle Identity Cloud Service notification templates you selected. Oracle recommends that you first test the customizations by sending the notifications to a limited number of recipients.

See Specify Recipients for Notifications.

To verify your notification customizations, use the following table.

Notification	Action (Administrator)	Action (User)
Welcome	Create an account for a user.	
Self-Registration Email Verification		Complete the self-registration process to create a user account.
Welcome Self-Registration User		Confirms email in the **Self-Registration Email Verification** notification.
Welcome Federated SSO User	Create an account for a federated SSO user.	
Welcome Delegated Authentication User	Create an account for a user whose authentication is delegated.	
Resend Welcome	Resend Invitation to user.	Receive another notification that the administrator resent the Welcome notification.
Resend Welcome To Delegated Authentication User	Resend Invitation to user authenticated by other than Oracle Identity Cloud Service.	Receive another notification that the administrator resent the Welcome notification for the user whose authentication is delegated.
Password Recovery Request		Request a password reset.
Recovery Email Verification	Create an account for a user and provide a password recovery email address for the user in the **Recovery Email** field.	Change the password recovery email address.
Primary Email Verification	Create an account for a user and provide a primary email address for the user in the **Email** field.	Change the primary email address.
Secondary Email Verification	Create an account for a user and provide a secondary email address for the user in the **Email** field.	Change the secondary email address.
Recovery Email Update	Updates Recovery Email field.	
Primary Email Update	Updates Email field.	
Secondary Email Update	Updates any secondary Email field.	
Password Change		Change the password.
Password Reset		Reset the password.
Password Has Been Changed by an Administrator to a Known Value	Change the password for a user to a known value.	
Admin Requesting a Password Reset on Behalf of a User	Initiate changing a user's password.	
User Activation	Activate a user's account.	

Notification	Action (Administrator)	Action (User)
User De-activation	Deactivate a user's account.	
User Account Locked		Lock the account by logging in to Oracle Identity Cloud Service unsuccessfully for a consecutive number of attempts.
Exceeded Maximum Number of Account Recovery Attempts		Exceed the maximum number of attempts to reset your password to recover your account.
User Account Unlocked		Unlock the account.
User Profile Updated by Administrator	Update the user's profile.	Update the profile.
User Profile Replaced by Administrator	Replace attribute values of the user's profile.	Replace attribute values of the profile.
Device Enrollment Request to Enable 2-Step Verification		Select the **Mobile App** option during enrollment and click the **Email** option to send the enrollment URL to the user's account.
2-Step Verification User Account Locked		Perform unusual activity on the account, such as entering an OTP incorrectly too many times, an SMS text code incorrectly too many times, or using an untrusted device to perform 2–Step Verification.
2-Step Verification Federated SSO User Account Locked		Perform unusual activity on the account, such as entering an OTP incorrectly too many times, an SMS text code incorrectly too many times, or using an untrusted device to perform 2–Step Verification.
2-Step Bypass Code Verification	Generate a bypass code and click the **Email** option.	Generate a bypass code and click the **Email** option.
Enable Kerberos Authentication Request		Log in to Oracle Identity Cloud Service. Use the Kerberos principal name provided in the notification and the Oracle Identity Cloud Service password for the Kerberos application.
New Access Request Submitted		Request access to groups or applications from the Catalog.
Access Request Fulfilled		Request access to groups or applications from the Catalog.
2–Step Email One-Time Passcode Verification		Enroll in 2–Step Verification by using a mobile number as an authentication method.
New Device Login Detected with Your Account		Log in to your account from a device, IP address, or web browser that Oracle Identity Cloud Service doesn't recognize is associated with your account.
Job Has Been Started	Start a job.	
Job Has Been Canceled	Cancel a job.	

Notification	Action (Administrator)	Action (User)
Job Is Complete	Start a job and wait for it to finish.	
Job Has Failed	Start a job that contains values that cause the job to fail.	
Quota Limit Exceeded	Exceed the quota limit for the Oracle Identity Cloud Service instance.	
From Email Domain Validation Initiated	Enter an email address in the **From Email Address** field of the **Notifications** page, and then click **Save**.	
Email Address Validation Initiated for From Email Address	Enter an email address in the **From Email Address** field of the **Notifications** page, and then click **Save**.	
Synchronization Job Summary	Synchronize users, groups, application accounts, or entitlements from an application into Oracle Identity Cloud Service.	

Deactivate Notifications

You prevent Oracle Identity Cloud Service from sending notifications to users and administrators by deactivating notifications.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Notifications**.
2. Turn **Off** the Status switch.
3. Click **Save**.
4. In the **Confirmation** window, click **Yes**.

13

Manage Oracle Identity Cloud Service Password Policies

Learn how to manage password policies for Oracle Identity Cloud Service.

Topics

Typical Workflow for Managing Oracle Identity Cloud Service Password Policies

With the password management feature in Oracle Identity Cloud Service, you can perform tasks such as setting, testing, modifying, and evaluating password policies.

Task	Description	Additional Information
Understand the types of password policies.	You can learn about password policies, including the three types of policies: Simple, Standard, and Custom.	Understand Password Policies
Set the password policies for your identity domain.	You can set multiple password policies for your identity domain using the **Password Policy** page.	Set the Password Policies for Your Identity Domain
Test a password policy.	You can test the criteria for a password policy using the **Password Policy** page.	Test a Password Policy
Modify the Custom password policy.	You can modify the Custom password policy using the **Password Policy** page.	Modify the Custom Password Policy
Evaluate password policies.	You can evaluate the Simple, Standard, or Custom password policy that you set for your identity domain.	Evaluate Password Policies

You can set, test, modify, and evaluate Simple, Standard, and Custom password policies by using:

- The Identity Cloud Service console
- SCIM-based APIs

In this section, you learn how to manage password policies by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Understand Password Policies

You can set up policies in Oracle Identity Cloud Service for an identity domain. You then attach a policy to a group and it is applicable to all users in a group.

You can create up to ten password policies in Oracle Identity Cloud Service and each is assigned a priority. A password policy is assigned to a group, and all users in the group will use that policy. When a user is a member of more than one group, the password policy with the highest priority applies.

When a user is created or when a user changes their password, Oracle Identity Cloud Service validates the password that's provided against the highest priority password policy for that user to ensure that it meets the criteria for the policy. A new user who is not a member of a group will use the default password policy. A user who is a member of a group which does not have a password policy assigned will use the default password policy. When a user logs in for the first time to change the password, or resets the password at any time, the password policy is evaluated.

Deleting Groups and Policies

When a group is deleted, the password policy attached to the group will no longer be assigned to users who had been members of the group. Instead, the highest priority password policy available will apply to users.

When a password policy is deleted, groups and therefore users of the group are no longer associated with it so the highest priority password policy available will apply to users.

Types of Password Policies

There are three types of password policies in Oracle Identity Cloud Service:

Simple
Used for your developer services and demos when you don't want to customize a policy for them. You can't modify this type of password policy.

Standard
Used when you don't want to use the Oracle-recommended password policy for your enterprise applications. You can't modify this type of password policy.

Custom
Used to tailor the strength of your password policy to meet the business and security requirements for your enterprise applications. As an administrator, it's your responsibility to make the minimal requirements of the Custom password policy strong.

Set the Password Policies for Your Identity Domain

You can create up to ten password policies in your identity domain, assign relative priorities to them, and attach them to groups. A group cannot be assigned to more than one password policy.

Prerequisite

Enable Group-based password policies. This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service.

To set the password policy for your identity domain, you must be assigned to either the identity domain administrator role or the security administrator role. See Add or Remove a User Account from an Administrator Role.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Password Policy**.
2. Click **Add**.
3. Enter a name and optionally a description. Choose the priority and click **Next**.

   The priority can be any integer between 1 and 10, where 1 is highest priority and 10 is lowest. If there is already a password policy with the priority you choose, that policy moves to the next priority number. For example, if there is a password policy with a priority of 2 and another with a priority of 3, and you create a new policy with a priority of 2, the other policies will have priorities of 3 and 4.
4. Click the button that represents the type of policy that you want to set for your identity domain (**Simple**, **Standard**, or **Custom**). Click **Next**.
5. Attach one or more groups to the password policy. Each group can only have one policy assigned to it. Click **Add** and select the group, then click **OK**.

   If a user has only one group assigned to them, then the password policy attached to that group is the password policy assigned to the user.

   If a user has more than one group assigned to them, then the password policy with the highest priority is the password policy assigned to the user.
6. Click **Finish**.
7. In the **Save Password Policy** dialog box, select the **Force all users to set a new password on their next login.** check box.

   Do this to ensure that their passwords meet the criteria for the policy before the users can use Oracle Identity Cloud Service.

   Otherwise, don't select the **Force all users to set a new password on their next login.** check box. The password policy applies to users only when they are created or when they reset their passwords.
8. Complete one of the following actions:
   - To save the updated password policy, click **OK**.
   - To reinstate the previously saved password policy, click **Cancel**.

Test a Password Policy

After setting the password policy for your identity domain, you can test the criteria for the policy to validate that the password policy has been set.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Password Policy**.
2. Select a policy which uses the password policy you want to test and go to the Password Rules tab.
3. Click **Test Your Password Policy**.
4. In the **New Password** field of the **Test Your Password Policy** dialog box, enter the password that you want to test.

   As the password meets a criterion for the policy, the associated red *X* mark changes to a green check mark. After all criteria are met, all red *X* marks appear as green check marks.
5. Close the **Test Your Password Policy** dialog box.

Modify the Custom Password Policy

Oracle Identity Cloud Service provides you with a Custom password policy that contains predefined settings. You can tailor the strength of this policy to meet the business and security requirements for your enterprise applications.

You can access the Customizing the Service infographic to see how to customize a password policy in Oracle Identity Cloud Service.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Password Policy**.
2. Select a policy which uses the Custom password policy and go to the Password Rules tab. Alternatively, you can create a new policy and change the Custom password policy on the Password Rules page.
3. Click **Change Your Password Policy**.
4. Click **Custom**.
5. To modify the Custom password policy, use the following list:

Field	Description
Password length (min size)	How many characters the password must contain **Note:** A password must contain at least one character.
Password length (max size)	How many characters are allowed for the password **Note:** A password can't exceed 500 characters.

Field	Description
Expires after (days)	How many days until the password expires **Note:** Setting this option to 0 means that the password never expires.
Account lock threshold	The number of consecutive, unsuccessful login attempts into Oracle Identity Cloud Service after which the user account is locked **Note:** If you enter **0** in the **Account lock threshold** field, then the user's account will never be locked.
Enable auto unlock account	To enable users to automatically unlock their accounts in Oracle Identity Cloud Service after a configured time.
Auto unlock account after (minutes)	The amount of time (in minutes), after which Oracle Identity Cloud Service will unlock an account automatically. You can set a value ranging between 5 minutes and 24 hours.
Previous passwords remembered	How many unique new passwords a user must use before an old password can be reused
Alphabetic (min)	How many alphabetic characters the password must contain
Numeric (min)	How many numeric characters the password must contain
Special (min)	How many special characters the password must contain
Lowercase (min)	How many lowercase characters the password must contain
Uppercase (min)	How many uppercase characters the password must contain
Unique (min)	How many unique characters a password must contain. Increasing the number of unique characters in a password can increase password strength by avoiding repetitive sequences that are easily guessed.
Repeated (max)	How many repeated characters are allowed for the password. This rule limits the use of repeating characters in a password. This value provides extra security by preventing users from specifying passwords that are easy to guess, such as the same character repeated several times.
Starts with (Alphabetic character)	To force the first character of the password to be an alphanumeric character, select this check box.

Field	Description
Required Characters	To activate the text field to the right of the check box, select the check box. Any alphanumeric or special characters that you enter into this field, separated by commas, are required characters for the password.
User attributes (The user's first name)	To prevent the user's first name from being used as all or part of the password, select this check box.
User attributes (The user's last name)	To prevent the user's last name from being used as all or part of the password, select this check box.
User attributes (The user name)	To prevent the user's user name from being used as all or part of the password, select this check box.
Characters not allowed	To activate the text field to the right of the check box, select the check box. Any alphanumeric or special characters that you enter into this field, separated by commas, are characters that aren't allowed for the password.
Whitespace Character	To prevent whitespace characters from being used as part of the password, select this check box. A **whitespace character** is a character that represents horizontal space in Oracle Identity Cloud Service. For example, for the display name of *John Smith*, the space between the first name of *John* and the last name of *Smith* is a whitespace character.
Restricted Words	If you select this check box, then you can screen all passwords against the following words: *Password,Qwerty,BaseBall,Dragon,Monkey,LetMeIn,Abc,Mustang,Access,Shadow,Master,Michael,Superman,BatMan,Trustno,Welcome,Fusion,Oracle,Orcl,ILoveYou,Paas,Admin,Administrator,Cloud,Princess,Azerty,Guest* Oracle Identity Cloud Service will reject any passwords that match the words in the list.

6. Click **Save**.
7. In the **Save Password Policy** dialog box, to force all users in your identity domain to set a new password upon their next login, select the **Force all users to set a new password on their next login.** check box.

   Otherwise, don't select the **Force all users to set a new password on their next login.** check box. The password policy applies to users only when they are created or when they reset their passwords.
8. Click **OK**.

> **Tip:**
>
> To reinstate the previously saved password policy, click **Cancel**. If you click **Cancel**, then all your changes will be lost.

Evaluate Password Policies

At any time, you can evaluate your current password policy.

You can evaluate the Simple, Standard, or Custom password policy that you set for a group when:

- Users in the group associated with the policy register themselves with Oracle Identity Cloud Service
- Users reset their passwords
- An administrator manually sets or changes a user's password

See Configure User Settings for more information about evaluating password policies.

14

Brand the Oracle Identity Cloud Service Interface

Learn how to customize the Oracle Identity Cloud Service web-based interface.

Topics

Typical Workflow for Branding the Oracle Identity Cloud Service Interface

With the customization feature in Oracle Identity Cloud Service, you can perform tasks such as customizing the **Sign In** page. You can brand the Identity Cloud Service console, My Profile console, **My Apps** page, **Catalog** page, **2–Step Verification** page, and notification templates by adding logos to them.

Task	Description	Additional Information
Customize the **Sign In** page.	You can customize the **Sign In** page using the **Branding** page.	Customize the Sign In Page
Brand the consoles.	You can brand the Identity Cloud Service console, My Profile console, **My Apps** page, **Catalog** page, and **2–Step Verification** page by adding a logo to them using the **Branding** page.	Brand the Consoles
Brand notification templates.	You can brand notification templates by adding logos to them using the **Branding** page.	Brand Notification Templates

You can access the Customizing the Service infographic to see how to customize the Oracle Identity Cloud Service interface.

You can customize the Oracle Identity Cloud Service interface by using:

- The Identity Cloud Service console
- SCIM-based APIs

In the following sections, you learn how to customize the interface by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Customize the Sign In Page

Use the default Branding page to customize the customer login in experience. For example, you can overwrite the existing CSS styles to customize the Sign-in Page.

You can customize this page by:

- Modifying the language of the text in the page
- Branding the page so that it displays the company name and logo
- Adding login text to the page. This text provides additional information that the user requires to log in to Oracle Identity Cloud Service.

To customize the **Sign In** page, use the **Branding** page in the Identity Cloud Service console. To open the **Branding** page, you must be assigned to either the identity domain administrator role or the security administrator role. See Add or Remove a User Account from an Administrator Role.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Branding**.
2. Because you want to customize the **Sign In** page, click **Custom Branding**.
3. Click **Translation**, and then select the language of the text for the **Sign In** page and the country for your identity domain. If you want English to be the text for the **Sign In** page, and your identity domain is located in the United States, then select **English (United States)**.

> **Important:**
>
> Make sure the language that you specify for the **Sign In** page matches the default language of your web browser.

4. In the **Company Name** field, enter the name of the company for the **Sign In** page.
5. In the **Login Text** field, enter and format the login text for the page.
6. In the **Sign In Page** region, click **Upload** for either the logo or the background image.
7. Select the company logo or background image that appears in the **Sign In** page when the user accesses the page through a web browser or mobile device.

> **Tip:**
>
> Make sure that the file you want to upload adheres to the recommended dimensions and file size before uploading it. See Customize the Interface.

8. In the **Console** region, click **Upload**.
9. Select the console logo that appears in the Admin console when the user accesses the page through a web browser or mobile device.

> **Tip:**
>
> Make sure that the file you want to upload adheres to the recommended dimensions and file size before uploading it. See Customize the Interface.

10. Preview your changes.
    a. Click **Preview Sign In** and web page opens that displays a preview of the customizations you made to the sign-in page
    b. Click **Preview Console** and web page opens that displays a preview of the customizations you made to the Admin console.
11. Verify your customizations to the **Sign In** page.
12. Reduce the size of the preview web page so that the dimensions of the page resemble the dimensions of a mobile device.
13. Verify that the logo appears properly again.
14. Click **Save**.

Brand the Consoles

You can brand the Identity Cloud Service console and the My Profile console to display the company logo in the header regions of the consoles. You can preview your customizations before saving them. This way, you can ensure that they meet your business requirements.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Branding**.
2. If **Oracle Branding (default)** is selected, then click **Custom Branding**.
3. In the **Sign In Page** section, upload a **Logo** or a **Background Image**.

> **Tip:**
>
> Make sure that the file you want to upload adheres to the recommended dimensions and file size before uploading it. See Customize the Interface.

4. In the **Console** region, click **Upload**.
5. Select the company logo that will appear in the header region of both the Identity Cloud Service console and the My Profile console when the user accesses the consoles using a web browser.

> **Tip:**
>
> Make sure that the file you want to upload adheres to the recommended dimensions and file size before uploading it. See Customize the Interface.

6. Click **Preview Console**.

   A web page opens that displays a preview of the logo you added.
7. Verify that the logo appears properly.
8. Reduce the size of the preview web page so that the dimensions of the page resemble the dimensions of a mobile device.

9. Verify that the logo appears properly again.
10. Click **Save**.

Brand Notification Templates

You can brand notification templates to display a logo in the header and footer regions of the templates.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Branding**.
2. If **Oracle Branding (default)** is selected, then click **Custom Branding**.
3. In the **Header Logo** pane of the **Email Templates** section, click **Upload**.
4. Select the logo that will appear in the header region for all notifications.

> **Tip:**
>
> Make sure that the file you want to upload adheres to the recommended dimensions and file size before uploading it. See Customize the Interface.

5. Click **Save**.

15

Create Hosted Sign In Pages

Instead of using the default sign in page, use a Hosted Sign In page to change the look and feel of the sign-in experience. You create a Hosted Sign In page by designing custom HTML code and specifying translations.

Topics:

What's a Hosted Sign In Page?

A Hosted Sign In page allows you to customize the look and feel of the Identity Cloud Service sign-in experience by using style classes, custom HTML, and translation support.

Prerequisite

Enable Hosted Sign In Page. This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service.

Overview

You can customize the login experience using one or both of the following methods.

Add your own **Background** image to the sign in page.	Background images provided for the sign-in page in the **Branding** settings apply for all the sign-in flows. See Customize the Sign In Page.
Provide custom HTML and custom translations using Hosted Sign In.	Hosted Sign In provides custom HTML and custom translations, in order to overwrite the current sign-in page definition. This customization applies to the main sign-in page. It doesn't affect all sign-in flows, for example, it doesn't affect the reset password and MFA flows. The Hosted Sign In page: • Allows you to change current styles and to add new HTML elements.

	• Supports translations for existing elements as well as new elements. See Create a Hosted Sign-In Page.

Assumptions

- The feature is based on JET 7.2. All browsers that support JET 7.2 can use this feature.
- Administrators are familiar with existing style classes in the sign-in page.

Limitations

- HTML Comments (`<!-- comment -->`) are not allowed.
- Adding custom JavaScript is not allowed.
- The `<style>` tag is not allowed. But you can only use an inline style of elements, for example `<div style="property: value, ...">`
- There is no code validation. Use the **Preview** option to verify that your custom code is valid.
- A limited set of style properties are available.

– align-items: – background-color: – background: – border-radius: – border: – box-shadow: – content: – display: – height: – justify-content: – left: – margin-top:	– margin: – max-height: – max-width: – min-height: – padding-left: – padding-right: – padding: – position: – text-align: – top: – width:

- A limited set of HTML tags are available.

– div – img – image – label – input	– h1 – h2 – h3 – span – style

Access the Hosted Sign In Feature

Hosted Sign In won't display unless the feature is enabled.

To enable this feature, file a Service Request with My Oracle Support. If you don't file a Service Request, then you won't be able to use Hosted Sign In.

Understand the Custom HTML

When Hosted Sign In is enabled for the first time, default HTML code is provided. You customize this code to define your Hosted Sign In page. This becomes your Custom HTML.

Use this default HTML code as a template for your custom HTML. The code is fully functional, which means that the Hosted Sign In page will work even if nothing is changed.

Default HTML

```
<!--
    NOTE:
    Comments on this code are not displayed in the Custom HTML area of Hosted
Sign-in page,
    they are just for documentation purposes
-->


<!-- classes starting with oj-flex are JET util classes for layout -->
<div class="oj-flex oj-sm-flex-direction-column">


  <!-- classes named oj-idaas-xxx , like oj-idaas-signin-app-shell or oj-
idaas-signin-app-shell-background are the default classes assigned to sign-in
page -->
  <div id="oj-idaas-signin-app-shell-background" class="oj-idaas-signin-app-
shell oj-idaas-signin-app-shell-background">
    <div class="oj-flex oj-idaas-signin-app-shell-wrapper oj-idaas-signin-app-
shell-wrapper-background">
      <div class="oj-flex-item oj-sm-12 oj-idaas-signin-app-shell-padding-
left oj-idaas-signin-app-shell-padding-right">
        <div class="oj-idaas-signin-app-shell-branding-logo-wrapper">

          <!--
               This image with id = custom-idaas-signin-branding-logo displays
               the sign-in page customer logo or the default logo (if no
custom logo is provided)
          -->
          <img id="custom-idaas-signin-branding-logo" class="oj-idaas-signin-
app-shell-branding-logo"/>

        </div>


        <!-- oj-idaas-signin-section displays the company name and welcome
text -->
        <oj-idaas-signin-section id="custom-idaas-signin-section"></oj-idaas-
signin-section>
```

```
      </div>
      <div class="oj-flex-item oj-sm-12 oj-idaas-signin-app-shell-content-
wrapper">


        <!-- oj-idaas-signin-message displays the error and warning messages,
DO NOT REMOVE this section -->
        <oj-idaas-signin-message id="custom-idaas-signin-message"></oj-idaas-
signin-message>

        <div class="oj-idaas-signin-app-shell-padding-left oj-idaas-signin-
app-shell-padding-right">


          <!--
            DO NOT REMOVE.
            oj-bind-slot tag, provides the core functionality of the sign-in
page, it is the main widget.
            This will generate the area where username and password are
entered
          -->
          <oj-bind-slot name="content"></oj-bind-slot>
        </div>
      </div>
    </div>
  </div>
</div>
```

Customized HTML Example

```
<!--
    NOTE:
    In this example two new labels are added by using a div tag. The text in
the div tags is dynamically inserted by
    providing translations (see translations section below).
-->
<div class="oj-flex oj-sm-flex-direction-column">
  <div id="idcs-app-shell-signin-background" class="oj-idaas-signin-app-shell
oj-idaas-signin-app-shell-background">

    <!-- Example of how the inline style can be applied -->
    <div class="oj-flex" style="margin:20px 0;width:520px;min-
height:400px;padding:50px 0;position:absolute;top:0px;left:16px;background-
color:#ffffff;border-radius:6px">
      <div class="oj-flex-item oj-sm-12 oj-idaas-signin-app-shell-padding-
left oj-idaas-signin-app-shell-padding-right">
        <div class="oj-idaas-signin-app-shell-branding-logo-wrapper">

          <!--
            These two div tags are new elements (labels) introduced in the
page, it is important to assign
            a data-idcs-text-translation-id id to each new element
introduced, this id will be used to apply the translated text to the element.
            If the text doesn't have to be translated, no id is required and
```

```
you can hardcode the text in the element.
          -->
          <h2 data-idcs-text-translation-id="welcometext" id="any1"> </h2>
          <h3 data-idcs-text-translation-id="welcometext2" id="any3"> </h3>

          <img id="custom-idaas-signin-branding-logo" class="oj-idaas-signin-
app-shell-branding-logo" />
        </div>
        <oj-idaas-signin-section id="custom-idaas-signin-section"></oj-idaas-
signin-section>
      </div>
      <div class="oj-flex-item oj-sm-12 oj-idaas-signin-app-shell-content-
wrapper">
        <oj-idaas-signin-message id="custom-idaas-signin-message"></oj-idaas-
signin-message>
        <div class="oj-idaas-signin-app-shell-padding-left oj-idaas-signin-
app-shell-padding-right">
          <oj-bind-slot name="content"></oj-bind-slot>
        </div>
      </div>

    </div>
  </div>
</div>
```

Understand How Translations Work

Hosted Sign In allows you to specify translations for existing elements as well as new elements for your custom HTML code.

The default translations value is `{}`, which means there are no translations provided for the custom HTML code.

The Structure of Translations

Each attribute represents a label, the key is `data-idcs-text-translation-id`, and the value is an object containing the different languages and the translated strings. The following example has translations for existing elements (`idcs-username-label`), as well as new elements (`welcometext`).

```
{
  "idcs-username-label": {
    "en": "Account ID",
    "es": "Cuenta de usuario o correo Electronico"
  },
  "idcs-username-placeholder": {
    "en": "Enter your Account ID",
    "es": "Introduzca su nombre de usuario o correo electronico"
  },
  "idcs-password-label": {
    "en": "Pass-word",
    "es": "Contra-sena"
  },
  "idcs-password-placeholder": {
    "en": "Enter your password",
```

```
    "es": "Introduzca su contrasena"
  },
  "welcometext": {
    "en": "Welcome to our Portal",
    "es": "Bienvenido"
  },
  "welcometext2": {
    "en": "This is great",
    "es": "Bienvenido"
  }
}
```

The following existing elements can be customized in the sign-in page. In order to customize the existing elements, the following reserve IDs must be used.

Element	Reserve ID
username label	idcs-username-label
username placeholder The ghost text inside the username field.	idcs-username-placeholder
password label	idcs-password-label
password placeholder The ghost text inside the password field.	idcs-password-placeholder

Translating New Labels

If a new label is introduced in the Hosted Sign In page, by using `<div>`, `<span>` or header tags like `<h1>`, `<h2>`, `<h3>`, and so on, a translation ID (`data-idcs-text-translation-id`) must be provided for them. For example, use `<div data-idcs-text-translation-id="instructions"></div>` where `data-idcs-text-translation-id` of the element is used to provide a translated text.

Use the Backup URL to Recover the Sign In Page

If changes made to the Hosted Sign-in code break the sign-in flow (for example, removing core components), administrators can use this URL to sign in with the default login page and gain access to Identity Cloud Service.

Create a backup URL like the following: `<hostname:[port]>/ui/v1/signin?`**`noBranding=true`**

> **Note:**
>
> Logos won't appear correctly in the recovery sign-in page as this is just a recovery mechanism.

Create a Hosted Sign-In Page

Create a Hosted Sign In page to customize the look and feel of the Identity Cloud Service sign-in experience by using style classes, custom HTML, and translation support.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, click **Hosted Sign In**, and then enable **Hosted Sign In**.

2. Edit the **Custom HTML**.

   A limited set of HTML tags are available.

• div • img • image • label • input	• h1 • h2 • h3 • span • style

   A limited set of style properties are available.

• align-items: • background-color: • background: • border-radius: • border: • box-shadow: • content: • display: • height: • justify-content: • left: • margin-top:	• margin: • max-height: • max-width: • min-height: • padding-left: • padding-right: • padding: • position: • text-align: • top: • width:

   > **Tip:**
   >
   > Need to start over? You can revert to the default HTML values by clicking **Reset**.

3. Specify translations.

   The following existing elements can be customized in the sign-in page. In order to customize the existing elements, the following reserve IDs must be used.

Element	Reserve ID
`username label`	`idcs-username-label`
`username placeholder` The ghost text inside the username field.	`idcs-username-placeholder`
`password label`	`idcs-password-label`
`password placeholder` The ghost text inside the password field.	`idcs-password-placeholder`

> **Tip:**
>
> Need to start over? You can clear the translations pane by clicking **Reset**.

4. Click **Preview Sign In** to view changes without saving them.
5. Click **Save**.

16

Manage Provisioning Bridges for Oracle Identity Cloud Service

Learn how to manage Provisioning Bridges for Oracle Identity Cloud Service.

Topics:

Typical Workflow for Managing Provisioning Bridges for Oracle Identity Cloud Service

With the Provisioning Bridge feature in Oracle Identity Cloud Service, you can create, manage, and remove Provisioning Bridges.

Task	Description	Additional Information
Understand the Provisioning Bridge.	You can receive an overview of the Provisioning Bridge for Oracle Identity Cloud Service, including how it provides a link between your on-premises apps (such as Oracle Internet Directory and Oracle E-Business Suite) and Oracle Identity Cloud Service. You can also learn why you should use the Provisioning Bridge, and how it's used to synchronize users and groups between your on-premises apps and Oracle Identity Cloud Service.	Understand the Provisioning Bridge Why Use the Provisioning Bridge?

Task	Description	Additional Information
Create a Provisioning Bridge.	You can create a Provisioning Bridge using the **Provisioning Bridges** page and the client for the bridge.	Create a Provisioning Bridge
Start a Provisioning Bridge.	You can start a Provisioning Bridge using the client for the bridge.	Start a Provisioning Bridge
View details about a Provisioning Bridge.	You can view details about a Provisioning Bridge using the **Provisioning Bridges** page.	View Details About a Provisioning Bridge
Activate and deactivate Provisioning Bridges.	You can activate and deactivate Provisioning Bridges using the **Provisioning Bridges** page.	Activate and Deactivate Provisioning Bridges
Modify a Provisioning Bridge.	You can modify a Provisioning Bridge using the **Provisioning Bridges** page.	Modify a Provisioning Bridge
Stop a Provisioning Bridge.	You can stop a Provisioning Bridge using the client for the bridge.	Stop a Provisioning Bridge
Remove Provisioning Bridges.	You can remove Provisioning Bridges using the **Provisioning Bridges** page.	Remove Provisioning Bridges
Manage log files for a Provisioning Bridge.	You can change the folder where all log files for the Provisioning Bridge are stored and the log level for these log files.	Manage Log Files for a Provisioning Bridge
Upgrade a Provisioning Bridge.	If you're using version 19.2.1 of the Provisioning Bridge, then upgrade to version 19.3.3 of the bridge.	Upgrade a Provisioning Bridge

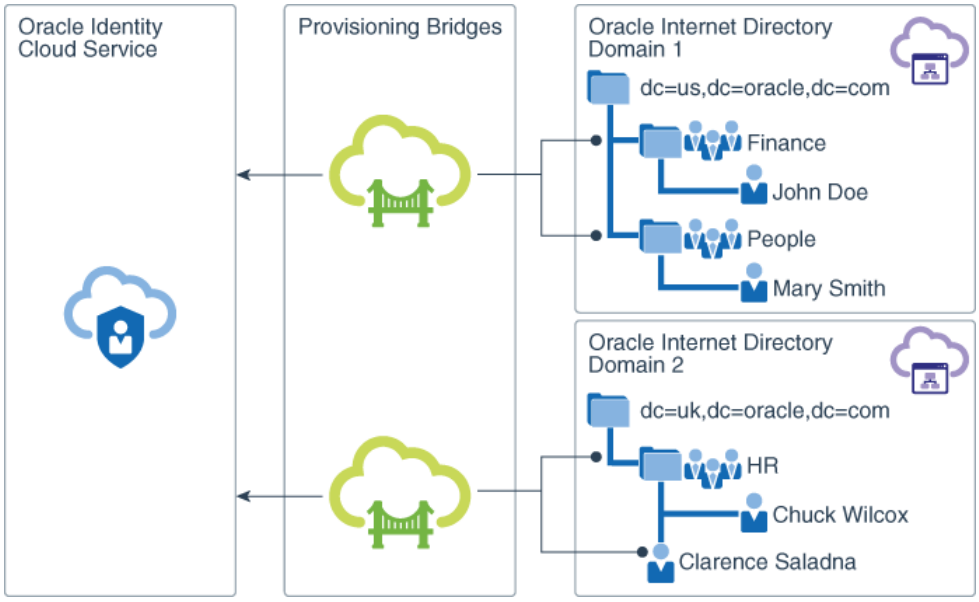
In the following sections, you learn how to use the Identity Cloud Service console to create, manage, and remove Provisioning Bridges.

Understand the Provisioning Bridge

The Provisioning Bridge provides a link between your on-premises apps and Oracle Identity Cloud Service. Through synchronization, account data that's created and updated directly on the apps is pulled into Oracle Identity Cloud Service and stored for the corresponding Oracle Identity Cloud Service users and groups. As a result, any changes to these records will be transferred into Oracle Identity Cloud Service. So, if a user is deleted in one of your apps, then this change will be propagated into Oracle Identity Cloud Service. Because of this, the state of each record is synchronized between your apps and Oracle Identity Cloud Service.

Suppose you're using an on-premises app such Oracle Internet Directory as an authoritative source for your company's users and groups. This app lies within your company's firewall. For a Provisioning Bridge to communicate with on-premises apps such as Oracle Internet Directory, it must leverage Identity Connector Framework (ICF) connectors to access the associated apps. As a result, the Provisioning Bridge can poll the on-premises apps for changes to users and groups in the apps, and synchronize these changes with Oracle Identity Cloud Service. You can configure a Provisioning Bridge so that Oracle Identity Cloud Service can synchronize users and groups from one or multiple apps.

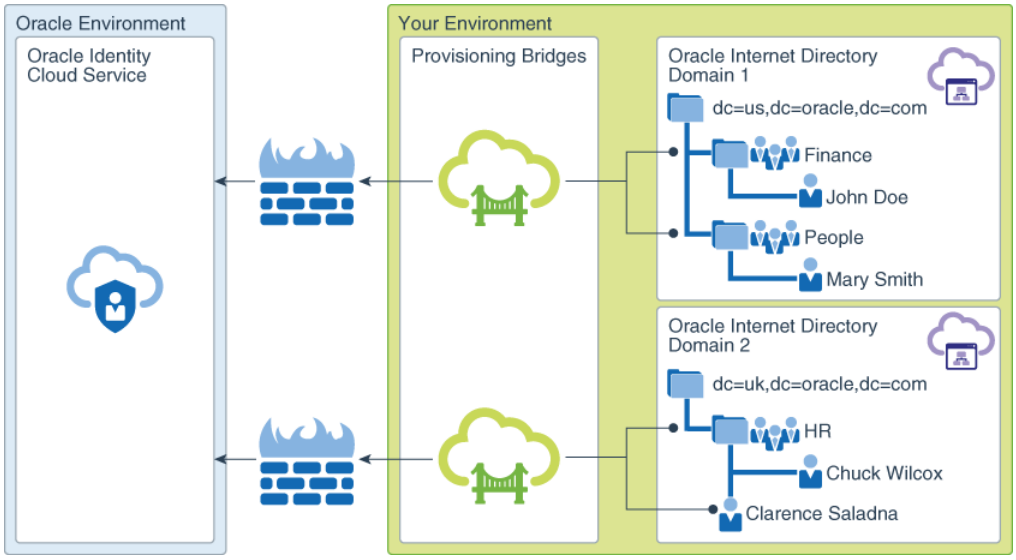
Figure 16-1 Directory Synchronization



Both the Provisioning Bridges and your on-premises apps are in your Microsoft Windows or generic environment. A generic environment consists of any machine that has Java 8 installed on it and supports Bash shell.

Each Provisioning Bridge uses a client network to access the on-premises apps with which you want to synchronize Oracle Identity Cloud Service users and groups. Because Oracle Identity Cloud Service is an Oracle Cloud service, it's in an Oracle environment.

Figure 16-2 Provisioning Bridge Security



The Synchronize Users from Oracle Internet Directory to Oracle Identity Cloud Service video shows you how to configure on-premises apps such as Oracle Internet Directory so that the Provisioning Bridge can use the associated ICF connectors to poll the apps for changes to

users and groups in the apps, and synchronize these changes with Oracle Identity Cloud Service.

Statuses

There are two statuses for a Provisioning Bridge client:

- Started: The Provisioning Bridge started successfully. See Start a Provisioning Bridge.
- Stopped: The Provisioning Bridge stopped unexpectedly or the identity domain administrator or security administrator stopped it. See Stop a Provisioning Bridge.

There are also two statuses for a Provisioning Bridge:

- Active: The Provisioning Bridge is installed, started, and activated. It's available to poll the apps to which the Provisioning Bridge is assigned for changes to users and groups in the apps, and synchronize these changes with Oracle Identity Cloud Service. See Activate Provisioning Bridges.
- Inactive: The Provisioning Bridge is installed and configured, but it's deactivated. It's not available to retrieve users and groups from the apps to which the Provisioning Bridge is assigned. For performance reasons, this is done. See Deactivate Provisioning Bridges.

Why Use the Provisioning Bridge?

Most customers have Microsoft Active Directory (AD) as their central directory service. These customers also use AD as their network directory. This directory is where all of their workstations are connected to and from where they manage their users.

In addition to AD, customers use:

- An enterprise LDAP to centralize all of their user identities. So, a customer uses AD to manage their employees, but in the centralized LDAP, the customer manages their partners, consumers, and any other users with which the customer has relationships.
- Business applications to manage and automate processes across their enterprise. These processes include customer relationship management (CRM), enterprise resource planning (ERP), and supply chain management (SCM) processes.

For these reasons, it's imperative that Oracle Identity Cloud Service can integrate with AD, an enterprise LDAP (for example, Oracle Internet Directory), and an on-premises business application to manage and automate the customer's CRM, ERP, SCM, and other business-related processes.

By using Oracle Identity Cloud Service, customers can control when they will migrate their directory-based applications to the cloud. In the interim, they can use one of the following:

- AD Bridge: This bridge provides a link between your AD enterprise directory structure and Oracle Identity Cloud Service. Oracle Identity Cloud Service can synchronize with this directory structure so that any new, updated, or deleted user or group records are transferred into Oracle Identity Cloud Service. Each minute, the bridge polls AD for any changes to these records and brings these changes into Oracle Identity Cloud Service. So, if a user is deleted in AD, then this change will be propagated into Oracle Identity Cloud Service. As a result, the state of each record is synchronized between AD and Oracle Identity Cloud Service. After the user is synchronized from AD to Oracle Identity Cloud Service, if you activate or deactivate a user, modify the user's attribute values, or change the group memberships for the user in Oracle Identity Cloud Service, then these changes are propagated to AD through the AD Bridge. See Manage Microsoft Active Directory (AD) Bridges for Oracle Identity Cloud Service.

- Provisioning Bridge: This bridge provides a link between your enterprise LDAP or on-premises business application (such as Oracle Internet Directory or Oracle E-Business Suite) and Oracle Identity Cloud Service. Through synchronization, account data that's created and updated directly on the LDAP or business application is pulled into Oracle Identity Cloud Service and stored for the corresponding Oracle Identity Cloud Service users and groups. Any changes to these records will be transferred into Oracle Identity Cloud Service. Because of this, the state of each record is synchronized between the LDAP or business application and Oracle Identity Cloud Service.

  After users are synchronized from the on-premises business application to Oracle Identity Cloud Service, you can also use the Provisioning Bridge to provision users to the application. Provisioning allows you to use Oracle Identity Cloud Service to manage the lifecycle of users in the application. This includes creating, modifying, deactivating, activating, and removing users and their profiles across the application. Any changes that you make to users or their profiles in Oracle Identity Cloud Service are propagated to the business application through the Provisioning Bridge.

This chapter focuses on using the Provisioning Bridge to synchronize and provision users between an enterprise LDAP (such as Oracle Internet Directory) or an on-premises business application (such as Oracle E-Business Suite) and Oracle Identity Cloud Service.

Create a Provisioning Bridge

Creating a Provisioning Bridge establishes a link between Oracle Identity Cloud Service and your on-premises apps.

To create this bridge, you must:

1. **Enable Provisioning Bridge.** Oracle must enable this feature for you. To learn about the features that Oracle must enable for you and how to enable them, see Standard License Tier Features for Oracle Identity Cloud Service.
2. Use the Identity Cloud Service console to add the bridge
3. Install the client for the bridge

To add a Provisioning Bridge, you must be assigned to either the identity domain administrator role or the security administrator role. See Add or Remove a User Account from an Administrator Role for more information about assigning administrator roles to users.

After adding the Provisioning Bridge, you install the client for the bridge on a Microsoft Windows or generic machine. A generic machine has Java 8 installed on it and supports Bash shell.

Installing the client for the Provisioning Bridge includes providing administrative credentials for Oracle Identity Cloud Service, including the URL for the Oracle Identity Cloud Service identity domain, Client ID, and Client Secret. The Provisioning Bridge requires these credentials to access Oracle Identity Cloud Service as an administrator.

Prerequisites

Part of creating a Provisioning Bridge is installing the client for the bridge. On the machine where you're installing this client, you must have:

- Java 8 installed
- Administrative rights to access the client network that the Provisioning Bridge uses to communicate with the apps that you want to monitor
- Permissions to run the scripts that are used to install and start the Provisioning Bridge

- Permissions to create, manage, and execute commands in the folders associated with the machine where you'll install the client for the Provisioning Bridge
- Permissions to manage log files associated with the Provisioning Bridge
- The ability to communicate with both the Oracle Identity Cloud Service server and the servers associated with the target apps (for example, the Oracle Internet Directory or Oracle E-Business Suite servers)
- Low network latency with these target servers

Create a Provisioning Bridge

In this procedure, you'll:

- Use the Identity Cloud Service console to add a Provisioning Bridge
- Install the client for this bridge on a Windows or generic machine. A generic machine has Java 8 installed on it and supports Bash shell.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Provisioning Bridges**.
2. If this is the first bridge you're creating, then click **Add a Provisioning Bridge**. Otherwise, click **Add**.
3. In the **Name** and **Description** fields of the **Add Provisioning Bridge** page, enter a name and descriptive information for the Provisioning Bridge. Then, click **Save**.

   A new page appears for the Provisioning Bridge. The name of this page is the name you provided for the bridge in this step. By default, this Provisioning Bridge is deactivated. See Activate Provisioning Bridges to learn how to activate it.

   This page contains three tabs:

   - **Details:** This tab contains high-level information about the Provisioning Bridge.
   - **Apps:** This tab displays the apps to which the Provisioning Bridge will
     - Poll for changes to users and groups in the apps, and synchronize these changes into Oracle Identity Cloud Service.
     - Use Oracle Identity Cloud Service to manage the lifecycle of users in the apps. This includes creating, modifying, deactivating, activating, and removing users and their profiles across the apps.

     See Assign a Provisioning Bridge to Apps.
   - **Connectors:** After you install the client for the Provisioning Bridge and start the bridge, this tab displays the connectors that the bridge uses to communicate with the apps. You can learn more about this tab in Start a Provisioning Bridge.
4. Make a note of the Identity Cloud Service URL, Client ID, and Client Secret.

   The Identity Cloud Service URL contains the name and port number for your Oracle Identity Cloud Service identity domain. The Client ID and Client Secret are used by the Provisioning Bridge to access Oracle Identity Cloud Service as an administrator.

> **Note:**
>
> The Client Secret is encrypted (for security purposes). To see the Secret in clear text, click **Show Secret**. To regenerate the Secret for the bridge, click **Regenerate**.

5. Click the **Downloads** link (because you want to download the client for the Provisioning Bridge).
6. In the **Downloads** page, click **Download** to the right of the **Identity Cloud Service Provisioning Bridge** client.

   Oracle Identity Cloud Service downloads the client for the Provisioning Bridge.
7. Verify that a **Success** status appears to the right of the **Identity Cloud Service Provisioning Bridge** client.
8. Launch the Windows or generic machine where you want to install the client for the Provisioning Bridge.

   > **Important:**
   >
   > Make sure that you have administrative rights for this machine. Also, this machine will communicate with the client network that the Provisioning Bridge uses to access the apps that you want to monitor.
9. On this machine, create a folder, and then unzip the file that you downloaded in step 6 of this procedure into this folder. This zipped file contains the client that you are to install for the Provisioning Bridge.

   After you unzip the file, the following folders are created:

   - `bin`: This folder contains the `crossplatform.jar` file. This file is used by the installer to install, start, and stop the Provisioning Bridge.
   - `bundle_home`: This folder contains the connector JAR files that Oracle ships with the bridge. These files are used by the bridge to communicate with the apps.
   - `conf`: This folder contains two properties files:
     - `BridgeRuntimeConfigurations.properties`: This file contains properties associated with the Provisioning Bridge communicating with Oracle Identity Cloud Service and the target apps. Oracle strongly recommends that you don't modify the contents of this file.
     - `log4j.properties`: This file contains properties associated with logging operations that are performed by the Provisioning Bridge. See Manage Log Files for a Provisioning Bridge.
   - `dependencies`: This folder contains the script files that the Provisioning Bridge uses to communicate with Oracle E-Business Suite for synchronization and provisioning purposes.
   - `logs`: This is the default folder is where all log files for the Provisioning Bridge are stored. You can change this folder and path by modifying the `log4j.properties` file. See Manage Log Files for a Provisioning Bridge.

   You'll also see three files:

   - `startup.bat`: Use this file to launch the client for the Provisioning Bridge on a Windows (`.bat`) machine.
   - `startup.sh`: Use this file to launch the client on a generic (`.sh`) machine.
   - `FileInfo.json`: This file contains version information about the zipped file that you downloaded. Oracle strongly recommends that you don't modify the contents of this file.

> **Tip:**
>
> While you're installing the client, Oracle Identity Cloud Service generates log files for the Provisioning Bridge automatically, and stores them in the `logs` folder.

10. If you're installing the Provisioning Bridge on a generic machine, then open a Terminal window, navigate to the folder that you created in step 9, and run the `/.startup.sh install` command.

    OR

    If you're installing the Provisioning Bridge on a Windows machine, then open Windows Explorer, navigate to the folder that you created in step 9, and double-click the `startup.bat` file.

11. At the **Enter a password for Oracle Wallet** prompt, enter your Oracle Wallet password. The wallet is a file that's used to store sensitive information such as the Identity Cloud Service URL, Client ID, and Client Secret for Oracle Identity Cloud Service securely.

12. At the **Re-enter your password** prompt, enter this password again.

> **Note:**
>
> After you install the Provisioning Bridge, a `wallet` folder is created, and the Oracle Wallet you created is stored in this folder. This way, when you start the Provisioning Bridge, instead of providing the Identity Cloud Service URL, Client ID, and Client Secret for Oracle Identity Cloud Service, you only have to supply the password you provided for your Oracle Wallet.

> **Important:**
>
> There's no mechanism to recover your Oracle Wallet password if you forget it. If this happens, then delete the `wallet` folder and install the Provisioning Bridge again.

13. At the **Enter the Identity Cloud Service URL**, **Enter the Client ID**, and **Enter the Client Secret** prompts, enter the Identity Cloud Service URL, Client ID, and Client Secret for Oracle Identity Cloud Service.

> **Tip:**
>
> These credentials appear on the **[Provisioning_Bridge_Name]** page of the Identity Cloud Service console.

14. For the following prompts:
    - **Enter the address for the proxy server**
    - **Enter the port number of the proxy server**
    - **Enter the name of the administrator who can connect to the proxy server**
    - **Enter the password of the administrator who can connect to the proxy server**

a. If your organization has a firewall in place and requires communication to be handled using an HTTP Proxy Server, then enter the full path (or address) of the proxy server, the port number reserved for this server, and the administrator credentials for connecting to the server.

b. If your organization doesn't require communication to be handled using an HTTP Proxy Server, then press **Enter** after each prompt to skip the prompt.

The bridge attempts to connect to the Oracle Identity Cloud Service server.

If a connection can be established, then information about the Provisioning Bridge you created appears. This information includes the name, description, version number, Identity Cloud Service URL of the identity domain, and the locations of the `log4j.properties` file and `bundle_home` folder.

Otherwise, you'll receive an error message, indicating that you entered an incorrect Identity Cloud Service URL, Client ID, or Client Secret. Modify the incorrect values, and try again. If the problem persists, then delete the Oracle Wallet you created, and repeat steps 10-14 of this procedure.

Start a Provisioning Bridge

To start a Provisioning Bridge on a Windows or generic machine, you must first start the client for the bridge. Then, use the Identity Cloud Service console to access the Provisioning Bridge to verify that:

- The Provisioning Bridge is started
- The connectors appear that the Provisioning Bridge uses to:
  - Poll the associated apps for changes to users and groups in the apps, and synchronize these changes into Oracle Identity Cloud Service
  - Manage the lifecycle of users in the apps. This includes creating, modifying, deactivating, activating, and removing users and their profiles across the apps.

Start the Provisioning Bridge on a Generic Machine

A generic machine has Java 8 installed on it and supports bash shell. For this type of machine, you can start the Provisioning Bridge in two modes:

- `normal`: The bridge starts in a Terminal window.
- `background`: The bridge starts as a process in the background in a Terminal window.

> **Important:**
>
> You can't start multiple Provisioning Bridges with the same configuration information. If you want to start another Provisioning Bridge, then use the **Provisioning Bridges** page to create a new bridge, and use the newly generated Client ID and Secret for Oracle Identity Cloud Service to start the bridge.

Start in Normal Mode

Start the Provisioning Bridge in `normal` mode.

1. Launch the generic machine where you installed the client for the Provisioning Bridge.

> **Important:**
>
> Make sure that you have administrative rights for this machine. Also, this machine will communicate with the client network that the Provisioning Bridge uses to access the apps that you want to monitor.

2. In a Terminal window, navigate to the folder you created that contains the files for the Provisioning Bridge. You created this folder in Create a Provisioning Bridge.
3. At the prompt, enter `./startup.sh normal`.
4. At the **Enter your password for Oracle Wallet** prompt, enter the password for Oracle Wallet that you created in Create a Provisioning Bridge.

   The Provisioning Bridge attempts to connect to the Oracle Identity Cloud Service server.
5. Verify that you see the `The Provisioning Bridge is started.` status message. A connection is established between the Provisioning Bridge and the Oracle Identity Cloud Service server.

> **Important:**
>
> Make sure that you keep this Terminal window open. If you close it, then you'll stop the Provisioning Bridge.

6. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Provisioning Bridges**.
7. Verify that the Provisioning Bridge that you created in Create a Provisioning Bridge has a status of **Started**.
8. Click this Provisioning Bridge, and then click the **Connectors** tab.
9. Verify that you see the names and versions of the connectors that are used by the Provisioning Bridge to communicate with the associated apps.

Start in Background Mode

Start the Provisioning Bridge in `background` mode.

1. Launch the generic machine where you installed the client for the Provisioning Bridge.

> **Important:**
>
> Make sure that you have administrative rights for this machine. Also, this machine will communicate with the client network that the Provisioning Bridge uses to access the apps that you want to monitor.

2. In a Terminal window, navigate to the folder you created that contains the files for the Provisioning Bridge. You created this folder in Create a Provisioning Bridge.
3. At the prompt, enter `./startup.sh background`.
4. At the **Enter your password for Oracle Wallet** prompt, enter the password for Oracle Wallet that you created in Create a Provisioning Bridge.

   The Provisioning Bridge attempts to connect to the Oracle Identity Cloud Service server.

5. Verify that you see the `The Provisioning Bridge is started. [Process_ID] is the process ID that's used to start this bridge.` status message. A connection is established between the Provisioning Bridge and the Oracle Identity Cloud Service server.

> **Note:**
>
> If you want to stop the Provisioning Bridge, then use the process ID to kill the process. You can also use this ID to check if the process is running properly, or if there are any errors associated with the process.

6. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Provisioning Bridges**.
7. Verify that the Provisioning Bridge that you created in Create a Provisioning Bridge has a status of **Started**.
8. Click this Provisioning Bridge, and then click the **Connectors** tab.
9. Verify that you see the names and versions of the connectors that are used by the Provisioning Bridge to communicate with the associated apps.

Start the Provisioning Bridge on a Windows Machine

In this procedure, you'll start the Provisioning Bridge on a Windows machine.

> **Important:**
>
> You can't start multiple Provisioning Bridges with the same configuration information. If you want to start another Provisioning Bridge, then use the **Provisioning Bridges** page to create a new bridge, and use the newly generated Client ID and Secret for Oracle Identity Cloud Service to start the bridge.

1. Launch the Windows machine where you installed the client for the Provisioning Bridge.

> **Important:**
>
> Make sure that you have administrative rights for this machine. Also, this machine will communicate with the client network that the Provisioning Bridge uses to access the apps that you want to monitor.

2. Open Windows Explorer, and then navigate to the folder you created that contains the files for the Provisioning Bridge. You created this folder in Create a Provisioning Bridge.
3. Double-click the `startup.bat` file.
4. At the **Enter your password for Oracle Wallet** prompt of the Command window, enter the password for Oracle Wallet that you created in Create a Provisioning Bridge.

   The Provisioning Bridge attempts to connect to the Oracle Identity Cloud Service server.
5. Verify that you see the `The Provisioning Bridge is started.` status message. A connection is established between the Provisioning Bridge and the Oracle Identity Cloud Service server.

> **Important:**
>
> Make sure that you keep this Command window open. If you close it, then you'll stop the Provisioning Bridge.

6. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Provisioning Bridges**.
7. Verify that the Provisioning Bridge that you created in Create a Provisioning Bridge has a status of **Started**.
8. Click this Provisioning Bridge, and then click the **Connectors** tab.
9. Verify that you see the names and versions of the connectors that are used by the Provisioning Bridge to communicate with the associated apps.

View Details About a Provisioning Bridge

By default, in the **Provisioning Bridges** page, you can see the name, description, and statuses for each Provisioning Bridge.

You can also see other information about a Provisioning Bridge, such as its Identity Cloud Service URL, version number, Client ID, and Client Secret, any apps assigned to the bridge, and any connectors that are used by the bridge to communicate between the apps and Oracle Identity Cloud Service.

> **Note:**
>
> See Create a Provisioning Bridge for more information about the Provisioning Bridge's Identity Cloud Service URL, Client ID, and Client Secret.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Provisioning Bridges**.
2. Click the bridge about which you want to view additional information.
3. Click **Details**.

   In this tab, you see information about the Provisioning Bridge, including its name, description, Identity Cloud Service URL, version number, and Client ID. By clicking **Show Secret**, you can see the Client Secret for the Provisioning Bridge in clear text. By clicking **Regenerate**, you can regenerate the Secret for this bridge.
4. Click **Apps**.

   In this tab, you can see a list of apps assigned to the Provisioning Bridge. You can assign additional apps to this bridge or change the bridge associated with the apps. See Assign a Provisioning Bridge to Apps and Change the Provisioning Bridge Assigned to Apps.

> **Note:**
>
> For load-balancing purposes, Oracle suggests that you don't assign more than 10 apps to a Provisioning Bridge. To maintain more apps, create another Provisioning Bridge.

5. Click **Connectors**.

   In this tab, you can see any connectors that the Provisioning Bridge uses to communicate with the apps. See Start a Provisioning Bridge.

Activate and Deactivate Provisioning Bridges

You can use Oracle Identity Cloud Service to activate and deactivate Provisioning Bridges.

- Activating a Provisioning Bridge enables the link between Oracle Identity Cloud Service and your on-premises apps. The Provisioning Bridge can use connectors to poll the corresponding apps for changes to users and groups in the apps, and synchronize these changes into Oracle Identity Cloud Service. The Provisioning Bridge can also use these connectors to provision users to the apps. This includes managing the lifecycle of users in the apps by creating, modifying, deactivating, activating, and removing users and their profiles across the apps.
- Deactivating a Provisioning Bridge disables the link between Oracle Identity Cloud Service and your on-premises apps. The Provisioning Bridge can't use connectors either to poll the associated apps for changes to users and groups in the apps, and synchronize these changes into Oracle Identity Cloud Service, or to provision users to the apps.

> **Note:**
>
> Oracle recommends that you deactivate a Provisioning Bridge before stopping the bridge or performing any maintenance activity on the machine where the client for the bridge is installed.

Activate Provisioning Bridges

You can use Oracle Identity Cloud Service to activate a single Provisioning Bridge. For efficiency purposes, you can also activate multiple Provisioning Bridges simultaneously.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Provisioning Bridges**.
2. Select the check box for each Provisioning Bridge that you want to activate.
3. Click **Activate**.
4. In the **Confirmation** window, click **OK**.

   The status of each Provisioning Bridge you selected changes from **Inactive** to **Active**.

Deactivate Provisioning Bridges

You can use Oracle Identity Cloud Service to deactivate either a single Provisioning Bridge or multiple Provisioning Bridges simultaneously (for efficiency purposes).

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Provisioning Bridges**.
2. Select the check box for each Provisioning Bridge that you want to deactivate.
3. Click **Deactivate**.

4. In the **Confirmation** window, click **OK**.

   The status of each Provisioning Bridge you selected changes from **Active** to **Inactive** .

Modify a Provisioning Bridge

You can change the following items for a Provisioning Bridge:

- The name, description, and Client Secret of the bridge
- The apps to which the bridge is assigned

> **Note:**
>
> You can also change the folder where all log files for the Provisioning Bridge are stored and the log level for these log files. See Manage Log Files for a Provisioning Bridge.

Modify a Provisioning Bridge

You can use the **Provisioning Bridges** page to modify a Provisioning Bridge.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Provisioning Bridges**.
2. Click the Provisioning Bridge that you want to modify.
3. Click **Details**.
   a. To edit the name or descriptive information about the Provisioning Bridge, enter the modifications in the **Name** or **Description** fields.
   b. To regenerate the Client Secret for this bridge, click **Regenerate**.

      > **Note:**
      >
      > If you have activated this Provisioning Bridge, then you can't regenerate a Client Secret for it because the bridge is using this Secret to access Oracle Identity Cloud Service as an administrator.
      >
      > To regenerate the Client Secret for this Provisioning Bridge, you must first deactivate the bridge, and then stop it. See Deactivate Provisioning Bridges and Stop a Provisioning Bridge.
      >
      > If you regenerate the Client Secret for a Provisioning Bridge, then you must delete the `wallet` folder and recreate the Oracle Wallet that you made in Create a Provisioning Bridge so that the wallet contains the regenerated Secret.

   c. Click **Save**.
   d. In the **Confirmation** window, click **OK**.

Assign a Provisioning Bridge to Apps

After creating a Provisioning Bridge, you can assign it to on-premises apps in the App Catalog. Because this bridge serves as a provisioning and synchronizing agent between Oracle Identity Cloud Service and your apps, the bridge can poll for changes to users or groups in the apps and synchronize those changes into Oracle Identity Cloud Service.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. Click the App Catalog app to which you want to assign a Provisioning Bridge.

> **Note:**
>
> For version 19.3.3 of Oracle Identity Cloud Service, you can assign the Provisioning Bridge to the Oracle Internet Directory or Oracle E-Business Suite apps. Also, see the Synchronize Users from Oracle Internet Directory to Oracle Identity Cloud Service and Synchronize and Provision Users Between Oracle E-Business Suite and Oracle Identity Cloud Service videos to learn more about configuring these apps.

3. Click **Deactivate**.
4. In the **Confirmation** window, click **OK**.

> **Note:**
>
> You must deactivate the app so that you can modify it by assigning a Provisioning Bridge to it.

5. Click **Provisioning**.
6. Turn on the **Enable Provisioning** switch.
7. From the **Associate with Provisioning Bridge** list, select the Provisioning Bridge that you want to assign to this app.

> **Note:**
>
> If the Provisioning Bridge has an inactive status, then activate it. See Activate Provisioning Bridges.

8. Click **Save**.
9. Click **Activate**.
10. In the **Confirmation** window, click **OK**.

> **Note:**
>
> By activating this app, the Provisioning Bridge that you assigned to it can be used either to poll the app for changes to users and groups in the app, and synchronize these changes into Oracle Identity Cloud Service, or to provision users to the app.

11. Repeat steps 1-10 for each app to which you want to assign the Provisioning Bridge.
12. In the **Navigation Drawer**, click **Settings**, and then click **Provisioning Bridges**.
13. Click the Provisioning Bridge that you assigned to apps, and then click the **Apps** tab.
14. Verify that you see each app to which you assigned the Provisioning Bridge.

Change the Provisioning Bridge Assigned to Apps

Only one Provisioning Bridge can be assigned to an app at any time. If you want to assign another bridge to the app, then you must replace the bridge that's already associated with the app with the designated Provisioning Bridge.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. Click the App Catalog app to which you want to assign another Provisioning Bridge.

> **Note:**
>
> For version 19.3.3 of Oracle Identity Cloud Service, you can assign a different Provisioning Bridge to the Oracle Internet Directory or Oracle E-Business Suite apps.

3. Click **Deactivate**.
4. In the **Confirmation** window, click **OK**.

> **Note:**
>
> You must deactivate the app so that you can modify it by changing the Provisioning Bridge assigned to it.

5. Click **Provisioning**.
6. Turn on the **Enable Provisioning** switch.
7. From the **Associate with Provisioning Bridge** list, select a different Provisioning Bridge than the one that’s assigned to this app.
8. Click **Save**.
9. Click **Activate**.
10. In the **Confirmation** window, click **OK**.

> **Note:**
>
> By activating this app, the other Provisioning Bridge that you selected for it can be used either to poll the app for changes to users and groups in the app, and synchronize these changes into Oracle Identity Cloud Service, or to provision users to the app.

11. Repeat steps 1-10 for each app to which you want to assign a different Provisioning Bridge.
12. In the **Navigation Drawer**, click **Settings**, and then click **Provisioning Bridges**.
13. Click the Provisioning Bridge that you unassigned from the apps, and then click the **Apps** tab.
14. Verify that you no longer see these apps in the tab.

Stop a Provisioning Bridge

You can stop a Provisioning Bridge that's running on a Windows or generic machine.

> **Important:**
>
> If you stop a Provisioning Bridge, then you must wait three minutes to restart the bridge. Also, before you stop a Provisioning Bridge, you must deactivate it. See Deactivate Provisioning Bridges.

Use the following table to guide you on how to stop a Provisioning Bridge.

Machine	Mode	Action
Generic	`normal`	Close the Terminal window or press `Ctrl + C`.
Generic	`background`	At the prompt of the Terminal window, kill the process by entering `kill -9 [Process_ID]`.

> **Note:**
>
> Because you started the Provisioning Bridge in `background` mode, even if you close the Terminal window, the bridge continues to run. For this reason, you must kill the process to stop the Provisioning Bridge.

> **Tip:**
>
> If you don't know the process ID, then run the following command: `ps -ef | grep CrossPlatformBridgeRunner`

Machine	Mode	Action
Windows	N/A	Close the Command window.

To verify that you stopped the Provisioning Bridge, complete the following steps:

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Provisioning Bridges**.
2. Verify that the Provisioning Bridge that you stopped has a status of **Stopped**.

> **Note:**
>
> If you still see a status of **Started** for the Provisioning Bridge, then wait three minutes, and click **Refresh**.

Remove Provisioning Bridges

You can remove unused Provisioning Bridges from Oracle Identity Cloud Service. You can remove either a single Provisioning Bridge or multiple bridges.

> **Important:**
>
> Before you remove any Provisioning Bridges, make sure that you:
>
> - Deactivate the Provisioning Bridges. See Deactivate Provisioning Bridges.
> - Assign different Provisioning Bridges to apps. See Change the Provisioning Bridge Assigned to Apps.
> - Stop the Provisioning Bridges. See Stop a Provisioning Bridge.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Provisioning Bridges**.
2. Select the check box for each Provisioning Bridge that you want to remove.
3. Click **Remove**.
4. In the **Confirmation** window, click **OK**.

Manage Log Files for a Provisioning Bridge

After you install and start a Provisioning Bridge, you may want to access the log files for troubleshooting purposes. You can locate these files in the `logs` folder.

The `logs` folder is contained in the directory that you created when you unzipped the file for the **Identity Cloud Service Provisioning Bridge** client in Create a Provisioning Bridge.

You can change the folder path where all log files for the Provisioning Bridge are stored and the log level for these log files. To do this, you modify the `log4j.properties` file.

The `log4j.properties` file is located in the `conf` folder of the directory that you created when you unzipped the file for the **Identity Cloud Service Provisioning Bridge** client, and contains properties associated with logging operations that are performed by the Provisioning Bridge.

1. Navigate to the `conf` folder.
2. Using a text editor, open the `log4j.properties` file.
3. In the file, locate the following line of code: `property.baseLocation = ./logs/`
4. Change the value of the `property.baseLocation` parameter to the folder path where you want all log files for the Provisioning Bridge to be stored.
5. Locate the following line of code: `filter.threshold.level = error`
6. Change the value of the `filter.threshold.level` parameter to one of the following log levels:

Log Level	Description
`all`	Capture all events
`debug`	Capture fine-grained informational events that are most useful to debug the Provisioning Bridge
`error`	Capture error events that might still allow the Provisioning Bridge to continue running
`info`	Capture informational events that highlight the progress of the Provisioning Bridge at a coarse-grained level

7. Save and close the `log4j.properties` file.

> **Note:**
>
> You must stop the Provisioning Bridge and restart it for the changes you made to the `log4j.properties` file to take effect. Also, after you stop the Provisioning Bridge, you must wait three minutes to restart it.

Upgrade a Provisioning Bridge

If you're using version 19.2.1 of the Provisioning Bridge, then upgrade to version 19.3.3 of the bridge.

1. If your Provisioning Bridge is started, then stop it. See Stop a Provisioning Bridge.
2. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Downloads**.
3. In the **Downloads** page, click **Download** to the right of the **Identity Cloud Service Provisioning Bridge** client.

   Oracle Identity Cloud Service downloads the client for the Provisioning Bridge.
4. Verify that a **Success** status appears to the right of the **Identity Cloud Service Provisioning Bridge** client.
5. Launch the Windows or generic machine where you installed the 19.2.1 version of the client for the Provisioning Bridge. See Create a Provisioning Bridge.

6. On this machine, create a **BRIDGE_NEW_HOME** folder, and then unzip the file that you downloaded in step 3 of this procedure into this folder.
7. Copy the **crossplatform.jar** file from the **BRIDGE_NEW_HOME/bin** folder to the **BRIDGE_OLD_HOME/bin** folder. For this procedure, **BRIDGE_OLD_HOME** represents the folder where you unzipped the file for the 19.2.1 version of the client for the Provisioning Bridge. See Create a Provisioning Bridge.
8. Copy the **org.identityconnectors.ebs-1.0.jar** file from the **BRIDGE_NEW_HOME/bundle_home** folder to the **BRIDGE_OLD_HOME/bundle_home** folder.
9. Copy the **log4j.properties** file from the **BRIDGE_NEW_HOME/conf** folder to the **BRIDGE_OLD_HOME/conf** folder.
10. Copy the **BRIDGE_NEW_HOME/dependencies** folder to the **BRIDGE_OLD_HOME** folder.
11. Start your Provisioning Bridge. See Start a Provisioning Bridge.

17

Manage Microsoft Active Directory (AD) Bridges for Oracle Identity Cloud Service

Learn how to manage Microsoft Active Directory (AD) Bridges for Oracle Identity Cloud Service.

Topics

Typical Workflow for Managing Microsoft Active Directory (AD) Bridges for Oracle Identity Cloud Service

With the Microsoft Active Directory (AD) Bridge feature in Oracle Identity Cloud Service, you can create, manage, and remove AD Bridges.

Task	Description	Additional Information
Understand the AD Bridge.	You can receive an overview of the AD Bridge for Oracle Identity Cloud Service. You can also learn why you should use the AD Bridge, and how it's used to synchronize users and groups between AD and Oracle Identity Cloud Service.	About the Microsoft Active Directory (AD) Bridge Why Use the Microsoft Active Directory (AD) Bridge?
Set permissions for your AD account.	Before creating an AD Bridge, you must set permissions for your AD domain administrator account. You must set these permissions so that you can install the bridge and configure delegated authentication for it.	Set Permissions for Your Microsoft Active Directory (AD) Account
Create an AD Bridge.	You can create an AD Bridge using the **Directory Integrations** page and the client for the bridge.	Create a Microsoft Active Directory (AD) Bridge
Configure an AD Bridge.	You can configure an AD Bridge using the **Directory Integrations** page.	Configure a Microsoft Active Directory (AD) Bridge
Define attribute mappings for an AD Bridge.	You can define attribute mappings for an AD Bridge using the **Directory Integrations** page.	Define Attribute Mappings for a Microsoft Active Directory (AD) Bridge
Run an AD Bridge.	You can run an AD Bridge manually or view a synchronization log about the bridge being run using the **Directory Integrations** page.	Run a Microsoft Active Directory (AD) Bridge
View details about an AD Bridge.	You can view details about an AD Bridge using the **Directory Integrations** page.	View Details About a Microsoft Active Directory (AD) Bridge
Activate and deactivate AD Bridges.	You can activate and deactivate AD Bridges using the **Directory Integrations** page.	Activate and Deactivate Microsoft Active Directory (AD) Bridges
Modify an AD Bridge.	You can modify an AD Bridge using the **Directory Integrations** page.	Modify a Microsoft Active Directory (AD) Bridge
Remove an AD Bridge.	You can remove an AD Bridge using the **Directory Integrations** page and the client for the bridge.	Remove a Microsoft Active Directory (AD) Bridge
Understand best practices for the AD Bridge.	You can learn about best practices for creating, managing, and maintaining the AD Bridge.	Log Files

You can create, manage, and remove AD Bridges by using:

- The Identity Cloud Service console
- SCIM-based APIs

In the following sections, you learn how to manage AD Bridges by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

About the Microsoft Active Directory (AD) Bridge

The Microsoft Active Directory (AD) Bridge provides a link between your AD enterprise directory structure and Oracle Identity Cloud Service.

Prerequisite

Enable AD Bridge. This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service.

Topics:

Understand the Microsoft Active Directory (AD) Bridge

The Microsoft Active Directory (AD) Bridge provides a link between your AD enterprise directory structure and Oracle Identity Cloud Service. Oracle Identity Cloud Service can synchronize with this directory structure so that any new, updated, or deleted user or group records are transferred into Oracle Identity Cloud Service. Each minute, the AD Bridge polls AD for any changes to these records and brings these changes into Oracle Identity Cloud Service. So, if a user is deleted in AD, then this change will be propagated into Oracle Identity Cloud Service. Because of this synchronization, the state of each record is synchronized between AD and Oracle Identity Cloud Service.

After users are synchronized from AD to Oracle Identity Cloud Service, if you activate or deactivate a user, modify the user's attribute values, or change the group memberships for the user in Oracle Identity Cloud Service, then these changes are propagated to AD through the AD Bridge.

> **Note:**
>
> The AD organizational units (OUs) contain the users and groups that are imported into Oracle Identity Cloud Service.

You can configure Oracle Identity Cloud Service to synchronize with one or multiple AD domains by installing an AD Bridge for each domain.

> **Note:**
>
> You must install the AD Bridge on the machine that’s attached to the Microsoft Active Directory domain for auto discovery. You don’t have to install the bridge on the domain controller.

Figure 17-1 Inbound Directory Synchronization

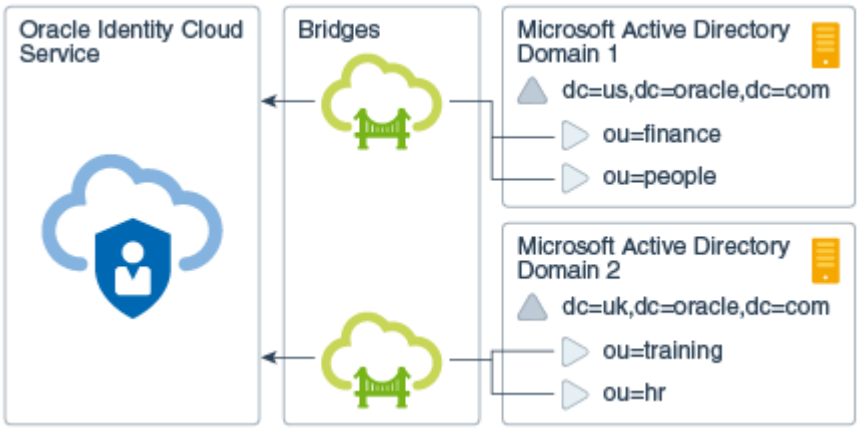
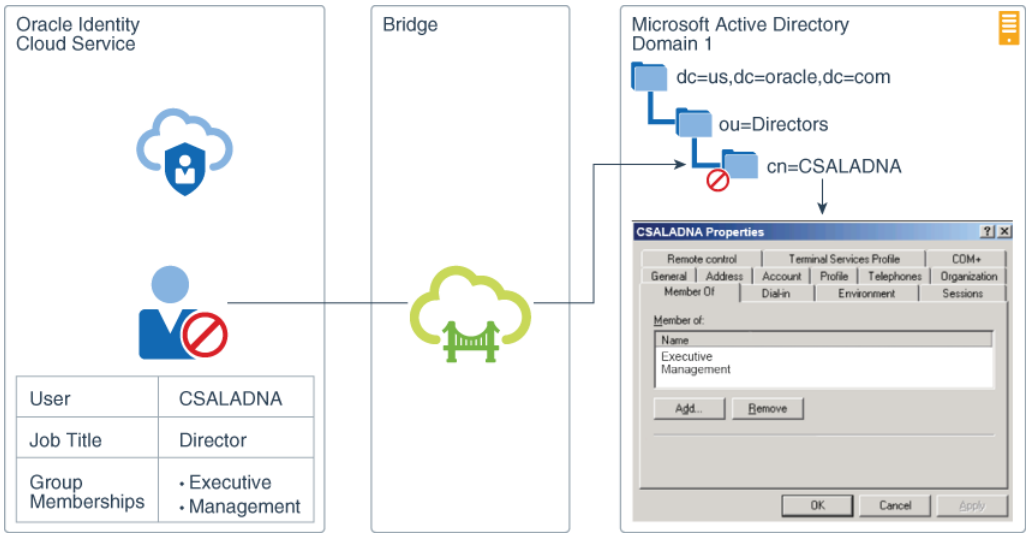


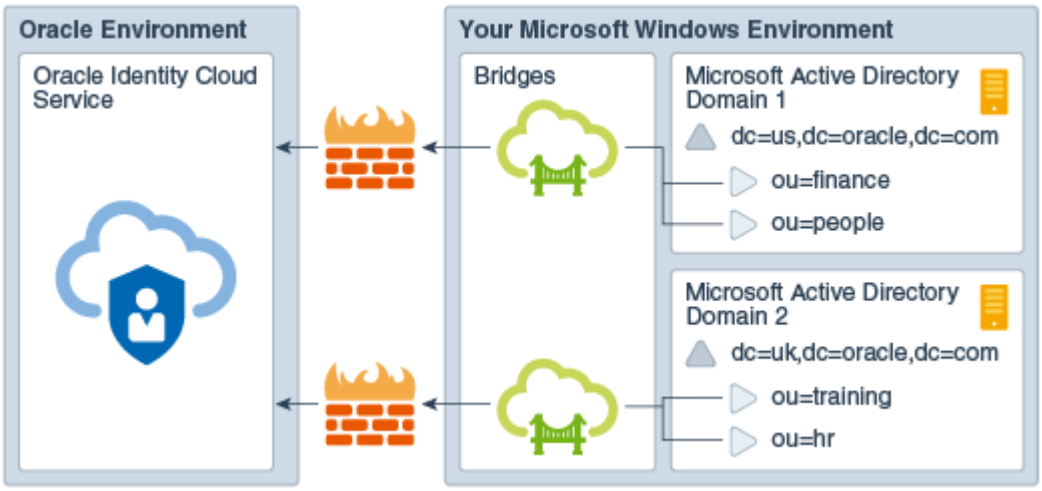
Figure 17-2 Outbound Directory Synchronization



In the diagram above, Clarence Saladna (CSALADNA) is a user who's been synchronized from AD to Oracle Identity Cloud Service through the AD Bridge. In Oracle Identity Cloud Service, an administrator deactivates Clarence's account because he's on vacation. Also, because Clarence received a promotion, he has a new job title of Director and belongs to different groups that are associated with his new role, including the Executive and Management groups. The AD Bridge can be used to propagate these changes to AD.

Both the AD Bridges and your AD enterprise directory structure are in your Microsoft Windows environment (for example, Microsoft Windows 2003). Because Oracle Identity Cloud Service is an Oracle Cloud service, it's in an Oracle environment.

Figure 17-3 Bridge Security



> **Note:**
>
> If an AD user attribute is multi-valued, then the AD Bridge will transfer only the first value of the attribute into Oracle Identity Cloud Service.

You can access the Integrating with Active Directory Using Identity Bridge tutorial to see how to integrate AD and Oracle Identity Cloud Service.

Certified Components

With the Microsoft Active Directory (AD) Bridge, Oracle Identity Cloud Service can connect to your AD enterprise directory structure.

The following table lists the certified versions for Oracle Identity Cloud Service, AD, your operating system, and the Microsoft .NET software framework (which is required for the AD Bridge to run).

Oracle Identity Cloud Service	AD	64–Bit	Operating System	.NET Framework
20.1.3	Microsoft Windows Server 2008 Microsoft Windows Server 2008 R2 Microsoft Windows Server 2012 Microsoft Windows Server 2012 R2 Microsoft Windows Server 2016 Microsoft Windows Server 2019	Yes	Windows 10 v1607 or later Windows Server 2016 or later	Version 4.6+

Statuses

Learn about the various statuses for Microsoft Active Directory (AD) and the AD Bridge.

There are two statuses for the AD domain with which the AD Bridge is communicating:

- Partially Configured: The AD Bridge is installed, but it's not configured to communicate with either the AD domain or Oracle Identity Cloud Service.
- Configured: The AD Bridge is installed and configured, and available to synchronize with the AD domain.

There are three statuses for the AD Bridge:

- Active: The AD Bridge is installed and configured, and available to synchronize with AD to retrieve user accounts and groups.
- Inactive: The AD Bridge is installed and configured, but it's not available to synchronize with AD. For performance reasons, this is done.
- Unreachable: The AD Bridge is installed and configured. However, one of the following conditions has occurred:
  - The back-end service used to establish communication between Oracle Identity Cloud Service and AD is stopped.
  - The Oracle Identity Cloud Service administrator uninstalled the client associated with the AD Bridge, but the bridge couldn't be removed from the **Directory Integrations** page of the Identity Cloud Service console because the client can't connect to the Oracle Identity Cloud Service server. Oracle Identity Cloud Service can't use the bridge to communicate with AD. See Remove a Microsoft Active Directory (AD) Bridge.
  - The administrator regenerated the Client Secret for the AD Bridge, and then uninstalled the client for the bridge.

Hardware Requirements

Learn about the minimum hardware requirements for setting up the Microsoft Active Directory (AD) Bridge.

The minimum hardware requirements are, as follows:

- 1 GB of RAM
- 1 GB of disk space
- A quad-core CPU

Why Use the Microsoft Active Directory (AD) Bridge?

Learn about why you should use the Microsoft Active Directory (AD) Bridge.

Most customers have AD as their central directory service. These customers also use AD as their network directory. This directory is where all of their workstations are connected to and from where they manage their users.

In addition to AD, customers use an enterprise LDAP to centralize all of their user identities. So, a customer uses AD to manage their employees, but in the centralized LDAP, the customer manages their partners, consumers, and any other users with which the customer has relationships.

For these reasons, it's imperative that Oracle Identity Cloud Service can integrate with both AD and an enterprise LDAP (for example, Oracle Internet Directory).

By using Oracle Identity Cloud Service, customers can control when they will migrate their directory-based applications to the cloud. In the interim, they can use one of the following:

- AD Bridge: This bridge provides a link between your AD enterprise directory structure and Oracle Identity Cloud Service. Oracle Identity Cloud Service can synchronize with this directory structure so that any new, updated, or deleted user or group records are transferred into Oracle Identity Cloud Service. Each minute, the bridge polls AD for any changes to these records and brings these changes into Oracle Identity Cloud Service. So, if a user is deleted in AD, then this change will be propagated into Oracle Identity Cloud Service. As a result, the state of each record is synchronized between AD and Oracle Identity Cloud Service. After the user is synchronized from AD to Oracle Identity Cloud Service, if you activate or deactivate a user, modify the user's attribute values, or change the group memberships for the user in Oracle Identity Cloud Service, then these changes are propagated to AD through the AD Bridge.
- Provisioning Bridge: This bridge provides a link between your enterprise LDAP (such as Oracle Internet Directory) and Oracle Identity Cloud Service. Through synchronization, account data that's created and updated directly on the LDAP is pulled into Oracle Identity Cloud Service and stored for the corresponding Oracle Identity Cloud Service users and groups. As a result, any changes to these records will be transferred into Oracle Identity Cloud Service. Because of this, the state of each record is synchronized between the LDAP and Oracle Identity Cloud Service. See Manage Provisioning Bridges for Oracle Identity Cloud Service.

This chapter focuses on using the AD Bridge to synchronize users and groups between AD and Oracle Identity Cloud Service.

About Multiple AD Bridges for High Availability and Load Balancing

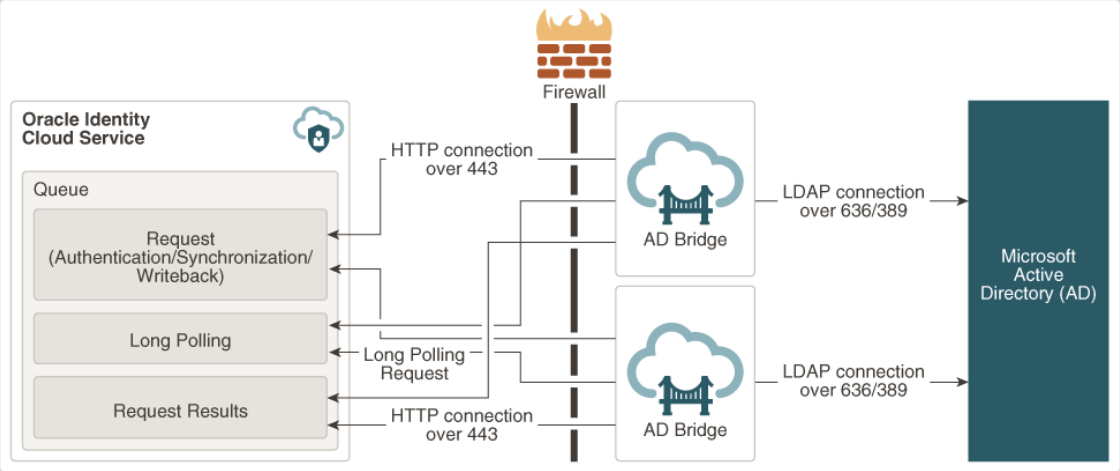
Learn how to set up High Availability and Load Sharing so that you don't have a single point of failure for your AD Bridge architecture.

If you only have one Microsoft Active Directory (AD) Bridge component in one Windows Service connecting to your Active Directory domain, it can be a single point of failure in the architecture.

To avoid this, Oracle Identity Cloud Service supports the installation of multiple AD Bridge instances mapping to the same Active Directory domain.

The maximum number of AD Bridges that an administrator can install per domain must not exceed five (5). In addition, the maximum number of domains that an administrator can configure per tenant must not exceed 10. To configure these limits, raise an SR with Oracle Support.

With a AD Bridge High Availability (HA) deployment of at least two AD Bridges per domain, delegated authentication and data synchronization loads can be shared among all the AD Bridges. The allocation of requests to a AD Bridge is completely random, depending on the availability of that particular AD Bridge. One delegated authentication request will be picked up by one AD Bridge. An AD Bridge can pick delegated authentication and full or incremental synchronization as well. Both AD Bridges have the capability to perform data synchronization and delegated authentication simultaneously. However, only one AD Bridge can perform data synchronization of a domain at a time.



Enable HA for an Existing Deployment

Learn about the prerequisites and the limitations for AD Bridge High Availability (HA) on an existing deployment before you enable AD Bridge HA.

Prerequisites

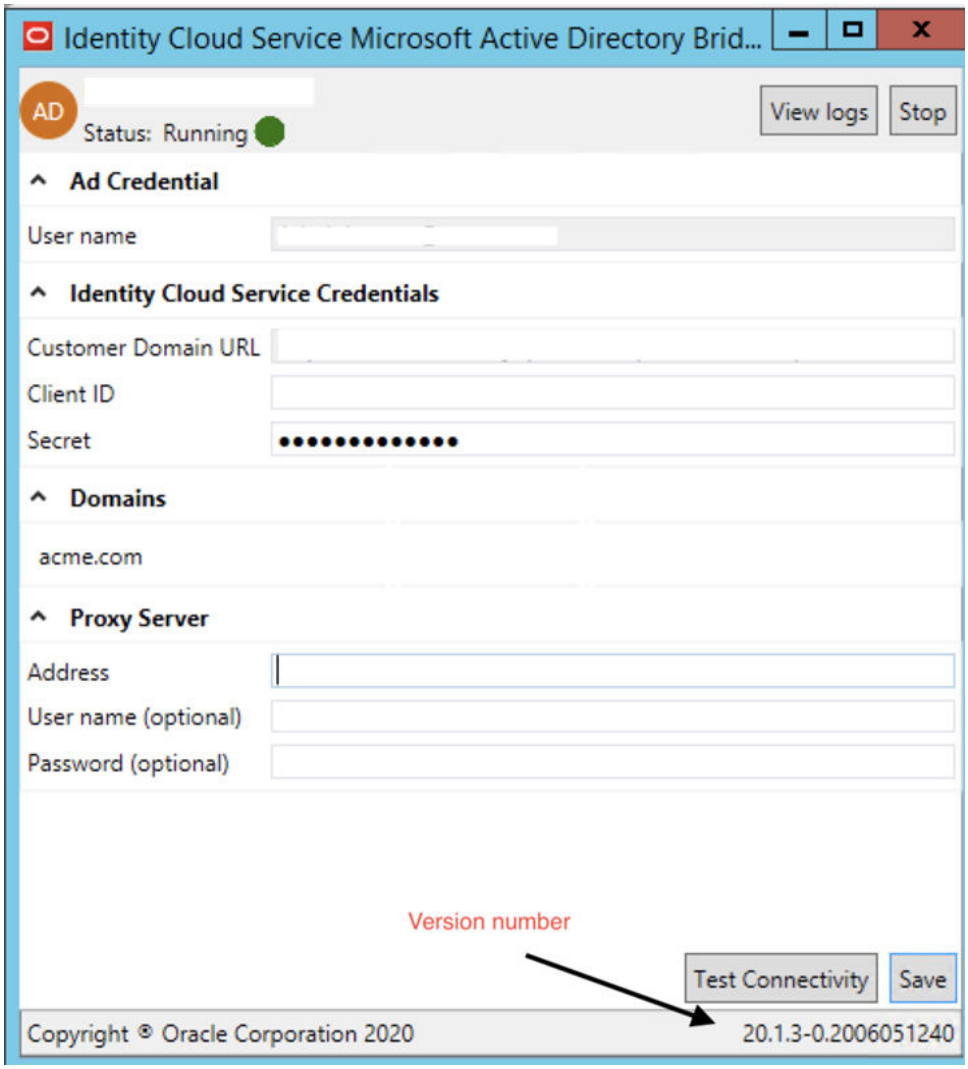
1. Upgrade all AD Bridges in all domains to version 20.1.3 or greater for every domain that you have configured.
2. **Enable High Availability.** This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service. Once HA is enabled, it is enabled for all configured domains.

Limitations

Note the following limitations for AD Bridge HA.

- Only one AD Bridge can be configured in one Windows machine. To configure multiple AD Bridges you have to use multiple Windows machines in the same domain. Note that without AD Bridge HA enabled by Oracle Support, installation of second AD Bridge for the domain will fail.
- Maximum of 5 AD Bridges per domain can be configured by an Administrator for HA and load sharing.
- AD Bridge HA won't work if any one of the AD Bridges installed for a domain is version 19.3.3 and below. To check the version of an AD bridge, open the AD Bridge user interface and note the version in the bottom right corner of the window, as seen in the screenshot below.

Figure 17-4 Check Whether AD Bridges Installed for a Domain is Version 19.3.3 and Below



Troubleshooting

If additional bridges won't install for a domain, make sure that all prerequisites are met and that none of the constraints apply to you.

Enable HA for a New Deployment

Learn how to enable AD Bridge High Availability (HA) for a new deployment as well as learn about the limitations for HA on a new deployment.

Prerequisite

Enable High Availability. This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service. Once HA is enabled, it is enabled for all configured domains.

Limitations

Note the following limitations for AD Bridge High Availability.

- AD Bridge HA will not work if any one of the AD Bridges installed for a domain is version 19.3.3 and below.
- Only one AD Bridge can be configured in one Windows machine. To configure multiple AD Bridges you have to use multiple Windows machines in the same domain. Note that without AD Bridge HA enabled by Oracle Support, installation of second AD Bridge for the domain will fail.
- Maximum of 5 AD Bridges per domain can be configured by an Administrator for HA and load sharing.

Troubleshooting

If additional bridges won't install for a domain, make sure that all prerequisites are met and that none of the constraints applies to you.

Check a Bridge Data Synchronization Status

Use the following steps to check whether a AD Bridge is running a data synchronization.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Directory Integrations**.
2. Check the status. You should see examples like the below.

Figure 17-5 No AD Bridges are running Active Sync.

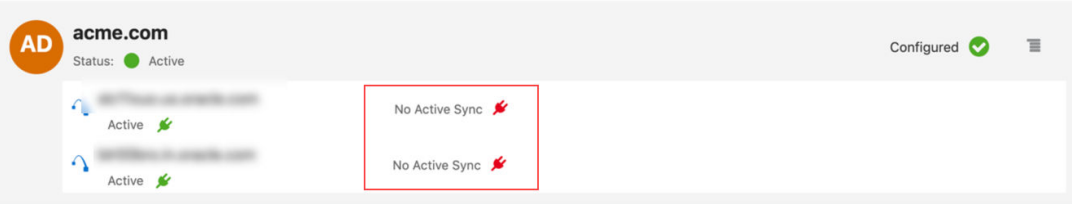
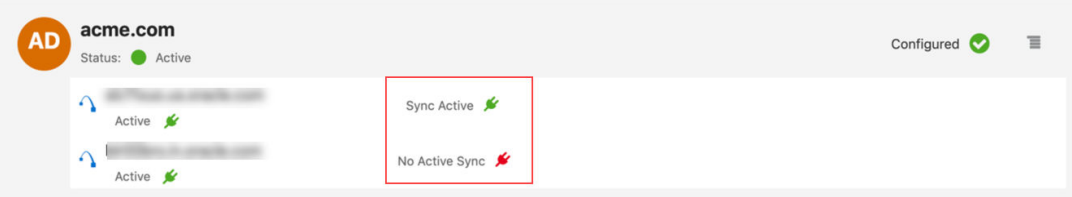


Figure 17-6 One AD Bridges is running Active Sync.



Test Active Directory Connectivity

Learn how to test connectivity or network related issues on a machine where the Active Directory (AD) Bridge client is installed.

There are 2 types of connectivity in AD Bridge:

- Connectivity between AD Bridge and LDAP server of Active Directory
- Connectivity between AD Bridge and Oracle Identity Cloud Service

To test the connectivity of AD Bridge follow below steps:

1. Open **ADBridgeUI.exe**. It's in the ADBridge installation folder. The default path is `C:\Program Files\Oracle\IDBridge`.
2. Click the **Test Connectivity** button.

AD Bridge Connectivity Notifications

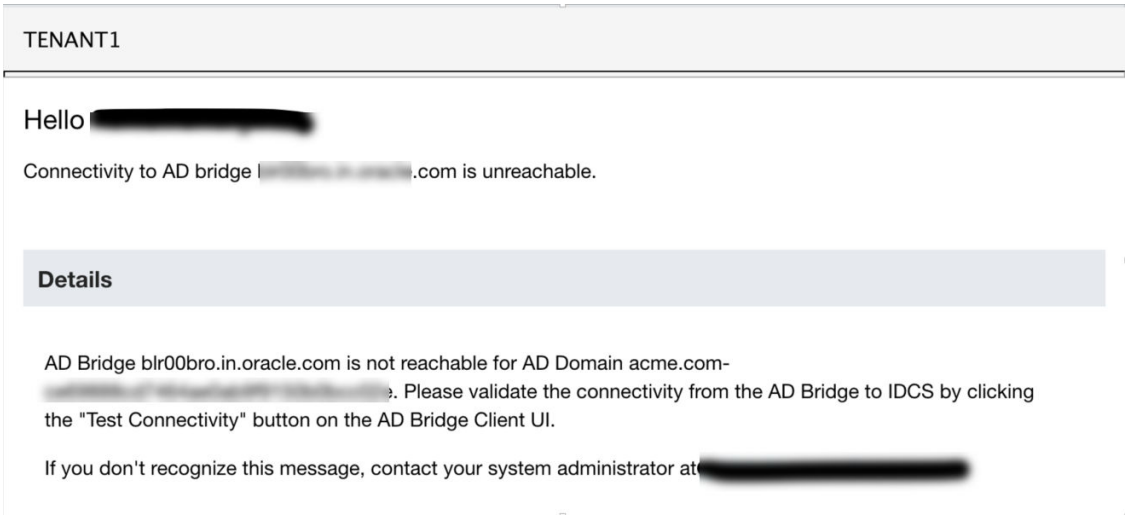
Learn about notifications that Oracle Identity Cloud Service sends to the tenant admin when connectivity between AD Bridge and the Oracle Identity Cloud Service server is broken and also when it is restored.

Notifications sent when connectivity of AD Bridge with Oracle Identity Cloud Service server is broken

Oracle sends notifications to the tenant admin when connectivity between AD Bridge and the Oracle Identity Cloud Service server is broken. Connectivity could be broken because of multiple reasons, for example if the AD Bridge is stopped or if the Oracle Identity Cloud Service is stopped on the Windows machine.

The notifications will have the email subject: *Connectivity to AD bridge <windows machine name> is unreachable.*

Figure 17-7 Email body for a broken connection.

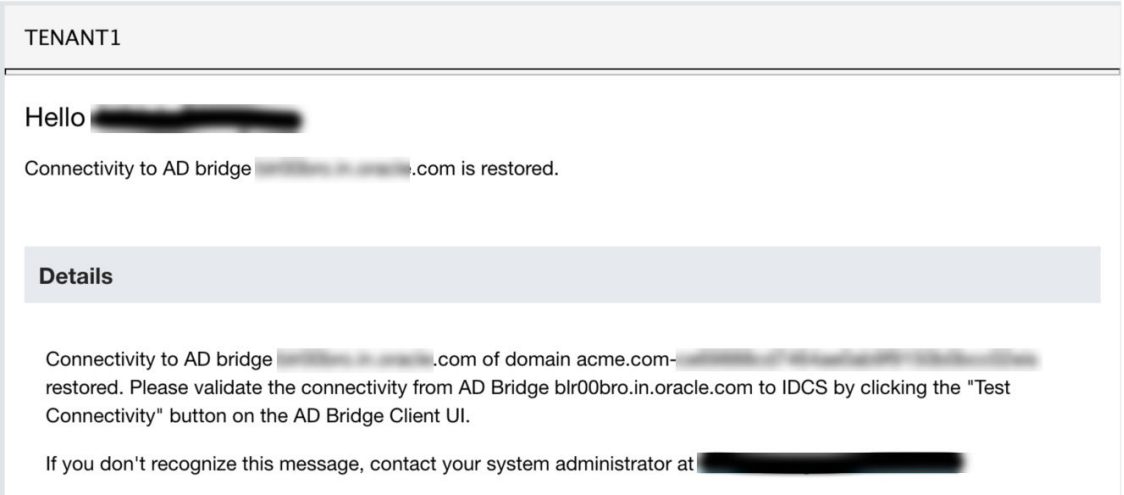


Notifications sent when connectivity of AD Bridge with Oracle Identity Cloud Service server is restored

Similarly, when connectivity is restored, an email will be sent from Oracle to the tenant administrator.

The notifications will have the email subject: *Connectivity to AD bridge <windows machine name> is restored.*

Figure 17-8 Email body for a restored connection.



Use REST API to Configure Email Notifications

Using the REST API, Administrators can also configure who should receive email notifications when connectivity is broken and restored.

Using REST API, the Administrator can rovide comma separated list of emails IDs to which to send the notififcations using `PATCH /admin/v1/Settings/Settings`.

Example payload:

```
{
    "schemas":[
      "urn:ietf:params:scim:api:messages:2.0:PatchOp"
    ],
    "Operations":[
       {
           "op": "replace",
           "path":         "contactEmails",
           "value": [
        "admin@oracle.com",
        "<emailid>@gmail.com"
      ]
       }

             ]
}
```

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Set Permissions for Your Microsoft Active Directory (AD) Account

You use your Microsoft Active Directory (AD) domain administrator account to create an AD Bridge. Before creating this bridge, you must set permissions for your account. You must set

these permissions so that you can install the bridge and configure delegated authentication for it.

With delegated authentication, Oracle Identity Cloud Service identity domain administrators and security administrators don't have to synchronize user passwords between AD and Oracle Identity Cloud Service. Users can use their AD passwords to sign in to Oracle Identity Cloud Service to access resources and applications protected by Oracle Identity Cloud Service.

See Understand Delegated Authentication for more information about delegated authentication.

Topics:

Set Permissions to Synchronize Users, Groups, and Group Membership

You set permissions for your Active Directory Bridge service account so that you can synchronize users, groups, or OUs between Microsoft Active Directory (AD) and Oracle Identity Cloud Service

1. Use your domain administrator credentials to sign in to the machine that contains your AD server.
2. Open a command window.
3. Set the **Generic Read** permissions for the users, groups, and organizational units (OU) in the AD domain that you want to import into Oracle Identity Cloud Service:

```
dsacls <AD_Domain_Name> /I:T /g "<AD_Domain_Name>\<User/Group_Name>:GR"
```

> **Note:**
>
> `<AD_Domain_Name>` is the name of the domain that you're associating with Oracle Identity Cloud Service and `<User/Group_Name>` is the username of your domain administrator account.
>
> `/I:T`: This parameter specifies the objects to which you are applying the permissions. T is the default, which means you can propagate inheritable permissions to this object and child objects down to one level only.
>
> `/g`: This parameter grants the permissions that you specify to the user or group. For example, `/g {<user> | <group>}:<permissions>`.
>
> `<permissions>`: This parameter specifies the type of permissions that you are applying.
>
> - `GR`: Generic Read
> - `GW`: Generic Write
> - `LC`: List the child objects of the object
> - `RP`: Read Property

4. Set the **List Children** and **Read** properties for the **cn=Deleted Objects** container with inheritance. This container is also in the AD domain that you're associating with Oracle Identity Cloud Service.

   ```
   dsacls "cn=deleted objects,<AD_Domain_Name>" /takeOwnership
   ```

   ```
   dsacls "cn=deleted objects,<AD_Domain_Name>" /I:T /g "<AD_Domain_Name>\<User/Group_Name>:LCRP"
   ```

   > **Note:**
   >
   > If you don't have the above permissions, then the AD Bridge won't be able to synchronize deleted users, groups, or OUs between AD and Oracle Identity Cloud Service. This will cause inconsistencies between AD and Oracle Identity Cloud Service.

Set Permissions to Propagate Changes to Microsoft Active Directory

You set permissions for your Active Directory Bridge service account so that you can propagate changes you have done in Oracle Identity Cloud Service to Microsoft Active Directory (AD) through the AD Bridge.

1. Use your domain administrator credentials to sign in to the machine that contains your AD server.
2. Open a command window.
3. Set the **Generic Write** permission for the users, groups, and organizational units (OU) in the AD domain, if you want to propagate the changes you have done in Oracle Identity Cloud Service to Active Directory.

   ```
   dsacls <AD_Domain_Name> /I:T /g "<AD_Domain_Name>\<User/Group_Name>:GW"
   ```

Set Permissions for Delegated Authentication

You set permissions for your Microsoft Active Directory (AD) domain administrator account so that you can configure delegated authentication for the AD Bridge.

1. Open **Active Directory Users and Computers**.
2. Right-click the user, group, or organizational unit (OU) that you want to delegate, and then click **Delegate Control**.
3. On the **Delegation of Control** wizard, click **Next**, and then click **Add**.
4. On the **Select Users, Computers, or Groups** dialog box, in the text area, enter the user name or group name that needs to be granted permissions to configure delegated authentication.
5. Click **Check Names** to verify that the user or group has been created in AD. If it hasn't been created, then create it.
6. Click **OK**, and then click **Next**.
7. Select the **Delegate the following common tasks** option, and then select **Reset user passwords and force password change at next logon**.
8. Click **Next**, and then click **Finish**.

The next steps explain how to set specific permissions to lock and unlock user accounts.

9. Right-click on the newly modified user or group, and select **Properties**.
10. Select the **Security** tab, click **Advanced**.
11. On the Advanced Security Settings, click **Add**.
12. On the Permission Entry wizard, click **Select a principal**, and enter the same user name or group name that has been granted reset permission.
13. Click **OK**.
14. In the **Applies to** field, select **Descendant User objects**.

    The list of permissions allowed for the user account (Principal) displays.
15. Scroll down and enable **Read lockoutTime**, and **Write lockoutTime**.
16. Click **OK** and continue to click **OK** until the end of the setup.

    The user account now has permissions to change passwords for all the user objects present in the high-level context.

Create a Microsoft Active Directory (AD) Bridge

To create a Microsoft Active Directory (AD) Bridge that provides a link between your AD enterprise directory structure and Oracle Identity Cloud Service, you must be assigned to either the identity domain administrator role or the security administrator role. You must also have administrative rights to access the AD domain that you want to monitor by using the bridge.

Part of creating the AD Bridge is providing administrative credentials for both AD and Oracle Identity Cloud Service. The bridge requires these credentials to communicate with AD and Oracle Identity Cloud Service as an administrator.

See Add or Remove a User Account from an Administrator Role for more information about assigning administrator roles to users.

Important:

The AD account used to install the AD Bridge should have the following permissions:

- **Generic Read** for the users and groups in the AD domain that you want to import into Oracle Identity Cloud Service
- **Generic Read** for all organizational units (OUs) in the domain
- **Generic Read** for the **cn=Configuration** container in the domain
- The **List Children** and **Read** properties for the **cn=Deleted Objects** container with inheritance

If this account is also used to configure delegated authentication for the AD Bridge, then the account should have the following permissions:

- **Change Password**
- **Reset Password**
- **Read pwdLastSet**
- **Write pwdLastSet**
- **Read lockoutTime**
- **Write lockoutTime**

See Set Permissions for Your Microsoft Active Directory (AD) Account.

You can access the Managing Security Settings infographic to see how to create an AD Bridge.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Directory Integrations**.
2. If this is the first AD Bridge you're creating, then click **Add a Microsoft Active Directory Bridge**. Otherwise, click **Add**.
3. In the **Install Bridge** page, make a note of the Identity Cloud Service URL, Client ID, and Client Secret.

   The Identity Cloud Service URL contains the name and port number for your Oracle Identity Cloud Service identity domain. The Client ID and Client Secret are used by the AD Bridge to access Oracle Identity Cloud Service as an administrator.

   Note:

   The Client Secret is encrypted (for security purposes). To see the Secret in clear text, click **Show Secret**. To regenerate the Secret for the AD Bridge, click **Regenerate**.

4. Click **Download**.

   Oracle Identity Cloud Service downloads the client for the AD Bridge.

> **Note:**
>
> Don't close the **Install Bridge** page. You'll need to reference the Identity Cloud Service URL, Client ID, and Client Secret when creating the AD Bridge.

5. To install the client for the AD Bridge, double-click the `ad-id-bridge.exe` file.

   The **Welcome to AD Bridge Installer** window appears.

6. In the **Language Selection** area, select the language that you want to use to install the client for the AD Bridge, and then click **OK**.

   The **Identity Cloud Service Microsoft Active Directory Bridge Installer** appears.

> **Tip:**
>
> While you're installing the client for the AD Bridge, Oracle Identity Cloud Service generates log files for the bridge automatically, and stores them in the `%Temp%` directory.

7. If the **Open File — Security Warning** dialog box appears, then click **Run**. Otherwise, go to step 8.
8. In the **Welcome** dialog box, click **Next**.
9. In the **Destination Folder** dialog box, choose one of the following install choices:
   - To install the client in the default directory, click **Next**.
   - To select another directory to install the client:
     a. Click **Browse**.
     b. In the **Browse For Folder** dialog box, select the directory where Oracle Identity Cloud Service will install the client.
     c. Click **OK**.
     d. Click **Next**.
10. In the **Specify Proxy Server** dialog box:
    a. If your organization has a firewall in place and requires communication to be handled using an HTTP Proxy Server, then select **Use Proxy Server**. If you select this check box, then provide the full path (or address) of the proxy server and the administrator credentials for connecting to the proxy server.
    b. If your organization doesn't require communication to be handled using an HTTP Proxy Server, then don't select **Use Proxy Server**.
    c. Click **Next**.
11. In the **Specify Identity Cloud Service Credentials** dialog box:
    a. Provide the Cloud Service URL, Client ID, and Client Secret.

> **Tip:**
>
> These credentials appear on the **Install Bridge** page of the Identity Cloud Service console.

b. Click **Test**.

The AD Bridge attempts to connect to the Oracle Identity Cloud Service server.

If a connection can be established, then a `Connection Successful!` confirmation message appears.

Otherwise, you'll receive an error message, indicating that you entered an incorrect Cloud Service URL, Client ID, or Client Secret. Modify the incorrect values, and click **Test** again.

c. Click **Next**.

12. In the **Specify Microsoft Active Directory Credentials** dialog box, provide the following connection details to the AD server:

a. **Username:** The AD account that the AD Bridge uses to access the AD server.

b. **Password:** The password for the AD account.

c. **Use SSL:** If you're connecting to the server via an SSL connection, then leave this check box selected. Otherwise, deselect it.

> **Note:**
>
> Oracle recommends that you keep the **Use SSL** check box selected because this results in a faster and more-secure connection. After you select or deselect this check box, and install the client for the AD Bridge, you can't modify this setting.

d. Click **Test**.

The AD Bridge attempts to connect to the AD server.

If a connection can be established, then a `Connection Successful!` confirmation message appears.

Otherwise, you'll receive an error message, indicating that:

- You entered an incorrect username or password. Modify the incorrect values, and click **Test** again.
- You're attempting to connect to the AD server via an SSL connection, but the certificate for the server isn't trusted. Make sure that this certificate is valid, and is present in the trust store of your machine. Then, click **Test** again.

e. Click **Next**.

13. In the **Summary** dialog box, click **Close**.

14. In the Identity Cloud Service console, access the **Directory Integrations** page.

The AD Bridge that you created for the AD domain appears with a status of **Partially Configured**. The bridge is created, but not configured. See Configure a Microsoft Active Directory (AD) Bridge for more information about configuring this bridge.

> **Note:**
>
> If you don't see the AD Bridge in the **Directory Integrations** page, then refresh your web browser. Also, you can create only one bridge per AD domain.

Configure a Microsoft Active Directory (AD) Bridge

After creating a Microsoft Active Directory (AD) Bridge, you configure it by:

- Selecting the AD organizational units (OUs) and groups with which you want Oracle Identity Cloud Service to synchronize using the AD Bridge. The OUs contain the users that you want to import into Oracle Identity Cloud Service. By synchronizing with AD, the bridge can transfer new, updated, or deleted user or group records into Oracle Identity Cloud Service.
- Specifying whether, after a user or group is synchronized from AD to Oracle Identity Cloud Service, if you activate or deactivate a user, modify the user's attribute values, or change the group memberships for the user in Oracle Identity Cloud Service, these changes will be propagated to AD.
- Scheduling how often you want Oracle Identity Cloud Service to use the AD Bridge to import users and groups from AD.
- Defining custom attribute mappings between AD and Oracle Identity Cloud Service.
- Specifying whether users can use their Oracle Identity Cloud Service or AD passwords, or their federated accounts, to authenticate into Oracle Identity Cloud Service to access resources that are protected by Oracle Identity Cloud Service, such as the My Profile console, the Identity Cloud Service console, or any apps assigned to the users.

You can access the Managing Security Settings infographic to see how to configure an AD Bridge.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Directory Integrations**.
2. Click the AD Bridge that you want to configure.

> **Note:**
>
> The bridge has a status of **Partially Configured**.

3. In the **Configure the Microsoft Active Directory Domain** page, configure the AD domain to poll for changes to users or groups in AD and import those changes into Oracle Identity Cloud Service.
   a. In the **Select organizational units (OUs) for users** and **Select organizational units (OUs) for groups** panes:
      i. Select the **Include Hierarchy** check box. If you select a parent OU, then all children OUs will be selected. The OUs contain the users and groups that you want to import into Oracle Identity Cloud Service.

         OR

         Deselect the check box. If you select a parent OU, then children OUs won't be selected.

      ii. Select the check box for each OU that contains users or groups with which you want Oracle Identity Cloud Service to synchronize using the AD Bridge.

> **Note:**
>
> If you don't see any OUs for users or groups in the **Select organizational units (OUs) for users** and **Select organizational units (OUs) for groups** panes, then refresh your web browser.
>
> To force a full synchronization between AD and Oracle Identity Cloud Service, deselect all check boxes for selected user or group OUs, click **Save**, and then in the **Save Configuration Changes?** dialog box, click **OK**. Then, click **Import** to import the users and groups from AD.

iii. Optional. In the **Filter** text box, enter a custom filter to search for user or group OUs. For example, entering `(sn=Smith)` will return all users with the last name of Smith. Or, enter `(department=IT)` to return the IT group.

> **Tip:**
>
> - To select all users or groups, select the **Include Hierarchy** check box, and then select the top-most check box in each pane.
> - In the **Filter** text box, you can't enter more than 4,000 characters.
> - The wildcard character `*` is allowed, except when the `AD Attribute` is a DN attribute. For more information about AD filters, click here.
> - You can use the **Filter** text box to synchronize users from AD to Oracle Identity Cloud Service based on their group memberships rather than their OUs. To do this, *don't* deselect the check boxes for the OUs. Instead, in the **Filter** text box, provide the custom group membership filters.
> - If there's a mismatch between the number of users or groups you're expecting to be transferred into Oracle Identity Cloud Service and how many users or groups are actually imported, then use Active Directory Users and Computers to test the custom filter in AD to verify that the users and groups brought into Oracle Identity Cloud Service are correct.
> - The names of the users that you want to import into Oracle Identity Cloud Service must contain at least three characters. The names of the groups that you want to import into Oracle Identity Cloud Service must contain at least five characters.
> - The telephone numbers of the users that you want to import must meet the requirements of the RFC 3966 specification.

b. In the **Supported Operations** area, choose which operations for Oracle Identity Cloud Service users or groups will be propagated to AD:

- If you activate or deactivate Oracle Identity Cloud Service users, and you want these user activation status changes to be reflected in AD, then select the **Activate/Deactivate Users** check box. Otherwise, leave this check box deselected.
- If you edit attribute values for Oracle Identity Cloud Service users, and you want these modifications to be passed to AD, then select the **Update Users Attributes** check box. Otherwise, leave this check box deselected.

- If you change the groups to which Oracle Identity Cloud Service users belong, and you want these group membership changes to be propagated to AD, then select the **Update Groups** check box. Otherwise, leave this check box deselected.

c. In the **Set import frequency** area, schedule how often, in hours and minutes, you want Oracle Identity Cloud Service to use the AD Bridge to import users and groups from AD.

> **Note:**
>
> During an incremental synchronization cycle, if there are more than 100,000 group membership changes in Microsoft Active Directory, then the synchronization cycle might take more than one hour. Microsoft Active Directory needs this time to process the change logs.

d. In the **Configure Attribute Mappings** area, click **Edit Attribute Mappings** to define custom attribute mappings between AD and Oracle Identity Cloud Service. See Define Attribute Mappings for a Microsoft Active Directory (AD) Bridge. Otherwise, go to step e.

e. In the **Authentication Settings** area, select **Enable local authentication** if you want users to use their Oracle Identity Cloud Service or their AD passwords to authenticate into Oracle Identity Cloud Service to access Oracle Identity Cloud Service-protected resources.

If you select this option, then configure delegated authentication for this AD Bridge. By activating delegated authentication, users transferred into Oracle Identity Cloud Service through the bridge will use their AD passwords to sign in to Oracle Identity Cloud Service. By deactivating delegated authentication, users must use their Oracle Identity Cloud Service passwords to authenticate into Oracle Identity Cloud Service. See Configure Delegated Authentication in Oracle Identity Cloud Service for more information about configuring delegated authentication for an AD Bridge.

Also, if you select **Enable local authentication**, then keep **Don't send Welcome Notifications** deselected to have Oracle Identity Cloud Service notify users by email that they must activate the Oracle Identity Cloud Service accounts that are created for them.

Otherwise, if you don't want users to be notified that Oracle Identity Cloud Service created accounts for them, then select the **Don't send Welcome Notifications** check box.

If you want users to use their federated accounts to authenticate into Oracle Identity Cloud Service, then select **Enable federated authentication**.

> **Note:**
>
> If you select this option, then configure SSO through the **Identity Providers** page. See Activate and Deactivate an Identity Provider.

> **Important:**
>
> By selecting **Enable federated authentication**, any user accounts that are transferred into Oracle Identity Cloud Service through the AD Bridge are classified as federated accounts. For referential integrity purposes, you can't deactivate, remove, or change the status of these user accounts to nonfederated.

f. Click **Save**.

4. In the **Confirmation** window, click **OK**.

   The status of the AD Bridge changes from **Partially Configured** to **Configured**. The bridge is created and configured.

> **Important:**
>
> Before you use the AD Bridge to import any AD user accounts into Oracle Identity Cloud Service, enable the **Password Never Expires** option for the accounts in AD. Otherwise, the passwords for the accounts will expire. If this occurs, then you can change the passwords. See Microsoft Active Directory (AD) Bridge Limitations in Known Issues for Oracle Identity Cloud Service.

> **Note:**
>
> If you use the AD Bridge to import a group into Oracle Identity Cloud Service, and then delete the group in Oracle Identity Cloud Service, you can re-establish a link between the group in AD and the group in Oracle Identity Cloud Service. To do so:
>
> a. In the **Select organizational units (OUs) for groups** pane, clear the check box for the designated group, and click **Save**.
>
> b. Select the check box for the group, and click **Save** again.
>
> c. Run the AD Bridge to synchronize the group between Oracle Identity Cloud Service and AD immediately. See Run a Microsoft Active Directory (AD) Bridge.

Define Attribute Mappings for a Microsoft Active Directory (AD) Bridge

By default, when you create a Microsoft Active Directory (AD) Bridge, attribute mappings are defined between AD and Oracle Identity Cloud Service. Attribute mappings enable the AD Bridge to pass values associated with user accounts between AD and Oracle Identity Cloud Service.

You can map attributes in two different ways: inbound and outbound. Inbound mappings allow you to map attributes from AD to Oracle Identity Cloud Service. Outbound mappings allow you to map any changes in Oracle Identity Cloud Service attributes to AD attributes.

For example, when you run the AD Bridge, the bridge can use the `givenName - First Name` mapping to transfer the first name of the user account from the **First name** field on the **General** tab of the **Properties** window of AD to the **First Name** field on the **Details** tab of the **Users** page of Oracle Identity Cloud Service. Similarly, you can perform an outbound mapping so that when you make any change to the first name of the user account in Oracle Identity Cloud Service, this change is reflected in AD. See Run a Microsoft Active Directory (AD) Bridge.

In addition to the predefined attribute mappings, you can define custom attribute mappings between AD and Oracle Identity Cloud Service.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Directory Integrations**.
2. Click the AD Bridge for which you want to define custom attribute mappings.
3. Click **Configuration.**
4. In the **Configure Attribute Mappings** area, click **Edit Attribute Mappings**. In the **Edit Attribute Mappings** window, two tabs appear:
   - **Microsoft Active Directory to Identity cloud:** This tab contains inbound attribute mappings from AD to Oracle Identity Cloud Service.
   - **Identity cloud to Microsoft Active Directory:** This tab contains outbound attribute mappings from Oracle Identity Cloud Service to AD.
5. If you want to define inbound attribute mappings, then click the **Microsoft Active Directory to Identity cloud** tab. Otherwise, go to step 9.

   You'll see predefined inbound mappings from AD to Oracle Identity Cloud Service. These mappings include:

List of predefined attributes	Required	Description
sAMAccountName	Yes	The user's user name.
givenName	No	The user's first name.
sn	Yes	The user's last name.
middleName	No	The user's middle name.
displayName	No	The user's display name.
title	No	The user's job title.
preferredlanguage	No	The user's preferred language (for example, English).
localeID	No	The user's language and region (locale).
mail	Yes	The user's email address.
telephonenumber	No	The user's telephone number.
homePhone	No	The user's home telephone number.
mobile	No	The user's mobile telephone number.
postalAddress	No	The user's postal address.
streetAddress	No	The user's street address.
l	No	The user's work location.
st	No	The state of the user's work address.
postalCode	No	The zip code of the user's work address.
c	No	The country of the user's work address.
usercertificate	No	This multi-valued attribute contains the DER-encoded X509v3 certificates issued to the user.

List of predefined attributes	Required	Description
userAccountControl	Yes	Specifies flags that control behavior for the user, such as whether the user has an **Active** or **Inactive** status, or whether the user's account is locked.

6. Click **Add Row** because you want to define an inbound attribute mapping from AD to Oracle Identity Cloud Service.
7. In the **Directory User Attributes** column, select the name of the AD attribute that contains a value which you want to transfer into Oracle Identity Cloud Service. If the attribute id is not available in the drop-down list, you can enter the new attribute name. After you save the changes, this new attribute will appear in the drop-down list.
8. In the **Oracle Identity Cloud Service User Attributes** column, enter or select the name of the Oracle Identity Cloud Service attribute that will contain the value transferred from AD.
9. If you want to define outbound attribute mappings, then click the **Identity cloud to Microsoft Active Directory** tab. Otherwise, go to step 13.

   You'll see predefined outbound mappings from Oracle Identity Cloud Service to AD. These mappings include:

List of predefined attributes	Required	Description
User Name	No	The user's user name.
Display Name	No	The user's display name.
Work Email	No	The user's work-related email address.
First name	No	The user's first name.
Last name	No	The user's last name.
Middle name	No	The user's middle name.
Title	No	The user's job title.
Locale	No	The user's language and region (locale
Preferred Language	No	The user's preferred language (for example, English).
Work Phone number	No	The user's work-related telephone number.
Mobile Phone number	No	The user's mobile telephone number.
Work Address Formatted	No	The user's work-related postal address.
Work Street Address	No	The user's street address.
Work Locality	No	The user's work location.
Work Address Region	No	The state or region of the user's work address.
Work Address Zip Code	No	The zip code of the user's work address.
Work Address Country	No	The country of the user's work address.
Home Phone number	No	The user's home telephone number.

10. Click **Add Row** because you want to define an outbound attribute mapping from Oracle Identity Cloud Service to AD.
11. In the **Oracle Identity Cloud Service User Attributes** column, enter or select the name of the Oracle Identity Cloud Service attribute that contains a value which you want to transfer into AD.
12. In the **Directory User Attributes** column, enter or select the name of the AD attribute that will contain the value transferred from Oracle Identity Cloud Service.
13. Click **Save**.

Understand Full and Incremental Sync

You can synchronize users and groups from selected organizational units (OUs) in Microsoft Active Directory (AD) into Oracle Identity Cloud Service. You can perform either an incremental sync or a full sync. Learn about syncing new OUs and read some example use cases.

Syncing New Organizational Units

Before 20.1.3, OU sync was triggered by the Bridge every minute so that newly added OUs in Active Directory were automatically available in Oracle Identity Cloud Service. Starting with the 20.1.3 release, when you add a new organizational unit (OU) in Active Directory, you must perform an incremental or full sync to see the newly created OU in Oracle Identity Cloud Service. Oracle recommends that you to run an incremental sync when adding new OUs.

Example Use Cases:

- Use Case: Unlink Users from Microsoft Active Directory (AD)
- Use Case: Delete Users and Groups from Microsoft Active Directory (AD)
- Use Case: Reattach an Unlinked User in Oracle Identity Cloud Service

Use Case: Unlink Users from Microsoft Active Directory (AD)

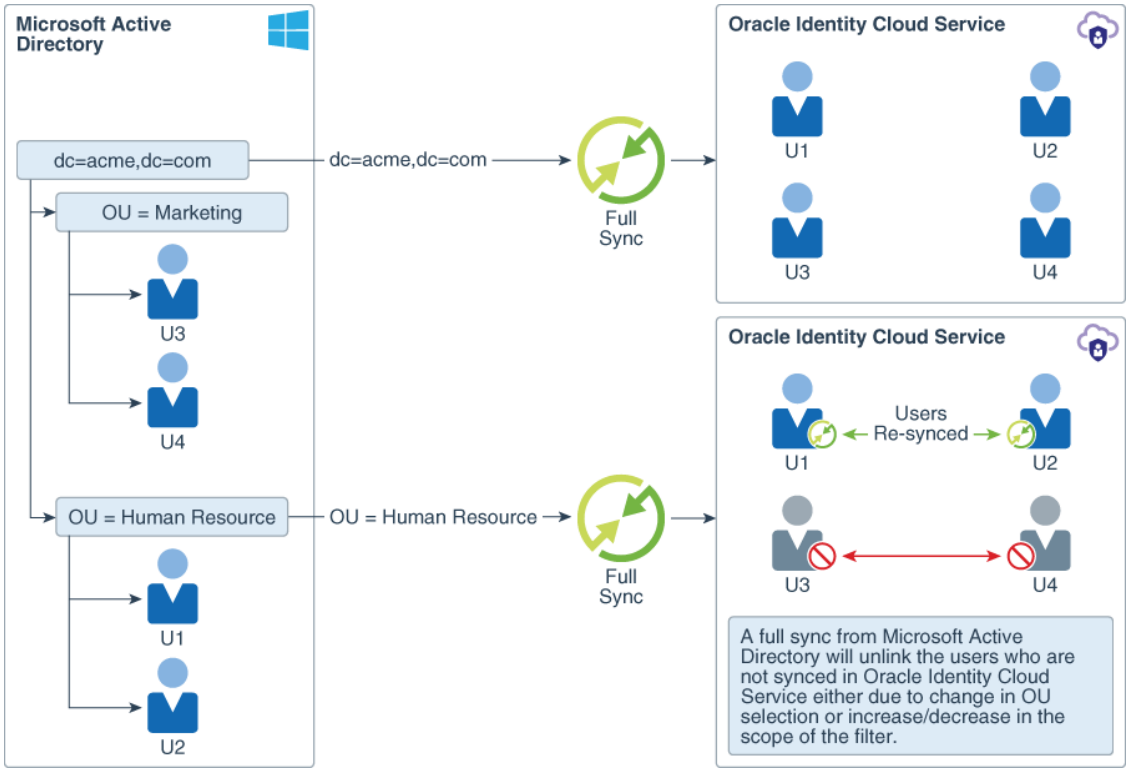
When you perform a full sync on users from organizational units (OUs), all users in the selected OUs are synchronized in Oracle Identity Cloud Service. The next time you apply a filter to synchronize a specific OU, you perform an incremental sync and the users in that OU are resynchronized in Oracle Identity Cloud Service.

The synchronized users who were not part of the filter will be unlinked from Microsoft Active Directory (AD). The unlinked users can no longer authenticate using delegated authentication because their link to AD is removed and their authentication falls back to Oracle Identity Cloud Service. Any new updates to these users won't be synced to Oracle Identity Cloud Service. You can use Oracle Identity Cloud Service to reset the passwords for these users. When you request a password change for the users, Oracle Identity Cloud Service sends a **Password Reset** notification to them so that they can provide their new passwords. See Reset Passwords for User Accounts.

If you remove the filter and synchronize these users again using full sync, then all of the users who were unlinked earlier will now be linked, and their authentication will fall back to AD.

Consider Human Resource and Marketing OUs with five users each. You are using full sync to sync them from AD to Oracle Identity Cloud Service. All of the users are synced in Oracle Identity Cloud Service.

If you want the Marketing users alone in Oracle Identity Cloud Service, then you can perform an incremental sync along with a filter to resync the Marketing users into Oracle Identity Cloud Service. All of the users who are part of the Human Resource OU are unlinked because they're not part of the filter that's used to resync users. The number of unlinked users appears in the UI.



Use Case: Delete Users and Groups from Microsoft Active Directory (AD)

Microsoft Active Directory (AD) is an authoritative source. Users that are deleted from AD are unlinked and deactivated in Oracle Identity Cloud Service. You can then remove these users from Oracle Identity Cloud Service.

When groups are deleted from AD, upon a full or incremental sync, these groups are also removed from Oracle Identity Cloud Service.

Use Case: Reattach an Unlinked User in Oracle Identity Cloud Service

Consider you want to create previously unlinked users in Microsoft Active Directory (AD) with the same usernames. When you next perform a full or an incremental sync, these users in AD are reattached to the associated users in Oracle Identity Cloud Service.

The reattached user's authentication will be delegated to AD if delegated authentication is activated in Oracle Identity Cloud Service. For example, a user is synced from multiple AD domains into Oracle Identity Cloud Service. All of these domains are authoritative because AD is an authoritative source. If you delete a user from one of the domains, then the user is unlinked in Oracle Identity Cloud Service. If you resync the user to a different AD domain, then this domain now becomes authoritative for the user.

Change Administrator Account Credentials for AD Bridge

Use the AD Bridge client to update administrator account details.

If the Active Directory credentials of the administrator changes, or if you want to use a diffferent administrator for AD Bridge, use the AD Bridge client to update them.

1. Open `ADBridgeUI.exe`. It's in the AD Bridge installation folder. The default path is `C:\Program Files\Oracle\IDBridge`.

2. Click **Update AD Credentials** and enter the changed user name and password, or the user name and password of another administrator.
3. Check the credentials by clicking **Test**.

   When you click **Test**, the AD Bridge checks the following:

   - Whether sync is running or not. If it is running, wait for it to finish then try again. You can see the sync status in the Oracle Identity Cloud Service Admin console. See Check a Bridge Data Synchronization Status.
   - That the credentials are valid. If they are not, check and re-enter them, then test again.
4. Click **Update**.

Locate a New Domain Controller

The domain controller is used to sync data between Active Directory (AD) and Oracle Identity Cloud Service. If the domain controller you have configured changes or you're having domain controller connectivity issues (for example, an LDAP Server Unavailable error), use the AD Bridge client to locate another domain controller to use.

1. Open `ADBridgeUI.exe`. It's in the AD Bridge installation folder. The default path is `C:\Program Files\Oracle\IDBridge`.
2. Click **Detect Domain Controller** and AD Bridge checks connectivity with the currently configured domain controller that's listed in the **AD Domain Controller** field.

   If you receive a *Sync already running* error, this error occurs when a sync is currently running by AD Bridge and you try to detect a new domain controller. Wait for current sync to finish and retry once it's completed.

   If you receive a *Connection was forcibly closed by remote host* error, this error occurs when the AD Bridge is not able to discover a new domain controller and the configured domain controller is also not working. Check that the machine on which the AD Bridge is installed is connected to the domain by using the **Test Connectivity** button.
3. If there's connectivity to this domain controller and it's working properly, a success message appears. Click **OK**.
4. If the configured domain controlled is no longer a domain controller or there are any connectivity issues with it, then AD Bridge searches for another available domain controller and displays a message listing the new domain controller found by AD Bridge.
   a. Click **Yes** and AD Bridge switches to the new domain controller and initiates a full sync.
   b. Click **No** and AD Bridge continues to use the currently configured domain controller.

Quit an Unresponsive Microsoft Active Directory (AD) Bridge Sync

You can quit an AD Bridge sync that is unresponsive (or stuck).

If syncing is taking longer than expected (for example, several hours), you might want to quit the AD Bridge sync. Always ensure that you are not quitting a Running sync. Once you have quit your current AD Bridge sync, the sync terminates in a Failure state, and you can start another.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Directory Integrations**.

2. To quit the AD Bridge sync, click **Abort**.
3. In the **Confirmation** window, click **OK**.

Run a Microsoft Active Directory (AD) Bridge

You can run a Microsoft Active Directory (AD) Bridge to synchronize Oracle Identity Cloud Service with AD immediately.

As part of configuring an AD Bridge, you specified how often, in hours and minutes, you want Oracle Identity Cloud Service to use the bridge to import users and groups from AD. You're synchronizing Oracle Identity Cloud Service with your AD enterprise directory structure.

When the interval you specified elapses, Oracle Identity Cloud Service synchronizes with the directory structure so that any new, updated, or deleted user or group records are transferred into Oracle Identity Cloud Service. Because of this, the state of each record is synchronized between AD and Oracle Identity Cloud Service.

For security purposes, you may want to import users and groups from AD immediately. There are two types of imports that you can run:

- Full import: The AD Bridge polls AD and retrieves data associated with all user and groups that you selected in the **Select organizational units (OUs) for users** and **Select organizational units (OUs) for groups** panes of the **Configuration** tab for the bridge. This data represents users and groups that were created, modified, or removed in AD. As a best practice, Oracle recommends that you perform a full import the first time you run the AD Bridge. See Configure a Microsoft Active Directory (AD) Bridge for more information about the **Configuration** tab.
- Incremental import: Similar to a full import, but for this type of import, the AD Bridge polls AD and retrieves only user and group data that changed since you last used the AD Bridge to import users and groups into Oracle Identity Cloud Service.

By running the AD Bridge, you can propagate changes for Oracle Identity Cloud Service users in AD. After users are imported into Oracle Identity Cloud Service through the bridge, if you activate or deactivate a user, modify the user's attribute values, or change the group memberships for the user in Oracle Identity Cloud Service, then these changes will be reflected in AD.

You can also use the AD Bridge to view a synchronization log of the communication between Oracle Identity Cloud Service and AD.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Directory Integrations**.
2. Click the AD Bridge that you want to use to import users and groups from AD.
3. Click **Configuration**.
4. In the **Configuration** tab:
    a. In the **Select organizational units (OUs) for users** and **Select organizational units (OUs) for groups** panes, select the check box for each OU that contains users or groups that you want to import.
    b. In the **Supported Operations** area, select check boxes to enable Oracle Identity Cloud Service to propagate a user's activation status, attribute values, or group memberships to AD. See Configure a Microsoft Active Directory (AD) Bridge for more information about the **Supported Operations** area.
    c. Click **Save**.

The AD Bridge propagates any changes to an Oracle Identity Cloud Service user's activation status, attribute values, or group memberships to AD.

5. In the **Confirmation** window, click **OK**.
6. Click **Import**.
7. In the **Import Type** window, choose whether you want to run an incremental import or a full import, and then click **OK**.

   Oracle Identity Cloud Service imports the users and groups from AD.

> **Note:**
>
> Based on how many users and groups you're importing, the job may take several minutes or even hours.

8. Click the **Import** tab. The status of the job Oracle Identity Cloud Service uses to import users and groups from AD is **Running**. After all users and groups are imported, the status changes to **Success**.

   Also, on this tab, you'll see a synchronization log of all traffic that occurs between Oracle Identity Cloud Service and AD for the current import job that ran. This includes the start date and time, and completion date and time, for the import job, how many users and groups were imported from AD successfully, and how many users and groups couldn't be imported.

> **Note:**
>
> If you don't see the status change after a few minutes, then click **Refresh**. Also, if the status of the job is **Failed**, then an error occurred while the AD Bridge was transferring users and groups from AD to Oracle Identity Cloud Service. See Troubleshoot Oracle Identity Cloud Service.

View Details About a Microsoft Active Directory (AD) Bridge

By default, you can see the domain name and status for each Microsoft Active Directory (AD) Bridge.

You might want to see other information about the AD Bridge, such as its configuration information, attribute mappings, and a synchronization log of the communication between Oracle Identity Cloud Service and AD.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Directory Integrations**.
2. Click the AD Bridge about which you want to see more information.
   - To view configuration information about the AD Bridge, click the **Configuration** tab. See Configure a Microsoft Active Directory (AD) Bridge.
   - To view attribute mappings for the AD Bridge, open the **Edit Attribute Mappings** window. See Define Attribute Mappings for a Microsoft Active Directory (AD) Bridge.
   - To view a synchronization log of all traffic between Oracle Identity Cloud Service and AD the last time the AD Bridge ran, click the **Import** tab. See Run a Microsoft Active Directory (AD) Bridge.

Activate and Deactivate Microsoft Active Directory (AD) Bridges

You can use Oracle Identity Cloud Service to activate and deactivate Microsoft Active Directory (AD) Bridges:

- Deactivate: Disable the link between your AD enterprise directory structure and Oracle Identity Cloud Service.
- Activate: Enable the link between Oracle Identity Cloud Service and AD.

Activate a Microsoft Active Directory (AD) Bridge

You can use Oracle Identity Cloud Service to activate a single Microsoft Active Directory (AD) Bridge.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Directory Integrations**.
2. Click the **Action** menu to the right of the domain that contains the AD Bridge that you want to activate.
3. Select **Activate**.
4. In the **Confirmation** window, click **OK**.

   By activating the domain, you're activating the AD Bridge associated with the domain. The status of the bridge changes from **Inactive** to **Active**.

Deactivate a Microsoft Active Directory (AD) Bridge

You can use Oracle Identity Cloud Service to deactivate a single Microsoft Active Directory (AD) Bridge.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Directory Integrations**.
2. Click the **Action** menu to the right of the domain that contains the AD Bridge that you want to deactivate.
3. Select **Deactivate**.
4. In the **Confirmation** window, click **OK**.

   By deactivating the domain, you're deactivating the AD Bridge associated with the domain. The status of the bridge changes from **Active** to **Inactive**.

Activate All Microsoft Active Directory (AD) Bridges

For efficiency purposes, you can use Oracle Identity Cloud Service to activate all Microsoft Active Directory (AD) Bridges simultaneously.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Directory Integrations**.
2. Click **Activate All**.
3. In the **Confirmation** window, click **OK**.

By activating all domains, you're activating the AD Bridge associated with each domain. The status of each bridge changes from **Inactive** to **Active**.

Deactivate All Microsoft Active Directory (AD) Bridges

For security purposes, you can use Oracle Identity Cloud Service to deactivate all Microsoft Active Directory (AD) Bridges simultaneously.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Directory Integrations**.
2. Click **Deactivate All**.
3. In the **Confirmation** window, click **OK**.

   By deactivating all domains, you're deactivating the AD Bridge associated with each domain. The status of each bridge changes from **Active** to **Inactive**.

Modify a Microsoft Active Directory (AD) Bridge

You can change the following items for a Microsoft Active Directory (AD) Bridge:

- The AD users and groups that you want Oracle Identity Cloud Service to import using the AD Bridge.
- Whether, after a user or group is synchronized from AD to Oracle Identity Cloud Service, if you activate or deactivate a user, modify the user's attribute values, or change the group memberships for the user in Oracle Identity Cloud Service, these changes will be propagated to AD.
- How often you want Oracle Identity Cloud Service to use the AD Bridge to import users and groups from AD.
- The predefined and custom attribute mappings defined between AD and Oracle Identity Cloud Service.
- Whether users can use their AD or their Oracle Identity Cloud Service passwords, or their federated accounts, to sign in to Oracle Identity Cloud Service to access resources protected by Oracle Identity Cloud Service, such as the My Profile console, Identity Cloud Service console, and apps assigned to the users.

> **Note:**
>
> You can upgrade the client for the AD Bridge. By doing this, you can install the latest client without removing the existing client that's installed.
>
> To upgrade the client, download it and follow the instructions in Create a Microsoft Active Directory (AD) Bridge. When you see the **Specify Identity Cloud Service Credentials** or the **Specify Microsoft Active Directory Credentials** dialog boxes, the client will use the credentials you provided in the previous installation. For this reason, the values are greyed out so they can't be edited.

Modify a Microsoft Active Directory (AD) Bridge

You can use the **Directory Integrations** page to modify a Microsoft Active Directory (AD) Bridge.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Directory Integrations**.
2. Click the AD Bridge that you want to modify.
3. To edit configuration information about the AD Bridge, go to step 4. Otherwise, go to step 5.
4. Click **Configuration**.
   a. In the **Select organizational units (OUs) for users** and **Select organizational units (OUs) for groups** panes, select or deselect check boxes to enable or prevent Oracle Identity Cloud Service from importing users and groups using the AD Bridge.

      See Configure a Microsoft Active Directory (AD) Bridge for more information about the **Select organizational units (OUs) for users** and **Select organizational units (OUs) for groups** panes.

   b. In the **Supported Operations** area, select or deselect check boxes to enable or prevent Oracle Identity Cloud Service from propagating changes for a user's activation status, attribute values, or group memberships to AD.

      See Configure a Microsoft Active Directory (AD) Bridge for more information about the **Supported Operations** area.

   c. In the **Set import frequency** area, change how often you want Oracle Identity Cloud Service to use the AD Bridge to import users and groups from AD.

   d. In the **Configure Attribute Mappings** area, click **Edit Attribute Mappings**. The **Edit Attribute Mappings** window opens and two tabs appear:

      - **Microsoft Active Directory to Identity cloud:** In this tab, you can modify inbound attribute mappings from AD to Oracle Identity Cloud Service.
      - **Identity cloud to Microsoft Active Directory:** Use this tab to modify outbound attribute mappings from Oracle Identity Cloud Service to AD.

      i. Click the **Microsoft Active Directory to Identity cloud** or **Identity cloud to Microsoft Active Directory** tab.

      ii. In the **Directory User Attributes** and **Oracle Identity Cloud Service User Attributes** columns, change the AD or Oracle Identity Cloud Service attribute used for the predefined or custom attribute mapping.

      iii. To remove an attribute mapping, click the **X** button to the right of the mapping.

      > **Note:**
      >
      > Inbound attribute mappings with asterisks in the **Microsoft Active Directory to Identity cloud** tab are required by the AD Bridge to pass values associated with AD user accounts into Oracle Identity Cloud Service so that the accounts can be created in Oracle Identity Cloud Service. You can't delete these mappings.

      iv. Click **Save** to close the **Edit Attribute Mappings** window.

      See Define Attribute Mappings for a Microsoft Active Directory (AD) Bridge for more information about the **Directory User Attributes** and **Oracle Identity Cloud Service User Attributes** columns of the **Microsoft Active Directory to Identity cloud** and **Identity cloud to Microsoft Active Directory** tabs of the **Edit Attribute Mappings** window.

e. In the **Authentication Settings** area, select the **Enable local authentication** option if you want users to use their Oracle Identity Cloud Service or their AD passwords to sign in to Oracle Identity Cloud Service to access Oracle Identity Cloud Service-protected resources.

   If you select this option, then configure delegated authentication for the AD Bridge. See Configure a Microsoft Active Directory (AD) Bridge.

   If you select **Enable local authentication**, then select or deselect **Don't send Welcome Notifications** to enable or prevent Oracle Identity Cloud Service from notifying users by email that they must activate the Oracle Identity Cloud Service accounts that are created for them.

   Otherwise, select **Enable federated authentication** to have users use their federated accounts to sign in to Oracle Identity Cloud Service.

f. Click **Save**.

g. In the **Confirmation** window, click **OK**.

See Configure a Microsoft Active Directory (AD) Bridge for more information about the areas of the **Configuration** tab.

Remove a Microsoft Active Directory (AD) Bridge

You can use Oracle Identity Cloud Service to remove a Microsoft Active Directory (AD) Bridge.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Directory Integrations**.
2. Click the **Action** menu  to the right of the domain that contains the AD Bridge that you want to remove.
3. Select **Remove**.
4. In the **Confirmation** window, click **OK**.

   By removing the domain, you're removing the AD Bridge associated with the domain. To ensure that your bridge is deleted cleanly and completely, you must delete the client associated with the bridge.
5. Double-click the `ad-id-bridge.exe` file.

   The **Identity Cloud Service Microsoft Active Directory Bridge Installer** appears.
6. In the **Welcome** dialog box, click **Next**.
7. In the **Removal Completed** dialog box, click **Close**.

Important:

If you can't remove the client for the AD Bridge or the bridge still appears in the **Directory Integrations** page, then complete the following steps:

1. Run the following CURL command to obtain the Client ID that you used to install the client for the AD Bridge:

   ```
   curl -X GET \
   <Identity_Cloud_Service_URL>/admin/v1/IdentityAgents \
   -H 'Authorization: Bearer <access_token>
   ```

   `<Identity_Cloud_Service_URL>` is a placeholder for the Identity Cloud Service URL that you used to install the client for the bridge, and `<access_token>` is a placeholder for the access token that contains the authorization credentials that are required to obtain the Client ID.

   See the Oracle Identity Cloud Service: First REST API Call tutorial to learn how to get this access token.

   A list of AD Bridge clients that are installed for your identity domain appears.
2. From this list, find the Client ID of the AD Bridge that you want to remove.
3. Run the following CURL command to remove the client for the AD Bridge:

   ```
   curl -X DELETE \
   <Identity_Cloud_Service_URL>/admin/v1/IdentityAgents/<Client_ID> \
   -H 'Authorization: Bearer <access_token>
   ```

   `<Client_ID>` represents the ID of the client for the AD Bridge that you want to remove.

   A `204 (No Content)` response appears, signifying that you removed the client for the bridge.

Transfer the Microsoft Active Directory (AD) Bridge

Maintaining the Microsoft Active Directory (AD) Bridge includes transferring the bridge to another machine and restarting the bridge.

Topics:

Transfer the Microsoft Active Directory (AD) Bridge

After you have setup a Microsoft Active Directory (AD) Bridge, you can transfer that bridge to another machine.

> **Note:**
>
> If you can't remove the client for the AD Bridge or the bridge still appears in the **Directory Integrations** page, then follow the procedure in Remove a Microsoft Active Directory (AD) Bridge to remove the bridge.

1. From the original machine, access the **Control Panel**, and uninstall the client for the AD Bridge.
2. On the other machine, install the client. See Create a Microsoft Active Directory (AD) Bridge.
3. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Directory Integrations**.
4. Verify that the AD Bridge appears in the other machine with an **Active** status. This bridge can now be used to synchronize with your AD enterprise directory structure.

Restart the Microsoft Active Directory (AD) Bridge

If the Microsoft Active Directory (AD) Bridge stops unexpectedly, then you can restart it.

1. Click **Start**.
2. In the text box, enter **Services**, and then press **Enter**.

   The **Services** window appears. This window contains a utility that's used to manage daemon processes within the Windows OS. These processes include the back-end service that's used to establish communication between Oracle Identity Cloud Service and AD.
3. Click **Services (Local)**, click the **Standard** tab, scroll down the list of services, right-click **Identity Cloud Service Microsoft Active Directory Bridge Service**, and then click **Start**.
4. Verify that **Running** appears as the status for the service.

Log Files

This section contains information about AD Bridge client log files for the AD Bridge:

- Create and Manage Log Files for the Microsoft Active Directory (AD) Bridge
- Allow My Oracle Support to Access Client Log Files

Create and Manage Log Files for the Microsoft Active Directory (AD) Bridge

After you install and configure the Microsoft Active Directory (AD) Bridge, you may want to access the log files for troubleshooting purposes. You can locate these files in the `%ProgramData%\Oracle\IDBridge\logs` directory.

To modify the log level of the log files for the AD Bridge:

1. Navigate to the `%ProgramFiles%\Oracle\IDBridge` directory.
2. Using a text editor, open the `log4net.config` file.
3. In the file, locate the following line of code: `<level value="info" />`
4. Change the value of the `level value` parameter to one of the following log levels:

Log Level	Description
`all`	Capture all events.
`debug`	Capture fine-grained informational events that are most useful to debug the AD Bridge.
`error`	Capture error events that might still allow the AD Bridge to continue running.
`fatal`	Capture severe error events that will result in the AD Bridge no longer running.
`info`	Capture informational events that highlight the progress of the AD Bridge at a coarse-grained level.
`off`	Turn off logging.
`trace`	Capture finer-grained informational events than the `debug` log level.
`warn`	Capture potentially harmful situations to the AD Bridge.

5. Save and close the `log4net.config` file.

> **Note:**
>
> You must restart the AD Bridge for the change you made to the log level to take effect.

Allow My Oracle Support to Access Client Log Files

Learn how to grant consent for Oracle Support to access client log files, the scope of consent, what it covers, and how long it lasts.

The default behavior is that My Oracle Support cannot access the client log files, which are on a machine at your premises. You have to add them to the support request. You can give your consent so that My Oracle Support can fetch the logs directly when they need to be analyzed to resolve an issue. This can reduce the time it takes for the support request to be resolved.

- **How long does consent last?**
  After you have given your consent, it remains effective until you remove your consent, or remove the AD Bridge domain.
- **Do I need to give separate consent for every AD Bridge?**
  No. Your consent applies at AD domain level. If you have more than one bridge under the same AD domain, the consent applies to all of them.
- **Do I need to provide consent for each AD domain?**
  If you have more than one AD domain, a separate consent is needed for each one.
- **Can Oracle fetch any file from the windows machine where the Microsoft AD Bridge client is installed?**
  No. Only Microsoft AD Bridge log files are fetched.

- **When is the log file fetched from the client machine?**
  Oracle only fetches logs files if they are needed so they can be analyzed as part of resolving a service request that you have raised. If you raise a service request and there is no need for the AD Bridge client log file to be examined, then it is not fetched.
- **Where are the log files stored?**
  They are uploaded to tenant Oracle cloud storage.
- **Do the log files stay in cloud storage indefinitely?**
  No. They will be removed from cloud storage after 24 hours, after Oracle has analyzed the logs. An automated purge job deletes all log files that are older than 24 hours.

Consent to Sharing Client Logs

Enable consent for Oracle to fetch AD Bridge client log files.

1. From the Windows Start menu, open the AD Bridge client from **Identity Cloud Service Microsoft Active Directory Bridge**.
2. Under **Share Client Logs**, check **Upload client logs** and click **Update**.

The Success window confirms that you have given consent for the AD Bridge logs to be uploaded to the Identity Cloud Server so that Oracle can view them to resolve a service request.

Remove Consent to Sharing Client Logs

Remove consent for Oracle to fetch AD Bridge client log files.

1. From the Windows Start menu, open the AD Bridge client from **Identity Cloud Service Microsoft Active Directory Bridge**.
2. Under **Share Client Logs**, uncheck **Upload client logs** and click **Update**.

The Success window confirms that you have removed consent for the AD Bridge logs to be uploaded.

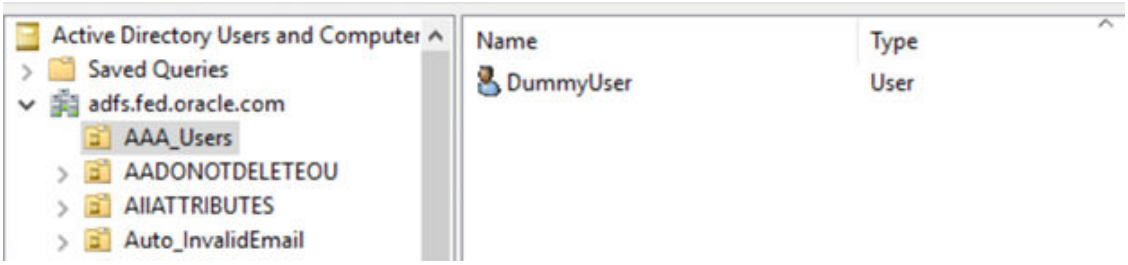
Troubleshooting and FAQs for Active Directory (AD) Bridge

Learn how to troubleshoot common Active Directory (AD) issues.

1. Why is my Active Directory (AD) Bridge client connecting to a different domain?

Answer: The domain to which the AD Bridge client is connected is determined from the domain of the signed-in user who is installing the AD Bridge client on the Windows Server. Check whether your user is present in the correct domain through the **Active Directory Users and Computers** utility.

The following screenshot shows the **DummyUser** is present in the domain **adfs.fed.oracle.com**.



2. Why can't I connect to Active Directory on an SSL port?

Answer: Active Directory must be configured for an SSL Connection. Try connecting **ldp.exe** with Active Directory on SSL. To verify the SSL connection:

1. Ensure that the Windows Support Tools is installed on the Active Directory machine.
   a. Select Start | All Programs | Windows Support Tools | Command Prompt.
   b. Start the **ldp tool** by typing **ldp** at the command prompt.
2. From the **ldp** window, select Connection | Connect and supply the host name and port number (636). Also, select the **SSL** check box.
   a. If the connection is successful, a window displays listing the information related to the Active Directory SSL connection.
   b. If the connection is unsuccessful, restart your system, and repeat this procedure. If Active Directory still doesn't connect, complete the following instructions to enable SSL: Enable LDAP over SSL with a third-party certification authority.

3. I received a "Connectivity to AD Bridge restored" email notification. What does it mean?

Answer: Because of network connectivity issues, the AD Bridge server might become disconnected to Oracle Identity Cloud Service. After connectivity is restored, you will get this email notification. **Note:** Any connectivity issues delay synchronization. Any new data will be synced after connectivity is restored.

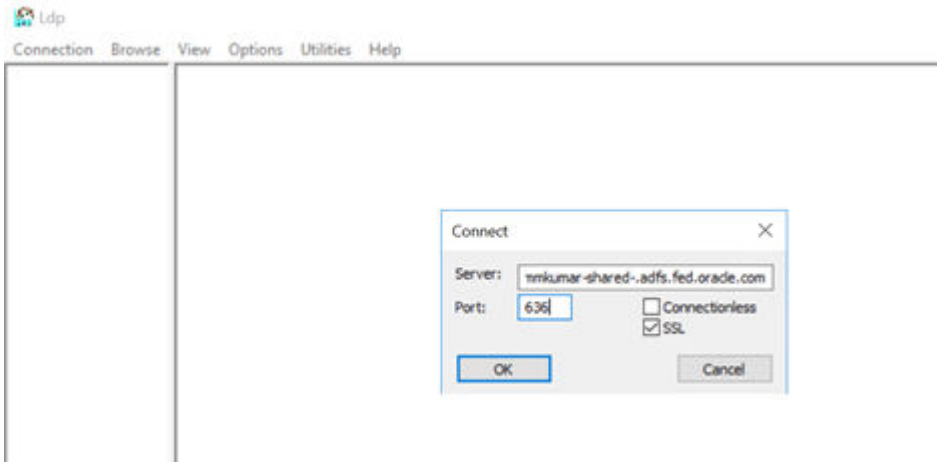
If you don't want to receive these email notifications, change the Notifications settings from the Oracle Identity Cloud Service Admin console. See About Administrator Notificationsto access Administrator notifications. You can choose to turn on the following Administrator AD Bridge connectivity notifications:

- Synchronization job summary
- Notify an administrator when connectivity between AD-ADbridge-Identity domain server is broken.
- Notify an administrator when connectivity between AD-ADbridge-Identity domain server is restored.
- Bridge update available
- Notify an administrator when sync between AD-ADbridge-Identity domain server has succeeded.
- Notify an administrator when sync between AD-ADbridge-Identity domain server has failed.

4. I see an "LDAP Server unavailable" error in the log file. What does it mean?

Answer: The "LDAP Server unavailable" error occurs when the server on which the AD Bridge client in installed is unable to connect to the Active Directory Domain Controller through LDAP. Verify that the Active Directory services are running (In Windows Services list, check the status for AD DS Domain Controller service.) and then try to connect using the client utility **ldp.exe**.

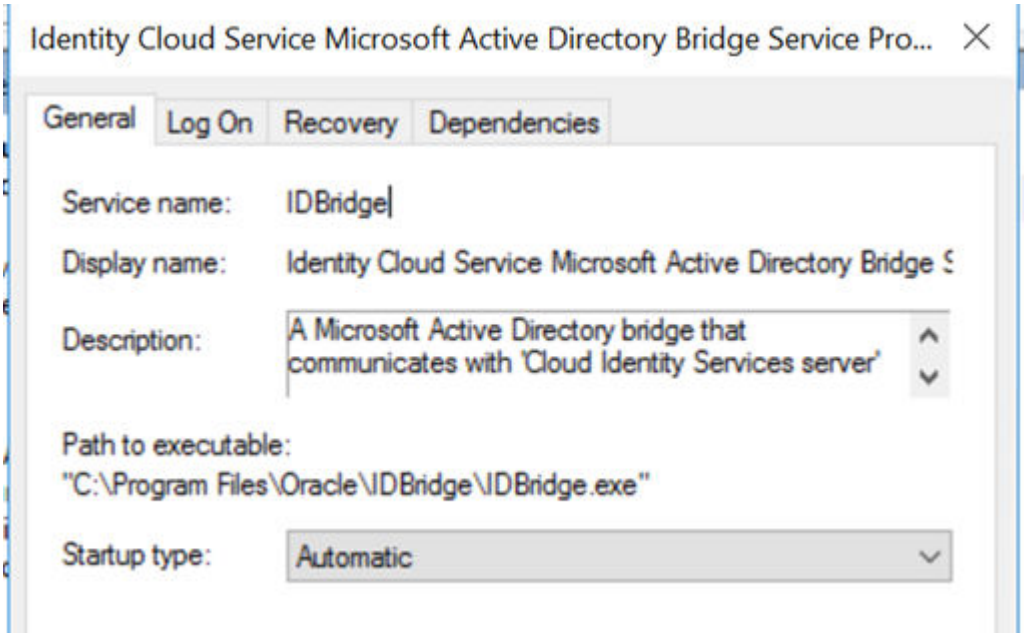
1. Open a run window from **Start**.
2. Enter **ldp** to open the client utility.
3. Select **Connection** and then **New Connection**. Complete the details and then check whether the connection is successful.



5. I see the message "ADBridge Unreachable" in the user interface. What does it mean?

Answer: AD Bridge has one-way communication with Oracle Identity Cloud Service. This means that Oracle Identity Cloud Service can't directly communicate with the server on which AD Bridge is installed. Instead, AD Bridge frequently polls Oracle Identity Cloud Service to check whether any operation (like sync) is pending. An "AD Bridge Unreachable" message means that the polling is not being performed. The following are some reasons that the AD Bridge might be unreachable.

- The AD Bridge is not installed.
- The AD Bridge is installed but unable to reach to Oracle Identity Cloud Service over the internet.
  - Check your connection/proxy settings.
  - Test the connectivity using the AD Bridge user interface.
- The background service is stopped.
  - Start *"Identity Cloud Service Microsoft Active Directory Bridge Service"* from Windows Services.
  - Ensure that the **Startup type** is *Automatic*.



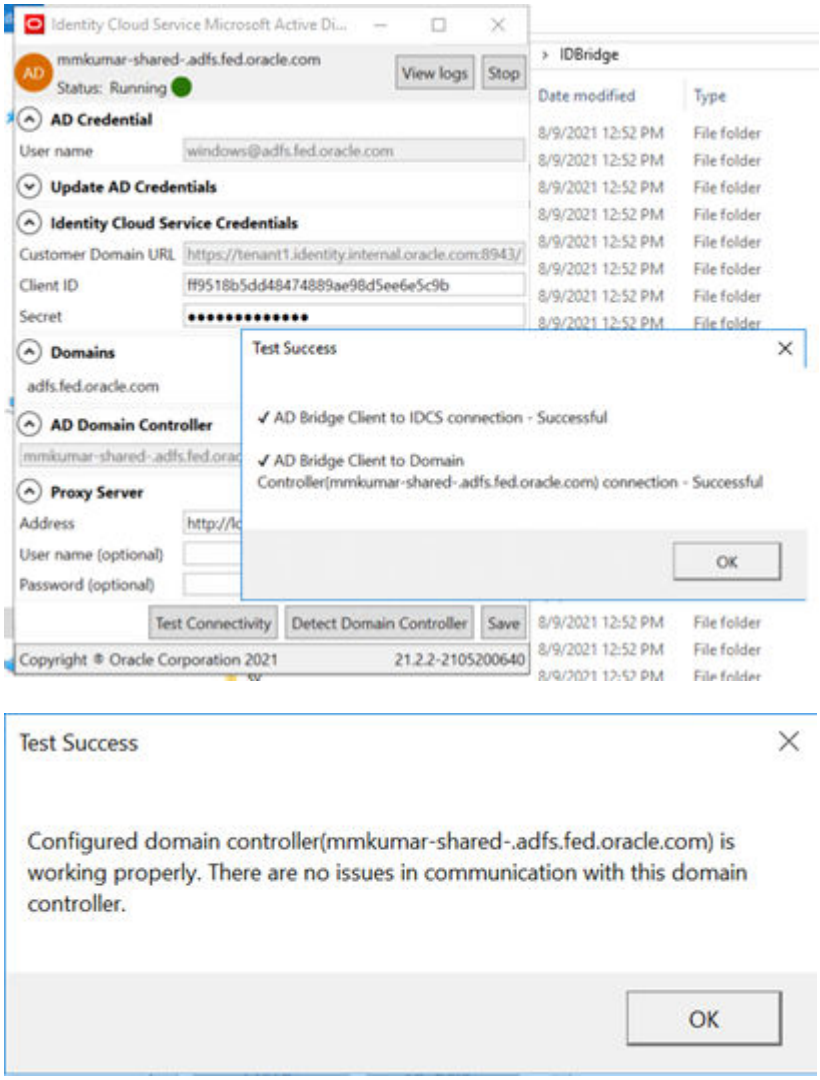
After you have determined the cause, restart the AD Bridge service, either from the AD Bridge user interface (Stop/Start buttons) or from Windows Services. **Important:** Before restarting the AD Bridge service, take a thread dump of the Oracle Identity Cloud Service process and share it with the Oracle Support Team. See 30. How to take thread dump of AD Bridge service on AD Bridge machine? You must resolve this issue for the AD Bridge to function properly. If you don't fix this issue, AD Bridge functionalities including Sync and Delegated Authentication will not work properly.

6. I see "No active sync" in the Admin console. What does it mean?

Answer: This message doesn't indicate an issue. This indicates that currently a sync is not in progress. The next sync will run according to the interval set for the domain through the configuration page. Or, it can be triggered manually. Since the incremental sync only reads changed data, a sync can happen very fast and it might appear that the “No active sync.” message never disappears. You can always verify the last sync status from the Import page for that particular domain.

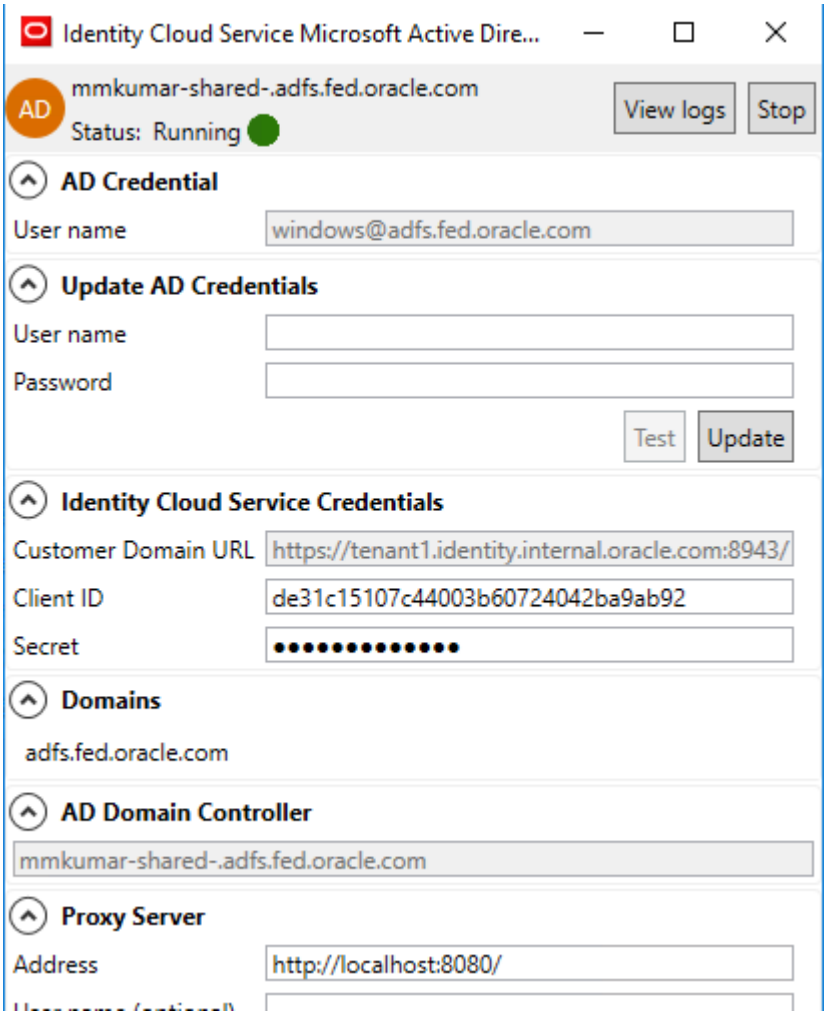
7. I have moved my Domain Controller from its current machine to another machine. What steps do I perform next?

Answer: Moving the Domain Controller should not cause any issues. Verify Domain Controller connectivity by using the **Test Connectivity** option in the AD Bridge user interface. If there's an issue in the AD Bridge to Domain Controller (LDAP) communication, then click **Detect Domain Controller** to further detect whether the Domain Controller is accessible. The following screen shots are examples of successful connection tests.



8. I have changed my User credentials to connect to Active Directory. How can I change the credentials in the Active Directory (AD) Bridge client?

Answer: After AD Bridge version 21.3.1, this feature is available in the user interface. Download and install the latest version of AD Bridge. **Note:** You don't need to uninstall the current binaries. The install upgrades them. See "Update AD credentials" in the following screenshot.

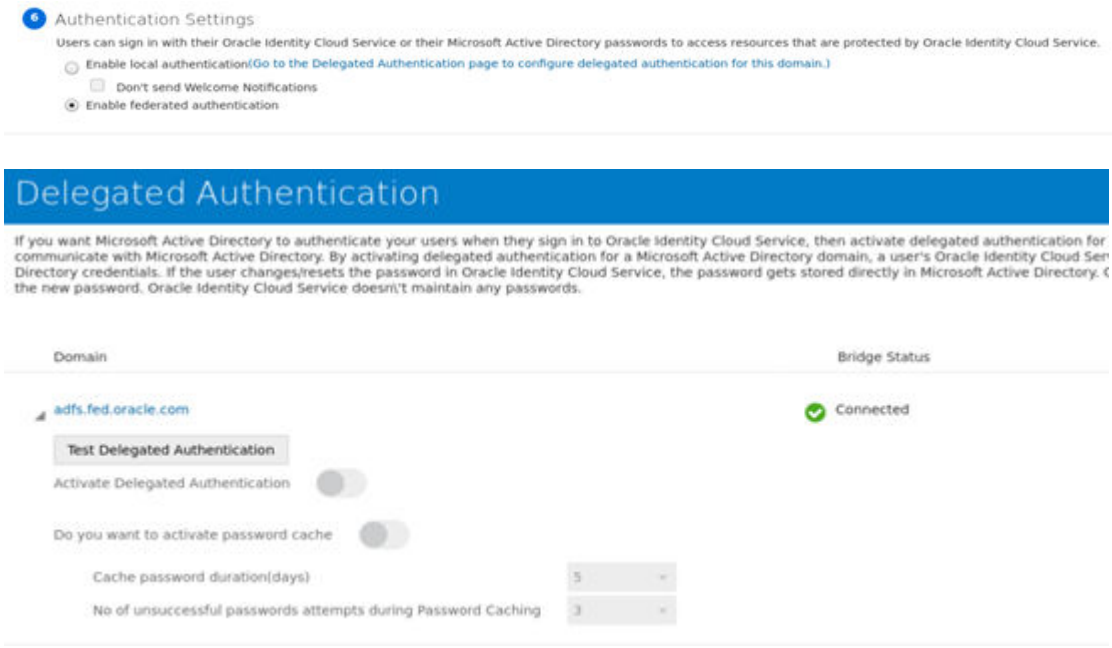


9. My Users are synced, but they are not able to sign in. What could be the problem?

Answer: This depends on which of the three authentication methods (listed below) are being used to sign in Active Directory (AD) users. These methods can be set using the domain configuration page. Sign in functionality works differently in each case.

- Local Authentication (default): After the sync, users will get a welcome notification to change the password for their account. They need to use the provided username (from AD) and password they set to sign in to their account. **Action to take:** Check whether the user is present in Oracle Identity Cloud Service. (The user sync might have failed Because of invalid data.) If the user exists, try resetting the password from Oracle Identity Cloud Service.
- Delegated Authentication: With local authentication, you can enable delegation from AD. In this method, users won't create a password but use their existing AD passwords to sign in. Oracle Identity Cloud Service delegates the user authentication to AD through AD Bridge. **Action to take:** Check whether the user is present in Oracle Identity Cloud Service. Also, check whether the user is active in AD and that the password is not expired.
- Federated Authentication: This method uses a third-party service like Microsoft AD FS to authenticate the user. **Action to take:** Check the configuration of the third-party service.

Use the following screen shots as a guide.



10. How long will Microsoft Windows Server 2012 be supported?

Answer: There is no pre-defined support period. Oracle provides six months' notice when compatibility is removed. Otherwise, presume that Oracle will support Windows Server 2012 as long as Microsoft supports it.

11. Why can't I enable Federation?

Answer: Check whether Delegated Authentication is enabled. If Delegated Authentication is enabled, Federated Authentication cannot be enabled. To switch from Delegated to Federated Authentication:

1. Deactivate Delegated Authentication. See Deactivate Delegated Authentication.
2. Turn on Federated Authentication in Directory Integrations.
3. Perform a Full Import.

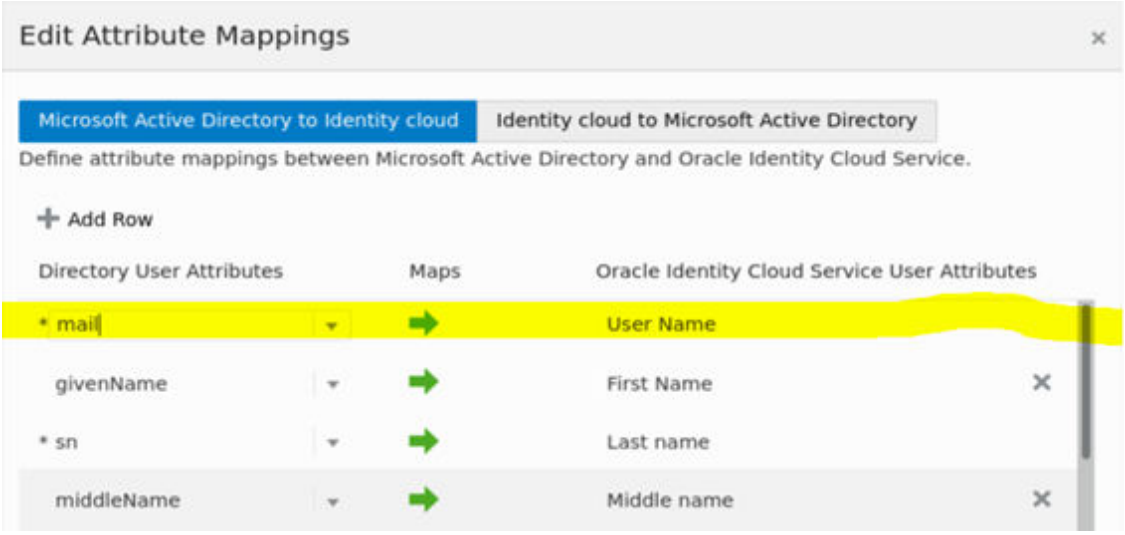
12. Why can't I enable Delegated Authentication?

Answer: Ensure that **Enable local authentication** is chosen on the Directory Integrations page. If you have Federated Authentication enabled, turn it off. Then go to the Delegated Authentication settings and activate it for a particular domain.

13. I want to change my sign-in username to an email-address or vice versa. How can I do it?

Answer: To allow sign in using email, you need to map the `mail` attribute of Active Directory (AD) to `User Name` in Oracle Identity Cloud Service inbound mapping as shown in screenshot below.

Note: You can either configure `sAMAccountName` or `mail` with `User Name` but not both at same time. If users are already synced, then you need to trigger a Full Import after changing this attribute mapping. A Full Import will sync all users again and this time store `mail` from AD to `User Name` in Oracle Identity Cloud Service.



14. We have AD Bridge configured to sync users into Oracle Identity Cloud Service. Sometimes few users are not syncing into Oracle Identity Cloud Service during scheduled sync job, but if we run full import then those missing users appear in Oracle Identity Cloud Service. Why?

Answer: AD Bridge records updates in Active Directory using synchronization tokens and an update sequence number (USN). The previous highest USN value is stored in Oracle Identity Cloud Service and any time an incremental sync is run; Oracle Identity Cloud Service reads the data from the stored USN to the latest USN. Sometimes, because of factors such as a Domain Controller change, USN numbers get corrupted (if a new DC has large USN value than previous DC) causing users not to sync. A Full sync doesn't use tokens that is why the users appear in a Full sync. To fix this issue, Oracle needs to reset the sequence number, which can be done by using the API. Contact Oracle support for the help.

Note: This issue is already handled and won't come in latest version of AD Bridge. Upgrading AD Bridge will resolve this automatically.

15. Can I use Active Directory (AD) Bridge client to sync with Azure AD?

Answer: No, Azure AD is not supported through AD Bridge. The AD Bridge only works with on-premise Active Directories. Azure AD is supported through Microsoft Azure integration as well as through Azure AD connector.

16. Can I change the attribute mapping at any time?

Answer: Yes, attribute mappings can be changed at any time. Ensure that you perform a Full sync after saving the new configuration. User data will be updated by the Full sync. If you don't do a Full sync, existing user data remains the same and new users will have updated data. It is **NOT** recommended that you change attribute mapping frequently.

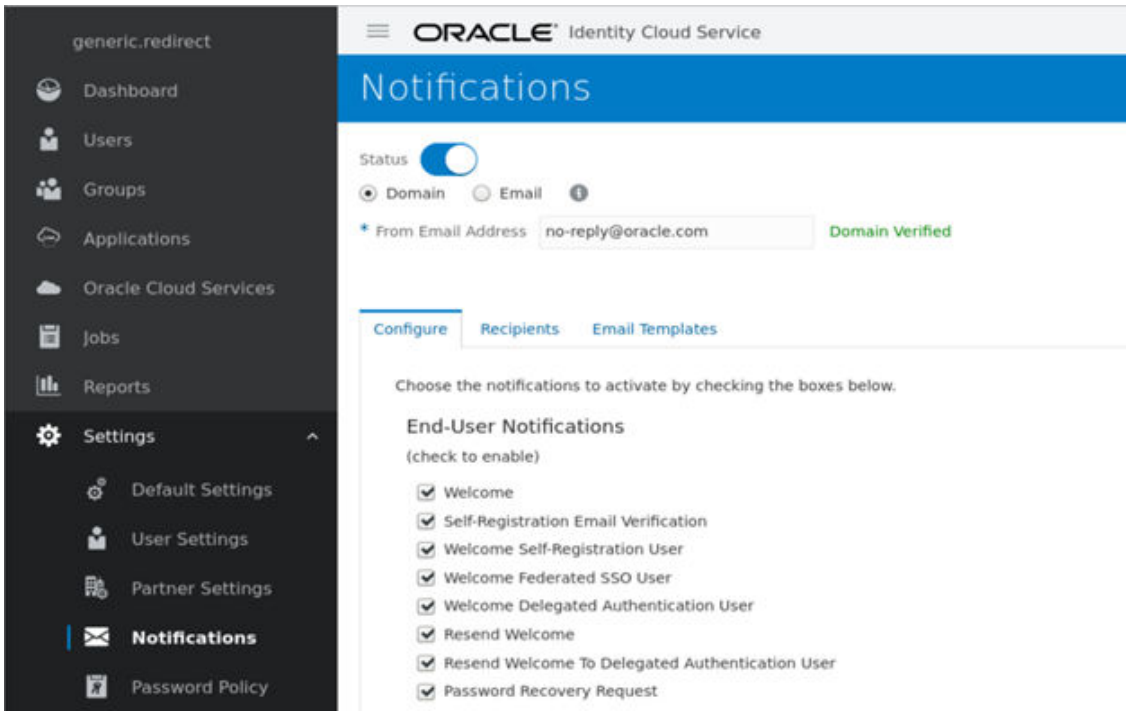
17. My sync hasn't completed for days. What should I do to terminate it?

Answer: Use the Abort option on the Import page to quit the unresponsive job. This will mark your previous stuck sync as Failed. Submit a new sync and then check connectivity from the Windows Server (on which AD Bridge is installed) to Oracle Identity Cloud Service. If the problem persists, contact Oracle support.

18. I want to suppress certain auto-generated emails / notifications. How can I do it?

Answer: Oracle Identity Cloud Service provides full control over notifications. Go to Settings, then Notifications. Here you can see three tabs:

- **Configure:** Select which notifications to send.
- **Recipients:** To limit users to send notifications to. Don't make changes here unless you are sure.
- **Email Templates:** Change the design or the contents of the email sent to the customers.



19. Where can I check to see which user/group failed to sync and the reason for the failure?

Answer: Currently, this can only be traced through AD Bridge Logs. You can find the log files from the AD Bridge client user interface. Search for your username or group name to see what failures occurred during the sync.

The following example shows one user that was successfully synced and another where the sync failed.

```
2021-08-09 13:00:53,019 [22] ERROR IDBridge - Failed to create/update User 'CN=u4
sr,OU=Noida,DC=adfs,DC=fed,DC=oracle,DC=com' with error code 400.
Error message: error.common.validation.invalidEmailFormat : "u 4@oracle.com" is an
invalid format for an email address . The format must be compliant with RFC 5322..
2021-08-09 13:00:53,020 [22] INFO  IDBridge - Adding user failed members from user
sync to refrain from retrying  :: 51330faa-d5ea-45b4-ab10-e033f9836ac7
2021-08-09 13:00:53,020 [22] INFO  IDBridge - Create on User 'CN=manojf
kumarl,OU=Noida,DC=adfs,DC=fed,DC=oracle,DC=com' succeeded on remote server.
```

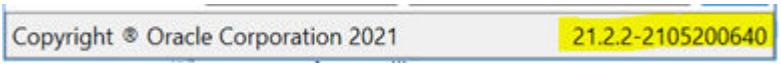
20. What does Delinking mean?

Answer: Oracle Identity Cloud Service keeps a mapping of all the AD users (Oracle Identity Cloud Service identifier mapped to AD identifier). When the user is removed from the active sync because of a new filter condition, for example, the record in Oracle Identity Cloud Service

is kept and just the mapping is removed. The removal of mapping is called Delinking. This case is different from deletion as user is not deleted from AD, if filters are reset, the user will be linked again.

21. A new version of Active Directory (AD) Bridge client is available. Should I install it?

Answer: You should always upgrade to a new version. Make sure you are not installing the current version again. Reinstalling the current version removes the existing Bridge and may lead to authentication and sync failures. Verify the version number from the AD Bridge user interface.



22. Do I need to uninstall the existing Active Directory (AD) Bridge installation in order to upgrade?

Answer: You do not need to uninstall the existing Active Directory (AD) Bridge to upgrade to a newer version.

23. How many Bridges can I install for a given domain?

Answer: A tenant can configure a maximum of 10 domains and for every domain a maximum of 5 Bridges can be configured, only when high availability (HA) is enabled for a tenant. This limit is defined in configuration at Oracle Identity Cloud Service.

24. Can I install more than one Bridge on the same Windows Server machine?

Answer: No, only a single Bridge can be installed, similar to a program in Windows. To use HA, you need multiple machines connected to the same AD Domain.

25. When we upgrade our Active Directory (AD) Bridge client does my first sync after that need to be a Full sync?

Answer: No, existing data will not be impacted because of an upgrade. You can perform an incremental sync. Also, the sync schedule won't be affected, and next sync will be performed as configured.

26. Can I downgrade my Active Directory (AD) Bridge client?

Answer: This is **not** recommended. If you want to downgrade the client, you need to uninstall the current one first. This leads to a downtime of services (sync, delegated authentication, etc.). You can then install the version you want.

27. A few of my users/groups are NOT getting synced. What should I do?

Answer: Use any of the following troubleshooting methods to determine the cause.

- Check the OU configuration on the Directory Integrations page. You need to select the OUs for groups and users separately. Even if you have the same OU for groups and users, select them separately. Make sure to save the configuration page after you make the changes.
- Confirm the filter used in users/groups on the configuration page. Use PowerShell to execute the filter and check whether your users are visible there.
- Check the network connectivity from AD Bridge client to Oracle Identity Cloud Service. (Only if some all records are failing.)
- Check the IDBridge log file ("View logs" from AD Bridge user interface). Look for an error like the following:

```
2021-08-09 13:00:53,019 [22] ERROR IDBridge - Failed to create/update User 'CN=u4
sr,OU=Noida,DC=adfs,DC=fed,DC=oracle,DC=com' with error code 400.
Error message: error.common.validation.invalidEmailFormat : "u 4@oracle.com" is an
invalid format for an email address . The format must be compliant with RFC 5322..
```

28. Which version of Windows Server do I need on my Windows machine. 2012, 2016?

Answer: Any version above 2012 R2 is supported. Recommendation is to use Windows Server 2016.

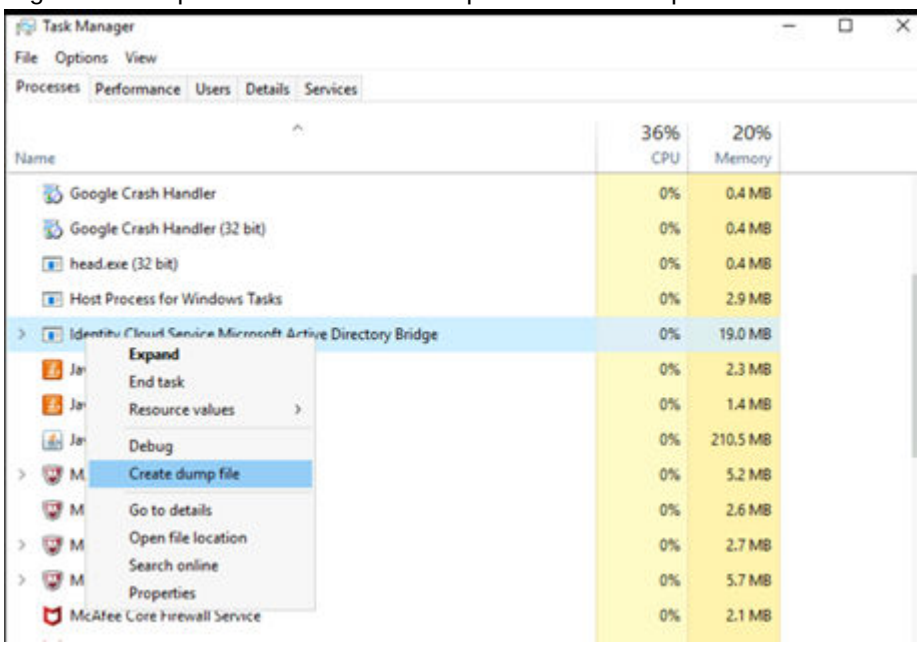
29. How do I enable AD Bridge trace mode logging?

To enable trace mode:

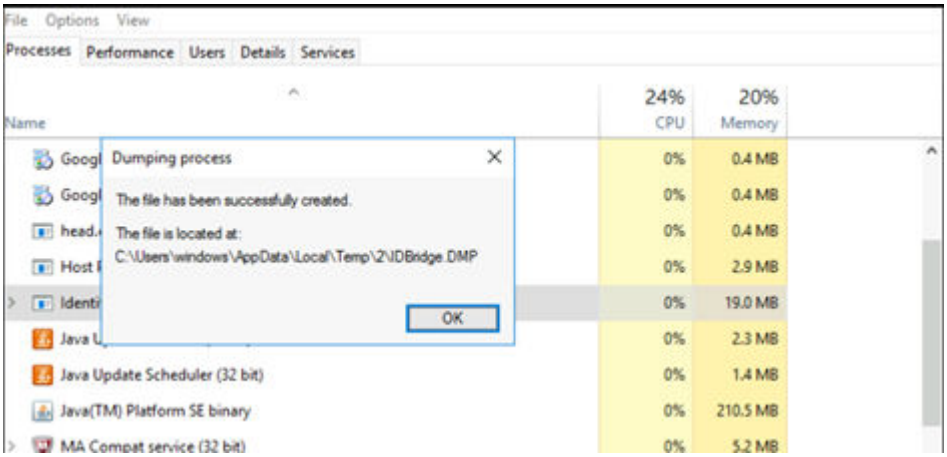
1. Go to the AD Bridge installation folder. The default location is: C:\Program Files\Oracle\IDBridge.
2. Open the file log4net.config.
3. Change this line <level value="**info**" /> to <level value="**trace**" />.
4. If you get a permissions error, open the editor with Administrator privileges. If you are using Notepad, search for Notepad in the Start menu, right click, and choose “Run as Administrator”, then open the log file to make changes. **Note:** The log level change does **NOT** require restart of AD Bridge client.

30. How do I take a thread dump of the AD Bridge service on an AD Bridge machine?

1. Open Task manager on A machine where the AD Bridge client binary is installed.
2. Go to the Processes tab.
3. Search for the process with the name "Identity Cloud Service Microsoft Active Directory Bridge" in the process list.
4. Right click the process and select the option Create dump file.



5. After a few seconds. the display dump location and dump file name display.



31. What additional steps I need to follow if I have changed my filter? Does changing the filter have an impact on my functionality?

Answer: Filters might prevent new users and groups from syncing into Oracle Identity Cloud Service. Complete the following tasks before adding or modifying filters:

1. Verify the filters by running them using PowerShell commands. Ensure that all data is included.
2. Always run a Full sync after changing filters. This will make sure any previously ignored entries are synced. Also, this will cleanup existing redundant mappings.
3. Existing users/groups will not be deleted. Even if they are out of filter, they will be delinked, but kept in Oracle Identity Cloud Service.

32. What will happen to my Delegated Authentication Request when any of below is true:

a. AD Bridge client is down

b. AD Bridge client is NOT able to connect to Oracle Identity Cloud Service Cloud.

c. Active Directory is down

d. AD Bridge client is busy processing other delegated authentication requests

Answer: In all the cases, the authentication request will fail, except if the password caching is enabled and the password is available in the cache. For first three scenarios (a,b,c), service will recover when the downstream system/connectivity issue resolves. For the last scenario (d), service will recover after the concurrent request load decreases.

33. If I have enabled password caching, then which password will be used for delegated authentication:

a. Cached Password or

b. Actual Password stored in Active Directory.

Answer: First, the actual password will be used to authenticate the users. The request will go to the Active Directory through AD Bridge and the Oracle Identity Cloud Service stored password will not be used. But, if this request fails because of any of reasons mentioned in previous question, then authentication will be tried using the password stored in cache. Fallback to the Oracle Identity Cloud Service cached password can be enabled or disabled from the Delegated Authentication settings.

34. When do we cache password in Oracle Identity Cloud Service and for how long it is kept in cache?

Answer: If password caching is enabled and there is no cached password or the cache password is expired, then, we store password next time when the user successfully logs in the system. Default expiry window of a password is five days but can be changed from delegated authentication settings.

35. Why is my AD Bridge installation failing with this message "ID Bridge Installer is failing"?

Answer: You've breached the number of domains or the number of Bridge clients allowed for your tenancy. Default limits are specified in question 23.

36. Where are installation log files are created, to triage issue with installation?

Answer: Installer logs from under %TEMP% folder on the Windows machine where the installation was attempted. From Windows start menu, open run prompt and enter "%TEMP%"

You will see three files per install:

- Identity_Cloud_Service_Microsoft_Active_Directory_Bridge_<timestamp>.log
- Identity_Cloud_Service_Microsoft_Active_Directory_Bridge_<timestamp>_Internal.log
- Identity_Cloud_Service_Microsoft_Active_Directory_Bridge_<timestamp>_ad_id_bridge.msi.log

Provide the latest files to Oracle support when you raise a service request.

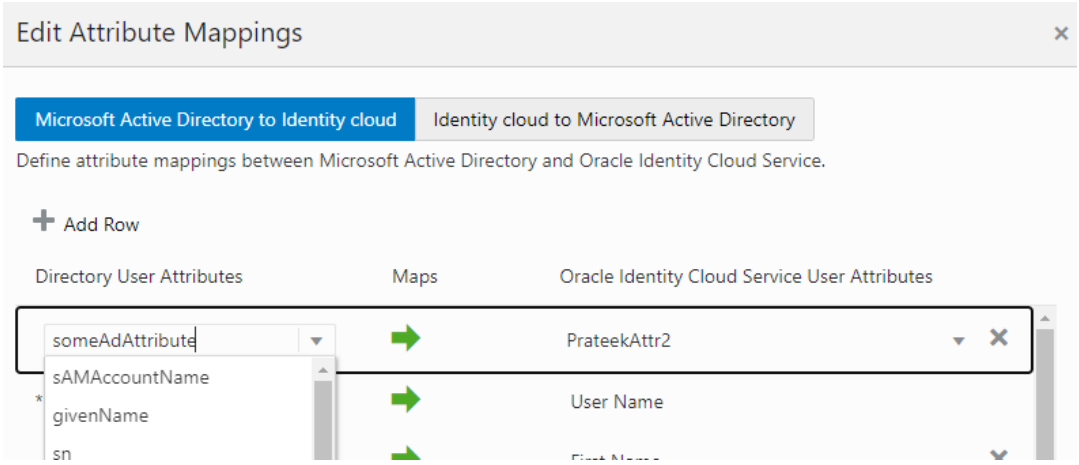
37. I'm unable to see my AD attribute in "Configure Attribute Mapping" section?

Answer: Note that the Directory User Attribute input is not a dropdown menu selection, but a suggestive text box. You can write anything to the text box, even if that attribute is not present in your AD. Ensure you type the correct attribute exactly (including the uppercase and lowercase characters) the way the attribute name appears in Active Directory. By not doing this, you will not get an error at mapping save time, but your AD sync will be impacted. It will not be able to pull this attribute from Active Directory.

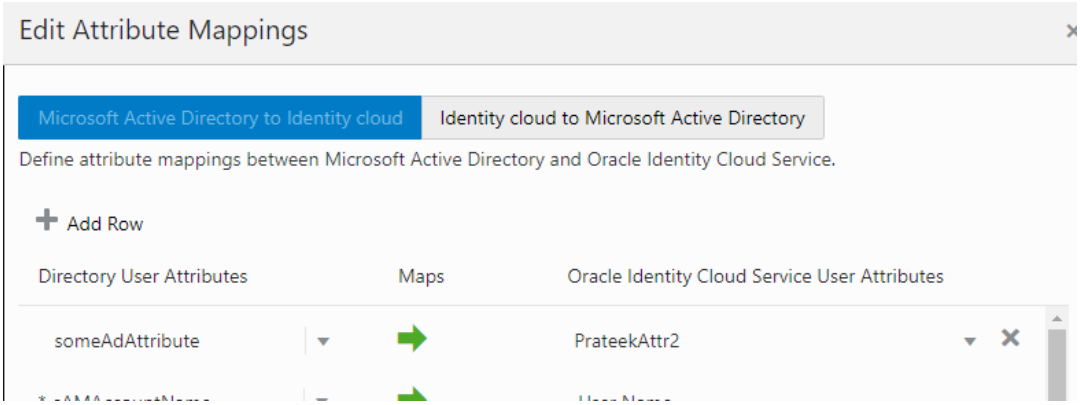
The suggestion are based on frequently used AD attributes only. The Oracle Identity Cloud Service attributes is a dropdown menu selection, and you will see all the attributes there.

Refer to following screen shots:

1. Write your attribute name, for example, "someAdAttribute".



2. Save your row.

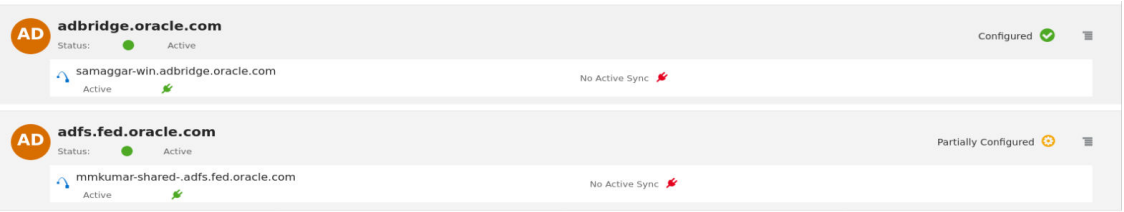


38. Why does my domain show that it's partially configured and the import option is disabled?

Answer: A partially configured domain indicates that no OU is selected on the configuration page. Any OU selection for users, groups or both is required for configuring domain for sync. Till then there is nothing to import and import will stay disabled.

To configure a domain:

1. Click the domain to open it.
2. Select any OU to fetch users and groups from. **Note:** Users and groups OU selection must be done separately.
3. You can choose a different set of OUs for users and groups.
4. Any OU selection for a user or a group will enable the import option.



18

Manage Oracle Identity Cloud Service Session Settings

Learn how to manage your default session settings for Oracle Identity Cloud Service.

To manage default identity domain settings, you must be assigned to the identity domain administrator role. See Add or Remove a User Account from an Administrator Role.

Topics:

Change Session Settings

Oracle Identity Cloud Service session settings include the session duration, URLs for login, logout, errors, and social callback, the authentication flow for accessing Oracle Identity Cloud Service, and CORS settings.

To open this page, you must be assigned the identity domain administrator role or the security administrator role.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Session Settings**.
2. In the **Session Duration** field, enter a duration in minutes.

> **Note:**
>
> The **Session Duration** is the duration in minutes for which the user's session is valid. The user's session will time out after the **Session Duration** has been reached regardless of actual user activity or inactivity.

3. In the **Login URL** field, enter the URl where you want the user redirected to log in.
4. Select the **Enable Custom Login Page For The Admin Console** switch to allow login-customization for the Admin Console.
5. Enter a **Logout URL**.

   For example, to redirect the user to the **My Profile** console, enter `/ui/v1/myconsole`.
6. In the **Error URL** field, enter the tenant specific Error page URL to which a user is redirected after an error.

   This URL is used when the Application specific Custom Error URL is not specified for an Application.
7. In the **Social Linking Callback URL** field, enter the URL that Oracle Identity Cloud Service redirects to after linking a user between social providers and Oracle Identity Cloud Service is complete.

   This URL is used when the Application specific Social Linking Callback URL is not specified for an Application.

8. (Optional) Select the **Enable User Name First** switch to allow the use of passwordless authentication.

   This setting changes the conventional user name and password login to user name, followed by another administrator-configured factor to log in.

   > **Note:**
   >
   > This option appears only if passwordless authentication is enabled. If this option doesn't appear, then contact Oracle Support to enable passwordless authentication.

   If you turn on the **Enable User Name First** switch, then users will be shown two pages when they sign in to Oracle Identity Cloud Service. In the first page, the user provides their user name, and then clicks **Sign In**. Oracle Identity Cloud Service evaluates the criteria in the identity provider policies to determine which identity providers and local authentication factors (such as **Email**, **Mobile App Notification**, **Mobile App Passcode**, **Text Message**, or **User Name-Password**) will be available to the user to sign in to Oracle Identity Cloud Service. These identity providers and local authentication factors appear in the second page. The user uses one of the identity providers or authentication factors to access Oracle Identity Cloud Service. See Add an Identity Provider Policy to see how you can configure login options for users.

   If you turn off this switch, then in the **Sign In** page, the user can authenticate into Oracle Identity Cloud Service either locally, by providing their credentials (user name and password), or by using a SAML or social identity provider.

9. (Optional) Turn on **Allow Cross-Origin Resource Sharing (CORS)**.

   If you turn this option on, you might also want to set the **Allowed CORS Domain Names** option.

   If you need to configure Cloud Gate CORS settings in Oracle Identity Cloud Service, then you use the Oracle Identity Cloud Service REST API. See Configuring Cloud Gate CORS Settings in Oracle Identity Cloud Service.

10. Leave the **Show The Specific Error Message For Login Policy Violation** switch on.

    This option is switched on by default and allows the system to display the specific policy-violation error-message if the login policy is violated. Although this option is less secure, but is more helpful. However, if the switch is turned off, the system displays the standard error message. This is the most secure behavior.

11. Click **Save.**

An additional session setting is to set device fingerprinting, where user device attributes are processed and the fingerprint is stored in a browser cookie to uniquely identify a user's system. See Use Device Fingerprints.

19

Manage Self-Registration Profiles in Oracle Identity Cloud Service

Create self-registration profiles to manage different sets of users, approval policies, and applications. For example, Identity domain administrators can create profiles that allow users to complete self-registration and gain access to specific applications without approval.

Topics

Typical Workflow for Managing Self-Registration Profiles

To start creating self-registration profiles, first complete the prerequisites in the following table.

Task	Description	Additional Information
Create groups.	Create the groups that you want to use for self-registration.	Manage Oracle Identity Cloud Service Groups
Review self-registration notification templates.	Review the self-registration email templates and make any necessary changes.	Typical Workflow for Customizing Oracle Identity Cloud Service Notifications

Once you have completed the prerequisites, complete the tasks in the following table.

Task	Description	Additional Information
Create Self-Registration Profiles	Add your customized header and footer logos, determine your allowed email domains, and add header, footer, success, and user consent text.	Create Self-Registration Profiles
Activate the profile.	By default, a profile is created in inactive state. Activate the profile before using it.	Create Self-Registration Profiles
Construct a self-registration URL.	Construct a self-registration URL to be sent to the users.	Create Self-Registration Profiles

Understand Self-Registration Profiles

Self-registration profiles give you the flexibility to define different mechanisms for users to register with an application.

Using self-registration profiles, you can:

- Create a self-registration **consumer flow** that allows users to create an account in a verified state. Use the REST API for Oracle Identity Cloud Service to turn off the `activationEmailRequired` option. The user can then directly log in to Oracle Identity Cloud Service using a user name and password to authenticate. See Self Registration Profiles REST Endpoints.
- Create a self-registration **partner flow** that allows users to create an account in an unverified state. Use the REST API for Oracle Identity Cloud Service to turn on the `activationEmailRequired` option so that a user receives a link in the welcome email to verify the user. Once the user clicks this link, the user's state is changed to verified and the user can log in to Oracle Identity Cloud Service. See Self Registration Profiles REST Endpoints.
  When `activationEmailRequired` is set to true the user is created in a pending state. Use a token embedded in the email notification using the `MeActivator` endpoint to change the user's state from pending to verified.

```
{
  "token": "<access-token>",
  "schemas": [
    "urn:ietf:params:scim:schemas:oracle:idcs:MeActivator"
  ]
}
```

- Specify whether users are prompted and must accept a user consent before self-registering.
- Assign groups to a profile so that users are assigned to all the groups that are part of that profile.
- Specify the domains allowed when accessing the self-registration process. Only users with these specific domains are allowed to register.
- Customize the self-registration login page with your header and footer logos.
- Customize the header, footer, success, and user consent text.
- Delete profiles using the REST API. See REST API for Oracle Identity Cloud Service.

Create Self-Registration Profiles

To manage self-registration for different sets of users, approval policies, and applications, create self-registration profiles.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then **Self Registration**.
2. Click **Add Profile**.
3. Complete the **Details** section.
   - Enter a unique **Profile Name**.
   - To require a user to accept the terms of use during self-registration, turn on the **User Consent Required** option.
   - To hide the terms of use from the user during self-registration, turn off the **User Consent Required** option.
   - To add groups to the profile, click **Add** in the **Assign to Group** section.

- Add the domains allowed during the self-registration process in the **Allowed Email Domains** field.

> **Note:**
>
> Enter `all` or leave this field blank to allow all email domains.

4. Upload footer and header logos or keep the default logos.
5. Complete the **Self-Registration Content** section.
   - Enter the **Registration Page Name** that you want to appear as a link on your customized login page.
   - Add header, footer, and success text or keep the default values.
   - If you have turned on the **User Consent Required** option, enter the text in the **User Consent Text** field.

> **Tip:**
>
> Click **Cancel** to discard your changes and return to the **Manage Self-Registration Profiles** page. Currently, you can only delete profiles using the REST API.

6. Click **Save**.

   The **Profile ID** that you need for the self-registration link is created.
7. On the **Manage Self-Registration Profiles** page, use the action menu to activate the profile.

Next, you must construct a self-registration URL. Click the profile that you created and use the **Profile ID** to construct a URL exactly like the following: `https://[instancename.idcs.internal.oracle.com:port]/ui/v1/signup?profileid=[ProfileID]`

If the URL is not constructed properly, you receive an error stating that your profile was not found. Verify that the syntax of the URL is correct.

This URL gives the user access to the self-registration page. After the user completes self-registration and clicks **Submit**, they are presented with a success page. The user must then click the link **Click here to continue** to go to the **My Apps** page in Oracle identity Cloud Service. If the user does not click the link within 1 hour, the token expires and user is presented with the **Login** page.

20

Download Oracle Identity Cloud Service SDKs and Applications

This section describes how to understand, download, and use Oracle Identity Cloud Service software development kits (SDKs) and applications.

Topics:

Typical Workflow for Downloading Oracle Identity Cloud Service SDKs and Applications

With the download feature in Oracle Identity Cloud Service, you can perform tasks such as downloading software development kits (SDKs) and applications.

Task	Description	Additional Information
Understand Oracle Identity Cloud Service SDKs and applications.	You can learn about SDKs, including how they're used to develop custom mobile and Web applications to authenticate and integrate them with Oracle Identity Cloud Service. You can learn about a Java application known as the Oracle E-Business Suite (EBS) Asserter, including how it's used to integrate Oracle E-Business Suite with Oracle Identity Cloud Service. You can learn about the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM), including how it's used to integrate your Linux environment with Oracle Identity Cloud Service to perform end user authentication with first and second factor authentication. You can learn about an admin client known as the Secure Form Fill Client, including how it's used to configure Secure Form Fill for your applications. You can learn about the Identity Cloud Service Provisioning Bridge, including how it's used to install, start, and stop the client for the Provisioning Bridge. The Provisioning Bridge provides a link between your on-premises apps and Oracle Identity Cloud Service. You can learn about the Identity Cloud Service Device Fingerprint Utility, including how it's used to enable the **Access for an unknown device** event of Adaptive Security for a custom sign-in page.	Understand Oracle Identity Cloud Service SDKs and Applications
Download Oracle Identity Cloud Service SDKs and applications.	You can download SDKs, the EBS Asserter, the Linux PAM, the Secure Form Fill Client, the client for the Provisioning Bridge, and the Identity Cloud Service Device Fingerprint Utility using the **Downloads** page.	Download Oracle Identity Cloud Service SDKs and Applications

Task	Description	Additional Information
Use Oracle Identity Cloud Service SDKs and applications.	You can access documentation, Oracle-by-Example (OBE) tutorials, and videos to learn how to use SDKs, the EBS Asserter, the Linux PAM, the Secure Form Fill Client, the Provisioning Bridge, and the Identity Cloud Service Device Fingerprint Utility.	Use Oracle Identity Cloud Service SDKs and Applications

You can download SDKs and applications by:

- The Identity Cloud Service console
- SCIM-based APIs

The following sections describe how to download SDKs and applications by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Understand Oracle Identity Cloud Service SDKs and Applications

You're an identity domain administrator or security administrator who wants to enable mobile and Web applications to authenticate with Oracle Identity Cloud Service. To do this, you use one of the options below.

You may want to integrate your Oracle E-Business Suite environment with Oracle Identity Cloud Service for authentication and password management purposes. To do this, you use a lightweight Java application known as the Oracle E-Business Suite (EBS) Asserter.

You may want to integrate your Linux environment with Oracle Identity Cloud Service to perform end user authentication with first and second factor authentication. To do this, you use the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM).

If your web application supports header based authentication, then use App Gateway to protect access to your application. App Gateway acts as a reserve proxy protecting web applications by restricting unauthorized network access to them.These applications are called Enterprise Applications in Oracle Identity Cloud Service.

Oracle Identity Cloud Service can be used to provide single sign-on for your applications. These applications can be integrated with Oracle Identity Cloud Service using one of the following options:

- **App Catalog:** The App Catalog contains ready-to-use templates to integrate with most of your cloud-based applications.
- **SAML 2.0:** Use Oracle Identity Cloud Service as an identity provider for applications that support the SAML standard.
- **SDKs:** Use SDKs to develop applications to use the Oracle Identity Cloud Service authentication mechanism.
- **Open ID Connect:** Use Oracle Identity Cloud Service as the authentication server for applications that support the Open ID Connect standard.
- **OAuth 2.0:** Use Oracle Identity Cloud Service as the authorization server for applications that support the OAuth standard.

When none of these methods apply to the applications you need to integrate for authentication, use Secure Form Fill. To help you configure Secure Form Fill for your applications, Oracle Identity Cloud Service provides you with an admin client known as the Secure Form Fill Client.

You may want to establish a link between your on-premises apps and Oracle Identity Cloud Service. To do this, create a Provisioning Bridge. Through synchronization, account data that is created and updated directly on the apps is pulled into Oracle Identity Cloud Service (through the Provisioning Bridge) and stored for the corresponding Oracle Identity Cloud Service users and groups. As a result, any changes to this data will be transferred into Oracle Identity Cloud Service. So, if a user is deleted in one of your apps, then this change will be propagated into Oracle Identity Cloud Service. As a result, the state of each record is synchronized between your apps and Oracle Identity Cloud Service.

Creating a Provisioning Bridge includes using the Identity Cloud Service console to add a bridge, and then installing the client for this bridge.

Adaptive Security is an advanced feature of Oracle Identity Cloud Service that provides strong authentication capabilities for your users, based on their behavior within Oracle Identity Cloud Service, and across multiple heterogeneous on-premises applications and cloud services.

When activated, the Adaptive Security feature can analyze a user's risk profile within Oracle Identity Cloud Service when they sign in to access the service, based on their historical behavior and real-time device context, such as access from unknown devices.

Adaptive Security uses the concept of risk providers to allow administrators to configure various contextual and threat events to be analyzed within Oracle Identity Cloud Service. A default risk provider within Oracle Identity Cloud Service is seeded automatically with a list of supported contextual and threat events, such as **Access from an unknown device**. For this event, if a user accesses Oracle Identity Cloud Service from a device that hasn't been previously used to access the service, then this event (commonly referred to as Device Fingerprinting) is triggered.

Although Oracle Identity Cloud Service has a sign-in page, you may prefer to use your own page. If so, then you can use the Identity Cloud Service Device Fingerprint Utility to enable the **Access for an unknown device** event of Adaptive Security for your custom sign-in page.

Oracle Identity Cloud Service has a centralized location in the Identity Cloud Service console where you can download SDKs, the EBS Asserter, the Secure Form Fill Client, the Linux PAM, the Identity Cloud Service Provisioning Bridge Client, or the Identity Cloud Service Device Fingerprint Utility.

For this version of Oracle Identity Cloud Service, the following SDKs and applications are available:

Name	Type	Description
Identity Cloud Service E-Business Suite Asserter	Application	Use this Java application to integrate Oracle E-Business Suite with Oracle Identity Cloud Service.
App Gateway for Identity Cloud Service	Application	Use this software appliance to integrate web applications, also known as enterprise applications with Oracle Identity Cloud Service for authentication purposes.
Identity Cloud Service Secure Form Fill Client	Application	Use this admin client to configure Secure Form Fill for your applications.

Name	Type	Description
Identity Cloud Service Provisioning Bridge	Application	Use this application to install, start, or stop the client for the Provisioning Bridge.
Identity Cloud Service Linux Pluggable Authentication Module (PAM)	Application	Use this module to integrate Linux environments with Oracle Identity Cloud Service for authentication purposes.
Identity Cloud Service Device Fingerprint Utility	Application	Use this utility to enable Adaptive Security for your custom UI.
Identity Cloud Service SDK for Java	SDK	Use this SDK to develop your web-based Java applications.
Identity Cloud Service SDK for Node.js	SDK	Use this SDK to develop web-based Node.js applications.
Identity Cloud Service SDK for Python	SDK	Use this SDK to develop web-based Python applications.
Identity Cloud Service SDK for .NET	SDK	Use this SDK to develop web based ASP.NET applications
Identity Cloud Service SDK for Android	SDK	Use this SDK to develop mobile Android applications.
Identity Cloud Service SDK for iOS	SDK	Use this SDK to develop mobile iOS applications.

Download Oracle Identity Cloud Service SDKs and Applications

You can download software development kits (SDKs) to develop your Web and mobile applications to authenticate and integrate them with Oracle Identity Cloud Service. You can also download the E-Business Suite Asserter to integrate Oracle E-Business Suite with Oracle Identity Cloud Service, the Secure Form Fill Client to configure Secure Form Fill for your applications, the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) to integrate your Linux environments with Oracle Identity Cloud Service, the App Gateway binary file to enable you to integrate your enterprise applications with Oracle Identity Cloud Service, the client for the Provisioning Bridge to establish a link between your on-premises apps and Oracle Identity Cloud Service, or the Identity Cloud Service Device Fingerprint Utility to enable the **Access for an unknown device** event of Adaptive Security for your custom sign-in page.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Downloads**.
2. In the **Downloads** page, click **Download** to the right of the SDK or application that you want to download.
3. Verify that a **Success** status appears to the right of the SDK or application that you downloaded.

Use Oracle Identity Cloud Service SDKs and Applications

You're a developer who wants to integrate Web applications with Oracle Identity Cloud Service. Oracle Identity Cloud Service provides you with software development kits (SDKs) that you can use to enable your Java, Node.js, Python, or .NET applications to authenticate with Oracle Identity Cloud Service. For more information about using these SDKs, refer to the following tutorials:

- Learn About Authenticating an Application with Oracle Identity Cloud Service by Using the Java SDK
- Learn About Authenticating an Application with Oracle Identity Cloud Service by Using the Node.js SDK
- Learn about Authenticating an Application with Oracle Identity Cloud Service by Using the Python SDK
- Use Oracle Identity Cloud Service's Software Development Kit (SDK) for Authentication in .NET Web Applications

If you're developing mobile applications, and you want to use Oracle Identity Cloud Service as an authentication server, then you can use SDKs to enable your mobile Android or iOS applications to authenticate with Oracle Identity Cloud Service. For more information on using these SDKs, refer to the following documentation:

- Learn About Authenticating an Android Mobile App with Oracle Identity Cloud Service
- Learn About Authenticating an iOS Mobile App with Oracle Identity Cloud Service

If your web application uses header variables to identitfy the user logged in, then you can use the App Gateway to integrate your application with Oracle Identity Cloud Service for authentication purposes. App Gateway acts as a reserve proxy protecting your application by restricting unauthorized network access to the application or ensuring that the users are authenticated in Oracle Identity Cloud Service before forwarding the request to the application. .

If you can't change the source code of your Web application or the application isn't based on headers, then use Oracle Identity Cloud Service's Secure Form Fill. The Secure Form Fill Client helps you map the sign in form for your Web application so Oracle Identity Cloud Service knows how to populate the user's user name and password automatically, and submit the user's credentials to the application's identity store. For more information about using the Secure Form Fill Client, see Use Secure Form Fill to Authenticate an Application with Oracle Identity Cloud Service.

You may have to integrate your Oracle E-Business Suite (EBS) environment with other cloud services in single-sign-on (SSO) mode using Oracle Identity Cloud Service. Oracle Identity Cloud Service provides a lightweight Java application known as the EBS Asserter that implements SSO. By using the EBS Asserter, you can integrate Oracle E-Business Suite with Oracle Identity Cloud Service for authentication and password management purposes. See Use the E-Business Suite Asserter to Enable SSO for Oracle E-Business Suite with Oracle Identity Cloud Service.

You may want to integrate your Linux environment with Oracle Identity Cloud Service to perform end user authentication with first and second factor authentication. Oracle Identity Cloud Service provides a module known as the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM). For more information about using the Linux PAM, see Manage Linux Authentication using the Linux-PAM Module.

If you want to establish a link between your on-premises apps and Oracle Identity Cloud Service, then create a Provisioning Bridge. By doing so, you're synchronizing user and group account data that's created and updated directly on the apps with Oracle Identity Cloud Service. Any changes to this data will be transferred into Oracle Identity Cloud Service and stored for the corresponding Oracle Identity Cloud Service users and groups. For more information about using the Provisioning Bridge, see Synchronize Users from Oracle Internet Directory to Oracle Identity Cloud Service.

If you have a custom sign-in page for Oracle Identity Cloud Service and you want to enable the **Access for an unknown device** event of Adaptive Security for your page, then use the Identity Cloud Service Device Fingerprint Utility. If a user uses your sign-in page to access

Oracle Identity Cloud Service from a device that hasn't been previously used to access the service, then this event is triggered. For more information about using the Identity Cloud Service Device Fingerprint Utility, see Enable the 'Access for an unknown device' Event for a Custom Sign-In Page.

21

Set Up and Validate RADIUS Proxy

Remote Authentication Dial In User Service (RADIUS) is a network protocol that defines rules and conventions for communication between network devices. RADIUS Proxy authenticates and authorizes users or devices and also trackes the usage of those services.

Topics:

Setup RADIUS Proxy

Learn how to setup and install RADIUS Proxy as well as to test that it's working.

Before You Begin

- **Enable RADIUS Proxy.** This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service.
- Install the latest Postman client.
- Download the Oracle Identity Cloud Service RADIUS Proxy Postman collection.
- Review these checkpoints. As you are setting up RADIUS Proxy, use the following checkpoints to verify that your configuration is correct at each step of the process.
    1. Check that the RADIUS Proxy and the RADIUS Proxy Client App are activated in Identity Cloud Service.
    2. Check the IP address of Database and port number of RADIUS Proxy are correctly configured in the RADIUS App.
    3. Check the RADIUS Agent is up and running.
    4. Check the proxy server is up and running.
    5. Check the database is up.

RADIUS Proxy Mapping

RADIUS Proxy and RADIUS Proxy Listener has a 1-1 mapping, for example for each RADIUS Proxy there is one RADIUS Proxy Listener. Multiple Oracle DB RADIUS clients can be mapped to one RADIUS Proxy, that is, a RADIUS Proxy has a 1-n mapping with Oracle DB RADIUS clients.

If an administrator configures multiple Oracle DB RADIUS clients, then that many Oracle Database RADIUS apps need to be created in Oracle Identity Cloud Service-one for each Oracle DB RADIUS client. For example, if an administrator has configured four Oracle DB RADIUS clients to one RADIUS Proxy, then in Oracle Identity Cloud Service there must be four Oracle Database RADIUS apps configured-one for each Oracle DB client.

To setup RADIUS Proxy:

1. Download the latest RADIUS Proxy Installer from Oracle Identity Cloud Service.
   a. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then **Downloads**.
   b. Choose **Identity Cloud Service RADIUS Proxy for Linux**, and then click **Download**.
2. Create the RADIUS App from the RADIUS App Template. Note: For REST go to **RADIUS Proxy**, **RADIUS App**, **Search**, and then **Search all apps** (with search criteria).
   a. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Applications**, **Add**, and then **App Catalog**.
   b. Search for the **Oracle Database Radius App Template** and click **Add**.
   c. Complete the App details similar to the example below.
      - **Name:** dbserver
      - **Description:** App representing the Oracle database server as a RADIUS client
      - **IP Address of Oracle Database server:** 10.242.230.122 (This is the IP address where the database is installed.)
      - **Port of RADIUS Proxy:** 1812 (The port number on which RADIUS Proxy should listen for requests from this Oracle database. The same port number should be configured in the RADIUS settings of Oracle Database.)
      - **Secret key:** testing123 (The secret key used to secure communication between RADIUS Proxy and the Oracle Database server. The same key should be configured in the RADIUS settings of Oracle Database.)
   d. Click **Add**, **Activate**, and then click the **Users** tab.

> **Note:**
>
> Assign the Users, who should be allowed to login to Oracle Database, to this RADIUS App by clicking **Assign Users**. Instead of assigning individual users, a Group which contains those users, can also be assigned. Click the Groups tab, and then **Assign Groups**.
>
> Note : Create the group name in the Oracle Identity Cloud instance according to the following format defined in Step 3C: Configure the RADIUS Server in Configuring RADIUS Authentication: `ORA_databaseSID_rolename[_[A]|[D]]`.
>
> For every role in Oracle database to be identified by Oracle Identity Cloud Service, create a corresponding group using the format above. Assign a user to this group in Oracle Identity Cloud Service so that the respective database user is associated with the respective database role.

3. Create a RADIUS Proxy in Oracle Identity Cloud Service.
   a. Register a Client Application. See Register a Client Application.
   b. Open Postman and import the Oracle Identity Cloud Service RADIUS Proxy.postman_collection.json collection to execute the REST requests in this section.
   c. Import the Oracle Identity Cloud Service RADIUS Proxy Example Environment with Variables.postman_environment.json environment file which contains the environment variables used in the collection.
   d. Set the following environment variables.

For **HOST**, use the Oracle Identity Cloud Service address, for example, https://example.identity.acmecorp.com.

For **CLIENT_ID** and **CLIENT_SECRET**, use the values that you copied in Step a above.

> **Note:**
>
> Other environment variables are automatically set when REST requests are executed. Just make sure that following REST requests are executed in the same order.

e. Obtain an access token. To make API calls to Oracle Identity Cloud Service, you must authenticate your client against Oracle Identity Cloud Service, and then obtain an OAuth access token. The access token provides a session between a client (in this case, Postman) and Oracle Identity Cloud Service. By default, the access token has a timeout interval of 60 minutes, and then you must request a new access token to perform additional REST API calls. To obtain an OAuth access token, execute the request in the Postman collection under **RADIUS Proxy**, **OAuth Token**, and then **Obtain access_token (client credentials)**.

f. Create the RADIUS Proxy by using a POST Operation. Go to **RADIUS Proxy**, **Create**, and then **Create a RADIUS Proxy**.

End point: `admin/v1/RadiusProxies/ {{RPid}}`

```
{
"displayName": "Acme RADIUS Proxy",
"description": "This is a RADIUS Proxy used for authentication of
database users",
"type":
"proxy",
"timeout": 20,
"noOfWorkerThreads": 10,
"schemas" :
["urn:ietf:params:scim:schemas:oracle:idcs:RadiusProxy"]
}
```

g. Use this Patch Operation to activate the RADIUS Proxy. Go to **RADIUS Proxy**, **Lifecycle**, and then **Activate a RADIUS Proxy**.

End point: `/admin/v1/RadiusProxies/{{RPid}}`

```
{
"Operations":[
{
"op": "replace",
"path": "active",
"value": true
} ],
"schemas": [
"urn:ietf:params:scim:api:messages:2.0:PatchOp" ] }
```

h. Create the RADIUS Proxy Listener using a POST Operation. Go to **RADIUS Proxy**, **RADIUS Proxy Listeners**, **Create**, and then **Create a RADIUS Proxy Listener**.

End point: `{{HOST}}/admin/v1/RadiusProxyListeners`

```
{
“description”: “Brief description for this RADIUS Proxy Listener.”,
“displayName”: “RP1_L1”,
“hostName”: “<HostName of the machine in which RADIUS Proxy will be
installed.>”,
“radiusProxySettings”:“{\”key1\”: \”value1\”, \”key2\”: \”value2\”}”,
“radiusProxy”:
{ “value” : “<ID of RadiusProxy which is created above.>”
},
“schemas” :
[“urn:ietf:params:scim:schemas:oracle:idcs:RadiusProxyListener”]

}
```

i. Get the `dbserver` App ID. Perform a GET call on `admin/v1/Apps?filter=displayName eq "dbserver"`. Fetch the App ID from the response of this GET call. Go to **RADIUS Proxy**, **RADIUS App**, **Search**, and then **Search all apps (with search criteria)**.

   You can also get the App ID from the URL of the `dbserver`.

j. Create a RADIUS Proxy Mapping using a POST Operation. Go to **RADIUS Proxy**, **RADIUS Proxy Mappings**, **Create**, and then **Create a RADIUS Proxy Mapping**.

   End point: `{{HOST}}/admin/v1/RadiusProxyMappings/`

> **Note:**
>
> For "value" below, the ID is the ID of Radius Proxy which you created above.

```
{
"description": "RADIUS Proxy mapping for Database server",
"radiusProxy": {
"value" : "<RadiusProxyID>"
},
"radiusApp": {
"value": "<<ID of RADIUS App obtained above.>"
},
"schemas" :
["urn:ietf:params:scim:schemas:oracle:idcs:RadiusProxyMapping"]
}
```

k. GET `client_id` and `clientSecret` of the RADIUS Proxy. This is required during RADIUS Proxy installation. RADIUS Proxy will use these credentials to authenticate with Oracle Identity Cloud Service. Go to **RADIUS Proxy**, **Search**, **Create**, **Get client ID, and client secret of the App corresponding to RADIUS Proxy**.

   End point: `{{HOST}}/admin/v1/Apps/{{RPOAuthClientAppId}}?attributes=clientSecret,name`

   `RPOAuthClientAppId`: is the ID of the App corresponding to RADIUS Proxy. You can find it in the response [response.oauthClient.value] in step 3f, *Create a RADIUS Proxy Mapping using a POST Operation*.

Response:

```
{
"isAliasApp": false,
"basedOnTemplate": {
"value": "RadiusProxyAppTemplateId"
},
"displayName": "Acme RADIUS Proxy",
"name": "<client id>",
"id": "75d525ce49ee469ba4dcac00bdfe6446",
"clientSecret": "<client secret>"
}
```

4. Execute the Installer.
   a. Unzip the downloaded `idcs_radius_proxy-xxxx.zip` file into a folder.
   b. Name the folder `<radius bin location-xxxx>`. Where `xxxx` is the version number (for example, 20.1.3).

      Three files are extracted: **FileInfo.json**, **idcs_radius_proxy_installer.bin**, and **InstallerValidation.jar**. The **InstallerValidation.jar** file and the **idcs_radius_proxy_installer.bin** file are located in the same directory post extraction. They must remain in the same directory.
   c. Login as root user or run the following command as sudo: `./idcs_radius_proxy_installer.bin`

      > **Note:**
      >
      > The installer supports only Graphical User Interface mode. It does not support console mode. So, if you see the error: "`Graphical installers are not supported by the VM.`", then make sure *X server* is configured properly. Then, run this command as non root user : `xhost +si:localuser:root` and run installer again.

5. Install RADIUS Proxy.
   a. Read the Welcome screen, and then click **Next**.
   b. Read the Information screen, and then click **Next**.
   c. Select the **Destination Folder** (default is `/root/oracle_radius_proxy`), where the RADIUS Proxy installer will be installed. Click **Next**.
   d. On the HTTP Proxy screen, select **Use HTTP Proxy** if RADIUS Proxy needs to use HTTP proxy to connect to Oracle Identity Cloud Service. If not, then leave this check box unchecked. Click **Next**.
   e. On the Oracle Identity Cloud Service screen, enter the **Cloud Service URL** in the following format: `https://tenant-base-url`. Provide the **Client ID** and **Client Secret** of the RADIUS Proxy created in Oracle Identity Cloud Service. (This is the RADIUS Proxy you created using the POST Operation above.) Click **Next**.
   f. On the RADIUS User and Group Information screen, provide the **Username** and user **Group** information, for example:
      - **Username**: <client>
      - **Group**: <dba>

Oracle Identity Cloud Service RADIUS Proxy daemon will run under the specified username and group.

g. Click **Next**.

h. On the pre-install screen, verify that all the information is correct. If it's correct, click **Install**.

i. When the installation is complete, click **Done**.

6. Check that the RADIUS Agent and RADIUS Proxy are running. The RADIUS Agent obtains configuration data from Oracle Identity Cloud Service at regular intervals. Then, it updates the configuration files used by RADIUS Proxy.

   a. Use the following RADIUS Agent commands to check whether the agent is running:

      - `python <RADIUS_PROXY_INSTALLER_LOCATION>/oracle_radius_proxy/radius_agent/scripts/src/radius_agent.py status`
      - You can also use `stop`, `start` and `restart` if needed.

   b. Use the following command to run RADIUS Proxy: `/sbin/service idcs_radiusd start`

   c. Run these RADIUS Proxy commands to verify the RADIUS service is running.

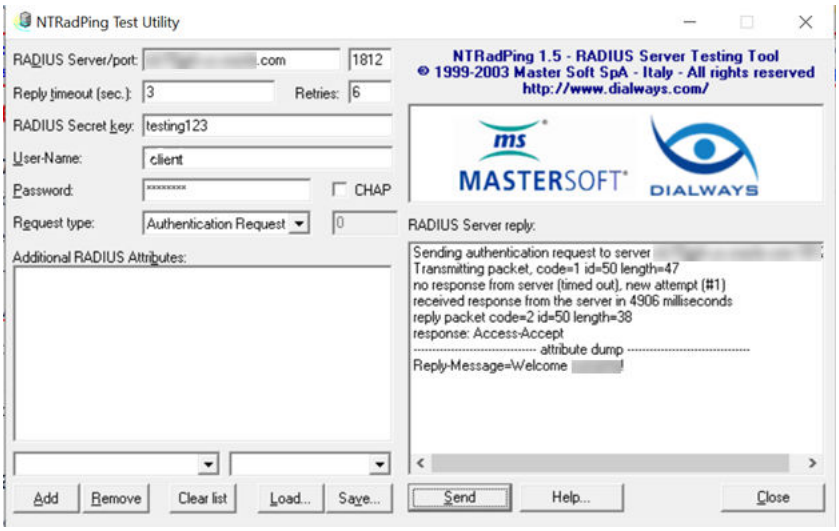
      - `/sbin/service idcs_radiusd status`
      - You can also use `stop`, `start` and `restart` if needed.

7. Optional: Use the NTRadPing Test Utility to validate that RADIUS proxy is working.

   a. Install the NTRadPing Test Utility in Windows, and then create a User in Oracle Identity Cloud Service.

   b. Use the below screenshot as an example. In the below screen shot **client** is the user created in Oracle Identity Cloud Service and **testing123** is the secret key given in RADIUS Settings, **Secret key** of **App Details** page.

Figure 21-1 NTRadPing Test Utility in Windows



8. Setup and Configure Oracle Database 12c. Follow the instructions at Configuring RADIUS Authentication and then use the following commands to create a user/role in the database.

9. Setup and Configure Oracle Database 12c. For more information see Configuring RADIUS Authentication. Follow the instructions in the *Configuring RADIUS Authentication* section to create a user and role in the database.

```
sqlplus /@orclpdb

Alter system set OS_ROLES=TRUE scope=spfile;
Alter system set OS_AUTHENT_PREFIX='' scope=spfile;
create user ckent identified externally;
create role dblogin identified externally;
create role dbreadtable identified externally;
grant create session to dblogin;
grant select on system.help to dbreadtable;
```

10. You can't add an IP address in CIDR format using the Oracle Identity Cloud Service user interface. If the IP address of the Oracle Database is in CIDR format, use the following request from the Postman collection. See Change an IP Address from CIDR Format.
11. Set up MFA. To set up MFA in Oracle Identity Cloud Service follow these instructions: Enable and Configure Multi-Factor Authentication (MFA).

Log Files and Configuration Information

Note the following file locations for log and configuration information.

Installer Logs	<RADIUS_PROXY_INSTALLER_LOCATION>/oracle_radius_proxy/_Oracle®\ Identity\ Cloud\ Service\ RADIUS\ Proxy_installation/Logs/
Radius Agent Logs	<RADIUS_PROXY_INSTALLER_LOCATION>/oracle_radius_proxy/radius_agent/logs/agent.log
Radius Proxy logs	<RADIUS_PROXY_INSTALLER_LOCATION>/oracle_radius_proxy/radius_proxy/log/radius_proxy.log
Radius Proxy Configuration	<RADIUS_PROXY_INSTALLER_LOCATION>/radius_proxy/conf/radius_proxy.conf
Radius Agent Configuration	<RADIUS_PROXY_INSTALLER_LOCATION>/radius_agent/conf/radius_agent.conf
Radius Client Configuration	<RADIUS_PROXY_INSTALLER_LOCATION>/radius_proxy/conf/radius_clients.conf

Trouble Shooting

Learn about common problems that you might encounter when using RADIUS Proxy and learn how to solve them.

`/sbin/service idcs_radiusd` is stopped

Use the following steps when you see that the status of `/sbin/service idcs_radiusd` is stopped.

1. Check the radius agent is running by uning the following Python command: `<RADIUS_PROXY_INSTALLER_LOCATION>/oracle_radius_proxy/radius_agent/scripts/src/radius_agent.py status`
2. If the status is running, check the agent logs at: `<RADIUS_PROXY_INSTALLER_LOCATION>/oracle_radius_proxy/radius_agent/logs/agent.log`
   If you see the below exception in the RADIUS Proxy logs (<RADIUS_PROXY_INSTALLER_LOCATION>/oracle_radius_proxy/radius_proxy/log/radius_proxy.log) file: *Exception in thread "main" java.net.BindException: Cannot assign requested address at sun.nio.ch.Net.bind0(Native Method)*

   The solution is to make sure the host entry is correct in RADIUS Proxy listener.

RADIUS Proxy Known Issues

Learn about RADIUS Proxy known issues you might encounter.

Changes in the RADIUS Proxy Configuration

If any RADIUS Proxy configuration is changed in Oracle Identity Cloud Service, restart RADIUS Agent and RADIUS Proxy by completing the following steps so that the new configuration is reflected:

1. `<RADIUS_PROXY_INSTALLER_LOCATION>/oracle_radius_proxy/radius_agent/scripts/src/radius_agent.py restart`.
2. Verify if the configuration is updated in: `<RADIUS_PROXY_INSTALLER_LOCATION>/radius_proxy/conf/radius_proxy.conf` or `<RADIUS_PROXY_INSTALLER_LOCATION>/radius_proxy/conf/radius_clients.conf`.
3. `/sbin/service idcs_radiusd restart`.

Change an IP Address from CIDR Format

You can't add an IP address in CIDR format using the Oracle Identity Cloud Service user interface. If the IP address of the Oracle Database is in CIDR format, use the following request from the Postman collection. Go to **RADIUS Proxy**, **RADIUS App**, **Modify**, and then **Update RADIUS App (IP Address in CIDR format)**.

```
PATCH: {{HOST}}/admin/v1/Apps/{{appid}}
{
 "schemas": [
 "urn:ietf:params:scim:api:messages:2.0:PatchOp"
 ],
 "Operations": [{
 "op": "replace",
 "path":
"urn:ietf:params:scim:schemas:oracle:idcs:extension:radiusApp:App:clientIP",
 "value": "10.34.0.0/16"
 }]
}
```

22

Customize Schemas in Oracle Identity Cloud Service

Learn how to add, edit, or remove custom schema attributes as well as change user permissions for out-of-the-box (base) schema attributes in Oracle Identity Cloud Service.

Topics:

Add a Custom Schema Attribute

If you're creating your own user interface, or you're using either the **Details** tab of the **Users** page of the Identity Cloud Service console or the **My Profile Details** tab of the **My Profile** console, and you don't see a field that you need, then you can create it.

To do this, extend the Oracle Identity Cloud Service user schema. This schema has two types of attributes: out-of-the-box (base) attributes and custom attributes.

If you don't find a schema attribute that you need from the list of base Oracle Identity Cloud Service schema attributes, then you can use the **Schema Management** page to add a custom attribute.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Schema Management**.
2. Click the **User** schema.
3. Click **Add**.

> **Important:**
>
> After you create a custom schema attribute, if data exists for that attribute, then you can't remove it.

4. Use the following table to populate the fields of the **Add attribute** window:

Field	Description
Display name	The attribute name that appears in the **Schema Management** page, the **Details** tab of the **Users** page of the Identity Cloud Service console, and the **My Profile Details** tab of the My Profile console.
Name	The attribute name that's recognized by the Oracle Identity Cloud Service server.

Field	Description
Description	Provide further information related to its usage and other details that helps the user identify this attribute.
Data type	If the value for this attribute can contain alphanumeric characters, special characters, or spaces, then select **String** for the data type. If you want to create a multivalued attribute, then select **String Array**.
Min length	Select the minimum length of the attribute value. The minimum value allowed is 1.
Max length	Select the maximum length of the attribute value. The maximum value allowed is 4,000.
Searchable	If this check box is selected, then the values for this attribute can be used in searches. If it's not selected, then the values can't be used for searches.
End-user permissions	Select the permission that you want to set for this attribute. Because this is a user permission, and not an administrator one, it applies to an attribute that's associated with the **My Profile Details** tab of the My Profile console only. You can grant the following permissions: • **Hide:** The attribute won't appear in the **My Profile Details** tab of the My Profile console. • **Set Once:** The user can provide a value for the attribute and save it, and then afterward, this becomes a read-only attribute. • **Read Only:** The user can see but can't modify the value associated with this attribute. • **Read-Write:** The user can see and modify the value associated with this attribute.

5. Click **OK**.

The custom attribute is created.

> **Tip:**
>
> Open the custom attribute to find its fully qualified name (FQN).

Edit a Custom Schema Attribute

After you create a custom attribute for the Oracle Identity Cloud Service user schema, you might need to change its settings. For example, you might need to adjust its minimum and maximum length.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Schema Management**.
2. Click the **User** schema.
3. In the **Filter attribute type** section, click **Custom**, and then click the display name of the custom attribute that you want to edit.

4. In the popup window that appears, you can edit values for the following fields: **Display name**, **Description**, **Min length**, **Max length**, and **End-user permissions**. See Add a Custom Schema Attribute for more information about these fields.

> **Note:**
>
> You can't increase the value of the **Min length** field or decrease the value of the **Max length** field. Also, if the value of the **Max length** field was set to below 40 when the custom schema attribute was added, then you can't increase it above 40. However, if the value was set to above 40, then you can increase the maximum length to 4,000.

5. After making your changes, click **OK**.

Remove a Custom Schema Attribute

If you no longer need a custom attribute for the Oracle Identity Cloud Service user schema, then you can remove it.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Schema Management**.
2. Click the **User** schema.
3. In the **Filter attribute type** section, click **Custom**, and then click the **X** button to the right of the custom attribute that you want to remove.
4. In the **Confirmation** window, click **OK**.

> **Note:**
>
> If data exists for the attribute, then you can't remove it.

Change User Permissions for a Base Schema Attribute

In addition to adding, editing, or removing custom schema attributes for the Oracle Identity Cloud Service user schema, you may want to change permissions for an out-of-the-box (base) attribute for this schema.

For example, to maximize the real estate of the **Details** tab of the **Users** page of the Identity Cloud Service console, you may want to hide base attributes that your subscribers don't use. Or, you may to protect the values associated with some base attributes so that they're not changed inadvertently.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Schema Management**.
2. Click the **User** schema.
3. In the **Filter attribute type** section, click **Base**, and then click the display name of the base attribute for which you want to change user permissions.
4. From the **End-user permissions** field, select the permission that you want to set for this attribute. See Add a Custom Schema Attribute.
5. After making your change, click **OK**.

Part IV
Configure Security Settings

Learn how to configure important security settings.

Chapters

Supported TLS Cipher-Suites

Oracle Identity Cloud Service supports the TLSv1.2 protocol with the cipher-suites listed in the following table.

Oracle Identity Cloud Service Release	TLS Protocol Version	TLS Cipher-Suite IANA Name	Open SSL Name
19.3.3	TLSv1.2	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDHE-RSA-AES256-GCM-SHA384
19.3.3	TLSv1.2	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	ECDHE-RSA-AES128-GCM-SHA256

Currently, Oracle Identity Cloud Service intends to continue supporting the TLS ciphers listed in the table above and may document support for other TLS ciphers in the future by adding them to the table. Oracle Identity Cloud Service will provide advance notice of any intention to discontinue future support for any currently supported TLS cipher.

> **Note:**
>
> Oracle Identity Cloud Service may expose, for reasons of backward-compatibility, additional TLS cipher-suites that are not documented as supported. However, you shouldn't rely on Oracle Identity Cloud Service to support a TLS cipher-suite other than those listed. Oracle Identity Cloud Service may remove, at any time, any cipher-suite that is not listed in the table above.

23

Manage Terms of Use

Oracle Identity Cloud Service allows the customer to present disclaimers and acceptable use policies, also known as Terms of Use, to its users. The customer can configure Terms of Use on an application basis and collect consent from users prior to allowing them access to the application. As a domain or security administrator, you can manage Terms of Use and perform the following:

Topics:

Understand Terms of Use

The Terms of Use are the rules that one must agree to abide in order to access an application.

The Terms of Use feature in Oracle Identity Cloud Service helps customer to set the terms and conditions for the users to access an application, based on the user's consent. This feature allows the identity domain administrator to set relevant disclaimers for legal or compliance requirements and enforce the terms by refusing the service.

In Oracle Identity Cloud Service, you can grant or deny access to the applications based on the consent provided by the user. When the user logs in for the first time, the relevant disclaimers for legal or compliance requirements are displayed. The user has the option of either accepting or denying his consent for accessing that particular application based on the content of the disclaimer. If the user does not provide his consent by accepting the Terms of Use, he will not be allowed to access that particular application. As a domain or security administrator, you can create or customize your disclaimers based on the need and language of your choice.

Add a Terms of Use

You create a Terms of Use that you can map it to an application or to multiple applications. When a user tries to access a particular application, the Terms of Use mapped to that application is presented to the user. When the user accepts the terms of use and provides his consent, he is allowed to access that application.

To create a Terms of Use, perform the following procedure:

1. From the navigation drawer, click **Security** and then click **Terms of Use**.
2. In the **Terms of Use** screen, click **Add**.
3. In the **Add Terms of Use** screen, enter the following details:

- **Name**: Enter a name for the Terms of Use to easily identify it.
- **Description**: Enter the description to help understand the purpose and usage of this Terms of Use.
- **Duration**: Enter the duration for which the Terms of Use consent will be valid. The value can range between 1 and 365 days. Select **Never Expires** if you do not want this Terms of Use to expire.

Click **Next** to proceed to the **Statements** screen. A confirmation message is displayed.

4. In the **Statements** screen, click **Add** and perform the following:
   - **Language**: A list of supported language is displayed. Select the language in which you want to create the statement for your Terms of Use.
   - **Statement**: Enter or paste the statement content.

   Click **Save**.
5. In the **Assign Applications** screen, click **Add** and perform the following:
   - Search and select the application that you want to assign to this Terms of Use. Alternatively you can select multiple applications to assign them with this Terms of Use.

   Click **OK** and the confirmation message is displayed.
6. Click **Finish**.

View Details of Terms of Use

You can view the details of a particular Terms of Use, like Name, Description, Duration, Statements and the applications that are assigned to the particular Terms of Use.

1. From the navigation drawer, click **Security** and then click **Terms of Use**. The list of Terms of Use that are already created is displayed.
2. Click the required Terms of Use to view the details.

Modify Terms of Use

You can modify the details of a particular Terms of Use, like Name, Description, Duration, Statements and the applications that are assigned to the particular Terms of Use.

1. From the navigation drawer, click **Security** and then click **Terms of Use**. The list of Terms of Use that are already created is displayed.
2. Click the required Terms of Use to modify the details.

> **Note:**
>
> You cannot edit or modify the name of the Default Terms of Use.

3. Click **Save**.

Remove Terms of Use

You can remove the Terms of Use that you do not need.

To remove a particular Terms of Use, perform the following procedure:.

1. To modify a Terms of Use, under **Security**, click **Terms of Use** and the list of Terms of Use is displayed.
2. Click the name of the required Terms of Use, then click the menu option, and then click **Remove**.

> **Note:**
>
> Alternatively, you can remove a particular Terms of Use by selecting the check box in front of the Terms of Use name, and then click **X Remove**.

Activate and Deactivate Terms of Use

Based on your requirement, you can activate or deactivate a particular Terms of Use.

> **Note:**
>
> Every newly created Terms of Use will be in the deactivated state.

To activate or deactivate a particular Terms of Use, perform the following procedure:.

1. From the navigation drawer, click **Security** and then click **Terms of Use**. A list of Terms of Use is displayed.
2. In the **Terms of Use** screen, perform the following:
   - **Activate**: Select the check box in front of the Terms of Use name, and then click **Activate**.
     Alternatively, you can click the menu option of the particular Terms of Use, and then click **Activate**.
   - **Deactivate**: Select the check box in front of the Terms of Use name, and then click **Deactivate**.
     Alternatively, you can click the menu option of the particular Terms of Use, and then click **Deactivate**.

24

Manage Adaptive Security in Oracle Identity Cloud Service

This section describes how to manage Adaptive Security in Oracle Identity Cloud Service.

Topics:

Typical Workflow for Managing Adaptive Security in Oracle Identity Cloud Service

With the Adaptive Security feature in Oracle Identity Cloud Service, you can perform tasks such as managing Adaptive Security and risk providers.

Task	Description	Additional Information
Understand Adaptive Security and risk providers.	You can learn about Adaptive Security, and how it's used to provide strong authentication capabilities for your users, based on their behavior within Oracle Identity Cloud Service, and across multiple heterogeneous on-premises applications and cloud services. You can also learn why you should use Adaptive Security, and how Adaptive Security uses risk providers to allow administrators to configure various contextual and threat events to be analyzed within Oracle Identity Cloud Service, and also to configure and consume user risk scores from third-party risk providers.	Understand Adaptive Security Why Use Adaptive Security? Understand Risk Providers
Activate and deactivate Adaptive Security.	You can activate and deactivate Adaptive Security using the **Adaptive Security** page.	Activate and Deactivate Adaptive Security
Configure Oracle Identity Cloud Service risk events.	You can modify risk events for the risk provider that's associated with Oracle Identity Cloud Service actions using the **Adaptive Security** page.	Configure the Default Risk Provider
View details about a risk provider.	You can view details about a risk provider using the **Adaptive Security** page.	View Details About a Risk Provider
Add a third-party risk provider.	You can add a third-party risk provider using the **Adaptive Security** page.	Add a Third-Party Risk Provider
Activate and deactivate risk providers.	You can activate and deactivate risk providers using the **Adaptive Security** page.	Activate and Deactivate Risk Providers
Modify third-party risk providers.	You can modify third-party risk providers using the **Adaptive Security** page.	Modify a Third-Party Risk Provider
Remove a third-party risk provider.	You can remove a third-party risk provider using the **Adaptive Security** page.	Remove a Third-Party Risk Provider

You can create, manage, and remove Adaptive Security and risk providers by using:

- The Identity Cloud Service console
- SCIM-based APIs

The following sections describe how to manage Adaptive Security and risk providers by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Understand Adaptive Security

Adaptive Security is an advanced feature that provides strong authentication capabilities for your users, based on their behavior within Oracle Identity Cloud Service, and across multiple heterogeneous on-premises applications and cloud services.

When activated, the Adaptive Security feature can analyze a user's risk profile within Oracle Identity Cloud Service based on their historical behavior, such as too many unsuccessful login attempts and too many unsuccessful MFA attempts, and real-time device context, such as access from unknown devices, impossible travel between locations, and so on. To evaluate the user's behavior across other systems with which Oracle Identity Cloud Service isn't directly involved, Adaptive Security allows you to configure your existing risk providers to obtain the user's risk score from third-party risk providers, such as Symantec CloudSOC Cloud Access Security Broker (CASB). With this enriched context and risk information, Adaptive Security risk profiles each user, and arrives at its own risk score and an overall consolidated risk level (High, Medium, Low) that can be used with Oracle Identity Cloud Service policies to enforce a remediation action, such as allowing or denying the user from accessing Oracle Identity Cloud Service and its protected applications and resources, requiring the user to provide a second factor to authenticate into Oracle Identity Cloud Service, and so on. Administrators can also view how the user's risk profile trended over a period of time, and drill down to see details associated with each event.

Why Use Adaptive Security?

Users are connected increasingly, accessing their accounts and applications from multiple locations, devices, and channels. Implementing overly restrictive and static controls to secure access (for example, prompting a user for a second factor for every authentication or blocking access to a user when the user is out of their base country) would result in a painful user experience with no overall improvements in security.

Adaptive Security can analyze contextual, risk, and threat information about the user, device, and network, and provide an intelligent, secure, and user-friendly way of providing access to corporate applications and resources. This also reduces the likelihood of online identity theft and fraud, which secures your business applications even if the user's device or the user's account password is compromised.

Activate and Deactivate Adaptive Security

You can activate or deactivate the Adaptive Security feature.

- Deactivating Adaptive Security stops Oracle Identity Cloud Service from performing contextual and threat event analytics, and obtaining user risk scores from third-party risk providers.
- Activating Adaptive Security allows Oracle Identity Cloud Service to start evaluating contextual and threat analysis, and obtain user risk scores from the configured third-party risk providers.

Activate Adaptive Security

You can use Oracle Identity Cloud Service to activate Adaptive Security.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Adaptive Security**.

2. In the **Adaptive Security** page, turn on the **Adaptive Security** switch.

Deactivate Adaptive Security

You can use Oracle Identity Cloud Service to deactivate Adaptive Security.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Adaptive Security**.
2. In the **Adaptive Security** page, turn off the **Adaptive Security** switch.

Understand Risk Providers

Adaptive Security uses the concept of risk providers to allow identity domain administrators and security administrators to configure various contextual and threat events to be analyzed within Oracle Identity Cloud Service, and also to configure and consume user risk scores from third-party risk providers.

A default risk provider within Oracle Identity Cloud Service is seeded automatically with a list of supported contextual and threat events, such as too many unsuccessful login attempts, too many unsuccessful MFA attempts, access from unknown devices, access from unfamiliar locations, access from suspicious IP addresses, and impossible travel between locations. Administrators can enable events of interest, and specify weighting or severity for each of these events. The system uses the configured weighting to compute the user's Oracle Identity Cloud Service risk score.

> **Example:**
>
> Consider a user who logs into Oracle Identity Cloud Service using a new device, say a laptop. Because the device is unknown, the system won't recognize the device, and will the trigger the **Access from an unknown login device** event and get the weighting from the configuration. There are six events in the risk provider configuration: **Access from an unknown device**, **Too many unsuccessful login attempts**, **Too many unsuccessful MFA attempts**, **Access from suspicious IP addresses**, **Access from an unfamiliar location**, and **Impossible travel between locations**.
>
> The administrator can assign weighting to these events that correspond to those risk ranges. Consider the weighting for each of the risks as follows: low risk range (0-25), medium risk range (26-75) and high risk range (76-100). If the administrator wants to consider the user login from an unknown device to be of low risk, then the administrator sets the weighting for that event to be less than 25. If the administrator wants to consider the same event to be of medium risk, then the administrator sets the weighting for that event to be between 26 and 75. Any value set above 75 for that event is considered as high risk. If the user hits more than one event, then the risk score will be a combination of two weightings and will correspond to whichever risk level the combination points. The user's risk scores are evaluated continuously and are reduced based on the remediation actions that are taken by the user, such as successful logins and password resets.

Administrators can add additional risk providers to obtain a user's risk score from the Symantec third-party risk engine. This risk engine provides additional intelligence on the user's behavior across heterogeneous systems with which Oracle Identity Cloud Service isn't directly involved.

To provide a consolidated risk profile of the user at any time, Oracle Identity Cloud Service takes the highest level of the risk scores of both the default Oracle Identity Cloud Service risk provider and the configured third-party risk providers, and qualifies the user as a high-risk, medium-risk, or low-risk user. For instance, if a user's risk score from the default risk provider is within the Low range, but the risk score from a third-party risk provider is within the Medium range, then the user's consolidated risk level is set to Medium.

Administrators can then use the Oracle Identity Cloud Service risk score, third-party risk score, or consolidated user risk level as conditions that can be used with Oracle Identity Cloud Service sign-on policies to enforce a remediation action, such as allowing or denying the user from accessing Oracle Identity Cloud Service and its protected applications and resources, requiring the user to provide a second factor to authenticate into Oracle Identity Cloud Service, and so on.

Configure the Default Risk Provider

You can modify the risk provider that's associated with Oracle Identity Cloud Service actions. When this risk provider, known as the default risk provider, is activated, it evaluates the following events that constitute risk-based activity for Oracle Identity Cloud Service users:

- **Access from an unknown device:** If a user accesses Oracle Identity Cloud Service from a device that hasn't been previously used to access the service, then this event (commonly referred to as Device Fingerprinting) is triggered.
- **Too many unsuccessful login attempts:** If the number of unsuccessful login attempts exceed the value specified for the **Account lock threshold** attribute for the password policy, then this event is triggered.

  > **Note:**
  >
  > See Modify the Custom Password Policy to learn how to set the maximum number of unsuccessful logins that the user can attempt in Oracle Identity Cloud Service before they're locked out of their account.

- **Too many unsuccessful MFA attempts:** If the number of unsuccessful login attempts using the factors configured exceed the value specified for the **Max Unsuccessful MFA attempts** attribute for MFA, then this event is triggered.

  > **Note:**
  >
  > See Configure Multi-Factor Authentication Settings to learn how to set the maximum number of unsuccessful MFA logins that the user can attempt in Oracle Identity Cloud Service using their MFA factors before they're locked out of their account.

> **Note:**
>
> If an event is disabled, then Oracle Identity Cloud Service won't use it to generate a risk score that can be used to evaluate risk-based activity for Oracle Identity Cloud Service users. Also, if the default risk provider is deactivated, then the user's risk score won't be increased.

Modifying the default risk provider includes:

- Changing the description of the risk provider.
- Setting the Low, Medium, and High risk range for this risk provider.
- Enabling or disabling the individual events for contextual and threat analytics.
- Setting a value (weighting) for each event that corresponds to the risk range for this risk provider. For example, suppose you set the Low risk range for the risk provider to be from 0-10, the Medium risk range to be from 11-80, and the High risk range to be from 81-100. If you set the weighting of the **Access from an unknown device** event to 20, and a low-risk user accesses Oracle Identity Cloud Service with a device that is previously not used which Oracle Identity Cloud Service doesn't recognize, then the user's risk range will change to Medium.

To modify the default risk provider:

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Adaptive Security**.
2. In the **Adaptive Security** page, click the **Action** menu to the right of the default risk provider.
3. Select **Edit**. The risk provider opens and displays three panes: **Details**, **Risk Range**, and **Events**. See Add a Third-Party Risk Provider for more information about the **Details** and **Risk Range** panes.
4. Change the values that you want to modify in the **Details** and **Risk Range** panes.
5. In the **Events** pane:
   a. Select or deselect a check box to enable or disable the event. By doing so, you're specifying whether Oracle Identity Cloud Service will use this event to generate a risk score that can be used to evaluate risk-based activity for Oracle Identity Cloud Service users.

      > **Note:**
      >
      > If you disable all events for the default risk provider, then you can't save it.

   b. Use the slider to set the weighting for each event to **Low**, **Moderate**, **Severe**, or **Critical**.
6. Click **Save**.
7. In the **Confirmation** window, click **Yes**.

View Details About a Risk Provider

By default, you can see the name, company, and activation status of each risk provider you added to Oracle Identity Cloud Service. You can also see other information, such as the risk levels and authentication information associated with the risk provider.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Adaptive Security**.
2. In the **Adaptive Security** page, click the risk provider about which you want to see more information.

The risk provider opens and displays two panes: **Details** and **Risk Range**. See Add a Third-Party Risk Provider for more information about these panes.

> **Note:**
>
> If you clicked the default risk provider, then you'll see a third pane: **Events**. See Configure the Default Risk Provider to learn more about this pane.

Add a Third-Party Risk Provider

You can add a risk provider to Oracle Identity Cloud Service that can be used to obtain a user's risk score from the Symantec third-party risk engine. This risk score provides additional intelligence on the user's behavior across heterogeneous systems with which Oracle Identity Cloud Service isn't directly involved. Administrators can then use this third-party risk score with Oracle Identity Cloud Service sign-on policies to enforce a remediation action, such as allowing or denying the user from accessing Oracle Identity Cloud Service and its protected applications and resources, requiring the user to provide a second factor to authenticate into Oracle Identity Cloud Service, and so on.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Adaptive Security**. The **Adaptive Security** page appears.

> **Note:**
>
> In the **Adaptive Security** page, Oracle Identity Cloud Service provides you with a default risk provider which can't be deleted. See Configuring the Default Risk Provider for more information about this risk provider.

2. Click **Add**. The **New Risk Provider** page appears.
3. Use the following table to populate the **Details** pane of the **New Risk Provider** page:

Field	Description
Company	Select the vendor of the risk provider solution.
Name	Enter the name of the risk provider.
Description	Provide a brief description of the risk provider.
Endpoint Configuration URL	Enter the risk provider URL that Oracle Identity Cloud Service can use to reach out to obtain the user's risk score.
Authentication Type	This menu contains two methods that Oracle Identity Cloud Service uses to authenticate against the risk provider: **BASIC** and **TOKEN**. If you select **BASIC**, then the **User Name** and **Password** fields appear. Enter the user name and password that Oracle Identity Cloud Service will use to authenticate against the risk provider. If you select **TOKEN**, then the **Scheme** and **Token** fields appear. Enter the name of the authentication scheme and the authentication token that Oracle Identity Cloud Service will use to pass a user's credentials to the risk provider.

Field	Description
User Identifier	Select the unique identifier for user accounts that Oracle Identity Cloud Service will use to link the user in the risk provider. This identifier can be either the user name or the primary email address.
Refresh Rate	Specify how often (in minutes or hours) Oracle Identity Cloud Service will make a call to the risk provider to check for refreshed scores.

4. To check whether the risk provider information is correct, click **Validate**.

   Verify that you see the **The connection to the {risk_provider_name} risk provider has been validated.** message.

> **Note:**
>
> If you receive an error message, then check the values you entered or selected for the **Endpoint Configuration URL** and **Authentication Type** fields.

5. In the **Risk Range** pane of the **Add Risk Provider** page, the risk levels configured in the risk provider will be shown automatically, if the provider supports an API to get this information. If the API is not available, then the administrator can specify the risk ranges manually, as configured in the risk provider. This is just to provide a reference to the configured risk ranges in the risk provider and has no significance in the risk calculations.
6. Click **Save**. The risk provider is added and saved with a deactivated status. See Activate a Risk Provider for more information about activating this risk provider.

Activate and Deactivate Risk Providers

You can activate or deactivate individual risk providers.

> **Note:**
>
> If the default risk provider is deactivated, then none of the events configured in this risk provider will be considered for the user's risk score analysis. Also, if third-party risk providers are deactivated, risk scores will not be fetched from these risk providers.

In addition to enabling and disabling the Adaptive Security feature, you can activate or deactivate one or more risk providers individually.

> **Note:**
>
> A green check mark  indicates an activated risk provider. A red circle with a red line through the circle  indicates a deactivated risk provider.

Topics:

Activate a Risk Provider

You can use Oracle Identity Cloud Service to activate a risk provider.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Adaptive Security**.
2. In the **Adaptive Security** page, click the **Action** menu to the right of the risk provider that you want to activate.
3. Select **Activate**.
4. In the **Confirmation** window, click **OK**. The status of the risk provider changes from deactivated to activated .

Deactivate a Risk Provider

You can use Oracle Identity Cloud Service to deactivate a risk provider.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Adaptive Security**.
2. In the **Adaptive Security** page, click the **Action** menu to the right of the risk provider that you want to deactivate.
3. Select **Deactivate**.
4. In the **Confirmation** window, click **OK**. The status of the risk provider changes from activated to deactivated .

Modify a Third-Party Risk Provider

After viewing details about, activating, or deactivating a risk provider that you added, you can modify it. Modifying this type of risk provider includes:

- Changing the name or description of the risk provider.
- Editing the endpoint configuration URL, authentication type, or authentication credentials of the risk provider.
- Specifying a different unique identifier for user accounts that the risk provider will use to evaluate risk-based activity for the users.
- Changing how often (in minutes or hours) the risk provider will evaluate risk-based activity for users.
- Modifying the Low, Medium, and High risk range for the risk provider.

To modify a third-party risk provider:

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Adaptive Security**.
2. In the **Adaptive Security** page, click the **Action** menu to the right of the risk provider that you want to modify.
3. Select **Edit**. The risk provider opens and displays two panes: **Details** and **Risk Range**. See Add a Third-Party Risk Provider for more information about these panes.
4. Change the values that you want to modify in the **Details** and **Risk Range** panes.
5. Click **Validate**. Verify that you see the **The connection to the {risk_provider_name} risk provider has been validated.** message.

> **Note:**
>
> If you receive an error message, then check the values you changed for the **Endpoint Configuration URL** and **Authentication Type** fields.

6. Click **Save**.
7. In the **Confirmation** window, click **OK**.

Remove a Third-Party Risk Provider

If a third-party risk provider is no longer needed to provide its user risk score to Oracle Identity Cloud Service, then you can remove it.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Adaptive Security**.
2. In the **Adaptive Security** page, if the risk provider that you want to remove is activated, then deactivate it. See Deactivate a Risk Provider.
3. Click the **Action** menu to the right of the risk provider that you want to remove.

> **Note:**
>
> Because the default risk provider is associated with Oracle Identity Cloud Service events, such as whether users access Oracle Identity Cloud Service with devices that aren't registered, or users exceed the number of consecutive, unsuccessful login attempts into Oracle Identity Cloud Service allowed, you can't remove this risk provider.

4. Select **Edit**.
5. Click **Delete**.

25

Manage Oracle Identity Cloud Service Identity Providers

Learn how to manage identity providers for Oracle Identity Cloud Service.

Topics

About Identity Providers

In this topic, you learn about Oracle Identity Cloud Service Identity providers.

An identity provider, also known as an "authentication authority", provides external authentication for users who want to sign into Identity Cloud Service using their external provider’s credentials.

For example, a customer may want its users to log in using their ADFS credentials and gain access to Oracle Cloud Services. In this case, MS ADFS acts as the identity provider and Oracle Identity Cloud Service functions as the service provider. MS ADFS authenticates the user and returns a token containing identity and authentication information to Oracle Identity Cloud Service (for example, the user name and the email address of the user). This security token is digitally signed by the IDP. The SP verifies the signature on the token and then uses the identity information to establish an authenticated session for the user. This is known as federated single sign-on where a user is challenged for credentials in one domain and is granted access to another domain.

About Digital Certificates

Learn what a digital certificate is and what to do when a certificate expires.

What is a Digital Certificate?

A digital certificate is like an electronic passport that helps a person, computer, or organization to exchange information securely over the Internet using public key cryptography. A digital certificate may be referred to as a public key certificate.

Just like a passport, a digital certificate provides identifying information, is forgery resistant, and can be verified because it is issued by an official, trusted agency. The certificate can contain the name of the certificate holder, a serial number, expiration dates, a copy of the certificate holder's public key (used for encrypting messages and verifying digital signatures) and the digital signature of the certificate-issuing authority (CA) so that a recipient can verify that the certificate is real.

In order to verify external identity providers' signatures, stores copies of their signing certificates. When receives a signed message from an identity provider, before the stored certificate is used to verify the signature, the certificate must be verified as valid. Part of validating the certificate is verifying that it has not expired. After the certificate has been validated, the certificate is used to verify the signature on the message.

In order for this operation to succeed, the public key embedded in the certificate must match the private key that the identity provider used to sign the message.

What if an Identity Provider's Certificate Expires?

If an identity provider's signing certificate expires, then certificate validation will fail, and will be unable to complete single sign-on operations for that identity provider's users. Therefore, when an identity provider's certificate nears its expiration date, you must make plans to replace it. The typical process is a follows:

1. Obtain the new signing certificate from the identity provider. This may be published by the identity provider for self-service download, or you may need to contact the identity provider administrator.
2. Load the new signing certificate into the Oracle Identity Cloud Service configuration for the identity provider.
3. If the identity provider has also rolled over its signing private/public key pair (rather than only re-issuing a new certificate for the existing key pair), then the identity provider must be updated to begin using the new keys to sign messages. Again, this may be self-service or require coordination with the identity provider administrator.

> **Note:**
>
> If the identity provider rolls over its signing key pair, then Single Sign-On will fail during the period of time between Step 2 and Step 3 above. For this reason, the certificate update is typically coordinated between the identity provider and Oracle Identity Cloud Service administrators, in order to minimize the downtime, as well as schedule it for a period of low user activity.

Understand SAML Just-In-Time Provisioning

SAML Just-In-Time (JIT) Provisioning automates user account creation when the user first tries to perform SSO and the user doesn't yet exist in Oracle Identity Cloud Service. In addition to automatic user creation, JIT allows granting and revoking group memberships as part of provisioning. JIT can be configured to update provisioned users so the users' attributes in the Service Provider (SP) store can be kept in sync with the Identity Provider (IDP) user store attributes.

Prerequisite

Enable SAML Just-In-Time Provisioning. Oracle must enable this feature for you. To learn about the features that Oracle must enable for you and how to enable them, see Standard License Tier Features for Oracle Identity Cloud Service.

Benefits

The advantages of JIT are:

- The footprint of user accounts in Oracle Identity Cloud Service is limited to those users who actually log in via federated SSO, rather than all users in the Identity Provider user directory.
- Reduced administrative costs as accounts are created on demand as part of the SSO process and the Identity Provider and Service Provider user stores don't have to be synchronized manually.
- Any new users added later to the Identity Provider user store won't require administrators to create corresponding Service Provider accounts manually (users will always be in sync).

How It Works

There are four runtime flows for JIT Provisioning:

When Signing In, The User:	Flow
Exists and JIT Provisioning is enabled.	Normal SSO flow.
Doesn't exist and JIT Provisioning is not enabled.	Normal SSO failure flow.
Doesn't exist and JIT creation is enabled.	User is created, and populated with the SAML assertion attributes, as mapped in the JIT configuration.
Exists and JIT update is enabled.	User attribute values are updated with the SAML assertion attributes, as mapped in the JIT configuration.

SAML JIT Provisioning can be configured only by using the `/admin/v1/IdentityProviders` REST API endpoint. See the following references to configure SAML JIT Provisioning:

- Create an Identity Provider
- Configuring SAML JIT Provisioning

Typical Workflow for Managing Identity Providers

With the identity provider (IDP) feature in Oracle Identity Cloud Service, you can perform tasks such as adding, managing, and using identity providers.

Task	Description	Additional Information
Add an IDP.	Add an IDP to your identity domain using the **Identity Providers** page.	Add a SAML Identity Provider
View details about an IDP.	View details about an IDP using the **Identity Providers** page.	View Details About an Identity Provider
Activate and deactivate an IDP.	Activate and deactivate an IDP using the **Identity Providers** page.	Activate and Deactivate an Identity Provider
Test an IDP.	Test an IDP to verify that you can use federated SSO credentials to log in to Oracle Identity Cloud Service through an external website.	Test an Identity Provider
Modify an IDP.	Modify an IDP using the **Identity Providers** page.	Modify an Identity Provider
Delete an IDP.	Delete an IDP using the **Identity Providers** page.	Delete an Identity Provider

You can add, manage, and use identity providers by using:

- The Identity Cloud Service console
- SCIM-based APIs

In the following sections, you learn how to manage identity providers by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Add a SAML Identity Provider

There are two ways that you can add a SAML 2.0 identity provider (IDP) in Oracle Identity Cloud Service:

- You can import metadata for the IDP. Identity provider metadata summarizes the basic information about data associated with the IDP. This metadata makes finding and working with this data easier. See Import Metadata for a SAML Identity Provider.
- You can enter metadata for the IDP. See Enter Metadata Manually for a SAML Identity Provider.

Oracle Identity Cloud Service provides you with a wizard to add a SAML 2.0 IDP. This wizard contains six panes:

- **Details**: Provide a name, description, and icon for the SAML IDP.

  > **Tip:**
  >
  > Make sure that the file you want to upload adheres to the recommended dimensions and file size before uploading it. See Customize the Interface.

- **Configure**: Configure SSO for the IDP by either importing metadata for it or entering metadata for it.

- **Map**: Map a user's attribute value received from the IDP to a corresponding attribute value for the user in Oracle Identity Cloud Service.

  After providing information in the **Map** pane of the wizard, Oracle Identity Cloud Service adds and deactivates the IDP. You may want to export metadata for the IDP, test it, or activate it. The wizard has the **Export**, **Test**, and **Activate** panes.

- **Export**: Export metadata for Oracle Identity Cloud Service and import this metadata into the IDP. The IDP requires this information to communicate with Oracle Identity Cloud Service for authentication purposes.

  > **Tip:**
  >
  > If the IDP doesn't support importing metadata, then the information for Oracle Identity Cloud Service appears in the **Export** pane. You can enter this metadata into the IDP manually.

  To learn about the other options that can be used to access SAML metadata, see Access SAML Metadata.

- **Test:** Test the configuration settings for the IDP to confirm that the IDP is working properly. You can use the credentials of the IDP to log in to Oracle Identity Cloud Service through an external website.

- **Activate:** Activate the IDP.

To add an IDP, you must be assigned to either the identity domain administrator role or the security administrator role. See Add or Remove a User Account from an Administrator Role.

Import Metadata for a SAML Identity Provider

You can use Oracle Identity Cloud Service to import metadata for a SAML 2.0 identity provider (IDP).

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Identity Providers**.
2. Click **Add SAML IDP**. The **Add Identity Provider** wizard appears.
3. Use the following table to populate the **Details** pane of the wizard, and click **Next**:

Task	Description
Name	Enter the name of the IDP.
Description	Enter explanatory information about the IDP.
Icon	Click **Upload** to add an icon that represents the IDP. The icon should be 48 x 48 pixels in size and have a transparent background. Supported file formats are png, fig, jpg, jpeg.

4. Use the following table to populate the **Configure** pane of the wizard, and click **Next**:

Task	Description
Import Identity Provider metadata	Click this button if you want to configure SSO for the IDP by importing metadata for it.
Metadata	Click **Upload**. Select the XML file that contains the metadata for the IDP that you want to import.

Task	Description
Signature Hashing Algorithm	Select the SHA-1 or SHA-256 hash algorithm to use when signing SAML messages to the Identity Provider.
Include Signing Certificate	To include the Oracle Identity Cloud Service signing certificate with signed SAML messages sent to the IDP, select this check box. If you don't want to include a signing certificate with your signed SAML messages, then leave the check box deselected.

5. Use the following table to populate the **Map** pane of the wizard, and click **Next**:

Task	Description
Identity Provider User Attribute	Select the element in the SAML assertion received from the IDP, where the unique user identifier will be found. • If you select Name ID, then Oracle Identity Cloud Service will match the user based on the value of the Subject NameID element in the assertion. • If you select SAML Attribute, then you must enter the name of an Attribute element in the SAML assertion. The user will be matched based on the value of that attribute.
Oracle Identity Cloud Service User Attribute	Select the user identity attribute in Oracle Identity Cloud Service that will be matched with the user identity attribute received in the SAML assertion from the IDP.
Requested NameID Format	Select the NameID format that Oracle Identity Cloud Service will specify in SAML authentication requests sent to the Identity Provider. If you don't want to provide a format, then select **<None Requested>**.

6. Use the following table to export the Oracle Identity Cloud Service SAML configuration details, and click **Next**:

Task	Description
Service Provider Metadata	To export metadata for Oracle Identity Cloud Service, click **Download**. Use this XML metadata to configure the Identity Provider service. If the Federation Partner into which you are importing Identity Cloud Service metadata does CRL validation (for example ADFS does CRL validation) instead of using the metadata exported from this button, download the metadata from: `https://[instancename.idcs.internal.oracle.com:port]/fed/v1/metadata?adfsmode=true` To learn about the other options that can be used to access SAML metadata, see Access SAML Metadata.

Task	Description
Provider ID	The URI that uniquely identifies the Oracle Identity Cloud Service identity domain as a SAML provider. (Provider ID is also known as Issuer ID or Entity ID.)
Assertion Consumer Service URL	The URL of the Oracle Identity Cloud Service SAML service to which the IDP will send SAML assertions.
Logout Service Endpoint URL	The URL of the Oracle Identity Cloud Service SAML service to which the IDP will send SAML logout requests.
Logout Service Return URL	The URL of the Oracle Identity Cloud Service SAML service to which the IDP will send SAML logout responses, after the Oracle SAML provider has sent it a SAML logout request.
Service Provider Signing Certificate	Click **Download** to retrieve the signing certificate of the Oracle Identity Cloud Service SAML provider. This certificate is used by the IDP to verify SAML requests and responses signed by Oracle Identity Cloud Service.
Service Provider Encryption Certificate	Click **Download** to retrieve the encryption certificate of the Oracle Identity Cloud Service SAML provider. This certificate can be used by the IDP to encrypt SAML assertions sent to Oracle Identity Cloud Service.

To get the issuing Oracle Identity Cloud Service root certificate, see Obtain the Root CA Certificate from Oracle Identity Cloud Service.

7. In the **Test** pane of the wizard, click **Test Login** to test the configuration settings for the IDP.
8. Click **Next**.
9. In the **Activate** pane of the wizard, click **Activate** to activate the IDP.
10. Click **Finish**.

Enter Metadata Manually for a SAML Identity Provider

You can use Oracle Identity Cloud Service to enter metadata for a SAML 2.0 identity provider (IDP).

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Identity Providers**.
2. Click **Add SAML IDP**.
3. Populate the **Details** pane of the **Add Identity Provider** wizard and click **Next**. See the table in Import Metadata for a SAML Identity Provider for more information about the options.
4. Use the following table to populate the **Configure** pane of the wizard, and click **Next**:

Field	Description
Enter Identity Provider metadata manually	Click this button if you want to configure SSO for the IDP by entering metadata for it.

Field	Description
Issuer ID	Enter the URI that identifies the Identity Provider in SAML messages. (Issuer ID is also known as Entity ID or Provider ID.)
Signing Certificate	To upload the Identity Provider's signing certificate, click **Upload** and select the file that contains the certificate.
SSO Service URL	Enter the URL of the Identity Provider's SAML SSO service, to which Oracle Identity Cloud Service will send SAML authentication requests.
SSO Service Binding	This menu contains two options for web-based SSO associated with the IDP: **Redirect** and **POST**. • Select **Redirect** to send SAML authentication requests to the IDP using the HTTP-Redirect binding. • Select **POST** to send SAML authentication requests to the IDP using the HTTP-POST binding.
Global Logout Activated	To activate SAML global logouts between Oracle Identity Cloud Service and the IDP, select this check box. Otherwise, leave the check box deselected. If you select the check box, then you must enter values for two URLs for the IDP: logout request and logout response, and specify whether you want Oracle Identity Cloud Service to initiate a logout with a HTTP-Redirect or HTTP-POST binding.
Logout Request URL	Enter the IDP service endpoint URL to which Oracle Identity Cloud Service will send SAML logout requests.
Logout Response URL	Enter the IDP service endpoint URL to which Oracle Identity Cloud Service will send SAML logout responses, after receiving a logout request from the IDP.
Logout Binding	This menu contains two options to initiate a logout: **Redirect** and **POST**. • To initiate a logout with the HTTP-Redirect binding, select **Redirect**. • To initiate a logout using the HTTP-POST binding, select **POST**.
Signature Hashing Algorithm	Select the SHA-1 or SHA-256 hash algorithm to use when signing SAML messages to the Identity Provider.
Include Signing Certificate	To include the Oracle Identity Cloud Service signing certificate with signed SAML messages sent to the IDP, select this check box. If you don't want to include a signing certificate with your signed SAML messages, then leave the check box deselected.

5. Populate the **Map** pane of the **Add Identity Provider** wizard, and click **Next**. See the table in Import Metadata for a SAML Identity Provider for more information about the options.
6. Export the Oracle Identity Cloud Service SAML configuration details, and click **Next**.

See the table in Import Metadata for a SAML Identity Provider for more information about the options.

To learn about the other options that can be used to access SAML metadata, see Access SAML Metadata.

7. In the **Test** pane of the wizard, click **Test Login** to test the configuration settings for the IDP.
8. Click **Next**.
9. In the **Activate** pane of the wizard, click **Activate** to activate the IDP.
10. Click **Finish**.

Add a Social Identity Provider

Administrators can add a social identity provider so that users can log in to Oracle Identity Cloud Service with their social credentials. Administrators can also allow users to self-register in Oracle Identity Cloud Service if they do not already have an account.

If users don't already have accounts in Oracle Identity Cloud Service, administrators can create an account by using a registration page.

You can configure the social identity provider that you're adding so that users can link to their social accounts manually. You can also prevent users from linking to their social accounts for security or organizational purposes. For example, if a hacker accesses the user's social account, the hacker can't sign in to Oracle Identity Cloud Service to access resources and applications that are protected by Oracle Identity Cloud Service. Or, you may want users to have separate profiles for their social accounts and Oracle Identity Cloud Service user accounts.

When adding an instance of a social identity provider, you can choose from any of the following predefined social identity provider types:

- Facebook
- Google
- LinkedIn
- Microsoft
- OpenID Connect
- Twitter

You can add an instance of an out-of-the-box social identity provider type by using either the Identity Cloud Service console or SCIM-based APIs. In this section, you learn how to add a social identity provider from a predefined type by using the Identity Cloud Service console. For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

If you don't see the social identity provider type for which you want to add an instance, then you can use SCIM-based APIs to create your own type and customize an icon for it. Through the API mechanism, you define the attributes for the social identity provider type, and then populate these attributes with values when you add an instance.

For example, you can define attributes for a custom social identity provider type that will enable it to retrieve an access token and user information from the social identity provider. When you add an instance of this social identity provider type, you provide the URLs that the social identity provider needs to retrieve this information.

You can also customize social identity provider types for particular identity domains. Suppose you have users in the United States accessing Oracle Identity Cloud Service from one identity domain, and users from India signing in to Oracle Identity Cloud Service from another identity domain. You want only the India-based users to be able to access Oracle Identity Cloud Service with their GitHub social credentials. So, you can customize a GitHub social identity provider type for the India identity domain only.

To remove a social identity provider type and the metadata associated with it cleanly and completely, first, remove the social identity provider type, and then, remove its metadata. Also, if you create a social identity provider type, add an instance of this social identity provider, and assign the instance to an identity provider policy, then don't update or remove the metadata associated with the social identity provider type. If you want to update or remove the metadata, then first remove the social identity provider type from the identity provider policy.

A social identity provider uses an access token to access a resource that's protected by Oracle Identity Cloud Service. This type of token has an expiration date and time. When the access token expires, a refresh token is used to obtain a renewed access token. Unlike access tokens, refresh tokens never expire.

For some custom social identity provider types (for example, Adobe e-Sign), separate URLs have to be provided for the access token endpoint and the refresh token endpoint. When this occurs, you must specify different URLs.

For more information about how to customize a social identity provider type, or to learn how to provide different URLs for the access token and refresh token endpoints, see REST API for Oracle Identity Cloud Service.

Some cloud services have applications that may have to connect to multiple instances of the same social identity provider. For example, for application A and application B, the Facebook social identity provider can be configured as an identity provider along with distinct configuration settings, such as a Client ID and Secret, social registration settings, and so on. To support such scenarios, Oracle Identity Cloud Service enables you to add multiple instances of the same social identity provider with different configuration settings for each instance.

After adding multiple instances of a social identity provider, you can choose which instances can be used to sign in to Oracle Identity Cloud Service by using an identity provider policy.

Prerequisites:

1. Create an application for the social identity provider; for example, go to the Google developer site to create a Google application.
2. Configure the `redirectUrl` in the application created in Step 1. The `redirectUrl` must have the format: `https://<IDCS tenant base URL>/oauth2/v1/social/callback`.

> **Note:**
>
> For social identity providers created before release 22.1.49, ensure that the `redirectUrl` doesn't contain port number `:443`. If it does, update the existing URL to remove the port number or add a new URL without the port number to the identity provider application using the external provider developers' website.
>
> For example, if your configuration looks like the following:
>
> `https://<IDCS tenant base URL>:443/oauth2/v1/social/callback`
>
> change it to:
>
> `https://<IDCS tenant base URL>/oauth2/v1/social/callback`.

At the time of this printing, each social identity provider calls these URLs by a different name. See the following list of the social identity providers and the names that they use for the URLs.

- Facebook: **Valid OAuth redirect URIs**
- Google and LinkedIn: **Authorized redirect URL**
- Microsoft: **Redirect URLs**
- Twitter: **Callback URL**

3. Ensure that you retain the `Client ID` and the `Client Secret` from the application that you created at the social identity provider. You use this ID and Secret when configuring a social identity provider in Oracle Identity Cloud Service.

To add a social identity provider:

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Identity Providers**.
2. Click **Add Social IDP**.
3. Choose a social login type.
4. In the **Name** and **Description** fields, enter a name and description for the social identity provider, and then click **Next**.

> **Note:**
>
> The social identity provider name can contain spaces. However, it can't contain special characters.

5. (Optional) For social login type OpenID Connect, upload an application icon, and then click **Next**.
6. Enter the **Client ID** and the **Client Secret** for the social login type.
7. For the **OpenID Connect** social login type, enter the **Discovery Service URL**.

   The discovery service URL is used to get authentication endpoints (URLs) to authenticate users for the social login type.
8. Set the **Account Linking** option.
   - To allow users to link to their social accounts, turn on this option.

- To prevent users from linking to their social accounts, turn off this option.

> **Note:**
>
> You can prevent users from linking to their social accounts for security or organizational purposes. For example, if a hacker accesses the user's social account, the hacker can't sign in to Oracle Identity Cloud Service to access resources and applications that are protected by Oracle Identity Cloud Service. Or, the administrator may want users to have separate profiles for their social accounts and Oracle Identity Cloud Service accounts.

9. Set the **Enable Registration** option.
   - To allow users to register their social identities with Oracle Identity Cloud Service, turn on this option.
   - To prevent users from registering their social identities with Oracle Identity Cloud Service, turn off this option.
10. Click **Finish**.
11. Locate the social identity provider that you created and use the **Action** menu to activate the social identity provider.
12. (Optional) Using the **Action** menu, click **Edit** and turn on **Enable Registration**.

> **Note:**
>
> After you add and activate the identity provider, you must add it to an identity provider policy. By doing so, it will appear in the **Sign In** page and can be used by a user who's trying to sign in to Oracle Identity Cloud Service, either when they're accessing a specific app or attempting to access resources that are protected by Oracle Identity Cloud Service, such as the My Profile console or the Identity Cloud Service console. See Add an Identity Provider Policy.
>
> If you no longer want to display the identity provider in the **Sign In** page, then remove the identity provider from all identity provider policies and deactivate the identity provider. See Remove Identity Providers from the Policy and Deactivate an Identity Provider.

> **Note:**
>
> User social identity profile information auto-populates the Oracle Identity Cloud Service registration page only if profile information exists in the user’s social identity profile. For example, if a user’s Twitter profile has only a Twitter handle and not a first name or last name, the user has to enter a first and last name on the Oracle Identity Cloud Service registration page to create an account.

13. Click **Save**.
14. Log in with the social identity provider.

> **Note:**
>
> You might encounter this error: “Not Logged In: You are not logged in. Please log in and try again.”
>
> The most likely cause is that the application you created on the social identity provider side has the wrong Client ID or Redirect URL in the configuration. Check the Client ID and the Redirect URL configuration, and try to log in again.

Add an X.509 Authenticated Identity Provider

Adding an X.509 authenticated identity provider allows users to login using two-way SSL.

Two-way SSL ensures that both the client and the server authenticate each other by sharing their public certificates and then verification is performed based on those certificates.

Prerequisites

- Enable X.509 certificate validation. See Enable X.509 Certificate Authentication.
- Import a trusted partner certificate. See Import a Trusted Partner Certificate.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Identity Providers**.
2. Click **Add X509 IDP**.
3. Select the **Signing Certificate Aliases**.
4. Choose a **Matching Attribute Type**.
   - **Default Filter**: Use the default filter to associate Oracle Identity Cloud Service user attributes to certificate attributes.
   - **Simple Filter**: Use the simple filter to select an Oracle Identity Cloud Service user attribute to associate it to a certificate attribute.
   - **Advanced Filter**: Use the advanced filter to create a custom filter to associate Oracle Identity Cloud Service user attributes to certificate attributes. For example, you can use username eq “(assertion.subject.cn)” or emails.primary sw “(assertion.serialNumber)”.
5. Click **Save**.

View Details About an Identity Provider

By default, you can see the name of each identity provider you added to Oracle Identity Cloud Service.

You can also see other information about the identity provider, such as its configuration settings.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Identity Providers**.
2. In the **Identity Providers** page, click the **Action** menu to the right of the identity provider about which you want to see more information.
3. Select **Edit**.

Activate and Deactivate an Identity Provider

You can use Oracle Identity Cloud Service to activate and deactivate an identity provider.

Deactivating an identity provider prevents users from being able to use the identity provider to access their Oracle Cloud services externally from a different login page than the one associated with their local Oracle Cloud account.

Activating an identity provider reinstates users to use the identity provider.

After you activate an identity provider, you can assign the identity provider to an identity provider policy. An identity provider policy allows you to define criteria that Oracle Identity Cloud Service uses to determine whether the identity provider appears for users on the **Sign In** page, either when they're accessing a specific app or attempting to access resources that are protected by Oracle Identity Cloud Service, such as the My Profile console or the Identity Cloud Service console.

See Understand Identity Provider Policies for more information about identity provider policies, and Add an Identity Provider Policy to learn more about assigning identity providers to an identity provider policy.

Activate an Identity Provider

You can use Oracle Identity Cloud Service to activate an identity provider.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Identity Providers**.
2. In the **Identity Providers** page, click the **Action** menu to the right of the identity provider that you want to activate.
3. Select **Activate**.
4. In the **Confirmation** window, click **OK**.

Deactivate an Identity Provider

You can use Oracle Identity Cloud Service to deactivate an identity provider.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Identity Providers**.
2. In the **Identity Providers** page, click the **Action** menu to the right of the identity provider that you want to deactivate.
3. Select **Deactivate**.

Test an Identity Provider

After adding and activating an identity provider, you can test it. You can verify that you can use your federated SSO credentials to log in to Oracle Identity Cloud Service through an external website.

1. If you assigned the identity provider to an identity provider policy, then go to step 2. Otherwise, assign the identity provider to an identity provider policy. See Assign Identity Providers to the Policy.

2. Log out of Oracle Identity Cloud Service.
3. In the **Sign In** page, verify that you see a link called **<Identity_Provider_Name>**.

   The **<Identity_Provider_Name>** placeholder represents the name you entered for the identity provider that you created.

   If, for example, you created an identity provider called Google, then the link appears as **Google**.

   See Add a SAML Identity Provider.
4. Click the **<Identity_Provider_Name>** link.
5. Log in to the external website with your federated SSO credentials.

   The identity provider evaluates the user's login credentials, verifies that the user is an authorized user, and returns this information to Oracle Identity Cloud Service. The user can access Oracle Identity Cloud Service.

> **Tip:**
>
> If you no longer want to display the link to the identity provider in the Sign In page, then remove the identity provider from all identity provider policies and deactivate the identity provider. See Remove Identity Providers from the Policy and Deactivate an Identity Provider.

Modify an Identity Provider

After viewing details about, activating or deactivating, and testing an identity provider (IDP), you can modify it.

See Import Metadata for a SAML Identity Provider and Enter Metadata Manually for a SAML Identity Provider.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Identity Providers**.
2. In the **Identity Providers** page, click the **Action** menu to the right of the IDP that you want to modify.
3. Select **Edit**.

   A window that displays configuration settings for the IDP opens.
4. Click **Edit**.
5. Modify the configuration settings for the IDP.
6. After editing the configuration settings for the IDP, click **Save**.

Delete an Identity Provider

You can use Oracle Identity Cloud Service to remove an identity provider.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Identity Providers**.
2. In the **Identity Providers** page, if the identity provider that you want to remove is assigned to an identity provider policy, then remove it from the policy. See Remove Identity Providers from the Policy.

3. Click the **Action** menu  to the right of the identity provider that you want to remove.
4. Select **Edit**.

   A window that displays configuration settings for the identity provider opens.
5. Click **Delete**.

> **Note:**
>
> Deleting a *social identity provider* removes the *user profiles* that are linked to *that social identity provider*. Alternatively, consider deactivating the social identity provider (which does not remove the user profiles) so that users can still see the accounts in **My Profile** but can't use them to log in.

6. In the **Confirmation** window, click **OK**.

26

Manage Oracle Identity Cloud Service Identity Provider Policies

This section describes how to manage Oracle Identity Cloud Service identity provider policies.

Topics:

Typical Workflow for Managing Oracle Identity Cloud Service Identity Provider Policies

With the identity provider policy feature in Oracle Identity Cloud Service, you can perform tasks such as creating, managing, and removing identity provider policies.

Task	Description	Additional Information
Understand identity provider policies.	You can learn about identity provider policies, including how they are used to restrict which identity providers appear in the **Sign In** page.	Understand Identity Provider Policies
Add an identity provider policy.	You can add an identity provider policy using the **Identity Provider Policies** page.	Add an Identity Provider Policy
View details about an identity provider policy.	You can view details about an identity provider policy using the **Identity Provider Policies** page.	View Details About an Identity Provider Policy
Modify an identity provider policy.	You can modify an identity provider policy using the **Identity Provider Policies** page.	Modify an Identity Provider Policy
Remove identity provider policies.	You can remove identity provider policies using the **Identity Provider Policies** page.	Remove Identity Provider Policies

You can create, manage, and remove identity provider policies by:

- The Identity Cloud Service console
- SCIM-based APIs

The following sections describe how to manage identity provider policies by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Understand Identity Provider Policies

An identity provider policy allows identity domain administrators, security administrators, and application administrators to define which identity providers are visible in the **Sign In** page either when they're accessing a specific app or attempting to access resources that are protected by Oracle Identity Cloud Service.

Oracle Identity Cloud Service also uses identity provider policies to determine whether users authenticate into Oracle Identity Cloud Service through identity providers or with a local authentication factor.

The following type of identity providers available with Oracle Identity Cloud Service:

- SAML identity provider: This type of identity provider supports the SAML 2.0 (Security Assertion Markup Language 2.0) standard. You use a SAML identity provider when you want to establish trust between an SAML-compatible identity provider such as Active Directory Federation Services so that users in your organization can access resources protected by Oracle Identity Cloud Service.

  If you want your users to be redirected to a specific SAML identity provider automatically so that they can access an app, then ensure that the identity provider policy associated with the app has only the SAML identity provider assigned to it. If multiple identity providers are assigned to the identity provider policy, then users will be prompted to select one of the identity providers from the **Sign In** page.
- Social identity provider: By linking an Oracle Identity Cloud Service user account to a user's social accounts, the user can access Oracle Identity Cloud Service using their social credentials, such as Facebook, Google, Linkedin, Microsoft, and Twitter.
- Passwordless authentication provider: Allows users to bypass the standard web-form-based authentication. Passwordless authentication allows access to the protected resource without the need for entering the user name and password every time. However, the first time **Sign In** uses the standard login form.
- Local identity provider (Local IDP): Authentication into Oracle Identity Cloud Service happens locally by the user providing their credentials (user name and password) in the **Sign In** page.

In addition to the identity providers, Oracle Identity Cloud Service comes equipped with the following out-of-the-box local authentication factors. To make these local authentication factors available so that you can assign them to identity provider policies, you must first enable them. See Configure Authentication Factors.

- **Email:** Oracle Identity Cloud Service sends a one-time passcode to the user’s primary email address for use as a verification method to authenticate into Oracle Identity Cloud Service. The user’s primary email address is defined in the user’s Oracle Identity Cloud Service account.
- **Mobile App Notification:** Oracle Identity Cloud Service sends a push notification that contains an approval request to allow or deny a login attempt. Push notifications are an easy and quick way to authenticate. After the user enters their user name and password, a login request is sent to the app on their phone. The user taps **Allow** to authenticate.
- **Mobile App Passcode:** An authenticator app such as the Oracle Mobile Authenticator (OMA) app generates a one-time passcode. This passcode can be generated even when

the user's device is offline. After the user enters their user name and password, a prompt appears for the passcode. The user obtains a generated passcode from the app, and then enters the code to access Oracle Identity Cloud Service.

- **Text Message:** After the user enters their user name and password, Oracle Identity Cloud Service sends a passcode as a text message to the user's device. The user enters the code to access Oracle Identity Cloud Service.
- **User Name-Password**: The user authenticates into Oracle Identity Cloud Service by user providing their credentials (user name and password) in the **Sign In** page.

The identity provider policy allows you to configure whether local authentication is displayed in the **Sign In** page for the user.

Suppose you've created several social identity providers and SAML identity providers, and you want to configure which of these identity providers will appear in the **Sign In** page when the user attempts to authenticate into Oracle Identity Cloud Service using a particular app. Without identity provider policies, you couldn't configure this. So, if you had all these SAML and social identity providers activated and set to appear in the **Sign In** page, they would all be displayed.

Oracle Identity Cloud Service provides you with a default identity provider policy that contains a default identity provider rule. This rule has the **User Name-Password** local authentication factor assigned to it. This way, at the bare minimum, users can authenticate into Oracle Identity Cloud Service with their user names and passwords. However, you can build upon this default policy by adding other identity provider rules to it. By adding these rules, you can prevent some of your identity providers from being available to users to authenticate into Oracle Identity Cloud Service. Or you can allow other identity providers to be available only to those users who access Oracle Identity Cloud Service from an IP address contained in one of your network perimeters. Both the My Profile console and the Identity Cloud Service console use the identity provider rules that are assigned to the default identity provider policy.

Suppose your company has restrictions as to who can sign in to Oracle Identity Cloud Service locally (by supplying their user names and passwords) and who must use an external identity provider. Company employees must authenticate into Oracle Identity Cloud Service by using Active Directory Federation Services (AD FS), contractors and partners must use their own SAML identity providers, and all other users (such as consumers) can use either their user names and passwords or a social identity provider.

To accomplish this goal, you can create two identity provider rules for the default identity provider policy. The first rule is applicable only to company employees. These users must sign in Oracle Identity Cloud Service with AD FS which is the employee identity provider. The second rule is applicable only to those users that have user names that end with @partner1.com. They can sign in with the partner 1 SAML identity provider. The third rule is applicable only to those users that have user names that end with @partner2.com. They can sign in with the partner 2 SAML identity provider. The final rule is a catch-all rule for all users (consumers). These users can use either their Oracle Identity Cloud Service passwords or a social identity provider.

Because you can define multiple identity provider rules for an identity provider policy, Oracle Identity Cloud Service must know the order in which the rules are to be evaluated. To do this, you can set the priority of the rules. For the example above, you can have the company employee rule evaluated first. If a user meets the criteria of this rule (that is, the user is an employee), then the user must authenticate into Oracle Identity Cloud Service with AD FS. Users who aren't employees don't meet the criteria of this identity provider rule, and so, the rule with the next highest priority is evaluated. For this example, this is the partner 1 rule where the user's user name must end with @partner1.com. These users must access Oracle Identity Cloud Service by using the partner 1 SAML identity provider. For users who aren't employees or who don't have user names that end with partner1@com, Oracle Identity Cloud Service evaluates the rule with the next highest priority number: the partner 2 rule where the user's

user name must end with @partner2.com. These users must use the partner 2 SAML identity provider to authenticate into Oracle Identity Cloud Service. All other users meet the criteria of the catch-all rule so they can use either their Oracle Identity Cloud Service passwords or a social identity provider to sign-in to Oracle Identity Cloud Service.

In addition to the default identity provider policy, you can create identity provider policies and associate them with specific apps. Suppose you have multiple apps and you want to assign different identity providers to each app. For example, you may have two apps, and you want users to authenticate into Oracle Identity Cloud Service from Facebook or Linkedin. So, you can have one identity provider policy specifically for one app and the Facebook social identity provider, and another identity provider policy exclusively for the second app and the Linkedin social identity provider.

Oracle Identity Cloud Service displays a maximum of four identity providers on the **Sign In** page. If you assign more than four identity providers to an identity provider policy, then a **View all** link appears on the page. Click the link and all identity providers associated with the policy appear.

Add an Identity Provider Policy

Oracle Identity Cloud Service provides you with a wizard to add an identity provider policy. As a result, you define criteria that Oracle Identity Cloud Service uses to determine which identity providers are available for users to authenticate against Oracle Identity Cloud Service when they're accessing particular apps.

Criteria that you can define for an identity provider policy include:

- The user name of the user
- The IP address that the user is using to sign in to Oracle Identity Cloud Service
- The identity providers that will be available to the user to access Oracle Identity Cloud Service

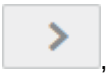
This wizard contains the following panes:

- **Details:** Provide the name and description for the policy.
- **Identity Provider Rules:** Assign or remove identity providers for this policy.
- **Apps:** Assign or remove apps for this policy.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, select **Security**, **IDP Policies**.

> **Tip:**
>
> In the **Identity Provider Policies** page, Oracle Identity Cloud Service provides you with a default identity provider policy. See Understand Identity Provider Policies for more information about this policy.

2. On the **Identity Provider Policies** page, click **Add**.
3. On the **Add Identity Provider Policy** wizard, **Details** pane, enter the name of the policy in the **Policy Name** field, then, click **Next**  .

After providing information in the **Details** pane and clicking **Next**  , Oracle Identity Cloud Service adds the identity provider policy.

You may want to assign or remove identity providers or apps for this policy. To do this, the wizard has the **Identity Provider Rules** and **Apps** panes.

4. In the **Identity Provider Rules** pane of the wizard, click **Add Rules** to assign identity providers to this policy.
5. Use the following table to populate the **Add Rules** dialog box:

Option	Description
Rule Name:	Enter the name of the identity provider rule.
If the user name:	Specify information about users' user names that Oracle Identity Cloud Service will use to determine whether users will meet the criteria of the rule. For example, if you want the rule to be applicable only to those users that have user names that end with `@example.com` , then select **Ends With** from the drop-down menu, and enter **@example.com** in the associated text field.
And is not one of these users:	Enter or select the users that will be excluded from the rule.
And the user's client IP address is:	There are two options associated with this field: **Anywhere** and **In one or more of these network perimeters**. If you select **Anywhere**, then the identity providers that you specify in this rule will be available to users that log in to Oracle Identity Cloud Service using any IP address. If you select **In one or more of these network perimeters**, then a text area appears. In this text area, you can enter or select network perimeters that you defined in Oracle Identity Cloud Service. See Add a Network Perimeter. The identity providers that you specify in this rule will be available to users that log in to Oracle Identity Cloud Service using only IP addresses that are contained in the defined network perimeters.
Assign Identity Providers:	Select the identity providers and local authentication factors that will be available to users to sign in to Oracle Identity Cloud Service if they meet the criteria of this rule.

Option	Description
	Note: In addition to the identity providers that you create in Oracle Identity Cloud Service, there are predefined local authentication factors available to you to assign to this identity provider rule. To use local authentication factors, you must first turn on the Enable User Name First switch in the Session Settings page, and then select the factors that you want to enable in the Multi-Factor Authentication (MFA) Settings page. See Understand Identity Provider Policies for more information about these authentication factors. See Change Session Settings and Configure Multi-Factor Authentication Settings. **Note:** You may have added incorrect identity provider rules to this policy inadvertently. If so, then you can remove them by clicking the "X" in the label for the rule in the **Assign Identity Providers** box.

6. Click **Save**.
7. To add another identity provider rule to this policy, repeat step 5 above.

   Note: If you have added multiple identity provider rules to this policy, then you can change the order that will Oracle Identity Cloud Service evaluate them. See Change the Priority of an Identity Provider Rule for the Policy.

8. When you are finished adding identity provider rules, click **Next**  .
9. In the **Apps** pane of the wizard, click **Assign** to assign apps to this policy.
10. In the **Assign Apps** dialog box, select the check box for each app that you want to assign to the policy, then, click **OK**.

> **Note:**
>
> You can assign only one identity provider policy to an app. If the app isn't assigned to any identity provider policy explicitly, then the default identity provider policy applies to the app.
>
> You can remove apps from the policy by selecting the check box for each app that you want to remove, clicking **Remove**, and then clicking **OK** from the confirmation window.

11. Click **Finish**.

View Details About an Identity Provider Policy

Learn about the details of an Identity Provider.

By default, you can see the name, description, and activation status of each sign-on policy you added to Oracle Identity Cloud Service. You can also see other information about the policy, such as the rules added to the policy and any apps assigned to the policy.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **IDP Policies**.
2. In the **Identity Provider Policies** page, click the **Action** menu to the right of the identity provider policy about which you want to see more information, and then select **Edit**.

   The policy opens and displays three tabs: **Details**, **Identity Providers**, and **Apps**.
3. To view high-level information about the identity provider policy, such as the policy name, click the **Details** tab.
4. To view applications assigned to the policy, click the **Apps** tab.
5. To view identity providers assigned to the policy:, click **Identity Provider Rules** or **Apps**, respectively.

   a. Click the **Action** menu for a rule and select **Edit**.

   b. In the **Edit...** dialog box, scroll down to the **Allowed Identity Providers** section.

   c. View assigned identity providers in the **Assign Identity Providers** box.

Modify an Identity Provider Policy

After viewing details about an identity provider policy, you can modify it.

Modifying an identity provider policy in Oracle Identity Cloud Service includes:

- Changing the name of the policy
- Adding, changing the priority of, editing, and removing identity provider rules for the policy
- Assigning identity providers and apps to the policy
- Removing identity providers and apps from the policy

To modify an identity provider policy:

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **IDP Policies**.
2. In the **Identity Provider Policies** page, click the identity provider policy that you want to modify.

   The policy opens and displays three tabs: **Details**, **Identity Provider Rules**, and **Apps**. See View Details About an Identity Provider Policy for more information about these tabs.

Change the Policy Name

You can change the name of an identity provider policy.

1. Click the **Details** tab.
2. In the **Policy Name** field, enter the new name of the identity provider policy.
3. Click **Save**.

Assign Identity Providers to the Policy

You can assign identity providers to an identity provider policy. These identity providers will appear in the **Sign In** page, and a user can use them to access resources that are protected by Oracle Identity Cloud Service, such as the My Profile console or the Identity Cloud Service console.

1. Click the **Identity Provider Rules** tab.
2. Click the **Action** menu for the rule to which you want to assign an identity provider, then select **Edit**.
3. In the **Edit...** dialog box, scroll down to the **Allowed Identity Providers** section.
4. Click in the **Assign Identity Providers** box and select the identity provider that you want to assign to this rule.

   Repeat this step to assign additional rules.
5. Click **Save**.

Remove Identity Providers from the Policy

You can remove identity providers from an identity provider policy. These identity providers will no longer appear in the **Sign In** page, and a user can't use them to access Oracle Identity Cloud Service-protected resources, such as the My Profile console or the Identity Cloud Service console.

1. Click the **Identity Provider Rules** tab.
2. Click the **Action** menu for the rule to which you want to assign an identity provider, then select **Edit**.
3. In the **Edit...** dialog box, scroll down to the **Allowed Identity Providers**
4. In the **Assign Identity Providers box**, click the "X" in the label for each identity provider that you want to remove from this rule.
5. Click **Save**.

Assign Apps to the Policy

You can assign apps to an identity provider policy. When a user attempts to authenticate into Oracle Identity Cloud Service through the apps, the only identity providers that appear in the **Sign In** page are the ones you assigned to the policy.

1. Click the **Apps** tab.
2. Click **Assign**.
3. In the **Assign Apps** window, select the check box for each app that you want to assign to the policy. Then, click **OK**.

Remove Apps from the Policy

You can remove apps from an identity provider policy. A user who uses these apps can no longer authenticate into Oracle Identity Cloud Service by using the identity providers assigned to the policy.

1. Click the **Apps** tab.
2. Select the check box for each app that you want to remove from the policy.
3. Click **Remove**.
4. In the **Confirmation** window, click **OK**.

Add Identity Provider Rules to the Policy

You can add identity provider rules to an identity provider policy.

By adding add identity provider rules rules, you can prevent some of your identity providers from being available to users to authenticate into Oracle Identity Cloud Service. Or you can allow other identity providers to be available only to those users who access Oracle Identity Cloud Service from an IP address contained in one of your network perimeters.

1. Click the Identity Provider Rules tab.
2. Click **Add**.
3. Add identity provider rules to the policy.

Change the Priority of an Identity Provider Rule for the Policy

You can change the priority of an identity provider rule for an identity provider policy to change the order that Oracle Identity Cloud Service will evaluate it.

1. Click the Identity Provider Rules tab.
2. Click the ellipsis button to the left of the identity provider rule for which you want to change the priority.
3. Drag-and-drop this rule to change the order that Oracle Identity Cloud Service will evaluate it.

   For example, if your identity provider rule has a priority of 4, and you want Oracle Identity Cloud Service to evaluate it first, drag the rule and drop it so that it appears directly above the identity provider rule with a priority of 1. Your rule will appear first in the list, and the other rule will now have a priority of 2.

4. Click **Save**.

Remove Identity Provider Policies

You can use Oracle Identity Cloud Service to remove multiple identity provider policies simultaneously.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **IDP Policies**.
2. In the **Identity Provider Policies** page, if the policy that you want to remove has apps assigned to it, then remove the apps from the policy. See Remove Apps from the Policy.
3. Select the check box for each identity provider policy that you want to remove.
4. Click **Remove**.
5. In the **Confirmation** window, click **Save**.

Edit an Identity Provider Rule for the Policy

You can modify configuration settings for an identity provider rule of an identity provider policy.

1. Click the Identity Provider Rules tab.
2. Click the **Action** menu to the right of the identity provider rule that you want to edit.
3. Select **Edit**. A window that displays configuration settings for the identity provider rule opens.
4. Modify a configuration setting for the identity provider rule:
   a. Enter the value in the attribute field (for example, changing the name of the identity provider rule in the **Rule Name** field).
   b. Select the value from the drop-down menu (for example, selecting **Starts With** from the **If the user name** menu).
   c. Select an option (for example, selecting the **Anywhere** option).
   d. Remove the value from the field (for example, removing an identity provider that appears in the **Assign identity providers** field by clicking the **X** button to the right of the identity provider name).
5. After editing the identity provider rule, click Save.

Remove Identity Provider Rules from the Policy

Remove identity provider rules from an identity provider policy and Oracle Identity Cloud Service no longer evaluates the rules to determine the identity providers available for users to authenticate when they're accessing particular apps.

1. Click the Identity Provider Rules tab.
2. Select the check box for each identity provider rule that you want to remove from the policy.
3. Click **Remove**.
4. In the **Confirmation** window, click **OK**.

27

Manage Oracle Identity Cloud Service Sign-On Policies

This section describes how to manage Oracle Identity Cloud Service sign-on policies.

Topics:

Typical Workflow for Managing Oracle Identity Cloud Service Sign-On Policies

With the sign-on policy feature in Oracle Identity Cloud Service, you can perform tasks such as creating, managing, and removing sign-on policies.

Task	Description	Additional Information
Understand sign-on policies.	You can learn about sign-on policies, including how they are used to allow or deny access to Oracle Identity Cloud Service for users.	Understand Sign-On Policies
Add a sign-on policy.	You can add a sign-on policy using the **Sign-On Policies** page.	Add a Sign-On Policy
View details about a sign-on policy.	You can view details about a sign-on policy using the **Sign-On Policies** page.	View Details About a Sign-On Policy
Activate and deactivate sign-on policies.	You can activate and deactivate sign-on policies using the **Sign-On Policies** page.	Activate and Deactivate Sign-On Policies
Modify a sign-on policy.	You can modify a sign-on policy using the **Sign-On Policies** page.	Modify a Sign-On Policy
Remove sign-on policies.	You can remove sign-on policies using the **Sign-On Policies** page.	Remove Sign-On Policies

You can create, manage, and remove sign-on policies by:

- The Identity Cloud Service console
- SCIM-based APIs

The following sections describe how to manage sign-on policies by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Understand Sign-On Policies

A sign-on policy allows identity domain administrators, security administrators, and application administrators to define criteria that Oracle Identity Cloud Service uses to determine whether to allow a user to sign in to Oracle Identity Cloud Service or prevent a user from accessing Oracle Identity Cloud Service.

Oracle Identity Cloud Service provides you with a default sign-on policy that contains a default sign-on rule. Oracle Identity Cloud Service evaluates the criteria of the rule for any user attempting to sign in to Oracle Identity Cloud Service. By default, this rule allows all users to sign in to Oracle Identity Cloud Service. This means whichever authentication the user uses, either local authentication, by supplying a user name and password, or authentication by using an external identity provider, will be sufficient. However, you can build upon this policy by adding other sign-on rules to it. By adding these rules, you can prevent some of your users from signing in to Oracle Identity Cloud Service. Or, you can allow them to sign in, but prompt them for an additional factor to access resources that are protected by Oracle Identity Cloud Service, such as the My Profile console or the Identity Cloud Service console.

For example, you can create two sign-on rules for the default sign-on policy. The first rule prevents any users from signing in to Oracle Identity Cloud Service if they’re using an IP address that falls within the range of a network perimeter that you defined. The second rule allows users who belong to a particular group (for example, the UA_Developers group) to sign in to Oracle Identity Cloud Service; however, they will be prompted for a second factor as part of the 2-Step Verification process. All other users will be able to sign in without being prompted for a second factor.

Because you can define multiple sign-on rules for a sign-on policy, Oracle Identity Cloud Service must know the order in which the rules are to be evaluated. To do this, you can set the priority of the rules. For the example above, you can have the network perimeter sign-on rule evaluated first, and the UA_Developers group rule evaluated next. If a user meets the criteria of the network perimeter sign-on rule (that is, the IP address used to attempt to sign in to Oracle Identity Cloud Service falls within the IP range that you defined in the network perimeter), the user is prevented from accessing Oracle Identity Cloud Service-protected resources. Users who attempt to sign in to Oracle Identity Cloud Service from IP addresses that don't fall within this range don't meet the criteria of this sign-on rule, and so, the rule with the next highest priority is evaluated. For this example, this is the UA_Developers group rule. Any users who attempt to sign in, and who also belong to the UA_Developers group, will be prompted for an additional factor to sign in to Oracle Identity Cloud Service. Users who aren't members of the UA_Developers group don't meet the criteria of this rule, and so, the rule with the next highest priority is evaluated. For this example, this is the default sign-on rule. Because, this rule, by default, allows all users to sign in to Oracle Identity Cloud Service, the user will be able to sign in without being prompted for a second factor.

> **Important:**
>
> For the default sign-on rule, never set access for all of your users to be denied because if users don't meet the criteria of any other rules you define that allow them to sign in to Oracle Identity Cloud Service, they will be prevented from accessing Oracle Identity Cloud Service-protected resources. Also, configure Oracle Identity Cloud Service to evaluate this sign-on rule last because, by default, it allows all users to sign in to Oracle Identity Cloud Service.

In addition to the default sign-on policy, you can create sign-on policies and associate them with specific apps. When a user uses one of these apps to attempt to sign in to Oracle Identity Cloud Service, Oracle Identity Cloud Service checks to see if the app has any sign-on policies associated with it. If so, then Oracle Identity Cloud Service evaluates the criteria of the sign-on rules assigned to the policy. If there are no sign-on policies for the app, then the default sign-on policy is evaluated by Oracle Identity Cloud Service.

Add a Sign-On Policy

Define criteria that Oracle Identity Cloud Service uses to determine whether to allow or deny access to users who are using apps to attempt to sign in to Oracle Identity Cloud Service.

Criteria that you can define for sign-on policies include:

- The identity providers that will be used to authenticate the user
- The groups of which the user is a member
- Whether the user is an Oracle Identity Cloud Service administrator
- The IP address that the user is using to sign in to Oracle Identity Cloud Service
- Whether the user will be forced to sign in to Oracle Identity Cloud Service again (for authentication purposes), or will be authenticated the next time they sign in to Oracle Identity Cloud Service
- Whether the user will be prompted for an additional factor to sign in to Oracle Identity Cloud Service

The sign-on policy wizard contains three panes:

- **Details:** Provide the name and description for the policy.
- **Sign-On Rules:** Assign or remove rules for this policy.
- **Apps:** Assign or remove apps for this policy.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Sign-On Policies**.

> **Note:**
>
> In the **Sign-On Policies** page, Oracle Identity Cloud Service provides you with a default sign-on policy. See Understand Sign-On Policies for more information about this policy.

2. Click **Add**.
3. Add a **Policy Name** and **Description**, and then click **Next**:

After providing information in the **Details** pane and clicking **Next**, Oracle Identity Cloud Service adds the sign-on policy and saves it in a deactivated state. You must activate the policy to use it.

You may want to assign or remove rules or apps for this policy. To do this, the wizard has the **Sign-On Rules** and **Apps** panes.

4. In the **Sign-On Rules** pane of the wizard, click **Add** to add a sign-on rule to this policy.
5. Use the following table to populate the **Add Rule** window, and then click **Save**:

Field	Description
Rule Name	Enter the name of the sign-on rule.
If the user is authenticated by	Enter or select all identity providers that will be used to authenticate the user accounts evaluated by this rule.
And is a member of these groups	Enter or select the groups that the user must be a member of to meet the criteria of this rule.
And is an administrator	If the user must be assigned to administrator roles in Oracle Identity Cloud Service to meet the criteria of this rule, then select this check box. See Add or Remove a User Account from an Administrator Role. Otherwise, leave the check box deselected.
And is not one of these users	Enter or select the user accounts that will be excluded from the rule.
And the user's client IP address is	There are two options associated with this field: **Anywhere** and **In one or more of these network perimeters**. • If you select **Anywhere**, then users can log in to Oracle Identity Cloud Service using any IP address. • If you select **In one or more of these network perimeters**, then a text area appears. In this text area, you can enter or select network perimeters that you defined in Oracle Identity Cloud Service. See Add a Network Perimeter. Users can log in to Oracle Identity Cloud Service using only IP addresses that are contained in the defined network perimeters. **For applications on OCI-C:** If your application is on OCI-C and Oracle Identity Cloud Service is the Identity Provider, ensure that you add the following OCI Service Gateway IP range to the network perimeter used by Sign-On policy: OCI Service Gateway IP CIDR 240.0.0.0/4.
Access is	There are two items in this menu: **Allowed** and **Denied**. Select whether a user will be allowed or prevented from accessing the apps that are assigned to them if the user account meets the criteria of this rule.
Prompt for reauthentication	Select this check box to force the user to log in to Oracle Identity Cloud Service again. By not selecting this check box, the user will be authenticated the next time they log in to Oracle Identity Cloud Service.

If you have activated Adaptive Security, then additional fields appear in the **Add Rule** window. You can use these fields to specify conditions that Oracle Identity Cloud Service will evaluate to determine whether a user who meets these conditions will be allowed to sign in to Oracle Identity Cloud Service or will be prevented from accessing Oracle Identity Cloud Service.

For example, you can specify that if a user's risk range is High and the risk score associated with the user from a risk provider is greater than a particular value, then the user is a security risk, and shouldn't be allowed to access resources that are protected by Oracle Identity Cloud Service, such as the My Profile console, the Identity Cloud Service console, or any apps assigned to the user.

Or, you can determine that if a user's risk range is Low, based on the risk score associated with a risk provider, then the user is not a risk, and therefore, should be able to sign in to Oracle Identity Cloud Service.

See Activate Adaptive Security for more information about activating Adaptive Security, and Understand Risk Providers to learn more about risk ranges, risk providers, and risk scores associated with users.

Field	Description
And if the user's risk level is	Select whether the user's risk range must be greater than, equal to, or less than a Low, Medium, or High risk range to meet the criteria of this rule.
And the risk provider name	Select the risk provider and the risk score that will be used to determine whether a user who meets the criteria of this rule will be allowed to sign in to Oracle Identity Cloud Service or will be prevented from accessing Oracle Identity Cloud Service. Click the **Plus** button  to add another risk provider to the **Add Rule** window or the **X** button  to remove the risk provider from this window.

Important:

Be careful when setting Adaptive Security conditions. For example, suppose you specify that a user who meets the criteria of this rule because their risk score meets or exceeds the risk score that you set is prevented from accessing Oracle Identity Cloud Service. Unless the user changes their password or Oracle Identity Cloud Service runs the **Time-based risk-score re-evaluation** event to lower the user's risk score, the user can't sign in to Oracle Identity Cloud Service.

If you have selected at least one factor for Multi-Factor Authentication, then additional fields appear in the **Add Rule** window. See Configure Multi-Factor Authentication Settings.

Field	Description
Prompt for an additional factor	Select this check box to prompt the user for an additional factor to log in to Oracle Identity Cloud Service. If you select this check box, then you must specify whether the user is required to enroll in Multi-Factor Authentication and how often this additional factor is to be used to log in to Oracle Identity Cloud Service. Select **Any Factor** to prompt the user to enroll and verify any factor enabled in the MFA tenant level settings. Select **Specific Factor** to prompt the user to enroll and verify a subset of factors enabled in the MFA tenant level settings. After you select **Specific Factor**, you can select factors that must be enforced by this rule.
Frequency	• Select **Once per Session (Default)**, so that for each session that the user has opened for accessing Oracle Identity Cloud Service from an authoritative device, they must use both their user names and passwords, and a second factor. • Select **Every time**, so that each time users log in to Oracle Identity Cloud Service from a trusted device, they must use their user names and passwords, and a second factor. • Select **Once every**, and then specify how often users must provide a second factor to log in to Oracle Identity Cloud Service. For example, if you want users to use this additional factor twice a month, then enter **15** in the text field and select **Days** from the drop-down menu to the right of the field.

Field	Description
Enrollment	This menu contains two options: **Required** and **Optional**. • Select **Required** to force the user to enroll in Multi-Factor Authentication. • Select **Optional** to give users the option of skipping enrolling in Multi-Factor Authentication. Users see the inline enrollment setup process after they enter their user name and password, but can click **Skip**. Users can then enable MFA later from the **2–Step Verification** tab of the My Profile console. Users are not prompted to set up a factor the next time that they sign in to Oracle Identity Cloud Service. **Note:** If you set **Enrollment** to **Required**, and later change it to **Optional**, the change only affects new users. Users already enrolled Multi-Factor Authentication will not be able to click **Skip** when logging in.

Note:

You may have added incorrect sign-on rules to this policy inadvertently. If so, then you can remove them. To do so, select the check boxes for each of the rules that you want to remove, click **Remove**, and then click **OK** from the confirmation window.

6. In the **Sign-On Rules** pane, click **Add** to add another sign-on rule to this policy. Otherwise, click **Next**.

Note:

If you have added multiple sign-on rules to this policy, then you can change the order that will Oracle Identity Cloud Service evaluate them. See Change the Priority of a Sign-On Rule for the Policy.

7. In the **Apps** pane of the wizard, click **Assign** to assign apps to this policy.
8. In the **Assign Apps** window, select the check box for each app that you want to assign to the policy. Then, click **OK**.

> **Note:**
>
> You can assign only one sign-on policy to an app. If the app isn't assigned to any sign-on policy explicitly, then the default sign-on policy applies to the app.
>
> You can remove apps from the policy by selecting the check box for each app that you want to remove, clicking **Remove**, and then clicking **OK** from the confirmation window.

9. Click **Finish**.

View Details About a Sign-On Policy

By default, you can see the name, description, and activation status of each sign-on policy you added to Oracle Identity Cloud Service. You can also see other information about the policy, such as the rules added to the policy and any apps assigned to the policy.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Sign-On Policies**.
2. In the **Sign-On Policies** page, click the sign-on policy about which you want to see more information.

   The policy opens and displays three tabs: **Details**, **Sign-On Rules**, and **Apps**.
3. To view high-level information about the sign-on policy, such as the policy name or description, click **Details**.
4. To view sign-on rules added to the policy or apps assigned to the policy, click **Sign-On Rules** or **Apps**, respectively.

> **Note:**
>
> For more information about changing high-level information about the sign-on policy, or the rules or apps associated with the policy, see Modify a Sign-On Policy.

Activate and Deactivate Sign-On Policies

You can use Oracle Identity Cloud Service to activate and deactivate sign-on policies.

- Deactivating a sign-on policy prevents Oracle Identity Cloud Service from using the criteria in the policy to allow or deny access to users who are attempting to sign in to Oracle Identity Cloud Service
- Activating a sign-on policy allows Oracle Identity Cloud Service to evaluate the criteria in the policy to determine whether to allow or deny users from accessing Oracle Identity Cloud Service-protected resources

> **Note:**
>
> A green check mark  indicates an activated sign-on policy. A red circle with a white line through the circle  indicates a deactivated sign-on policy.

Activate Sign-On Policies

You can use Oracle Identity Cloud Service to activate sign-on policies.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Sign-On Policies**.
2. In the **Sign-On Policies** page, select the check box for each sign-on policy that you want to activate.
3. Click **Activate**.
4. In the **Confirmation** window, click **OK**. The status of each sign-on policy changes from deactivated  to activated  .

Deactivate Sign-On Policies

You can use Oracle Identity Cloud Service to deactivate sign-on policies.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Sign-On Policies**.
2. In the **Sign-On Policies** page, select the check box for each sign-on policy that you want to deactivate.
3. Click **Deactivate**.
4. In the **Confirmation** window, click **OK**. The status of each sign-on policy changes from activated  to deactivated  .

Modify a Sign-On Policy

After viewing details about, activating, or deactivating a sign-on policy, you can modify it. Modifying a sign-on policy in Oracle Identity Cloud Service includes:

- Changing the name or description of the policy
- Adding, editing, changing the priority of, and removing sign-on rules for the policy
- Assigning apps to the policy
- Removing apps from the policy

To modify a sign-on policy:

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Sign-On Policies**.
2. In the **Sign-On Policies** page, click the sign-on policy that you want to modify.

The policy opens and displays three tabs: **Details**, **Sign-On Rules**, and **Apps**. See View Details About a Sign-On Policy for more information about these tabs.

Change the Policy Name and Description

You can change the name and description for a sign-on policy.

1. Click the **Details** tab.
2. In the **Policy Name** and **Description** fields, enter the new name and description for the sign-on policy.
3. Click **Save**.

Add a Sign-On Rules to the Policy

You can add sign-on rules to a sign-on policy. By adding these rules, you can prevent some of your users from signing in to Oracle Identity Cloud Service. Or, you can allow them to sign in, but prompt them for an additional factor to access resources that are protected by Oracle Identity Cloud Service, such as the My Profile console or the Identity Cloud Service console.

1. Click the **Sign-On Rules** tab.
2. Click **Add**.
3. Add a sign-on rule to the policy.

Change the Priority of a Sign-On Rule for the Policy

You can change the priority of a sign-on rule for a sign-on policy to change the order that Oracle Identity Cloud Service will evaluate it.

1. Click the **Sign-On Rules** tab.
2. Click the ellipsis button  to the left of the sign-on rule for which you want to change the priority.
3. Drag-and-drop this rule to change the order that Oracle Identity Cloud Service will evaluate it.

   For example, if your sign-on rule has a priority of 4, and you want Oracle Identity Cloud Service to evaluate it first, drag the rule and drop it so that it appears directly above the sign-on rule with a priority of 1. Your sign-on rule will appear first in the list, and the other rule will now have a priority of 2.
4. Click **Save**.

Edit a Sign-On Rule for the Policy

You can modify configuration settings for a sign-on rule of a sign-on policy.

1. Click the **Sign-On Rules** tab.
2. Click the **Action** menu  to the right of the sign-on rule that you want to edit.
3. Select **Edit**. A window that displays configuration settings for the sign-on rule opens.
4. Modify a configuration setting for the sign-on rule:

a. Enter the value in the attribute field (for example, changing the name of the sign-on rule in the **Rule Name** field).

b. Select the value from the drop-down menu (for example, selecting **Allowed** from the **Access is** menu).

c. Select or clear a check box or option (for example, selecting the **Prompt for reauthentication** check box or the **Anywhere** option).

d. Remove the value from the field (for example, removing a group that appears in the **And is a member of these groups** field by clicking the **X** button to the right of the group name).

5. After editing the sign-on rule, click **Save**.

Remove Sign-On Rules from the Policy

If you remove sign-on rules from a sign-on policy, Oracle Identity Cloud Service will no longer evaluate the criteria in the rules to determine whether to allow a user to sign in to Oracle Identity Cloud Service or prevent a user from accessing Oracle Identity Cloud Service-protected resources.

1. Click the **Sign-On Rules** tab.
2. Select the check box for each sign-on rule that you want to remove from the policy.
3. Click **Remove**.
4. In the confirmation window, click **OK**.

Assign Apps to the Policy

When a user uses an app assigned to a sign-on policy to attempt to sign in to Oracle Identity Cloud Service, Oracle Identity Cloud Service evaluates the criteria of the sign-on rules that are also assigned to the policy to determine whether to allow or deny the user from accessing resources that are protected by Oracle Identity Cloud Service.

1. Click the **Apps** tab.
2. Click **Assign**.
3. In the **Assign Apps** window, select the check box for each app that you want to assign to the policy. Then, click **OK**.

Remove Apps from the Policy

You can remove apps from a sign-on policy. Oracle Identity Cloud Service will no longer evaluate the criteria of the sign-on rules assigned to the policy to determine whether to allow or deny the user from signing in to Oracle Identity Cloud Service from one of these apps.

1. Click the **Apps** tab.
2. Select the check box for each app that you want to remove from the policy.
3. Click **Remove**.
4. In the confirmation window, click **OK**.

Remove Sign-On Policies

You can use Oracle Identity Cloud Service to remove multiple sign-on policies simultaneously.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Sign-On Policies**.
2. In the **Sign-On Policies** page, if the policy that you want to remove has apps assigned to it, then remove the apps from the policy.
3. Select the check box for each sign-on policy that you want to remove.
4. Click **Remove**.
5. In the confirmation window, click **OK**.

28

Manage Oracle Identity Cloud Service Network Perimeters

This section describes how to manage Oracle Identity Cloud Service network perimeters.

Topics:

Typical Workflow for Managing Oracle Identity Cloud Service Network Perimeters

With the network perimeter management feature in Oracle Identity Cloud Service, you can perform tasks such as creating, managing, and removing network perimeters.

Task	Description	Additional Information
Understand network perimeters.	You can learn about network perimeters, including how they are used to restrict the IP addresses that users can use to log in to Oracle Identity Cloud Service.	Understand Network Perimeters
Add a network perimeter.	You can add a network perimeter using the **Network Perimeters** page.	Add a Network Perimeter
View details about a network perimeter.	You can view details about a network perimeter using the **Network Perimeters** page.	View Details About a Network Perimeter
Modify a network perimeter.	You can modify a network perimeter using the **Network Perimeters** page.	Modify a Network Perimeter
Remove network perimeters.	You can remove network perimeters using the **Network Perimeters** page.	Remove Network Perimeters

You can create, manage, and remove network perimeters by:

- The Identity Cloud Service console
- SCIM-based APIs

The following sections describe how to manage network perimeters by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Understand Network Perimeters

For security purposes, identity domain administrators, security administrators, and application administrators can define network perimeters in Oracle Identity Cloud Service. A network perimeter contains a list of IP addresses.

> **Note:**
>
> Enable Network Perimeters. This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service.

After creating a network perimeter, you can prevent users from signing in to Oracle Identity Cloud Service if they use one of the IP addresses in the network perimeter. This is known as blacklisting. A blacklist contains IP addresses or domains that are suspicious. As an example, a user may be trying to sign in to Oracle Identity Cloud Service with an IP address that comes from a country where hacking is rampant.

An IP address is a string of numbers that identifies the network of any device connected to the internet. It's like a return address on an envelope, and is associated with a human-readable domain. Since the IP address tells other devices where data is coming from, it can be a good way to track bad content.

Blacklists can list a single IP address or a (set) range of IPs. Oracle Identity Cloud Service can use this information to block users who attempt to sign in from suspicious IP addresses.

You can also configure Oracle Identity Cloud Service so that users can log in, using only IP addresses contained in the network perimeter. Users who attempt to sign in to Oracle Identity Cloud Service with these IP addresses will be accepted. You can also create a list of IP addresses from which users can't log in.

You can configure Oracle Identity Cloud Service so that only users who use a particular IP address or IP address in a specific range will be allowed to sign in to Oracle Identity Cloud Service. Or, you can configure Oracle Identity Cloud Service to monitor for suspicious IP addresses or IP address ranges, and prevent users who use these IP addresses from signing in to Oracle Identity Cloud Service.

With a network perimeter, you can define, in a standard format, an exact IP address, a range of IP addresses, or a set of masked IP addresses. Both Internet Protocol version 4 (IPv4) and Internet Protocol version 6 (IPv6) protocols are supported.

Detailed information about these three formats appears below.

- Exact IP address. You can enter a single IP address or multiple IP addresses. If you enter multiple exact IP addresses, then put a comma between each one.
- Two IP addresses, separated by a hyphen, which is an IP range. For example, if you specify the IP range of `10.10.10.1-10.10.10.10`, any user who attempts to sign in to Oracle Identity Cloud Service with an IP address from `10.10.10.1` through `10.10.10.10` will be using an IP address that falls within the IP range.

- Masked IP address range. Each number of an IP address is 8 bits. For example, if you have a masked range of `10.11.12.18/24`, then the first three numbers (24 bits) is the mask that must be applied to see if an IP address falls in this range. For this example, valid IP addresses will be those that begin with `10.11.12`.

> **Note:**
>
> The examples listed above are using IP addresses with the IPv4 protocol. However, you can apply the same formats to IP addresses that use the IPv6 protocol (for example, `B138:C14:52:8000:0:0:4D8`).

After defining your network perimeters, you can assign them to a sign-on policy, and configure the policy so that if you're trying to sign in to Oracle Identity Cloud Service using an IP address that's defined in the network perimeter, you can log in to Oracle Identity Cloud Service or you'll be prevented from accessing Oracle Identity Cloud Service.

See Add a Sign-On Policy for more information about assigning network perimeters to a sign-on policy.

Add a Network Perimeter

You can add a network perimeter in Oracle Identity Cloud Service, and then configure Oracle Identity Cloud Service to restrict the IP addresses that users can use to log in to Oracle Identity Cloud Service.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Network Perimeters**.
2. In the **Network Perimeters** page, click **Add**. The **Add Network Perimeter** window appears.
3. Use the following table to populate the **Add Network Perimeter** window, and then click **Save**:

Field	Description
Network Perimeter Name	Enter the name of the network perimeter.
List of IP Addresses	Enter the exact IP address or IP addresses, IP range, or masked IP address range for the network perimeter. See Understand Network Perimeters for more information about these IP address formats.

You added a network perimeter. See Add a Sign-On Policy for more information about using this network perimeter in sign-on policy rules.

View Details About a Network Perimeter

By default, you can see the name of each network perimeter you defined in Oracle Identity Cloud Service. You can also see other information about the policy, such as the IP addresses for the network perimeter.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Network Perimeters**.

2. In the **Network Perimeters** page, click the **Action** menu  to the right of the network perimeter about which you want to see more information, and then click **Edit**.

   A window opens and displays the name and IP addresses associated with the network perimeter. From this window, you can modify this information. See Modify a Network Perimeter.

Modify a Network Perimeter

After viewing details about a network perimeter, you can modify it. Modifying a network perimeter in Oracle Identity Cloud Service includes changing the name of the network perimeter or the IP addresses associated with the network perimeter.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Network Perimeters**.

2. In the **Network Perimeters** page, click the **Action** menu  to the right of the network perimeter about which you want to modify, and then click **Edit**.

   A window opens and displays the name and IP addresses associated with the network perimeter.

3. In the **Network Perimeter Name** field, enter the new name of the network perimeter.
4. In the **List of IP Addresses** text area, modify the IP addresses associated with the network perimeter by adding, editing, or removing IP addresses for the network perimeter. If you enter multiple new IP addresses, then put a comma between each one.
5. Click **Save**.

Remove Network Perimeters

You can use Oracle Identity Cloud Service to remove multiple network perimeters simultaneously.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Network Perimeters**.
2. In the **Network Perimeters** page, select the check box for each network perimeter that you want to remove.
3. Click **Remove**.
4. In the **Confirmation** window, click **OK**.

29

Manage Oracle Identity Cloud Service App Gateways

This section describes how to manage Oracle Identity Cloud Service App Gateways.

Service Change Announcement

App Gateway Replaces App Gate

The software appliance **App Gate** has been replaced with **App Gateway**. As of **August 2019**, App Gate has been replaced with App Gateway. Both the App Gate and the App Gateway solutions are software appliances that you can use to provide Single Sign-On (SSO) and authorization for your on-premises applications. This enables you to use one appliance to provide SSO for multiple applications by allowing external users to access internal applications securely without needing a VPN client. There's no change in functionality between the old App Gate and the new App Gateway solution. However, as a customer you will need to replace App Gate with App Gateway and reconfigure your supported applications. Technical support for App Gate will end after **August 15, 2021**. See Manage Oracle Identity Cloud Service App Gateways and see Download and Extract the App Gateway Binary File to download the App Gateway and to ensure that you are using the latest version of App Gateway.

Topics:

Typical Workflow for Managing App Gateways

With the App Gateways feature in Oracle Identity Cloud Service, you can perform tasks such as creating, managing, and removing App Gateways.

Task	Description	Additional Information
What is App Gateway	You learn what an App Gateway is.	What is App Gateway?
Add an App Gateway	You learn how to register an App Gateway in Oracle Identity Cloud Service before you install and configure the App Gateway software.	Register an App Gateway
View Details About an App Gateway	You can view details about an App Gateway using the **App Gateways** page.	View Details About an App Gateway
Activate and Deactivate App Gateways	You can activate and deactivate App Gateways using the **App Gateways** page.	Activate and Deactivate App Gateways
Modify an App Gateway	You can modify an App Gateway using the **App Gateways** page.	Modify an App Gateway
Remove App Gateways	You can remove App Gateways using the **App Gateways** page.	Remove App Gateways

You can create, manage, and remove App Gateways by:

- The Identity Cloud Service console
- SCIM-based APIs

The following sections describe how to manage App Gateways by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Understand App Gateway

Understand what an App Gateway is, why you should use it, and how App Gateway works to to protect access to your web applications.

- What is App Gateway?
- Why Should You Use App Gateway?
- How Does App Gateway Work?
- Set Up High Availability

Enable App Gateway. This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service.

What is App Gateway?

App Gateway is a software appliance that enables you to integrate applications hosted either on a compute instance, in a cloud infrastructure, or in an on-premises server with Oracle Identity Cloud Service for authentication purposes.

App Gateway acts as a reverse proxy protecting web applications by restricting unauthorized network access to them. App Gateway intercepts any HTTP request to these applications and ensures that the users are authenticated with Oracle Identity Cloud Service before forwarding

the request to these application. App Gateway propagates the authenticated user's identity to the applications.

If the user isn't authenticated with Oracle Identity Cloud Service, then App Gateway redirects the user to Oracle Identity Cloud Service's **Sign In** page for credential validation.

Why Should You Use App Gateway?

App Gateway is a non-intrusive integration method that uses a middle-tier layer to integrate web applications with Oracle Identity Cloud Service for authentication purposes.

Use App Gateway to:

- Integrate enterprise applications hosted either on-premises or in a cloud infrastructure with Oracle Identity Cloud Service for authentication purposes.

  For example, if you have a web applications hosted on-premises or in a cloud infrastructure, you can ingrate this application with any other cloud-based applications for single sign-on. Use App Gateway to integrate your web application with Oracle Identity Cloud Service, and then make sure that the other cloud-based applications use Oracle Identity Cloud Service as their authentication mechanism. All these applications will make use of the single sign-on provided by Oracle Identity Cloud Service.

- Expose intranet web applications to internet access.

  If your web application is hosted and accessed over your intranet and you want to expose access to this application over the internet, use App Gateway to proxy any internet request and to require users to authenticate with Oracle Identity Cloud Service before accessing your intranet web application. In this case, you deploy App Gateway in your network DMZ while your application remains in the intranet zone.

- Integrate with applications that lack a native authentication mechanism and don't support SAML federation, OAuth, or OpenID Connect integration methods.

  If your application doesn't support the standards for authentication that Oracle Identity Cloud Service supports (`SAML`, `OAuth`, and `OpenID Connect`), and you can't use Oracle Identity Cloud Service's SDKs in your application, then you can use App Gateway to integrate your web application with Oracle Identity Cloud Service.

- Integrate with applications that support the HTTP Header-based authentication.

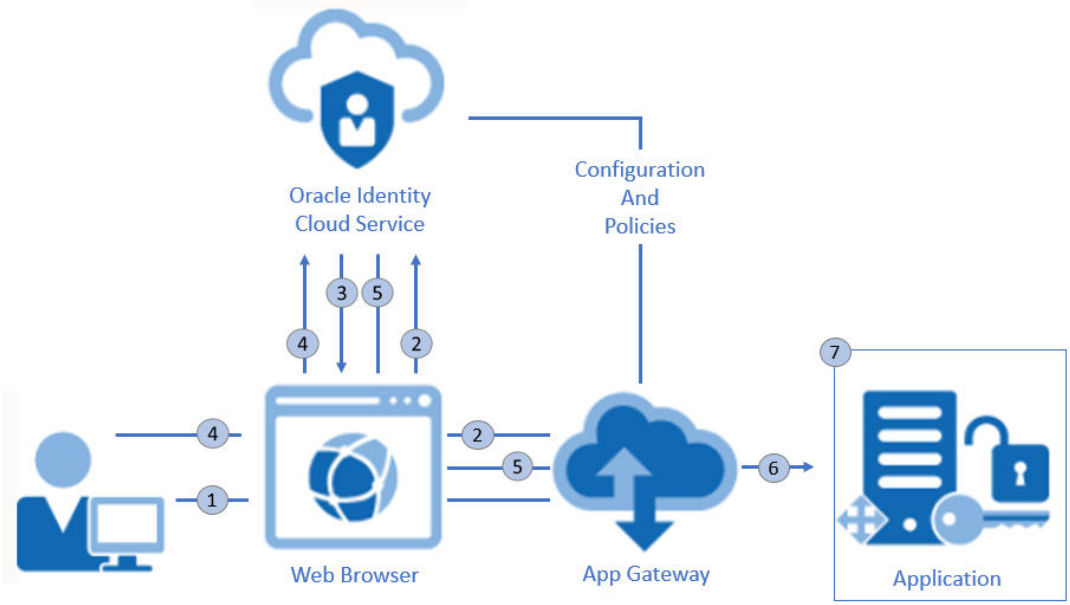
  For web applications that support HTTP Header-based authentication, the App Gateway integration method requires no change to the web application's source code. You need to configure the application's authentication policies in Oracle Identity Cloud Service to add header variables in the request before App Gateway forwards the request to the application. By doing so, the application can identify the user authenticated with Oracle Identity Cloud Service.

How Does App Gateway Work?

The App Gateway is deployed within a customer’s infrastructure, regardless of whether the infrastructure is in the cloud, on-premises, or a hybrid one.

The App Gateway works as a reverse proxy, intercepting all requests from the client to the application. The App Gateway then verifies if a user is already logged in to Oracle Identity Cloud Service. If the user has logged in, then App Gateway adds header variables to the request so that the application being protected can access the header variable The application trusts App Gateway has identified the signed in user in Oracle Identity Cloud Service values and create the user session.

Ensure that the communication between App Gateway and application is secure to avoid changes in the header variable values before the request is sent to the application.



The following steps explain the form-based authentication flow between the web browser, App Gateway, and an enterprise application:

1. In a web browser, a user requests access to an application through a URL exposed by App Gateway.
2. App Gateway intercepts the request, verifies the user doesn't have a session with Oracle Identity Cloud Service, and then redirects the user's browser to the Oracle Identity Cloud Service's **Sign In** page.
   In step 2, if the user has a session with Oracle Identity Cloud Service, it means that the user has already signed in to Oracle Identity Cloud Service. If so, then an access token is sent to App Gateway, and then the remaining steps are skipped.
3. Oracle Identity Cloud Service presents the **Sign In** page or whichever sign-in mechanism has been configured for Oracle Identity Cloud Service.
4. The user signs in to Oracle Identity Cloud Service.
5. Upon successful authentication, Oracle Identity Cloud Service creates a session for the user and issues an access token to App Gateway.
6. App Gateway uses the token to identify the user. It then adds header variables to the request and forwards the request to the application.
7. The application receives the header information, validates the user's identity, and starts the user session.

Any subsequent request to the application's protected resources is intercepted by App Gateway. App Gateway identifies the user, adds header variables to the request, and forwards the request to the application.

To sign out, the user calls an application's logout URL. The App Gateway identifies the logout URL and redirects the user to the Oracle Identity Cloud Service's OAuth logout endpoint (`/oauth2/v1/userlogout`). After Oracle Identity Cloud Service signs the user out, Oracle Identity Cloud Service can redirect the user's browser to a URL of the application which can then remove the application's user session.

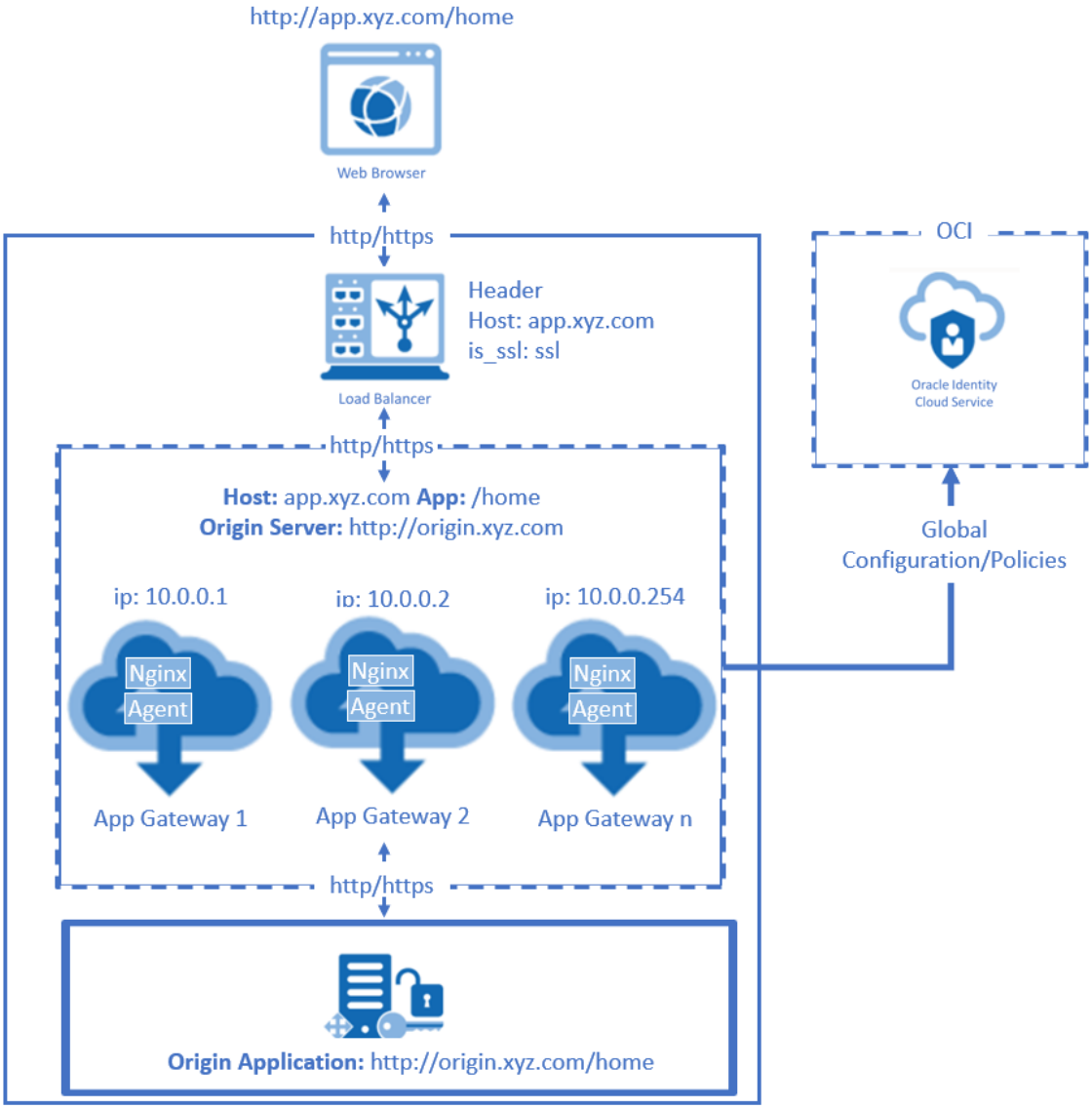
Set Up High Availability

Use a load balancer to achieve high availability for multiple instances of App Gateway.

If high-availability is a requirement to access your web application, you can have multiple App Gateways, configure each of them to integrate with Oracle Identity Cloud Service, and use a load balancer to balance the request among the App Gateway instances.

The following architecture diagram shows the components required for high availability.

Figure 29-1 App Gateway Load Balancer Diagram



This architecture requires that you install and configure more than one instance of App Gateway. Each App Gateway instance is configured to link to the same Oracle Identity Cloud Service URL, and to use the same **Client ID** and **Client Secret** from the App Gateway registration in Oracle Identity Cloud Service console.

Use a load balancer to distribute request between the App Gateway instances.

Additionally, the load balancer must perform health checks via HTTP with `HTTP keepalives` enabled for a duration that exceeds the health check interval. This prevents the load balancer from redirecting browser requests to an offline App Gateway instance.

The health check endpoint of App Gateway is `/cloudgate/v1/about`.

Set Up an App Gateway

Download the App Gateway binary file, install the App Gateway server, register the App Gateway using Identity Cloud Service console, configure the App Gateway server, assign an enterprise application, start the App Gateway server, and test the access to the application through App Gateway.

Download and Extract the App Gateway Binary File

The App Gateway binary file you download from Identity Cloud Service console is a compressed (.zip) file. This file contains an Open Virtual Appliance (.ova) file which you use to install the App Gateway server.

Service Change Announcement

App Gateway Replaces App Gate

The software appliance **App Gate** has been replaced with **App Gateway**. As of **August 2019**, App Gate has been replaced with App Gateway. Both the App Gate and the App Gateway solutions are software appliances that you can use to provide Single Sign-On (SSO) and authorization for your on-premises applications. This enables you to use one appliance to provide SSO for multiple applications by allowing external users to access internal applications securely without needing a VPN client. There's no change in functionality between the old App Gate and the new App Gateway solution. However, as a customer you will need to replace App Gate with App Gateway and reconfigure your supported applications. Technical support for App Gate will end after **August 15, 2021**. See Manage Oracle Identity Cloud Service App Gateways and see Download and Extract the App Gateway Binary File to download the App Gateway and to ensure that you are using the latest version of App Gateway.

To download and extract the App Gateway binary file:

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Downloads**.
2. In the **Downloads** page, click **Download** to the right of **App Gateway for Identity Cloud Service**.
3. Verify that a **Success** status appears to the right of **App Gateway for Identity Cloud Service**.
4. Extract the content of the zip file you downloaded to a location on your desktop. For Example, `c:\temp`.

   The `c:\temp\app-gateway-<version>.ova` file will be created.

> **Note:**
>
> The **App Gateway for Identity Cloud Service** doesn't replace the **App Gate for Identity Cloud Service**. The **App Gateway for Identity Cloud Service** software is based on NGINX web server and is used to protect access of enterprise applications. The **App Gate for Identity Cloud Service** software is an OEM product that has similar but not the same features.

Configuring Cloud Gate CORS Settings in Oracle Identity Cloud Service

Learn how to configure Cloud Gate CORS settings in Oracle Identity Cloud Service.

If you need to configure Cloud Gate CORS settings in Oracle Identity Cloud Service, then you use the Oracle Identity Cloud Service REST API. See Configuring Cloud Gate CORS Settings in Oracle Identity Cloud Service.

Install App Gateway

Use the App Gateway Open Virtual Application (OVA) file or Docker to install the App Gateway server. You can run the server in a compute instance on Oracle Cloud Infrastructureor in a virtual machine hosted in your network environment.

To install the App Gateway server, see the following topics:

- Install App Gateway on Oracle Cloud Infrastructure
- Install App Gateway Using Oracle VM Virtual Box Software
- Deploy the Oracle App Gateway Docker Container

Install App Gateway on Oracle Cloud Infrastructure

To install App Gateway on Oracle Cloud Infrastructure, you need to upload the App Gateway virtual disk image file to a **Bucket** in Oracle Cloud Infrastructure, create a **Custom Image** using the App Gateway virtual disk image file, and then create a **Compute** instance based on this custom image.

- Upload the App Gateway Virtual Machine Disk Image File to an Object Storage bucket in Oracle Cloud Infrastructure
- Create a Custom Image in Oracle Cloud Infrastructure Based on the App Gateway Virtual Machine Disk Image File
- Create a Compute Instance Using App Gateway's Custom Image

Upload the App Gateway Virtual Machine Disk Image File to an Object Storage bucket in Oracle Cloud Infrastructure

Before creating a compute instance on Oracle Cloud Infrastructure to run App Gateway, you need create a `Virtual Machine Disk Image` (`VMDK`) file using the App Gateway `Open Virtual Appliance` (`OVA`) file, and then upload this `VMDK` file to Oracle Cloud Infrastructure.

To create the `VMDK` file, follow the procedure to import the App Gateway's `Open Virtual Appliance` (`OVA`) file using **Oracle VM Virtual Box** software, but don't start the machine or

configure port forward rule for it. See Import the Open Virtual Appliance Image File in Virtual Machine Software.

1. Sign in to Oracle Cloud.
2. In the Oracle Cloud console, expand the **Navigation Drawer**, move the mouse over **Object Storage**, and then click **Object Storage**.
3. In the **Object Storage** page, select the compartment where the bucket will be to upload the image. Click **Create Bucket**, click **Create Bucket** in the **Create Bucket** dialog, and then click the name of the bucket you created.

   Contact your Oracle Cloud Infrastructure administrator for more information about which compartment to create buckets.
4. On the **Bucket Detail** page, click **Upload Object** in the **Objects** section.
5. Click **select files** to browse and open the App Gateway's `VMDK` file, and then click **Upload Objects**.
6. After the file uploads, click **Close**.
7. Click the menu on the right for your object entry, and then record the **URL Path (URI)** value.

Create a Custom Image in Oracle Cloud Infrastructure Based on the App Gateway Virtual Machine Disk Image File

To create a compute instance on Oracle Cloud Infrastructure to run App Gateway, you need to create a custom image from the App Gateway's `Virtual Machine Disk Image` (`VMDK`) file you uploaded to a bucket on Oracle Cloud Infrastructure.

Make sure your Oracle Cloud Infrastructure account has compartments, a virtual cloud network, and subnets previously set up.

Make sure you have selected a compartment in Oracle Cloud console, before proceding.

> **Note:**
>
> The components design should align with your Oracle Cloud Infrastructure operational model. Contact your Oracle Cloud Infrastructure administrator for more information.

1. In the Oracle Cloud console, click the top-left menu, mouse over **Compute**, and then click **Custom Images**.
2. In the **Images** page, select the same compartment where you uploaded your VMDK file, and then click **Import Image**.
3. In the **Import Image** dialog box, enter or select the following values, and then click **Import Image**.
   - **CREATE IN COMPARTMENT**: Select the compartment to import the image. The compartment must be the same where your compute instance will be created.
   - **NAME**: `App Gateway Custom Image`
   - **OPERATING SYSTEM**: Select **Linux**.
   - **OBJECT STORAGE URL**: Enter the URL path you recorded after you uploaded the VMDK file.

- **IMAGE TYPE**: Select **VMDK**.
- **LAUNCH MODE**: Select **EMULATED MODE**.

Wait until the custom image creation finishes.

Create a Compute Instance Using App Gateway's Custom Image

After you uploaded the App Gateway's `Virtual Machine Disk Image` (`VMDK`) file to a bucket in Oracle Cloud Infrastructure and created a custom image using this `VMDK` file, you can create a compute instance to run App Gateway.

1. In the Oracle Cloud console, click the top-left menu, mouse over **Compute**, and then click **Instances**.
2. In the **Instances** page, click **Create Instance**.
3. In the **Create Compute Instance** page, enter `My App Gateway Server` in the **Name your instance** field, and then click **Change Image Source**.
4. In the **Browse All Images** dialog, click **Custom Images**, select the appropriate compartment, select **App Gateway Custom Image**, and then click **Select Image**.
5. In the **Add SSH Key** section, add a public SSH key, by either uploading a public key file or pasting the public key value in the **SSH Key** field.

   See **Creating an SSH Key Pair Using PuTTY Key Generator** section in Managing Key Pairs on Linux Instances.
6. In the **Configure networking** section, select a compartment in **Virtual cloud network compartment**.

   If your compartment doesn't have virtual cloud network configured, then enter `App Gateway VNC` as **Name** in the **New virtual cloud network** section. If your compartment has virtual cloud network configured, then select the values for **Virtual cloud network**, **Subnet compartment**, and **Subnet** in which your compute instance will be created.

   > **Note:**
   >
   > The component design should align with your Oracle Cloud Infrastructure operational model. Contact your Oracle Cloud Infrastructure administrator for more information.
7. Click **Create**, and wait until your compute instance is provisioned and running.
8. Record the value of the **Public IP Address** assigned to this compute instance.

Make sure that you have a **Security List** configured so that you can connect to the `My App Gateway Server` compute instance using a SSH client software such as `PuTTY`. Contact your Oracle Cloud Infrastructure administrator for more information.

Install App Gateway Using Oracle VM Virtual Box Software

To install App Gateway using Oracle VM Virtual Box, import the App Gateway `Open Virtual Appliance` (`OVA`) file in a **Oracle VM Virtual Box**, and then configure the App Gateway virtual machine to receive HTTP request.

- Import the Open Virtual Appliance Image File in Virtual Machine Software
- Configure Port Forwarding Rules

Import the Open Virtual Appliance Image File in Virtual Machine Software

To run App Gateway in a virtual machine, import the App Gateway `Open Virtual Appliance` (`OVA`) image file in virtual machine software such as Oracle VM Virtual Box.

The following procedure requires access to a Windows server as administrator. This server must have **Oracle VM Virtual Box** software installed.

1. Log in to the Windows server, and upload the App Gateway `OVA` file from your desktop to a working folder in the server. For example, `c:\temp`.

   See Download and Extract the App Gateway Binary File.
2. Launch the **Oracle VM Virtual Box Manager** software, and then select **Import Appliance** from the **File** menu.
3. Locate the `OVA` file on the Windows server, and then click **Next**.
4. In the **Import Virtual Appliance** window, update the **Name** field with the value `App Gateway Server`.
5. To define a new MAC address to the App Gateway server network component, select **Reinitialize the MAC address of all network cards**.
6. Click **Import**.
7. Verify `App Gateway server` is listed in the **Oracle VM Virtual Box Manager**.

After you import App Gateway, a virtual disk image file (`VMDK`) will be created in the Windows server.

To locate this file, select `App Gateway Server` in **Oracle VM Virtual Box Manager**, click **Settings**, click **Storage**, and then click the name that appears under **Controller: SATA** in the **Storage Devices** section. The location of the `VMDK` file appears in the **Location** field under **Information**.

Configure Port Forwarding Rules

Create a port forwarding rule to allow the requests received by the Windows server hosting the App Gateway virtual machine to be forwarded to the App Gateway server.

1. In the **Oracle VM Virtual Box Manager** software, select the App Gateway server, and then click **Settings**.
2. Select **Network** on the left menu, expand **Advanced**, and then click **Port Forwarding**.
3. In the **Port Forwarding Rules** window, click  , configure a rule to forward the requests from the host port to the guest port, and then click **OK**.

   For example, if your App Gateway is configure to use port `4443`, then enter `4443` in both **Host Port** and **Guest Port** columns.

   The port number must be the same as the port value that you provided during App Gateway registration.
4. In the **Port Forwarding Rules** window, click  , configure a rule to forward the requests from the host port `22` to the guest port `22`, and then click **OK**.

   This enable you to use a `SSH` client such as `PuTTY` to login to the App Gateway server later.
5. In the **Port Forwarding Rules** dialog box, click **OK**.

6. In the **App Gateway Settings** dialog box, click **OK**.
7. Select the App Gateway server, and then click **Start**.

Deploy the Oracle App Gateway Docker Container

App Gateway can be deployed by using OVA or using Docker. Learn how to deploy the Oracle App Gateway Docker container.

Prerequisites

- Download the App Gateway docker image. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Downloads**.
- Create a wallet file containing the Client ID and Client Secret of the App Gateway that was created in the Admin Console. The wallet file should be named **cwallet.sso** and should be copied to the local folder, so that the container can uptake the file. **Note:** The wallet tool can be downloaded from the Identity Cloud Service console. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Downloads**.
- Install Docker (Command: `$ yum install docker-engine`).
- Add the current user to the Docker group (Command: `$ sudo usermod -a -G docker $USER`).

Extract the Docker Image

If the Docker image is in `.tar.gz` format, then you must use the following commands to extract the image before you can create the container.

1. Load the `.tar.gz` file to the local Docker registry. Command: `$ docker load -i <.tar.gz file>`
2. Verify that you see the image in the local Docker registry. Command: `$ docker images`

Create the App Gateway Container

Set the App Gateway Environment Variables

To run the App Gateway Docker container, the following environment variables must be set in the `appgateway-env` file. **Important:** No validation is performed on these values. If you configure invalid values, App Gateway Docker container creation will fail.

- `CG_APP_TENANT=<tenant name>`
- `IDCS_INSTANCE_URL=<idcs instance url>`. The URL required to access the Identity Cloud Service instance.
- `NGINX_DNS_RESOLVER=<resolver ip>`. Configure the nameserver found in the file `/etc/resolv.conf`. The default value is `127.0.0.1`.

Run Docker

Use the following command to run Docker.

> **Note:**
>
> The local folder will be mounted as volume and is accessible within the Docker container.
>
> The wallet file (which contains the Client ID and Client Secret) you created as a prerequisite (`cwallet.sso`) should be copied to the local folder, so that the container can reference the file.

```
$ docker run -it -d
--name <container name>
--env-file <path to env file>
--env HOST_MACHINE=`hostname -f`
--volume <local folder>/cwallet.sso:/usr/local/nginx/conf/cwallet.sso
--net=host/<User-defined bridge name> <image name>
```

Example Container with Host Networking with No Port Mapping

The following is an example of Host Networking with no port mapping. This is only for port numbers greater than 1024.

> **Note:**
>
> If the port number configured for the App Gateway host is less than 1024, then you must use Bridge Networking for the Docker, along with the port mapping. See the *Bridge Networking with Port Mapping* command example below to run the Docker container.

```
$ docker run -it -d
--name appgateway
--env-file appgateway-env
--env HOST_MACHINE=`hostname -f`
--volume /opt/appgateway/cwallet.sso:/usr/local/nginx/conf/cwallet.sso
--net=host opc-delivery.docker.example.com/idcs/appgateway:RELEASE-BUILDNUMBER
```

Example Bridge Networking with Port Mapping

The following is an example of Bridge Networking with port mapping (ports 80 to 65535).

Prerequisite: Before you use the Bridge Network configuration, add/update `iptables` to `true`, in the file `/etc/docker/daemon.json`. This allows the Docker daemon to edit the iptables filter rules required for port mapping.

```
$ docker run -it -p 80:9000  -d
--name appgateway
--env-file /home/<username>/dev/appgateway_pool/appgateway_env --env
HOST_MACHINE=`hostname -f`
--volume /opt/appgateway/cwallet.sso:/usr/local/nginx/conf/cwallet.sso
--net=bridge-net  idcs.docker.example.com/idcs/appgateway: RELEASE-BUILDNUMBER
```

Note: Docker internally updates the iptables/firewalld with the routes for the port, when the above command is run.

Post-Requisite Container Step

If the host is configured as HTTPS, the following additional steps are required to copy the certificates to the container.

1. Configured SSL certificates need to be copied to the location that is specified in **Additional Properties**. Go to **Security**, **App Gateways**, **<Gateway>**, **Hosts**, **Additional Properties** and note the location.
2. Run commands like the following. **Note:** The location of the certificate depends on the location you specified in the App Gateways Host.

```
$ docker cp deploy/docker/nginx/build/test-config/certs/my-appgateway.cert
appgateway:/scratch/docker/cloudgate/certs/my-appgateway.cert
```

```
$ docker cp deploy/docker/nginx/build/test-config/certs/my-appgateway.key
appgateway:/scratch/docker/cloudgate/certs/my-appgateway.key
```

Upgrade to a New App Gateway Version

To upgrade to a new App Gateway version, delete the existing container and recreate the container with a new version of the image. The wallet files are automatically used by the container, provided the files are not deleted in the local folder, and the same local folder is used for the volume mount.

FAQs

- **How do I know if my container was created successfully?**

  Run the command: `$ docker ps -a` and ensure that the `STATUS` is **Up** in the list corresponding to your container name.

- **If the container STATUS shows exited, how do I check the logs to determine why the container was terminated?**

  Run the command: `$ docker logs <container name>`. This command prints the log messages, which will contain the log messages printed by App Gateway.

- **How to do edit the cloudgate.config file inside the container?**

  Run the command: `$ docker exec -it <container name> bash`.

  Run this command to access the container if the container is running with a Bash shell. Once inside the container, you can edit the files using Nano editor.

- **Can we print the access logs in JSON format?**

  Yes, you can print the access logs in JSON format. Add the lines below to the file `/usr/local/nginx/conf/nginx.conf`, inside an HTTP block and then restart App Gateway.

```
log_format jsonf escape=json '{"remote_addr": "$remote_addr",
"remote_user":
      "$remote_user", "time": [$time_local], "request": "$request",
"status": $status,
      "body_bytes_sent": $body_bytes_sent, "http_referer":
"$http_referer", "user_agent":
      "$http_user_agent", "x_forwarded_for": "$http_x_forwarded_for"}';
```

```
access_log
      /usr/local/nginx/logs/access.log jsonf;
```

Note: You can edit the JSON fields that you're interested in by removing or adding the NGINX variable.

Register an App Gateway

Before installing the binary file for App Gateway that appears on the **Downloads** page, you must register your App Gateway using the Identity Cloud Service console.

To register an App Gateway you must add hosts and associate each host to an enterprise application your App Gateway will protect:

- In the **Hosts** pane, you define host identifiers. Each host identifier represents a domain name and port number App Gateway uses to proxy an enterprise application.
- In the **Apps** pane, you associate an enterprise application with a host identifier.

To register an App Gateway, you must be assigned to either the **Identity Domain Administrator** role or the **Security Administrator** role.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, click **App Gateways**, and then click **Add**.
2. In the **Details** pane, specify the name of your App Gateway, and then click **Next** (**>**).
3. In the **Hosts** pane, click **Add**.
4. In the **Add Host** dialog, provide a name in the **Host Identifier** field.
5. Enter the **Host** and **Port** values that the App Gateway server will respond to HTTP requests.

   The port number you provide in this step is used by the App Gateway server to respond to HTTP requests.
6. To have your App Gateway listen to HTTP requests in secure mode (HTTPS), select the **SSL Enabled** check box. Otherwise, clear this check box and your App Gateway will listen to non-secure HTTP requests only.
7. If you select the **SSL Enabled** check box, then populate the **Additional Properties** text area with the following values to specify the certificate key pair the App Gateway server will use, protocols and ciphers for SSL:

   ```
   ssl_certificate /usr/local/example.com.rsa.crt;
   ssl_certificate_key /usr/local/example.com.rsa.key;
   ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
   ssl_ciphers HIGH:!aNULL:!MD5;
   ```

   The **/usr/local/example.com.rsa.crt** is the full path of a certificate file in the App Gateway server. The **/usr/local/example.com.rsa.key** is the secret key of that certificate file. You must upload both files to the App Gateway server after you install the App Gateway binary file.

> **Note:**
>
> Starting with App Gateway OVA version 20.4.1-4.0.0, App Gateway will work only in SSL/HTTPS mode. Note the following considerations:
>
> a. If there is no load balancer in front of App Gateway, then populate the **Additional Properties** as specified above.
>
> b. If App Gateway is configured to be running behind a load balancer, then the load balancer must be listening over SSL/HTTPS.
>
> c. If the load balancer is listening over SSL/HTTPS and SSL is not enabled in the App Gateway settings, then the load balancer must pass the header (Name: `is_ssl` Value: `ssl`) to App Gateway.

8. In the **Add Host** dialog, click **Save**.
9. In the **Hosts** pane, click **Next >**.
10. If you have previously registered an enterprise application in Oracle Identity Cloud Service, then in the **Apps** pane, click **Add**. See Assign an Enterprise Application to an App Gateway.
11. Click **Finish**.
12. In the **App Gateway Details** page, note the value of the **Client ID**.
13. Click **Show Secret** and note the value of the **Client Secret**.

    The **Client ID** and **Client Secret** are equivalent to a credential (for example, an ID and password) that your App Gateway server uses to communicate with Oracle Identity Cloud Service. You'll need these values when you configure the App Gateway server.
14. In the **Navigation Drawer**, click **App Gateways**.
15. In the **App Gateways** page, select your App Gateway, click **Activate**, and then click **OK** in the **Confirmation** window to activate your App Gateway.

Configure the App Gateway Server

Before you start the App Gateway server for the first time, you need to configure the server to connect with Oracle Identity Cloud Service.

1. Use a SSH client such as `PuTTY` and the following credentials to log in to the App Gateway server.
   - **Localhost login**: `oracle`
   - **Password**: `cloudgateR0X!`
     You are required to change the provisioned password on the first login.
2. Execute the `sudo yum updateinfo list security all` command and provide sudo password.

   This command lists the security errata for your App Gateway Oracle Linux server. To update all packages for which security-related errata are available to the latest versions of the packages enter `sudo yum --security update`.
3. Execute the `telnet <idcs-tenant>.identity.oraclecloud.com` command to confirm that the App Gateway server can reach the Oracle Identity Cloud Service instance.
4. Restart the App Gateway server after applying the updates.

5. Navigate to the `/scratch/oracle/cloudgate/ova/bin/setup` folder, and then edit the `cloudgate-env` file present in this folder (`vi cloudgate-env`).
6. Enter values for the following parameters, and then save the file:
   - **IDCS_INSTANCE_URL**: The URL of your Oracle Identity Cloud Service instance.

     For example, `https://idcs-123456789.identity.oraclecloud.com`
   - **CG_APP_TENANT**: The tenant name of the Oracle Identity Cloud Service instance. For example, `idcs-123456789`
   - **CG_APP_NAME**: The client ID value you made note during the App Gateway registration in Identity Cloud Service console.
   - **CG_APP_SECRET**: The client secret value you made note during the App Gateway registration in Identity Cloud Service console.
   - **CG_CALLBACK_PREFIX**: If App Gateway is configured in SSL mode (HTTPS), then set the value to `https://%hostid%`. Otherwise, use `http://%hostid%` as the value for this parameter.

   > **Note:**
   >
   > Starting with App Gateway OVA version 20.4.1-4.0.0, the **CG_CALLBACK_PREFIX** value must be `https(https://%hostid%)`.

7. Confirm that the resolver entry in `/usr/local/nginx/conf/nginx-cg-sub.conf` has the right DNS server IP address.

   Execute the `nslookup <your_identity_cloud_service_domain>` command, and verify the `Server` IP Address is the same one of the `resolver` entry in the `/usr/local/nginx/conf/nginx-cg-sub.conf` file. If not, then update this file accordingly.
8. Check the OVA version being installed. If the OVA version is 20.4.1-4.0.0 or 21.2.1-5.0.0, run the following commands.
   a.
   ```
   mkdir -m 700 $NGINX_INSTALL/
   {uwsgi_temp,scgi_temp,proxy_temp,fastcgi_temp,client_body_temp}
   ```
   b.
   ```
   sed -i '/Starting nginx server successful/a setfacl -Rm
   u::rwx,d:u::rwx,g::rwx,d:g::rwx ${CLOUDGATE_NGINX}/*_temp' $
   {CLOUDGATE_HOME}/bin/cg-start; sed -i '/Reloading nginx server
   successful/a setfacl -Rm u::rwx,d:u::rwx,g::rwx,d:g::rwx $
   {CLOUDGATE_NGINX}/*_temp' ${CLOUDGATE_HOME}/bin/cg-reload; sed -i 's/
   NGX_PROCESS_WORKER="^nginx: worker process .+"/NGX_PROCESS_WORKER="^nginx:
   worker process"/g' $CLOUDGATE_OVA_BIN/jobs/watchdog-cloudgate.sh
   ```
9. In the `/scratch/oracle/cloudgate/ova/bin/setup` folder, execute `./setup-cloudgate` command.

   When prompted, enter `y` to proceed with the configuration.

   For OVA version 20.4.1-4.0.0 and greater, after running the `./setup-cloudgate` command, the values for `CG_APP_NAME` and `CG_APP_SECRET` are automatically deleted for security reasons.

App Gateway service and agent service will start after the configuration finishes.

Assign an Enterprise Application to an App Gateway

Update the App Gateway registration in Oracle Identity Cloud Service console and assign an enterprise application.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, click **App Gateways**, and then click the name of your App Gateway.
2. Click the **Apps** tab, and then click **Add**.
3. In the **Assign an App to gate** window, map App Gateway to an enterprise application using the values below, and then click **Save**.
   - **Application**: Select the enterprise application you want to protect using this App Gateway. See About Enterprise Applications.

     > **Note:**
     >
     > The enterprise application must be in activated status.

   - **Select a Host**: Select the host identifier to which the App Gateway will proxy the enterprise application.
   - **Resource Prefix**: Enter the URL prefix used by App Gateway to proxy the enterprise application.

     For example, use `/` to represent that every request since root path will be forwarded to the enterprise application you've selected.
   - **Origin Server**: This is the actual base URL where the application is hosted. If the application is not directly accessible, but accessible through a web proxy, then enter the URL of the web proxy. See example diagram below.
   - **Additional Properties**: This field is used to provide additional configuration for the application. The values specified into the field are nginx directives or statements which will be part of location block in `nginx.conf`. Examples when you would do this are:

     a. If protected applications need to do further redirects or to access resources after successful authentication with App Gateway, you can use this field to populate the host header with correct value and pass it to the application.

        For example, if a user accesses the application using `https://myappgateway.example.com:4443/home`, the browser passes the host header to App Gateway with the value set to `Host: myappgateway.example.com:4443`. This value is passed by App Gateway to the downstream application, and you can achieve this by setting either of the values below in Additional Properties:

        ```
        proxy_set_header host "myappgateway.example.com:4443";
        ```

        or

        ```
        proxy_set_header host $http_host;
        ```

        `$http_host` is a variable and its value is populated with the host header App Gateway receives from the browser or from any client.

> **Note:**
>
> If the there are load balancers sitting behind App Gateway, then it is the job of the load balancers to forward the actual host header to app gateway so that `$http_host` is populated with the correct value and App Gateway can forward it to the application.

b. If the application is accessible through a web proxy, then enter the values below:

```
proxy_set_header host "myapp.internal.example.com";
```

The `"myapp.internal.example.com"` domain is the domain name where the application is hosted, also known as origin server.

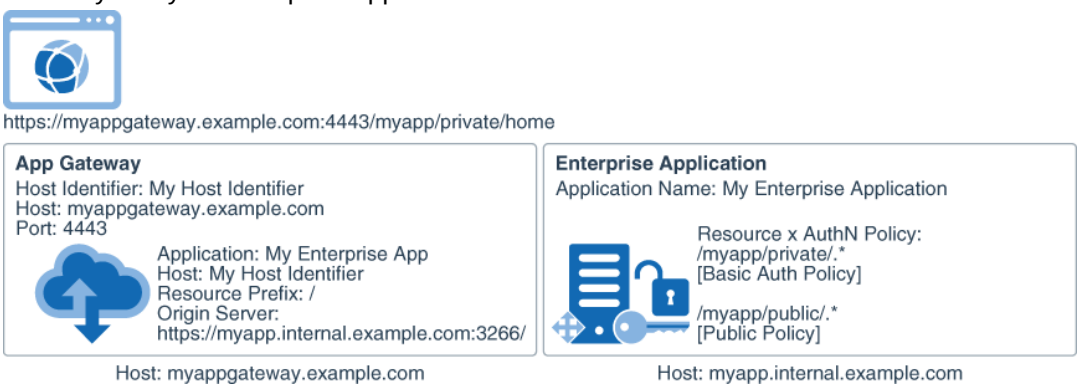
In this case, App Gateway can't pass the host header received from browser or other client and applications cannot do further redirects via App Gateway.

c. To pass through the upstream header, enter the following:

```
proxy_pass_header Server;
```

The App Gateway server header isn't used, and the upstream header is used instead.

The following figure provides examples of the mappings that you configure between App Gateway and your enterprise application:



> **Note:**
>
> You can assign multiple enterprise applications to the same App Gateway, and consequently the same App Gateway server will protect these applications.
>
> Make sure for each application, the value of **Resource Prefix** differs. For example, if you have `http://myapp.internal.example.com:3266/myapp1/page.jsp` and `http://myapp.internal.example.com:6355/myapp2/page.jsp`, both accessible through `http://myappgateway.example.com:4443/` App Gateway URL, then enter `/myapp1` as **Resource Prefix** when you register application 1, and `/myapp2` as **Resource Prefix** when you register application 2.

After you assign the application to your App Gateway, you may need to restart the App Gateway server for the changes to be effective immediately.

Enable Session Persistence with Sticky Cookies

Follow these steps to enable persistent sessions using cookies in App Gateway. The sticky cookie will always be forwarded to the same backend server.

You only need to use sticky support when you have multiple origins, and you do this by creating a nginx upstream block .

1. Enable the sticky module in App Gateway by editing the file `/usr/local/nginx/conf/nginx.conf`.
   - Below the line `load_module /scratch/oracle/cloudgate/home/lib/idcs_cloudgate_ngx.so;`, add

     ```
     load_module /scratch/oracle/cloudgate/home/lib/
     ngx_http_sticky_module.so;
     ```

   - Below the line `include /usr/local/nginx/conf/agent_conf/*.conf;`, add

     ```
     include /usr/local/nginx/conf/origin_conf/*.conf;
     ```

2. Create a nginx upstream block using

   ```
   $ vi /usr/local/nginx/conf/origin_conf/myupstream.conf

   Add below entry to myupstream.conf
   upstream weblogic {
       sticky;
       server 100.111.190.221:7003;
       server 100.111.190.220:7003;
   }
   ```

3. Change the origin server.
   a. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, click **App Gateways**, and then click the name of your App Gateway, and in the Apps tab select the App.
   b. Change the Origin Server to `http://weblogic`.

Sticky Parameters

```
upstream {
  sticky;
  server 127.0.0.1:9001;
  server 127.0.0.1:9002;
}

  sticky [hash=index|md5|sha1] [no_fallback]
       [name=route] [domain=.example.com] [path=/] [expires=1h] [secure]
[httponly];
   or
  sticky [hmac=md5|sha1 hmac_key=<foobar_key>] [no_fallback]
       [name=route] [domain=.example.com] [path=/] [expires=1h] [secure]
[httponly];
   or
```

```
  sticky [text=raw] [no_fallback]
       [name=route] [domain=.example.com] [path=/] [expires=1h] [secure]
[httponly];
```

Server Selection Algorithm

Algorithm	Description
`hash`	The hash mechanism used to encode upstream server. It can't be used with `hmac` or `text`. • `md5|sha1`. Standard cryptographic hash functions to encode the information. • `index`. The information is not hashed and instead an in-memory index is used. This is quicker and the overhead is shorter, but the matching against upstream servers list is inconsistent and if the upstream server has changed index values may not correspond to the same server. Only use `index` if you are certain you want to use it despite this. The default is `md5`.
`hmac`	The HMAC hash mechanism used to encode upstream server It's like the hash mechanism but it uses `hmac_key` to secure the hashing. It can't be used with `hash` or `text`.
`hmac_key`	The cryptographic key to use with `hmac`. You must set a `hmac_key` if you use `hmac`.
`no_fallback`	Set this flag so that if a request comes with a cookie and the corresponding backend is unavailable, a 502 (Bad Gateway or Proxy Error) is returned. You can set it to the upstream block, or set `sticky_no_fallback` in a server or location block.

Cookie Settings

Setting	Description
`name`	The name of the cookie used to track the persistant upstream server. The default is `route`.
`domain`	The domain in which the cookie will be valid. The default is `none` when the browser handles the domain.
`path`	The path in which the cookie is valid. The default is `/`.
`expires`	The validity duration of the cookie. The default is `nothing` which means that it's a session cookie and deleted when the client shuts down. Enter a value to have the cookie expire after the specified time. The value is set relative to the client, and it must be for a period greater than one second.
`secure`	Enable secure cookies (transferred only using https).

Setting	Description
`httponly`	Tells the browser that the cookie can only be accessed by the server.

Start and Stop App Gateway

To start and stop App Gateway server and App Gateway agent, you can use scripts or the services installed in the server where your App Gateway runs.

Use Script to Start and Stop App Gateway

You can start and stop the App Gateway server and agent using scripts provided in the server.

Login to the App Gateway server and then run the following command:

1. To start App Gateway server.

   ```
   /scratch/oracle/cloudgate/home/bin/cg-start
   ```

2. To start App Gateway agent.

   ```
   /scratch/oracle/cloudgate/home/bin/agent-start
   ```

3. To stop App Gateway server.

   ```
   /scratch/oracle/cloudgate/home/bin/cg-stop
   ```

4. To stop App Gateway agent.

   ```
   /scratch/oracle/cloudgate/home/bin/agent-stop
   ```

When you start the App Gateway server, App Gateway contacts Oracle Identity Cloud Service to retrieve the port number you configured during the App Gateway registration in Oracle Identity Cloud Service console. The App Gateway server starts using this port number.

The App Gateway agent is responsible for synchronizing the App Gateway configuration (hosts and applications) from Oracle Identity Cloud Service to the App Gateway server.

To check the running status of the App Gateway server, run the following command: `/scratch/oracle/cloudgate/home/bin/cg-status`

Use Service to Start and Stop App Gateway

You can start and stop the App Gateway server and agent as services running on the server.

> **Note:**
>
> Starting with App Gateway OVA version 20.4.1-4.0.0, these commands can't be used. Instead, use the commands listed in Use Script to Start and Stop App Gateway.

Login to the App Gateway server and then run the following command:

1. To start App Gateway server.

   ```
   service cloudgate-nginx start
   ```

2. To start App Gateway agent.

   ```
   service cloudgate-agent start
   ```

3. To stop App Gateway server.

   ```
   service cloudgate-nginx stop
   ```

4. To stop App Gateway agent.

   ```
   service cloudgate-agent stop
   ```

To check the running status of the App Gateway server, run the following command: `/scratch/oracle/cloudgate/home/bin/cg-status`

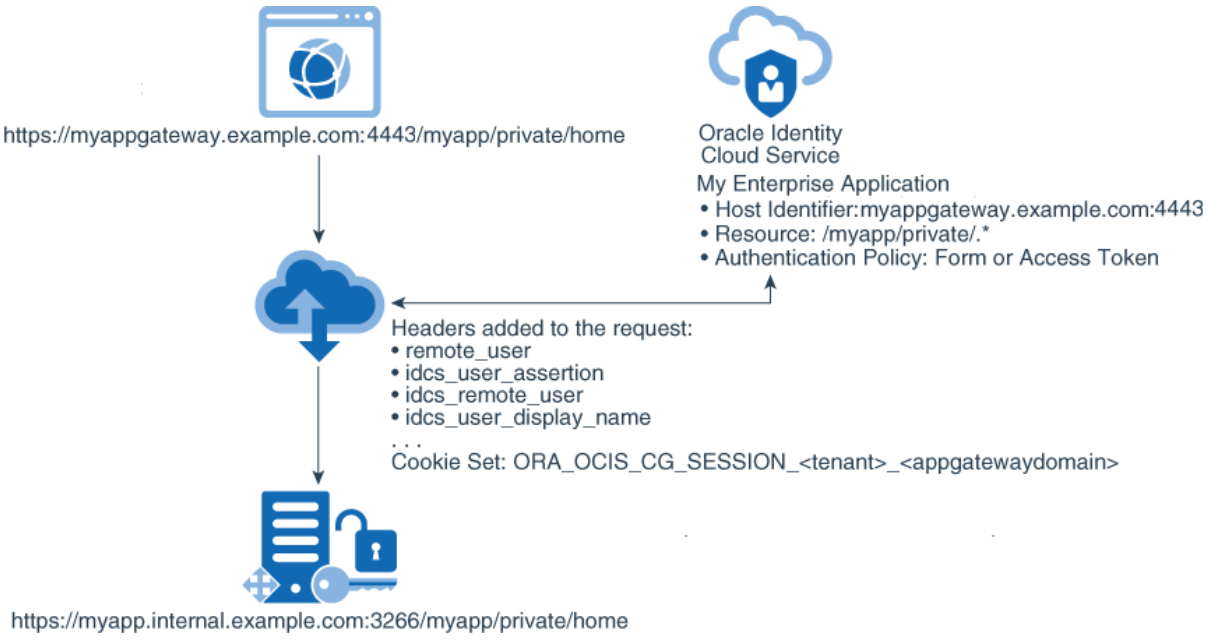
Test Access to Your Application Using App Gateway

After you configure the App Gateway server to communicate with your Oracle Identity Cloud Service instance, and start the server, test access to your enterprise application.

The following diagram provides an example of how App Gateway and Oracle Identity Cloud Service interact when an HTTP request to an application resource is sent by the user browser through App Gateway.

Because App Gateway proxies your web application, use the App Gateway base URL to access the application instead of the application actual URL.

Figure 29-2 Workflow of protecting an application using App Gateway



1. Open a new web browser and access your application using the App Gateway URL.

   In this example, the URL is: `https://myappgateway.example.com:4443/myapp/private/home`

   The actual application `https://myapp.internal.example.com:3266/myapp/private/home` isn't accessible by the user browser.
2. App Gateway intercepts the request and communicates with Oracle Identity Cloud Service to verify if the URL corresponds to an enterprise application.

   In this example, `My Enteprise Application` is registered, and the authentication policy for this enterprise application is **Form or Access Token**.
3. App Gateway verifies the request contains a valid Oracle Identity Cloud Service's access token in the `Authorization Bearer` header or Oracle Identity Cloud Service's session cookie, indicating the user has already signed in to Oracle Identity Cloud Service.
4. If the user hasn't signed in to Oracle Identity Cloud Service, then App Gateway redirects the user browser to Oracle Identity Cloud Service **Sign In** page.
5. If the user has signed in, then App Gateway adds header variables and a cookie to the request, and then forwards the request to the application.

The application receives the request, uses the header variables to identify the user and to present the content of the `/myapp/private/home` page.

How App Gateway Logout Works?

Users can log out from the applications protected by App Gateway using two different mechanisms: App Gateway Logout URL or by calling an resource protected by a logout authentication method.

Use App Gateway Logout URL

App Gateway provides a central logout URL which can be used to log the user out from the single sign-on provided by Oracle Identity Cloud Service. Any call to this endpoint triggers the logout process. After the user is logged out, then any subsequent access to a protected application resource will require the user to sign in to Oracle Identity Cloud Service again.

This endpoint supports two parameters appended to the URL:

- **postlogouturl**: The URL of a post-logout landing page. This value must be URL-encoded. If the parameter isn't specified, then App Gateway redirects the user browser to the **Logout URL** specified in theOracle Identity Cloud Service console's **Session Settings**.
- **state**: This is an optional parameter to be used by the enterprise application, after the logout process finishes.

Syntax

```
http(s)://<appgateway_host>:<appgateway_port>/cloudgate/logout.html?
postlogouturl=<url_encoded>&state=<state_value>
```

Example 29-1 Log out Endpoint With Parameters

If the App Gateway base URL is `https://myappgateway.example.com:4443`, then use the following URL to log the user out from the single sign-on: `https://myappgateway.example.com:4443/cloudgate/logout.html?postlogouturl=http%3A%2F%2Fwww.oracle.com&state=123`

Use Resource Protected by Logout Authentication Method

You can create a resource in your enterprise application and configure an authentication policy for this resource using **Forms+Logout** authentication method. When the user accesses this resource, App Gateway invokes the log out process and logs the user out from the single sign-on provided by Oracle Identity Cloud Service.

Syntax

```
http(s)://<appgateway_host>:<appgateway_port>/<logout_resource>
```

Example 29-2 Resource Protected by Logout Authentication Method

If you created `/myapp/logout` resource in your enterprise application, and assigned **Forms+Logout** as **Authentication Method** for this resource in **Authentication Policy** section, then when users access the URL `https://myappgateway.example.com:4443/myapp/logout`, they will be logged out from the single sign-on provided by Oracle Identity Cloud Service.

Run App Gateway in SSL Mode on Port `1024` or Lower

You can configure App Gateway to run in SSL mode on port number `1024` or lower.

> **Note:**
>
> To run your App Gateway server in Secure Sockets Layer (SSL) mode, you need to have an valid certificate.

Configure App Gateway in Identity Cloud Service Console

Update your App Gateway configuration to enable the server to listen on port number 443 and in Secure Sockets Layer (SSL) mode.

1. Sign in to Oracle Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, click **App Gateways**, and then click the name of your App Gateway..
2. In the **Hosts** tab, click the name of the host you created.
3. in the Edit Hosts window, update the following parameters as per the example below:

Parameter	Value
Port	`443`
SSL Enabled	Selected.

Additional Properties	`ssl_certificate /scratch/myappgateway.example.com.cert;`
	`ssl_certificate_key /scratch/myappgateway.example.com.key;`
	`ssl_protocols TLSv1 TLSv1.1 TLSv1.2;`
	`ssl_ciphers HIGH:!aNULL:!MD5;`

> **Note:**
>
> You need to generate a valid certificate to be used as the SSL certificate. The certificate file (`myappgateway.example.com.cert`) and the certificate key file (`myappgateway.example.com.key`) are referenced as an example.

4. Click **Save**.

Configure the App Gateway Server

Execute the following procedure to enable your App Gateway server to run on port 443 in SSL mode.

> **Note:**
>
> Generate a valid certificate to your App Gateway to run on SSL mode, and copy the certificate file and the certificate key file to your desktop.

1. Use a SSH client such as `PuTTY` to log in to the App Gateway server.
2. Execute the following commands to update a privileged user.
   - For App Gateway OVA 20.4.1-4.0.0 and higher.

     ```
     sed -i "s/touch \$source_log/touch \$source_log \&\& chown
     \$NGINX_USER:\$NGINX_USER \$source_log/g" /scratch/oracle/
     cloudgate/ova/bin/jobs/manage-logs.sh
     sudo sed -i "s/ oracle / root /g" /etc/cron.d/cloudgate-jobs
     sudo sed -i "/User=oracle/d" /etc/systemd/system/cloudgate-nginx.service
     sudo sed -i "/User=oracle/d" /etc/systemd/system/cloudgate-agent.service
     ```

   - For earlier App Gateway OVA versions.

     ```
     sed -i "s/touch \$source_log/touch \$source_log \&\& chown
     \$NGINX_USER:\$NGINX_USER \$source_log/g" /scratch/oracle/
     cloudgate/ova/bin/jobs/manage-logs.sh
     sudo sed -i "s/ oracle / root /g" /etc/cron.d/cloudgate-jobs
     sudo sed -i "s/sudo -u oracle//g" /etc/init.d/cloudgate-nginx
     sudo sed -i "s/sudo -u oracle//g" /etc/init.d/cloudgate-agent
     ```

3. Execute the following commands to change permission of the folders.

   ```
   sudo chmod -R 755 /scratch/
   sudo chown root:root /scratch/oracle/cloudgate/home/bin/nginx
   cd /usr/local/nginx/sbin/
   rm nginx
   sudo ln -sf /scratch/oracle/cloudgate/home/bin/nginx
   ```

4. Copy the certificate file (for example, `myappgateway.example.com.cert`) and the certificate key file (for example, `myappgateway.example.com.key`) from your desktop to the `/scratch/` folder.

5. Add user `oracle` to the `nginx.conf` file by executing the following command.

```
sudo sed -i "/working_directory.*/a user oracle;" /usr/local/nginx/conf/
nginx.conf
```

6. Edit the `/scratch/oracle/cloudgate/ova/bin/setup/cloudgate-env` file. You can use the following command or any other text editor of your choice: `vi /scratch/oracle/cloudgate/ova/bin/setup/cloudgate-env`
7. Replace the value of the `CG_CALLBACK_PREFIX` parameter with the following `https://%hostid%`
8. Save the `/scratch/oracle/cloudgate/ova/bin/setup/cloudgate-env` file.
9. Run the following `sed` commands to enable running the server with `sudo` command:

```
sed -i s/verify_running_as_user/#verify_running_as_user/g /scratch/oracle/
cloudgate/ova/bin/setup/setup-cloudgate
sudo sed -i "/create_wallet || .*/a chmod -R 755 /scratch/oracle/cloudgate/
wallet/" /scratch/oracle/cloudgate/ova/bin/setup/setup-cloudgate
```

10. Confirm the `setup-cloudgate` file is configured with the values of your Oracle Identity Cloud Service tenant, and the values of the `CG_APP_NAME` and `CG_APP_SECRET` of the App Gateway you registered in Oracle Identity Cloud Service console.
11. The `setup-cloudgate` script runs in `sudo` mode. Run the following `sed` commands to remove sudo from the commands in `setup-cloudgate` script.

```
sed -i 's/$SUDO $SYSTEMCTL/$SYSTEMCTL/g' /scratch/oracle/cloudgate/ova/bin/
setup/setup-cloudgate
sed -i 's/$SUDO \"$TEE\"/\"$TEE\"/g' /scratch/oracle/cloudgate/ova/bin/
setup/setup-cloudgate
```

12. Run the following command to reconfigure App Gateway according to the parameters registered in the Oracle Identity Cloud Service console (in this case, port number `443` and **SSL Enabled**.

```
sudo -E /scratch/oracle/cloudgate/ova/bin/setup/setup-cloudgate
```

After the `setup-cloudgate` script finishes, the App Gateway server starts automatically. You can access any application protected by your App Gateway using HTTPs, App Gateway domain, and port number `443` (default HTTPs port). For example, `https://myappgateway.example.com/myapp/index`

Start and Stop App Gateway Server Using `sudo` Command

Because you set up your App Gateway server to run on port `443`, you need to start and stop App Gateway server and agent using `sudo` command.

1. To stop the App Gateway server and agent use the following command:

```
sudo -E /scratch/oracle/cloudgate/home/bin/cg-stop
sudo -E /scratch/oracle/cloudgate/home/bin/agent-stop
```

2. To start the App Gateway server and agent use the following command:

```
sudo -E /scratch/oracle/cloudgate/home/bin/cg-start
sudo -E /scratch/oracle/cloudgate/home/bin/agent-start
```

How to Enable and Access App Gateway Logs

App Gateway provides log files to help you monitor App Gateway's behavior. Learn how to configure and access thes log files.

- Configure App Gateway Logs
- View App Gateway Logs

Configure App Gateway Logs

Enable App Gateway log files and configure logging levels.

To disable logs or change the log levels of the App Gateway server, login to the server, edit the `/usr/local/nginx/conf/cloudgate.config` file, and then under the `general` section, change the value of the `logLevel` attribute, and then save the file.

The following are default values for App Gateway:

```
"general":{
      "disableAuthorize":false,
      "logLevel":"warn",
      "logFolder":"",
      "policyMode":"gateway",
      "policyRefreshTime":300,
      "policyStaleTime":3600,
      "policyExpiryTime":604800
    }
```

> **Note:**
>
> Values for the `logLevel` attribute are: `off | crit | security | config | fail | warn | info | trace1 | trace2 | trace3`.

By Default, the log files are located in the `/usr/local/nginx/logs` folder. If you want to change the default log folder, then update the value of the `logFolder` attribute under the `general` section of the `/usr/local/nginx/conf/cloudgate.config` file.

To change the log level for the agent service of the App Gateway, modify the `/usr/local/nginx/conf/cloudgate.config` file, and set the `logLevel` and `logFolder` attributes under the `agentConfig` section as follows:

For example, to change the log level to `trace3` and the log folder to `/tmp`, update the `/usr/local/nginx/conf/cloudgate.config` file with the following values, and then save the file.

```
"agentConfig":{
   "pollIntervalSecs":60,
   "daemon":true,
```

```
 "logFolder":"/tmp",
 "logLevel":"trace3"
}
```

The log level and log folder changes takes effect next time you start App Gateway. See Start and Stop App Gateway.

View App Gateway Logs

Learn about the different log files App Gateway uses.

App Gateway is based on a `NGINX` Server. The following `NGIX` native log files are located in the `/usr/local/nginx/logs/` directory:

Table 29-1 NGINX Native Log Files

Log File	Description
`access.log`	`NGINX` Native access log contains information about all HTTP requests received by `NGINX`, and by App Gateway.
`error.log`	`NGINX` Native debug log.
`nginx.pid`	Contains the `NGINX` Server process ID number.

The following App Gateway specific log files are located in the `/usr/local/nginx/logs/` directory:

Table 29-2 App Gateway Log Files

Log File	Description
`cg-trace-main.log`	App Gateway main log file.
`cg-trace-policy.log`	Logs information about a policy refresh, when App Gateway contacts Oracle Identity Cloud Service.
`cg-trace-session.log`	Logs information about the sessions created and handled by App Gateway.
`cg-trace-token.log`	Logs information about the access tokens received by App Gateway.
`cg-trace-agent.log`	Agent logging file.
`cg-trace-init.log`	Contains information about the initialization process.

View Details About an App Gateway

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **App Gateways**.
2. In the **App Gateways** page, click the App Gateway row to view the app gateway details.

   You can modify the App Gateway configurations.

Activate and Deactivate App Gateways

You can use Oracle Identity Cloud Service to activate and deactivate app gateways.

- Deactivating an App Gateway prevents Oracle Identity Cloud Service from working with the App Gateway software.
- Activating an App Gateway enable Oracle Identity Cloud Service working with the App Gateway software.

> **Note:**
>
> A green check mark  indicates an activated App Gateway. A red circle with a white line through the circle  indicates a deactivated App Gateway.

Activate App Gateways

You can use Oracle Identity Cloud Service to activate app gateways.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **App Gateways**.
2. In the **App Gateways** page, select the check box for each App Gateway that you want to activate.
3. Click **Activate**.
4. In the **Confirmation** window, click **OK**. The status of each App Gateway changes from deactivated  to activated  .

Deactivate App Gateways

You can use Oracle Identity Cloud Service to deactivate app gateways.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **App Gateways**.
2. In the **App Gateways** page, select the check box for each App Gateway that you want to deactivate.
3. Click **Deactivate**.
4. In the **Confirmation** window, click **OK**. The status of each app gateway changes from activated  to deactivated  .

Modify an App Gateway

After viewing details about, activating, or deactivating an App Gateway, you can modify it.

Modifying an App Gateway in Oracle Identity Cloud Service includes:

- Changing the name or description of the App Gateway
- Show or regenerate the client secret
- Add or remove hosts
- Add or remove enterprise applications

To modify an App Gateway:

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **App Gateways**.
2. In the **App Gateways** page, click the App Gateway that you want to modify.

   The App Gateway page opens and displays three tabs: **Details**, **Hosts**, and **Apps**.
3. After you modify any App Gateway configuration, click **Save** to save the modification.

Remove App Gateways

You can use Oracle Identity Cloud Service to remove multiple app gateways simultaneously.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **App Gateways**.
2. In the **App Gateways** page, select the check box for each App Gateway that you want to remove.
3. Click **Remove**.
4. In the **Confirmation** window, click **OK**.

Upgrade and Patch App Gateway

Learn how to upgrade and patch App Gateway.

If you are performing a patch upgrade, the App Gateway patch will be installed when you run the upgrade script. As patches become available they will be listed in the *Oracle Identity Cloud Service What's New*.

> **Note:**
>
> The following applies only to App Gateway version 19.3.3 onwards. If you are running App Gateway version 19.2.1, then download and install a new App Gateway 19.3.3 (or newer version) server and replace the old-version server with this new-version server.

App Gateway versioning uses the following convention: `<release version>-<major version>.<minor version>.<build number>`. For example, App Gateway version `19.3.3-1.0.1`, means release `19.3.3`, major version `1`, minor version `0`, and patch version `1`.

If you have multiple App Gateway instances, then repeat the following procedure for each App Gateway server.

1. Use a SSH client such as `PuTTY` to log in to the App Gateway server.
2. Execute `cd /scratch/oracle/cloudgate`, and verify two information in this folder:

- In the command prompt, execute the following command `cat /scratch/oracle/cloudgate/INSTALLED_VERSION` to verify the version of the App Gateway.

  The following example shows that the version of the App Gateway is `19.3.3-1.0.0`:

  ```
  $ cd /scratch/oracle/cloudgate
  $ cat INSTALLED_VERSION
  OVA Base Version: 19.3.3-1.0.0
  OVA Patch Version:
  Cloud Gate Version: 19.3.3-1910012252
  ```

- Run the following command `ls -la` and verify that the `home` folder links to the folder named the App Gateway version:

  The following example shows that the `home` folder is linked to the `19.3.3.-1.0.0` folder:

  ```
  $ cd /scratch/oracle/cloudgate
  $ ls -la
  total 16
  drwx------. 6 oracle oracle 4096 Oct  2 00:23 19.3.3-1.0.0
  lrwxrwxrwx. 1 oracle oracle   38 Oct  2 01:38 home -> /scratch/oracle/
  cloudgate/19.3.3-1.0.0
  -rw-------. 1 oracle oracle   89 Oct  2 01:38 INSTALLED_VERSION
  drwx------. 3 oracle oracle 4096 Oct  2 01:38 ova
  drwxr-x---. 2 oracle oracle 4096 Oct  7 09:45 wallet
  ```

3. Execute `cd /scratch/oracle/cloudgate/home/bin`, and then `./cg-upgrade` to start the upgrade process.

   During the upgrade process, App Gateway contacts Oracle Identity Cloud Service to verify if a patch for your App Gateway is available. If so, then the process downloads the patch and applies the patch to your App Gateway server.

4. After the upgrade process finishes, execute the commands described in step 2 and verify whether their return refers to the App Gateway patch or the upgraded version.

5. Optional: Configure App Gateway in SSL mode. If you ran the `cg-upgrade` script, and App Gateway was configured in non-SSL mode, then after running the `cg-upgrade` script, complete the following steps. **Note:** If App Gateway was already configured in SSL mode, then don't complete the following steps.

   a. Get the SSL certificates.

   b. Log in to App Gateway and copy the certificates, for example, to `/scratch/certificates/`.

   c. Log in to the Oracle Identity Cloud Service Admin console, navigate to **App Gateway**, **Hosts**, and then open the required host and select **Enable SSL**.

   d. Navigate to **App Gateway**, **Hosts**, and then open the required host, navigate to **Additional Properties** and then add the path of the certificates and other information, such as `ssl_protocols` and `ssl_ciphers`.

   ```
   ssl_certificate /scratch/certificates/myappgateway.example.com.cert;
   ssl_certificate_key /scratch/certificates/myappgateway.example.com.key;
   ssl_protocols TLSv1 TLSv1.1 TLSv1.2; ssl_ciphers HIGH:!aNULL:!MD5;
   ```

e. Open `/usr/local/nginx/conf/cloudgate.config`, search for `callbackPrefix` and change its value from `HTTP` to `HTTPS`.

f. Execute the following commands so that you can see all changes reflected in App Gateway:

   i. cg-stop

   ii. cg-start

   iii. agent-stop

   iv. agent-start

   Now the application can be accessed only through the HTTPS protocol and not the HTTP protocol.

During this procedure App Gateway restarts. Access to your application through this App Gateway server may be affected.

Upgrade Path for High Availability Deployments

Cloud Gate has updated its Block Cipher mode of operation which changes how data is encrypted. The change is being rolled out over three patch releases, R1, R2, and R3 so that you can upgrade without service interruptions.

> **Note:**
>
> This upgrade path only applies when you have enabled high availability and are using multi-node deployments.

If you are using high-availability with multiple App Gateways and using a load balancer, you must follow a specific upgrade path. If you perform the upgrades in the wrong order, or miss an upgrade, then you might have problems, such as:

- Unexpected redirects to Oracle Identity Cloud Service login, because of Cloud Gate failing to decrypt its session cookie.
- Failures after login, because of Cloud Gate being unable to decrypt its state cookie or the data returned by Oracle Identity Cloud Service.
- Incomplete logouts, because of Cloud Gate being unable to decrypt the data sent by Oracle Identity Cloud Service.

R1 Patch Release

The R1 patch release encrypts using the old Block Cipher mode of operation, but it adds fail over logic to Cloud Gate's decryption operation. If Cloud Gate fails to decrypt using the current Block Cipher mode of operation, it tries again using the new Block Cipher mode of operation. This fail over allows Cloud Gate to maintain backward compatibility with session data created by older Cloud Gate clients, and support decrypting new session data created by Cloud Gate clients running the R2 or R3 patch release of this upgrade path.

R2 Patch Release

The R2 patch release encrypts and decrypts using the new Block Cipher mode of operation. Decryption supports failing over using the old Block Cipher mode of operation. The R2 patch release is not backward compatible with Cloud Gate clients from before the R1 patch release.

These older Cloud Gate clients cannot decrypt the new session data created by R2 release Cloud Gate clients.

R3 Patch Release

The table shows how the patch release relates to the Cloud gate release, and to the App Gateway Docker image. Contact Oracle Support to open a support ticket and ask to have the appropriate patch made available to you.

> **Note:**
>
> Only the patch release downloads for R1 and R2 are currently available. When R3 is available, this page will be updated.

Patch Release	Cloud Gate Release	Cloud Gate Build	App Gateway Docker
R1	22.1.49	22.1.49-2201171005	22.1.49-2201040708
R2	22.2.63	22.2.63-2203141550	22.2.57-2202180045
R3	To be announced	To be announced	To be announced

Configuration Override

You can disable the encryption change in Cloud Gate using the configuration setting:`encryptWithGcm`. This is a boolean setting that is set to `false` to disable the encryption change.

After making the change, restart the NGINX server. For example, in a WTSS deployment, use

`/u01/data/idcs-cloudgate/bin/cg-reload.`

Example of a cloudgate.config file

```
{
  "cloudgateConfig" : {
    "version"                 : "2.9",
    "comment"                 : "Sample Cloud Gate Configuration (HTTPS)",
    "enabled"                 : true,
...
    "general" : {
      "encryptWithGcm": false,
...
    },
...
}
```

Troubleshooting

This section describes some of the errors you might see if the Cloud Gate deployment mixes incompatible releases, for example, patching directly to R2, or to R3, without patching first to R1.

Failed Login

After successfully signing into Oracle Identity Cloud Service, the Cloud Gate callback (`cloudgate/v1/oauth2/callback`) will return 401 if Cloud Gate is unable to decrypt the State cookie.

Sample Logging from cg-trace-main.log

```
# First, the Cloud Gate State cookie ( ORA_OCIS_CG_ST_ ) will fail to decrypt:
[2022-04-07T18:42:34.618617+00:00] [trace3] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [SessionKeyManager.cpp] [390]
[decryptSessionData] [] decryptSessionData: using explicit crypto key
[2022-04-07T18:42:34.618693+00:00] [crit] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [SessionKeyManager.cpp] [628]
[logSessionKey] [] CRITICAL - SESSIONKEY(REGION-CRYPTO) - - TUPLE: all keys
failed (may be expected if old data)
[2022-04-07T18:42:34.618705+00:00] [crit] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [SessionKeyManager.cpp] [643]
[logSessionKey] [] CRITICAL - SESSIONKEY(REGION-CRYPTO) - -
prevSHA256=9E30456BA34D76BD5CCBFF74DBF03C734EF4097B1A8725B146E76E99000984B0
[2022-04-07T18:42:34.618713+00:00] [crit] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [SessionKeyManager.cpp] [645]
[logSessionKey] [] CRITICAL - SESSIONKEY(REGION-CRYPTO) - -
currSHA256=9E30456BA34D76BD5CCBFF74DBF03C734EF4097B1A8725B146E76E99000984B0
[2022-04-07T18:42:34.618722+00:00] [crit] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [SessionKeyManager.cpp] [647]
[logSessionKey] [] CRITICAL - SESSIONKEY(REGION-CRYPTO) - -
nextSHA256=321846513AE2657C6E0AA36EDDE38AFE8BD1B10169D204CF495EDFFE50F1AEC2
[2022-04-07T18:42:34.618733+00:00] [crit] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [SessionKeyManager.cpp] [649]
[logSessionKey] [] CRITICAL - SESSIONKEY(REGION-CRYPTO) - - expiry-loc-
svr=2022-08-05 17:39:05
[2022-04-07T18:42:34.618741+00:00] [crit] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [SessionKeyManager.cpp] [651]
[logSessionKey] [] CRITICAL - SESSIONKEY(REGION-CRYPTO) - - expiry-loc-
fix=2022-08-05 17:39:05
[2022-04-07T18:42:34.618748+00:00] [crit] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [SessionKeyManager.cpp] [653]
[logSessionKey] [] CRITICAL - SESSIONKEY(REGION-CRYPTO) - - expiry-loc-
pad=2022-08-05 17:41:06
[2022-04-07T18:42:34.618758+00:00] [fail] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [SessionKeyManager.cpp] [438]
[decryptSessionData] [ORA_CG_2302 ORA_CG_2650 ORA_CG_2653 ORA_CG_2651
ORA_CG_1621 ORA_CG_1664 ORA_CG_2652 ORA_CG_2656 ORA_CG_2302 ORA_CG_2650
ORA_CG_2653 ORA_CG_2651 ORA_CG_1621 ORA_CG_1664 ORA_CG_2652 ORA_CG_2656
ORA_CG_2302 ORA_CG_2650 ORA_CG_2653 ORA_CG_2651 ORA_CG_1621]
decryptSessionData: FAIL - all keys failed - SESSIONKEY(REGION-CRYPTO)
dataKeyID=9E30456BA34D76BD5CCBFF74DBF03C734EF4097B1A8725B146E76E99000984B0
[2022-04-07T18:42:34.618786+00:00] [fail] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [EncodingEncryptor.cpp] [104]
[decrypt] [ORA_CG_2302 ORA_CG_2650 ORA_CG_2653 ORA_CG_2651 ORA_CG_1621
ORA_CG_1664 ORA_CG_2652 ORA_CG_2656 ORA_CG_2302 ORA_CG_2650 ORA_CG_2653
ORA_CG_2651 ORA_CG_1621 ORA_CG_1664 ORA_CG_2652 ORA_CG_2656 ORA_CG_2302
ORA_CG_2650 ORA_CG_2653 ORA_CG_2651 ORA_CG_1621 ORA_CG_1664] decrypt failed
(bad key/data?)
[2022-04-07T18:42:34.618793+00:00] [trace3] [P:290] [T:0]
```

```
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [EncodingEncryptor.cpp] [108]
[decrypt] [] decrypt: b64=547 cry=410 out=0 bytes
[2022-04-07T18:42:34.618801+00:00] [trace2] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [CookieBase.cpp] [117]
[initializeFromRequest] [] Cookie decryption failed
[name=ORA_OCIS_CG_ST_cgdev-tenant1_cgdev-tenant1.cgdevcloud.test]
[2022-04-07T18:42:34.618815+00:00] [trace2] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [CookieManager.cpp] [41]
[createCookie] [] Added cookie [name=ORA_OCIS_CG_ST_cgdev-tenant1_cgdev-
tenant1.cgdevcloud.test] [type=REQUEST_STATE] [initialized=0]
[existsInRequest=1] [valid=0] [last-status=ERR_Decrypt_Failed] to
CookieManager

# Next, the ID Token from IDCS will fail to decrypt:
[2022-04-07T18:42:34.624873+00:00] [trace3] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [OAuthFlows.cpp] [3398]
[getIdTokenInImplicitFlow] [] IDCS_CG_ENC isSecretKey=1
[2022-04-07T18:42:34.625048+00:00] [trace3] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [SessionKeyManager.cpp] [628]
[logSessionKey] [] SESSIONKEY(REGION-SECRET) - - TUPLE: attempting decrypt
with keys
[2022-04-07T18:42:34.625061+00:00] [trace3] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [SessionKeyManager.cpp] [637]
[logSessionKey] [] SESSIONKEY(REGION-SECRET) - - currSHA256=
[2022-04-07T18:42:34.625069+00:00] [trace3] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [SessionKeyManager.cpp] [395]
[decryptSessionData] [] decryptSessionData: using explicit regional secret
[2022-04-07T18:42:34.625154+00:00] [crit] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [SessionKeyManager.cpp] [628]
[logSessionKey] [] CRITICAL - SESSIONKEY(REGION-SECRET) - - TUPLE: all keys
failed (may be expected if old data)
[2022-04-07T18:42:34.625168+00:00] [crit] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [SessionKeyManager.cpp] [637]
[logSessionKey] [] CRITICAL - SESSIONKEY(REGION-SECRET) - - currSHA256=
[2022-04-07T18:42:34.625177+00:00] [fail] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [SessionKeyManager.cpp] [438]
[decryptSessionData] [ORA_CG_2302 ORA_CG_2650 ORA_CG_2653 ORA_CG_2651
ORA_CG_1621 ORA_CG_1664 ORA_CG_2652 ORA_CG_2656 ORA_CG_2302 ORA_CG_2650
ORA_CG_2653 ORA_CG_2651 ORA_CG_1621 ORA_CG_1664 ORA_CG_2652 ORA_CG_2656
ORA_CG_2302 ORA_CG_2650 ORA_CG_2653 ORA_CG_2651 ORA_CG_1621 ORA_CG_1664
ORA_CG_2652 ORA_CG_2656 ORA_CG_2641 ORA_CG_2642 ORA_CG_1621]
decryptSessionData: FAIL - all keys failed - SESSIONKEY(REGION-SECRET)
dataKeyID=28C0F2CE5E3B385F9F28E7BED8EC26C5B171E1D65E66FCD2400112EB6B743EAD
[2022-04-07T18:42:34.625220+00:00] [fail] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [EncodingEncryptor.cpp] [104]
[decrypt] [ORA_CG_2302 ORA_CG_2650 ORA_CG_2653 ORA_CG_2651 ORA_CG_1621
ORA_CG_1664 ORA_CG_2652 ORA_CG_2656 ORA_CG_2302 ORA_CG_2650 ORA_CG_2653
ORA_CG_2651 ORA_CG_1621 ORA_CG_1664 ORA_CG_2652 ORA_CG_2656 ORA_CG_2302
ORA_CG_2650 ORA_CG_2653 ORA_CG_2651 ORA_CG_1621 ORA_CG_1664 ORA_CG_2652
ORA_CG_2656 ORA_CG_2641 ORA_CG_2642 ORA_CG_1621 ORA_CG_1664] decrypt failed
(bad key/data?)
[2022-04-07T18:42:34.625225+00:00] [trace3] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [EncodingEncryptor.cpp] [108]
[decrypt] [] decrypt: b64=2748 cry=2061 out=0 bytes
[2022-04-07T18:42:34.625233+00:00] [trace1] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [OAuthFlows.cpp] [3421]
```

```
[getIdTokenInImplicitFlow] [ORA_CG_2302 ORA_CG_2650 ORA_CG_2653 ORA_CG_2651
ORA_CG_1621 ORA_CG_1664 ORA_CG_2652 ORA_CG_2656 ORA_CG_2302 ORA_CG_2650
ORA_CG_2653 ORA_CG_2651 ORA_CG_1621 ORA_CG_1664 ORA_CG_2652 ORA_CG_2656
ORA_CG_2302 ORA_CG_2650 ORA_CG_2653 ORA_CG_2651 ORA_CG_1621 ORA_CG_1664
ORA_CG_2652 ORA_CG_2656 ORA_CG_2641 ORA_CG_2642 ORA_CG_1621 ORA_CG_1664
ORA_CG_2551] id_token decode-and-decryption failed
[2022-04-07T18:42:34.625246+00:00] [trace1] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [OAuthFlows.cpp] [3683]
[completeOauthBrowserFlow] [ORA_CG_2302 ORA_CG_2650 ORA_CG_2653 ORA_CG_2651
ORA_CG_1621 ORA_CG_1664 ORA_CG_2652 ORA_CG_2656 ORA_CG_2302 ORA_CG_2650
ORA_CG_2653 ORA_CG_2651 ORA_CG_1621 ORA_CG_1664 ORA_CG_2652 ORA_CG_2656
ORA_CG_2302 ORA_CG_2650 ORA_CG_2653 ORA_CG_2651 ORA_CG_1621 ORA_CG_1664
ORA_CG_2652 ORA_CG_2656 ORA_CG_2641 ORA_CG_2642 ORA_CG_1621 ORA_CG_1664
ORA_CG_2551 ORA_CG_2534] Could not get ID token

# Cloud Gate will attempt to retry the login flow, but this fails as the
State cookie failed to decrypt:
[2022-04-07T18:42:34.625251+00:00] [trace1] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [OAuthFlows.cpp] [3686]
[completeOauthBrowserFlow] [] Retrying /authorize
[2022-04-07T18:42:34.625255+00:00] [trace2] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [OAuthFlows.cpp] [3314]
[retryOauthBrowserFlow] [] Entry
[2022-04-07T18:42:34.625263+00:00] [trace1] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [OAuthFlows.cpp] [3276]
[validateStateCookie] [ORA_CG_2302 ORA_CG_2650 ORA_CG_2653 ORA_CG_2651
ORA_CG_1621 ORA_CG_1664 ORA_CG_2652 ORA_CG_2656 ORA_CG_2302 ORA_CG_2650
ORA_CG_2653 ORA_CG_2651 ORA_CG_1621 ORA_CG_1664 ORA_CG_2652 ORA_CG_2656
ORA_CG_2302 ORA_CG_2650 ORA_CG_2653 ORA_CG_2651 ORA_CG_1621 ORA_CG_1664
ORA_CG_2652 ORA_CG_2656 ORA_CG_2641 ORA_CG_2642 ORA_CG_1621 ORA_CG_1664
ORA_CG_2551 ORA_CG_2534 ORA_CG_2539] State cookie invalid
[2022-04-07T18:42:34.625268+00:00] [trace2] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [OAuthFlows.cpp] [3347]
[retryOauthBrowserFlow] [] Second retry
[2022-04-07T18:42:34.625272+00:00] [trace2] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [OAuthFlows.cpp] [3354]
[retryOauthBrowserFlow] [] Exit, success=0
[2022-04-07T18:42:34.625284+00:00] [trace3] [P:290] [T:0]
[E:1.0448b21d2f58bc605049585de68f149d;kXjE] [PlatformUtil.cpp] [602]
[addResponseHeader] [] www-authenticate: Bearer error="invalid_session",
error_description="Authentication Failure"
```

Login Loop

If there is an existing Cloud Gate Session, or Oracle Identity Cloud Service SSO session, or both, you might see a login loop, similar to the loop caused by Cloud Gate Session cookies being too large.

When Cloud Gate cannot decrypt the existing Cloud Gate Session cookie, it will redirect to Oracle Identity Cloud Service to kick off authentication (see the `/oauth2/v1/authorize` request).

The initial request to `/smoke/test/oauth/echo` goes to a Cloud Gate node that hasn't been patched to R1. As it cannot detect a valid Cloud Gate Session, the unpatched Cloud Gate redirects to Oracle Identity Cloud Service to log in.

The Cloud Gate callback goes to the R2 Cloud Gate node. As the R2 release supports both Block Cipher modes of operation, it is able to decrypt the Cloud Gate State cookie and create a new Cloud Gate Session (encrypted using the new Block Cipher mode of operation).

The `/smoke/test/oauth/echo` replay request goes to the unpatched Cloud Gate node. And, again, it fails to decrypt the Cloud Gate Session cookie.

This is the login loop.

Sample Logging from cg-trace-main.log

```
#  The login loop is caused by the Cloud Gate Session cookie
( ORA_OCIS_CG_SESSION_ ) failing to decrypt:
[2022-04-07T20:10:04.971799+00:00] [trace3] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [SessionKeyManager.cpp] [357]
[decryptSessionData] [] decryptSessionData: using default regional session key
[2022-04-07T20:10:04.971815+00:00] [trace3] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [SessionKeyManager.cpp] [628]
[logSessionKey] [] SESSIONKEY(REGION-CRYPTO) - - TUPLE: attempting decrypt
with keys
[2022-04-07T20:10:04.971823+00:00] [trace3] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [SessionKeyManager.cpp] [643]
[logSessionKey] [] SESSIONKEY(REGION-CRYPTO) - -
prevSHA256=9E30456BA34D76BD5CCBFF74DBF03C734EF4097B1A8725B146E76E99000984B0
[2022-04-07T20:10:04.971832+00:00] [trace3] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [SessionKeyManager.cpp] [645]
[logSessionKey] [] SESSIONKEY(REGION-CRYPTO) - -
currSHA256=9E30456BA34D76BD5CCBFF74DBF03C734EF4097B1A8725B146E76E99000984B0
[2022-04-07T20:10:04.971840+00:00] [trace3] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [SessionKeyManager.cpp] [647]
[logSessionKey] [] SESSIONKEY(REGION-CRYPTO) - -
nextSHA256=321846513AE2657C6E0AA36EDDE38AFE8BD1B10169D204CF495EDFFE50F1AEC2
[2022-04-07T20:10:04.971853+00:00] [trace3] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [SessionKeyManager.cpp] [649]
[logSessionKey] [] SESSIONKEY(REGION-CRYPTO) - - expiry-loc-svr=2022-08-05
17:39:05
[2022-04-07T20:10:04.971871+00:00] [trace3] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [SessionKeyManager.cpp] [651]
[logSessionKey] [] SESSIONKEY(REGION-CRYPTO) - - expiry-loc-fix=2022-08-05
17:39:05
[2022-04-07T20:10:04.971879+00:00] [trace3] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [SessionKeyManager.cpp] [653]
[logSessionKey] [] SESSIONKEY(REGION-CRYPTO) - - expiry-loc-pad=2022-08-05
17:41:06
[2022-04-07T20:10:04.971887+00:00] [trace3] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [SessionKeyManager.cpp] [390]
[decryptSessionData] [] decryptSessionData: using explicit crypto key
[2022-04-07T20:10:04.971978+00:00] [crit] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [SessionKeyManager.cpp] [628]
[logSessionKey] [] CRITICAL - SESSIONKEY(REGION-CRYPTO) - - TUPLE: all keys
failed (may be expected if old data)
[2022-04-07T20:10:04.971988+00:00] [crit] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [SessionKeyManager.cpp] [643]
[logSessionKey] [] CRITICAL - SESSIONKEY(REGION-CRYPTO) - -
prevSHA256=9E30456BA34D76BD5CCBFF74DBF03C734EF4097B1A8725B146E76E99000984B0
[2022-04-07T20:10:04.971996+00:00] [crit] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [SessionKeyManager.cpp] [645]
```

```
[logSessionKey] [] CRITICAL - SESSIONKEY(REGION-CRYPTO) - -
currSHA256=9E30456BA34D76BD5CCBFF74DBF03C734EF4097B1A8725B146E76E99000984B0
[2022-04-07T20:10:04.972004+00:00] [crit] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [SessionKeyManager.cpp] [647]
[logSessionKey] [] CRITICAL - SESSIONKEY(REGION-CRYPTO) - -
nextSHA256=321846513AE2657C6E0AA36EDDE38AFE8BD1B10169D204CF495EDFFE50F1AEC2
[2022-04-07T20:10:04.972027+00:00] [crit] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [SessionKeyManager.cpp] [649]
[logSessionKey] [] CRITICAL - SESSIONKEY(REGION-CRYPTO) - - expiry-loc-
svr=2022-08-05 17:39:05
[2022-04-07T20:10:04.972034+00:00] [crit] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [SessionKeyManager.cpp] [651]
[logSessionKey] [] CRITICAL - SESSIONKEY(REGION-CRYPTO) - - expiry-loc-
fix=2022-08-05 17:39:05
[2022-04-07T20:10:04.972041+00:00] [crit] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [SessionKeyManager.cpp] [653]
[logSessionKey] [] CRITICAL - SESSIONKEY(REGION-CRYPTO) - - expiry-loc-
pad=2022-08-05 17:41:06
[2022-04-07T20:10:04.972050+00:00] [fail] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [SessionKeyManager.cpp] [438]
[decryptSessionData] [ORA_CG_2302 ORA_CG_2650 ORA_CG_1621]
decryptSessionData: FAIL - all keys failed - SESSIONKEY(REGION-CRYPTO)
dataKeyID=9E30456BA34D76BD5CCBFF74DBF03C734EF4097B1A8725B146E76E99000984B0
[2022-04-07T20:10:04.972062+00:00] [fail] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [EncodingEncryptor.cpp] [104]
[decrypt] [ORA_CG_2302 ORA_CG_2650 ORA_CG_1621 ORA_CG_1664] decrypt failed
(bad key/data?)
[2022-04-07T20:10:04.972067+00:00] [trace3] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [EncodingEncryptor.cpp] [108]
[decrypt] [] decrypt: b64=3174 cry=2380 out=0 bytes
[2022-04-07T20:10:04.972074+00:00] [trace2] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [CookieBase.cpp] [117]
[initializeFromRequest] [] Cookie decryption failed
[name=ORA_OCIS_CG_SESSION_cgdev-tenant1_cgdev-tenant1.cgdevcloud.test]
[2022-04-07T20:10:04.972087+00:00] [trace2] [P:290] [T:0]
[E:1.ae9c054fa6fe4419bc5e1857e94958ed;kXjE] [CookieManager.cpp] [41]
[createCookie] [] Added cookie [name=ORA_OCIS_CG_SESSION_cgdev-tenant1_cgdev-
tenant1.cgdevcloud.test] [type=SESSION] [initialized=0] [existsInRequest=1]
[valid=0] [last-status=ERR_Decrypt_Failed] to CookieManager
```

Cross-Domain Log out Failure

The cross-domain logout flow might fail when third party cookies are disabled and there is a Cloud Gate NGINX server which has been patched to R2, and two Cloud Gate NGINX servers which haven't been patched.

When the R2 server initiates logout, the unpatched nodes fail to decrypt the `LOGOUT_DATA` post body submitted to Cloud Gate by Oracle Identity Cloud Service.

The cg-trace-main.log file notes decryption failures, such as:

- `all keys failed (may be expected if old data)`
- `decrypt failed (bad key/data?)`

Troubleshoot App Gateway

Learn about common problems that you might encounter when setting up App Gateway and how to solve them:

- I Made Changes in Oracle Identity Cloud Service but the App Gateway Server Doesn't Reflect the Changes
- Error Log Files Contain Invalid_session Message
- Error Log Files Contain GET 127.0.0.1:53 Command Responding Error Number 500
- App Gateway Server Can't Communicate With Oracle Identity Cloud Service

My Response Error Message Contains: 400 Bad Request: invalid header value

Learn the common cause when a response error message contains: `400 Bad Request: invalid header value`.

App Gateway adds headers to the requests that are proxied to an upstream Application Server. One of these headers, `idcs_user_display_name`, might have invalid characters as defined by the newer RFC - depending on the values set for the **First Name** and **Last Name** of the Oracle Identity Cloud Service user. This new RFC limits the allowed characters to printable US-ASCII characters (that is, 0x21 - 0x7E and the space and horizontal tab characters). See RFC 7230 HTTP/1.1 Message Syntax and Routing.

Application Servers that enforce the newer RFC will reject the request with the response: `400 Bad Request: invalid header value`. **Note:** The exact response depends on the Application Server being used.

To resolve this issue, remove any nonprintable characters.

I Made Changes in Oracle Identity Cloud Service but the App Gateway Server Doesn't Reflect the Changes

Changes you make to enterprise applications and App Gateway definitions in Oracle Identity Cloud Service may not be reflected immediately on App Gateway because App Gateway caches Oracle Identity Cloud Service information, such as resources, authentication policies, and header values of enterprise applications.

Explanation: App Gateway contacts Oracle Identity Cloud Service using agents to collect host and port information. When you start App Gateway, its NGINX server is automatically configured with this information. Any changes to Oracle Identity Cloud Service is periodically polled by the agents.

By default the policy and headers refresh time are 3600 seconds (1 hour) each. To change these values, log in to the App Gateway server, and edit the `/usr/local/nginx/conf/cloudgate.config` file. Change the `ttl` value for `policy` and `headers` in the `caching` section as per the following example, and then restart both App Gateway server and the agent.

```
"caching" : {
  "minimumTtl"            : 300,
  "headers"               : { "ttl": 3600 },
  "discovery"             : { "ttl": 3600 },
```

```
  "policy"                : { "ttl": 3600},
  "tenantKeys"            : { "ttl": 86400 }
}
```

You can also change the poll interval of the agents. By default, the agent's refresh time to get new App Gateway configuration from Oracle Identity Cloud Service is 60 seconds, which is the minimum amount of time supported. In the `/usr/local/nginx/conf/cloudgate.config` file, change the `pollIntervalSecs` value in the `agentConfig` section as per the example:

```
"agentConfig": {
    "pollIntervalSecs"    : 60,
    "daemon"          : true,
    "logLevel"         : "warn",
    "logFolder"         : ""
}
```

If you want the changes in the Enterprise Application configuration to be reflected immediately, stop the App Gateway server and then start the server.

```
/scratch/oracle/cloudgate/home/bin/cg-stop
/scratch/oracle/cloudgate/home/bin/cg-start
```

If you want the changes in the App Gateway configuration to be reflected immediately, stop the agent and then start the agent.

```
/scratch/oracle/cloudgate/home/bin/agent-stop
/scratch/oracle/cloudgate/home/bin/agent-start
```

See Start and Stop App Gateway

Error Log Files Contain Invalid_session Message

When App Gateway can't communicate correctly with Oracle Identity Cloud Service, you'll find `invalid_session` messages in the App Gateway error log files.

The following is an example of an `invalid_session` messages in `error.log` file:

```
www-authenticate: Bearer error="invalid_session",
error_description="Authentication Failure
```

This can be because of the way App Gateway processes a client request to a protected resource. App Gateway uses `NGINX` sub requests to make requests to Oracle Identity Cloud Service, and then App Gateway requires Linux `NGINX` resolver to be configured appropriately to allow these sub requests to function correctly.

1. Verify that the resolver setting in the file `/usr/local/nginx/conf/nginx-cg-sub.conf` is set to the correct IP.
2. Verify that the tenant name in `/usr/local/nginx/conf/cloudgate.config` file is configured correctly.

Error Log Files Contain GET 127.0.0.1:53 Command Responding Error Number 500

Because App Gateway makes sub requests to an internal servlet, App Gateway requires your virtual machine to listen to port `53`.

The App Gateway server must communicate to itself through IP address `127.0.0.1` and port `53`.

If you're running App Gateway in a virtual machine software, configure port forward for this port from the host to the guest. See Configure Port Forwarding Rules

App Gateway Server Can't Communicate With Oracle Identity Cloud Service

Use a SSH client such as `PuTTY` and the following credentials to log in to the App Gateway server:

1. Execute the `sudo su -` command to login as `root`, and when prompted provide the oracle password.
2. Install telnet by running the following command:

   ```
   yum install telnet
   ```

3. Run the following telnet command and try to establish a connection to your Oracle Identity Cloud Service instance and your application from the App Gateway server.

   ```
   telnet <idcs-tenant>.identity.oraclecloud.com 443
   ```

   If telnet can't connect to your Oracle Identity Cloud Service, then contact your network administrator to apply any other network configuration to enable the App Gateway server to establish connection with your Oracle Identity Cloud Service instance.
4. Execute the `exit` command, to log out from root account.

Configuring Cloud Gate CORS Settings in Oracle Identity Cloud Service

Learn how to configure Cloud Gate CORS settings in Oracle Identity Cloud Service.

If you need to configure Cloud Gate CORS settings in Oracle Identity Cloud Service, then you use the Oracle Identity Cloud Service REST API. See Configuring Cloud Gate CORS Settings in Oracle Identity Cloud Service.

30

Manage Account Recovery in Oracle Identity Cloud Service

This section describes how to manage account recovery in Oracle Identity Cloud Service.

Topics:

Typical Workflow for Managing Account Recovery in Oracle Identity Cloud Service

With the account recovery feature in Oracle Identity Cloud Service, you can perform tasks such as configuring account recovery. This way, if users have trouble signing in, they're locked out, or they forget their passwords, then they can regain access to their accounts.

Task	Description	Additional Information
Configure account recovery.	Configure account recovery using the **Account Recovery Settings** page.	Configure Account Recovery

You can configure account recovery by:

- The Identity Cloud Service console
- SCIM-based APIs

The following sections describe how to manage account recovery by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Configure Account Recovery

Account recovery is an automated process designed to help users regain access to their accounts if they have trouble signing in, they're locked out, or they forget their passwords.

There are three account recovery factors that identity domain administrators and security administrators can configure for users:

- Security questions: You can allow a user to select and answer security questions, and provide hints for answers to these questions, to verify their identity. If they have to recover their account, then they must answer these questions correctly to regain access.
- Email: By default, a user's primary email address has been set as the email address that Oracle Identity Cloud Service will use to help the user recover their account. If the user has

to regain access, then Oracle Identity Cloud Service will send a notification to this email address. The user follows the instructions in the notification to recover their account.

Instead of their primary email address, you can allow the user to specify an alternate (recovery) email address to regain access to their account.

- Text message (SMS): You can allow a user to provide a mobile number that Oracle Identity Cloud Service will use to help them recover access to their account. This way, if they have to regain access, then Oracle Identity Cloud Service will send a passcode in a text message (SMS) to this mobile number. The user enters this passcode to recover their account.

> **Tip:**
>
> This account recovery factor is useful for users without Internet connectivity.

> **Important:**
>
> Because you want users to be able to regain access to their accounts, you must set at least one account recovery factor for them.

In addition to setting account recovery factors, identity domain administrators and security administrators can specify:

- How many consecutive, unsuccessful account recovery attempts a user can make before the user's account is locked.
- How long the user's account will be locked before they can attempt to recover their account again.

You can access the Manage Account Recovery in Oracle Identity Cloud Service infographic to see how to configure account recovery factors for users.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Account Recovery**. The **Account Recovery Settings** page appears.
2. Use the following table to populate the **Account Recovery Settings** page:

Field	Description
Security Questions	If you want users to be able to configure security questions to recover their accounts, then select this check box. Otherwise, deselect this check box, and in the **Deactivate Security Questions?** dialog box, click **Deactivate Security Questions**. If you select this check box, then click **Configure** to set up security questions that users can manage for their accounts. See Set Up a Mobile Number As An Authentication Method.

Field	Description
Email	If you want users to be able to specify an email address other than their primary email address to recover their accounts, then select this check box. Otherwise, deselect this check box, and in the **Deactivate Email?** dialog box, click **Deactivate Email**. If you select this check box, then click **Configure** to define the settings for the notification that's sent to the user. See Configure Email Settings.
Text Message (SMS)	If you want users to be able to provide a mobile number to recover their accounts, then select this check box. Otherwise, deselect this check box, and in the **Deactivate SMS?** dialog box, click **Deactivate SMS**. If you select this check box, then click **Configure** to define the settings in Oracle Identity Cloud Service for sending a passcode as a text message (SMS) to the user. See Configure One-Time Passcode Text Messages.
Maximum consecutive unsuccessful recovery attempts	Specify the number of consecutive, unsuccessful account recovery attempts after which the user's account is locked.
Lockout duration	Specify (in minutes) how long the user's account will be locked (because they exceeded the setting in the **Maximum consecutive unsuccessful recovery attempts** field) before the user can attempt to recover their account again.

3. Click **Save**.
4. In the **Confirmation** window, click **OK**.

   Users can set up account recovery for their accounts. See Set Your Account Recovery Options.

31

Manage Oracle Identity Cloud Service Multi-Factor Authentication Settings

Learn how to configure security settings such as Multi-Factor Authentication (MFA) for your environments.

Topics

Typical Workflow for Managing Oracle Identity Cloud Service Multi-Factor Authentication Settings

Use the **MFA** and **Factors** pages within the **Security** tab in Oracle Identity Cloud Service to perform tasks such as configuring MFA settings and configuring the authentication factors that you want to use.

Task	Description	Additional Information
Understand MFA	Learn about MFA and the types of authentication factors that are supported in Oracle Identity Cloud Service.	Understand Multi-Factor Authentication
Enable MFA	Enable MFA by adding a sign-on rule for MFA.	Add a Sign-On Policy
Configure MFA	Configure overall MFA policy settings such as the type of factors that you want to allow and compliance policies using the **Multi-Factor Authentication (MFA) Settings** page.	Configure Authentication Factors
Disable MFA	Disable MFA by deactivating the MFA sign-on rule.	Deactivate Sign-On Policies

You can access the Configure Multi-Factor Authentication (MFA) infographic to see how to enable and configure MFA.

You can enable, manage, and disable MFA by using:

- The Oracle Identity Cloud Service administration console
- SCIM-based APIs

In the following sections, you learn how to manage MFA by using the Oracle Identity Cloud Service administration console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Understand Multi-Factor Authentication

Multi-Factor Authentication (MFA) is a method of authentication that requires the use of more than one factor to verify a user's identity.

With MFA enabled in Oracle Identity Cloud Service, when a user signs in to an application, they are prompted for their user name and password, which is the first factor – something that they know. The user is then required to provide a second type of verification. This is called 2-Step Verification. The two factors work together to add an additional layer of security by using either additional information or a second device to verify the user's identity and complete the login process.

MFA may include any two of the following:

- Something that you **know**, like a password.
- Something that you **have**, like a device.
- Something that you **are**, like your fingerprint.

Users are increasingly connected, accessing their accounts and applications from anywhere. As an administrator, when you add MFA on top of the traditional user name and password, that helps you to protect access to data and applications. This also reduces the likelihood of online identity theft and fraud, which secures your business applications even if an account password is compromised.

Configure Multi-Factor Authentication Settings

Configure tenant-specific Multi-Factor Authentication (MFA) settings and compliance policies that define which authentication factors that you want to allow.

To define MFA settings in Oracle Identity Cloud Service, you must be assigned to either the identity domain administrator role or the security administrator role.

1. In the Oracle Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then **MFA**.
2. Under **Select the factors that you want to enable**, select each of the factors that you want to be available for your users to select.
3. Edit an existing sign-on rule or add a new sign-on rule. In order for a user to be prompted for an additional authentication factor, the sign-on policy rule that is applied to that user must have these settings in the **Edit...** dialog box, **Actions** section:
   - **Access is** set to **Allowed**.
   - **Prompt for additional factor** selected.
   - **Enrollment** set to **Required**, or the user will be allowed to skip the additional authentication factor.

   See Add a Sign-On Policy.
4. (Optional) Click the **Configure** link for MFA factors you have selected to configure them individually.

You can do this later. If you want to configure these settings now, see Configure Authentication Factors.

If you're configuring a one-time passcode **Phone Call**, see Configure One-Time Passcode Phone Calls for configuration information.

5. Use the **Trusted Device(s)** section to configure trusted device settings.

   Similar to “remember my computer,” trusted devices don’t require the user to provide secondary authentication each time that they sign in (for a defined time period).

6. In the **Factors** section, set the **Maximum number of enrolled factors** that your users can configure.
7. In the **Login Rules** section, set the **Maximum unsuccessful MFA attempts** that you want to allow a user to incorrectly provide MFA verification before being locked out.
8. Click **Save**, and then click **OK** in the **Confirmation** window.
9. Ensure that any sign-in policies that are active allow two-step authentication:
   a. In the Oracle Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then **Sign-In Policies**.
   b. On the **Sign-In Policies** page, click **Default Sign-In Policy**.
   c. On the **Default Sign-In Policy**, select the **Sign-On Rules** tab.
   d. In the **Default Sign-On Rule** row, click the **Menu** icon and select **Edit**.
   e. In the **Edit Default Sign-On Rule** dialog box, ensure that **Actions** is set to **Allowed** and **Prompt for an additional factor** is selected.
   f. If you changed any settings, click **Save**.
   g. If other sign-on policies have been added, follow steps c-d above for each of those policies to ensure that MFA is enabled under all conditions where you want it to be enabled.

> **Note:**
>
> The settings for the default sign-in rule enable MFA globally. Settings for other sign-in rules may override the default sign-in rule for users and groups specified by conditions for those rules. See Manage Oracle Identity Cloud Service Sign-On Policies.

Configure Authentication Factors

Oracle Identity Cloud Service offers a variety of Multi-Factor Authentication (MFA) factors that you can configure.

The following is a brief overview of the authentication factors available for use with 2-Step Verification.

- **Mobile App Passcode**: Use an authenticator app, such as the Oracle Mobile Authenticator (OMA) app to generate an OTP. An OTP can be generated even when the user's device is offline. After the user enters their user name and password, a prompt appears for the passcode. The user obtains a generated passcode from the app, and then enters the code as the second verification method.

Oracle Identity Cloud Service also works with any third-party authentication app that adheres to the TOTP: Time-Based One-Time Password Algorithm specification, such as the Google Authenticator.

- **Mobile App Notification**: Send a push notification that contains an approval request to allow or deny a login attempt. Push notifications are an easy and quick way to authenticate. After the user enters their user name and password, a login request is sent to the app on their phone. The user taps **Allow** to authenticate.
- **Security Questions**: Prompt the user to answer security questions to verify their identity. After the user enters their user name and password, they must answer a defined number of security questions as the second verification method.
- **Text Message (SMS) or Phone Call**: Send a passcode as a text message (SMS) or as a phone call to the user. This method is useful for users without Internet connectivity. After the user enters their user name and password, Oracle Identity Cloud Service sends a passcode to their device for use as a second verification method.
- **Recovery Email**: Send a one-time passcode in an email to the user's recovery email address. After the user selects **Recovery Email** as the authentication method, Oracle Identity Cloud Service sends a one-time passcode to the user's recovery email address for use as a second verification method. The user's **Recovery Email** address is defined in the user's Oracle Identity Cloud Service account.
- **Email**: Send a one-time passcode in an email to the user. After the user selects **Email** as the authentication method, Oracle Identity Cloud Service sends a one-time passcode to the user's primary email address for use as a second verification method. The user's primary email (**Email**) address is defined in the user's Oracle Identity Cloud Service account.
- **Bypass Code**: Use the Oracle Identity Cloud Service self-service console to generate bypass codes. The ability to generate a bypass code is available to the user after the user enrolls in 2-Step Verification. Users can generate bypass codes and save for use later. User-generated bypass codes never expire, but can only be used once. Users also have the option to contact an administrator to obtain a bypass code for access.
- **Duo Security**: Enable Duo Security as an MFA Factor so that users use the Duo App or other Duo factors to authenticate. If Duo Security is enabled, users that have not enrolled are prompted to do so when a Sign-On policy triggers an MFA verification.
- **FIDO Authentication**: Enable FIDO Authentication as an MFA Factor so that users use platform authentication, such as Windows Hello or Mac Touch ID, or cross platform authentication, using devices such as Yubikeys.

Learn About Using Mobile Authenticator Apps with MFA

Using a mobile authenticator application for MFA provides a second factor of authentication in the form of a time-based one-time passcode (OTP) or push notification, and offers multiple options for implementing app protection and compliance policy.

A mobile authenticator app is a soft token that is installed on a mobile device. A mobile authenticator app uses either OTP or push notifications to prove that the user has possession of the mobile device. Only the mobile authenticator app that is in possession of the user's secret key can generate a valid OTP. During MFA enrollment, when a user scans the Quick Response (QR) code or uses the enrollment URL, the mobile authenticator app is automatically configured with the Oracle Identity Cloud Service server. The mobile authenticator app retrieves a secret key, which is required to generate the OTP and to receive push notifications on the mobile authenticator app. That secret key is then shared between the client and the Oracle Identity Cloud Service server.

A user can use the mobile authenticator app to generate an OTP both online or offline. However, registering for push notifications and performing device compliance checks (jailbreak detection/PIN protection) can only be done while online.

- **Mobile App Passcode**: Use a mobile authenticator app, such as the Oracle Mobile Authenticator app, to generate an OTP. A new OTP is generated every 30-60 seconds and is valid for 90-180 seconds. After the user enters their user name and password, a prompt appears for the passcode. After generating the passcode using the mobile authenticator app, the user enters that code as the second verification method.
- **Mobile App Notification**: Send a push notification to the OMA app that contains an approval request to allow or deny a login attempt. After the user enters their user name and password, a login request is sent to their phone. The user taps **Allow** to authenticate.

The OMA app is available for Android, iOS, and Windows operating systems.

> **Note:**
>
> During MFA enrollment, a user must enter the key manually or use the enrollment URL when using the OMA app on a Surface Pro or Windows Desktop device. The QR code scanner can't be used due to a camera limitation. When a user enters that key manually, the OMA app supports only BASE32 encoding.

When you enable both the **Mobile App Passcode** and **Mobile App Notification** factors and a user is enrolled in Mobile App as a second method of verification, the Mobile App Notification factor is the default that is presented to the user. Users can change which factor they want to use by either selecting a different backup verification method when logging in or by selecting a different method as their default option. Oracle Identity Cloud Service users can use the OMA app or any supported third-party authenticator app that they want to generate OTPs. However, users must use the OMA app to receive push notifications.

Oracle Identity Cloud Service works with any third-party authenticator app (such as Google Authenticator) that adheres to the TOTP: Time-Based One-Time Password Algorithm specification. There are no special administrator configuration steps for third-party authenticator apps. When a user enrolls in MFA and selects **Mobile App** as the method, the user can either select the **Enter Key Manually** or **Scan offline QR code** options to set up third-party authenticators. We recommend the use of the OMA app as it supports notifications and security features such as app protection policy, compliance policy, and silent key refresh.

Configure Mobile OTP and Notifications

Configure policy for the time-based one-time passcode (OTP), and protection and compliance policies for the Oracle Mobile Authenticator (OMA) app.

1. In the Oracle Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, **Factors**. and the **Mobile App** tab appears.
2. In the **Passcode Policy** section, make changes to these settings, if necessary.

   The default values are the industry-recommended settings:

   - The value in the **New Passcode Generation** box indicates the number of seconds before a new passcode must be generated.
     To avoid clock skew, which is the time difference between the server and the device, the user must make sure that their device clock is synchronized. The maximum allowed time difference between the server and the device is 90 seconds.

- The value in the **Secret Key Refreshed** box indicates the number of days before you want to refresh the shared secret.
  Each time that a user enrolls a mobile device, a secret key is pushed and securely stored on the device via the scanned Quick Response (QR) code or when the user enters the key manually. This key is the input to the OTP algorithm that is used to generate the OTP. The key is refreshed silently, so no user action is required.

3. In the **Notification Policy** section, select **Enable pull notifications** to allow the OMA App to pull pending notification requests from the server.

   Pull notifications are updates that are delivered to a mobile device or computer in response to a user who is manually checking for login request notifications. You can only enable this option if you enabled the **Mobile App Notification** factor on the **Multi-Factor Authentication (MFA) Settings** page.

   Pull notifications are useful in scenarios where the GCM service (Android), APNS Service (iPhone), or WMS service (Windows) doesn't work. For example, China blocks the GCM service, so users don't receive notifications that are pushed to their device. However, if pull notifications are available, the user can manually pull notifications from a server using the OMA app. Also, offering pull notifications is useful in situations where push notifications are not 100% reliable.

4. In the lower section of the page, configure **App Protection Policy** and **Compliance Policy** for the OMA app.

   Compliance policy checks are performed each time that the OMA app launches.

Configure Security Questions

Configure security questions settings, select the security questions that a user may use as a second verification method during log in, and add custom security questions.

1. In the Oracle Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, **Factors**. and then select the **Security Questions** tab.
2. In the **Security Questions Settings** section, under **Manage security questions for MFA and account recovery**, set options for answer length and the number of questions a user is asked.
   - **Number of security questions a user must set up**: The minimum number of security questions a user must configure.
   - **Minimum answer length**: The minimum number of characters that must be contained in each security question answer.
   - **Number of security questions a user is asked**: The default number of security questions that a user must set up is set to three.

   > **Note:**
   >
   > This value can't be changed using the UI. If you need to change this number, you must use the `/SecurityQuestionSettings` endpoint.

3. In the **Manage Security Questions** section:, select the check boxes for the questions that you want to use. To disable a default security question, deselect the check box for that question.
   - Select the **Language** in which you want to view the questions.
   - To add a custom security question:

a. Click **Add Question**.

b. In the **Add a Security Question** dialog box, enter the custom security question in the default language row, indicated with an asterisk (*).

c. Click **Save**.
Your custom question is added at the bottom of the **Security Questions** list.

Optionally, enter the translated question text into the appropriate language row. When you view the custom security questions in a different language, those questions appear in your default language if you don't provide translated question text.

- To edit a custom question:
  a. Locate the custom question in the **Security Questions** list.
  b. Click the menu to the right of the question and select **Edit**.
  c. Make your changes and click **Save**.
- To remove a custom question:
  a. Locate the custom question in the **Security Questions** list.
  b. Click the menu to the right of the question and select **Remove**.
  c. Click **Remove** in the confirmation dialog box.

Configure One-Time Passcode Text Messages

Configure settings for sending a one-time passcode (OTP) as a text message (SMS) to users in Oracle Identity Cloud Service.

1. In the Oracle Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, **Factors**. and then the **Phone Number** tab.
2. Make any necessary changes to the settings for the one-time passcode that is sent to the user's device.
   - **Passcode Length**: The number of characters in the passcode.
   - **Passcode Validity Duration**: The number of minutes for which the passcode will be valid, after it is sent.
3. Select the language in which you want to view the text message, and then click **View** to preview the text message.
4. Use the text message template to create the message that's sent to the user.

   Oracle Identity Cloud Service provides a fixed list of message variables for your use. Click **Message Variables** to view the available variables and variable definitions.

   > **Note:**
   >
   > When you use the `${companyName}` variable, be sure to add your company name to the **Company Name** field on the **Branding** page in **Settings.** If you don't, your company details don't appear in email notifications, SMS notifications, or in the Oracle Mobile Authenticator (OMA) app when a user completes MFA enrollment.

5. Click **Save.**

Configure One-Time Passcode Phone Calls

Configure settings for sending a one-time passcode (OTP) as a phone call to users in Oracle Identity Cloud Service.

Prerequisite: Enable the **Phone Call** as a factor feature. This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service.

1. In the Oracle Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, **Factors**. and then the **Phone Number** tab.
2. Make any necessary changes to the settings for the one-time passcode phone call to the user's device.
   - **Passcode Length**: The number of characters in the passcode.
   - **Passcode Validity Duration**: The number of minutes for which the passcode will be valid, after it is sent.
3. Add an external notification provider using the REST API `/admin/v1/ExternalNotificationProviders` endpoint. See Create the External Notification Provider.
4. In the Oracle Identity Cloud Service console, on the **Phone Number** tab, enable the factor **Phone Call**.
5. Click **Save**.
6. Create a phone call template using the REST API. There is a default phone call template in Oracle Identity Cloud Service that must be modified. If you don't modify the template, you'll get unwanted results. For example, for OTPs, you want the passcode spoken as digits (one, two, three, four) and not as a cardinal number (one thousand two hundred and thirty-four). Use SSML tags to make changes such as these in your template. See the Nexmo developer site at Customizing Spoken Text. When creating your template, use the following variables.
   - `${tenantName}` - The name of the identity domain (or tenant).
   - `${companyName}` - The name of the company that will appear in the notification.
   - `${userName}` - The user's user name.
   - `${maskedUserName}` - The masked user name. For example, if the user name is Jhony, then maskedUserName is Jhoxxxx.
   - `${OTP}` - The OTP that's sent to a user for the user to complete 2–Step Verification.
   - `${validity}` - The amount of time (in minutes), after which the OTP will no longer be valid. As a result, the user can't use it to enroll in 2–Step Verification.
7. Base64 encode the phone call template.
8. Configure the OTP phone call template using the following REST API PATCH request.

> **Note:**
>
> The value for `localizedBody` must be base64 encoded.
>
> The value for `externalNotificationProvider` is the `id` attribute returned from POST `/admin/v1/ExternalNotificationProviders`.

```
curl --location --request PATCH
 ' https://tenant-base-url.com/admin/v1/PhoneCallTemplates/
AuthenticationPhonecallRequestNEXMO'
--header 'Content-Type: application/json' \
--header 'Authorization: Bearer <ADMIN_TOKEN' \
--header 'X-RESOURCE-IDENTITY-DOMAIN-NAME: <TENANT NAME' \
--data-raw '{
    "Operations": [
        {
            "op": "replace",
            "path": "localizedBody[locale eq \"en\"]",
            "value": [
                {
                    "value":
"PHNwZWFrPlRoZSA8cHJvc29keSByYXRlPSJzbG93Ij50ZW5hbnQxPC9wcm9zb2R5PiBsb2dpbi
BwYXNzY29kZSBmb3IgdGhlIGFjY291bnQgJHt1c2VyLnVzZXJOYW1lfSBpcyA8cHJvc29keSByY
XRlPSJ4LXNsb3ciPjxzYXktYXMgaW50ZXJwcmV0LWFzPSJkaWdpdHMiPiR7T1RQfTwvc2F5LWFz
PjwvcHJvc29keT4gVGhpcyBwYXNzY29kZSBpcyB2YWxpZCBmb3IgJHt2YWxpZGl0eX0gbWludXR
lcy48L3NwZWFrPg==",
                    "locale": "en"
                }
            ]
        },
        {
            "op": "add",
            "path": "externalNotificationProvider",
            "value": {
                "value": "24e8b100b1b2461bada230541f3ac535"
            }
        },
        {
            "op": "add",
            "path": "eventId",
            "value": "authentication.phonecall.notification"
        }
    ],
    "schemas": [
        "urn:ietf:params:scim:api:messages:2.0:PatchOp"
    ]
}
```

Configure Recovery Email Settings

Configure settings for the one-time passcode (OTP) that is sent by Oracle Identity Cloud Service in an email to the user's recovery email address.

1. In the Oracle Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, **Factors**. and then select the **Email** tab.
2. In the **Configure the email settings for account recovery and activation**section, configure valid time frames for the account recovery and email activation notifications and whether you want to allow the user to add an alternate email address for account recovery.
3. In the **Configure the mail settings for MFA** section, make changes to the settings for the one-time passcode that is sent in an email to the user.
4. Click **Save.**

   To access the email template that's sent to the user's primary email account:

   a. Expand the **Navigation Drawer** and click **Settings**, **Notifications**.

   b. Select the **Email Templates** tab.
   The template name is **2-Step Email One-Time Passcode Verification**.

Configure Email Settings

Configure settings for the one-time passcode (OTP) that is sent by Oracle Identity Cloud Service in an email to the user's primary email address.

1. In the Oracle Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, **Factors**. and then select the **Email** tab.
2. In the **Configure the email settings for account recovery and activation**section: and whether you want to allow the user to add an alternate email address for account recovery.

   a. Select the check box for **Allow the user to add an alternate email address for account recovery**.

   b. Configure valid time frames for the account recovery and email activation notifications.
3. In the **Configure the mail settings for MFA** section, make changes to the settings for the one-time passcode that is sent in an email to the user.
4. Click **Save.**

   To access the email template that's sent to the user's primary email account:

   a. Expand the **Navigation Drawer** and click **Settings**, **Notifications**.

   b. Select the **Email Templates** tab.
   The template name is **2-Step Email One-Time Passcode Verification**.

Configure Duo Security Settings

If you have implemented or want to implement Duo Security as a third-party multi-factor authentication (MFA) solution, and Oracle Identity Cloud Service manages your primary authentication and identity management, you can connect to and integrate with Duo to secure Oracle IaaS, PaaS, or SaaS applications or to secure applications already secured by Oracle Identity Cloud Service.

Prerequisites

- **Enable Duo.** This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service.
- Download and install the Duo Mobile app from the Google Play Store or the Apple Store.

1. Subscribe to Duo and a create a Duo administrator account.

Go to https://duo.com/ to set up your subscription and to set up your Duo administrative account. Refer to the Duo documentation for the latest instructions.

2. Create and activate the Duo-protected Web SDK app.

   To create and activate the Duo-protected Web SDK app, refer to the Duo documentation for the latest instructions.

3. Note the credentials and connecting host information.

   These values were generated when you created and activated the Duo-protected Web SDK app. You need the values for **Integration key**, **Secret key**, and **API hostname**. Refer to the Duo documentation for the latest instructions.

4. In the Oracle Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, **Factors**. and then **Duo Security**.

5. Enter the credentials and connecting host information (**Integration key**, **Secret key**, and **API hostname**) that was generated from your Duo Administrative account, and then choose a **User Identifier**.

   The User Identifier that you choose must map to the user identifier set in the Duo user account. For example, **User Name** in the Oracle Identity Cloud Service user account must map to **Username** in the Duo security user account.

6. In the Oracle Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, **MFA**, turn on **Duo Security**, and then click **Save**.

   You may have to log out and log in again to see **Duo Security**.

Post Requirement: Understand the user Duo enrollment workflow.

1. User accesses the login screen.
2. Duo Security prompts the user to enroll.
3. Duo sends the User a notification asking them to enroll in Duo. Options are PUSH, TOTP, SMS, or SECURITY_QUESTIONS.
4. User accepts the enrollment verification.
5. User is created in Duo.
6. Optional. User sets up an additional factor. Options are PUSH, TOTP, SMS, or SECURITY_QUESTIONS. Or click Done.
7. User is logged in to Oracle Identity Cloud Service.
8. User can now use Duo Security MFA factors to sign in to Oracle Identity Cloud Service.

Configure FIDO Security

Configure FIDO authentication so that users can use their FIDO authentication device, for example an external authentication device such as a YubiKey, or an internal device such as Windows Hello or Mac Touch ID, to authenticate to Oracle Identity Cloud Service.

Enable FIDO. This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service.

1. In the Oracle Identity Cloud Service console, expand the Navigation Drawer, click **Security** then **MFA**.

   The Multi-Factor Authentication (MFA) Settings page opens.

2. Select **FIDO Authenticator** and click **Configure**.

   The FIDO Authenticator tab opens.

3. Configure the FIDO Authenticator settings:
   - **Timeout**: The length of time the user has to take action. If the user doesn't take action within this period, there is an authentication failure. The default is 60,000 milliseconds (6 seconds).
   - **Attestation**: Not supported.
   - **Authenticator Selection Attachment**: Controls what type of authenticator user can use during Registration.
     - Platform. Windows Hello and Mac Touch ID.
     - Cross Platform. Choose to use a cross-platform authenticator such as YubiKey.
     - Both (default).
   - **Authenticator Selection Resident Key**: Whether Resident key support should be enabled.
     - Required.
     - Preferred.
     - Discouraged.
     - None (default). The private key is encrypted and stored on the server.
   - **Authenticator Selection User Verification**: Relying Party's requirements regarding user verification during Registration:
     - Required.
     - Preferred (default).
     - Discouraged.
   - **Public Key Types**: The cryptographic algorithm used to generate a public key pair during Registration. Oracle Identity Cloud Service certifies the ES256 (default) and RS256 algorithms. **Note:** The RS256 algorithm is mandatory for Windows Hello FIDO authentication.
   - **Exclude Credentials**: Used by Relying Parties to limit the creation of multiple credentials for the same account on a single authenticator. Default value is `false`.

FIDO Authentication is now an additional sign-in factor

Multi-Factor Authentication Authorization Flow

The authorization flows that support Oracle Identity Cloud Service Mutli-Factor Authentication are the Authorization Code Grant Type and SAML2 Assertion.

Authorization Flows that Don’t Support Oracle Identity Cloud Service Multi-Factor Authentication:

- Resource Owner Password Credentials Grant Type
- Client Credentials Grant Type
- Assertion Grant Type
- Implicit Grant Type

32

Manage Oracle Identity Cloud Service OAuth Settings

Learn how to configure OAuth settings for your environments.

Topics

Configure OAuth Settings

You can configure OAuth settings to allow all resources.

Prerequisite: OAuth settings won't display unless the feature is enabled. To enable this feature, file a Service Request with My Oracle Support. If you don't file a Service Request, then you won't be able to configure OAuth settings.

To configure OAuth settings:

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Security**.
2. Click **OAuth**.
3. In the **OAuth Settings** page, select **Always Allow All Resources**.

   This allows the client to access any resource within the tenant regardless of the Trust Scope settings at the application level.
4. (Optional) In the **Issuer** field, enter a custom issuer value. This issuer value will be used in the newly issued tokens.

   The default IDCS issuer, `https://identity.oraclecloud.com/` is used if you do not specify a custom issuer.

   > **Caution:**
   >
   > Only one previous issuer value is stored. If you make frequent changes in the issuer value, the old token validation may fail.
   >
   > After changing the Issuer value at Oracle Identity Cloud Service, the Issuer may be different on the client side based on the Tenant configuration. Make sure you validate the issuer value logic on the client side to use the new Issuer value.
5. Click **Save**.

Update the `idcs.iss.url` value in the EBS Asserter properties file. See Update the E-Business Suite Asserter Configuration File.

33

Configure Delegated Authentication in Oracle Identity Cloud Service

This section describes how to configure delegated authentication in Oracle Identity Cloud Service.

Topics:

Typical Workflow for Managing Delegated Authentication in Oracle Identity Cloud Service

With the delegated authentication feature in Oracle Identity Cloud Service, you can perform tasks such as viewing, deactivating, testing, and activating delegated authentication for a Microsoft Active Directory (AD) Bridge associated with an AD domain.

Task	Description	Additional Information
Understand delegated authentication.	You can learn about delegated authentication, including how administrators can use it so that users can use their AD passwords to sign in to Oracle Identity Cloud Service to access Oracle Identity Cloud Service-protected resources and applications.	Understand Delegated Authentication
View details about delegated authentication.	You can view details about an AD Bridge, such as whether it's activated or deactivated for delegated authentication, by using the **Delegated Authentication** page.	View Details About Delegated Authentication
Deactivate delegated authentication.	You can deactivate delegated authentication for an AD Bridge associated with an AD domain by using the **Delegated Authentication** page.	Deactivate Delegated Authentication

Task	Description	Additional Information
Test delegated authentication.	You can verify that a user's AD credentials from a domain associated with an AD Bridge can be used to sign in to Oracle Identity Cloud Service by using the **Delegated Authentication** page.	Test Delegated Authentication
Activate delegated authentication.	You can activate delegated authentication for the AD Bridge by using the **Delegated Authentication** page.	Activate Delegated Authentication
Handle Network Failure in Delegated Authentication.	You can login into Oracle Identity Cloud Service even if Active Directory is not reachable.	Handle Network Failure in Delegated Authentication

You can manage delegated authentication for an AD Bridge by:

- The Identity Cloud Service console
- SCIM-based APIs

The following sections describe how to manage delegated authentication in Oracle Identity Cloud Service by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

Understand Delegated Authentication

With delegated authentication, identity domain administrators and security administrators don’t have to synchronize user passwords between an on-premises Microsoft Active Directory (AD) enterprise directory structure and Oracle Identity Cloud Service. Users can use their AD passwords to sign in to Oracle Identity Cloud Service to access resources and applications protected by Oracle Identity Cloud Service.

Prerequisite

Enabling Delegated Authentication. This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service.

Suppose you have an AD domain that contains user accounts that you want to import into Oracle Identity Cloud Service. To transfer these accounts, install and configure an AD Bridge for this domain. The AD Bridge provides a link between the domain and Oracle Identity Cloud Service. Oracle Identity Cloud Service can synchronize with this domain so that any new, updated, or deleted user records are transferred into Oracle Identity Cloud Service. Because of this, the state of each record is synchronized between AD and Oracle Identity Cloud Service. See Manage Microsoft Active Directory (AD) Bridges for Oracle Identity Cloud Service for more information about installing and configuring AD Bridges in Oracle Identity Cloud Service.

After using an AD Bridge to transfer user accounts from the AD domain into Oracle Identity Cloud Service, you want to configure Oracle Identity Cloud Service so that users from this domain must use their AD passwords to sign in to Oracle Identity Cloud Service. To do this, activate delegated authentication for the AD Bridge. However, first, you may want to verify that the AD credentials from a user in the domain can be used to sign in to Oracle Identity Cloud Service. This way, if there are any issues, then you can resolve them before activating delegated authentication.

After you activate delegated authentication in Oracle Identity Cloud Service, if you change or reset a password in Oracle Identity Cloud Service, then the password is stored directly in AD. The AD password policies are applicable for the new password. Password policies configured in Oracle Identity Cloud Service aren't applicable for this password. Oracle Identity Cloud Service doesn't maintain the password.

Statuses

Find here the three statuses of the Microsoft Active Directory (AD) Bridge.

There are three statuses for an AD Bridge that Oracle Identity Cloud Service uses to communicate with an AD domain to delegate responsibilities for authenticating users of that domain into Oracle Identity Cloud Service:

- Connected: The AD Bridge is installed and configured, and can communicate with the domain.
- No Clients Found: You installed or configured an AD Bridge without installing the client for the bridge. Click the **Click here to download the client.** link to download the client for the bridge.
- Incompatible Client Found: You used an outdated version of the client to install or configure an AD Bridge. Click the **Click here to download the client.** link to download the updated client for the bridge.

View Details About Delegated Authentication

By default, in the **Delegated Authentication** page, you can see the name and status of each Microsoft Active Directory (AD) Bridge that Oracle Identity Cloud Service uses to communicate with an AD domain. You can also see other information about the bridge, such as whether it's activated or deactivated for delegated authentication.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Delegated Authentication**.
2. Expand the node to the left of the AD Bridge about which you want to view additional information. You'll see an **Activate Delegated Authentication** switch, which indicates whether the bridge is activated or deactivated for delegated authentication.

> **Tip:**
>
> Click the AD Bridge to see detailed configuration information about it, as well as how many users and groups were transferred by the bridge from AD into Oracle Identity Cloud Service.

Note:

The **Activate Delegated Authentication** switch may be "greyed out" (that is, you can't turn the switch on or off) for one of the following reasons:

1. The status of the AD Bridge is **No Clients Found**. You can't activate delegated authentication for the bridge because the bridge won't work until you install a client for the bridge. Click the **Click here to download the client.** link to download the client for the bridge.
2. The status of the AD Bridge is **Incompatible Client Found**. You can't activate delegated authentication for the bridge because the bridge won't work until you install the correct version of the client for the bridge. Click the **Click here to download the client.** link to download the updated client for the bridge.
3. The AD Bridge isn't configured for delegated authentication. To configure it:
   a. Click the bridge.
   b. Click **Configuration**.
   c. In the **Configure the Microsoft Active Directory Domain** page, scroll down until you see the **Authentication Settings** area.
   d. Select **Enable local authentication**.
   e. Click **Save**.
   f. In the **Save Configuration Changes?** dialog box, click **OK**.

Deactivate Delegated Authentication

You can deactivate delegated authentication for a Microsoft Active Directory (AD) Bridge associated with an AD domain. Users transferred into Oracle Identity Cloud Service through this bridge must use their Oracle Identity Cloud Service passwords to authenticate into Oracle Identity Cloud Service. Also, by deactivating delegated authentication, you can verify that the AD credentials from a user in that domain can be used to sign in to Oracle Identity Cloud Service before activating delegated authentication for the bridge.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Delegated Authentication**.
2. Expand the node to the left of the AD Bridge for which you want to deactivate delegated authentication.
3. Turn **Off** the **Activate Delegated Authentication** switch.
4. In the **Deactivate Delegated Authentication** window:
   a. Select the **Send a Password Reset Notification (recommended)** option if you want users in the AD domain associated with the AD bridge to receive notifications to reset the passwords for their accounts. This is recommended for security purposes.
   b. Select the **Create a Password** option if you want to manually reset passwords for the users in the domain associated with the bridge. No notification is sent to users. Selecting **Create a Password** means that the Users in the domain, who were previously able to sign in using Delegated Authentication, will not be able to sign in to the system. To allow them to sign into the system, reset their passwords using the reset passwords option on the Users tab. See Reset Passwords for User Accounts.
5. Click **OK**.

Test Delegated Authentication

You can verify that a user's Microsoft Active Directory (AD) credentials from a domain associated with an AD Bridge can be used to sign in to Oracle Identity Cloud Service. This way, if there are any issues, then you can resolve them before activating delegated authentication for the bridge.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Delegated Authentication**.
2. Expand the node to the left of the AD Bridge for which you want to test delegated authentication.
3. Click **Test Delegated Authentication**.
4. In the **Test Delegated Authentication** window, enter the AD user name and password that you want to use to sign in to Oracle Identity Cloud Service.
5. Click **Test**.

Activate Delegated Authentication

After verifying that the Microsoft Active Directory (AD) credentials of a user in the domain associated with an AD Bridge can be used to sign in to Oracle Identity Cloud Service, activate delegated authentication for the bridge. Users transferred into Oracle Identity Cloud Service through this bridge will use their AD passwords to authenticate into Oracle Identity Cloud Service.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Delegated Authentication**.
2. Expand the node to the left of the AD Bridge for which you want to activate delegated authentication.
3. Turn **On** the **Activate Delegated Authentication** switch.
4. In the **Confirmation** window, click **Yes**.

Handle Network Failure in Delegated Authentication

Most organizations still rely on Microsoft Active Directory (AD) for managing their user accounts, and users rely on Active Directory for authentication and access to various systems. If for some reasons, users are not be able to authenticate themselves with Active Directory credentials. This will have a huge impact on the daily operations and business of the organizations.
To avoid these kind of situations, Oracle Identity Cloud Service provides you with a network failure handling functionality. This functionality helps users to login with Active Directory credentials even when Oracle Identity Cloud Service is not able to reach the Oracle Identity Cloud Service Active Directory (AD) Bridge.

You configure delegated authentication for a bridge in Oracle Identity Cloud Service so that a user can use their Active Directory password to authenticate into Oracle Identity Cloud Service.

If AD Bridge is not reachable, then users are unable to validate their credentials with Active Directory and therefore cannot login into Oracle Identity Cloud Service. Your Active Directory is not reachable for a number of reasons. This could be due to network connectivity between AD Bridge and Oracle Identity Cloud Service is down.

To avoid this situation, Oracle Identity Cloud Service provides the local password caching functionality to perform local authentication in case AD Bridge is not reachable. This functionality helps delegated users to login into Oracle Identity Cloud Service even if AD Bridge is not reachable. For security reasons, this password is stored in hashed form in Oracle Identity Cloud Service.

It is important to make sure that the lifetime of this cache password in Oracle Identity Cloud Service is limited. You can configure the maximum duration (5 days) you set to cache the password on Oracle Identity Cloud Service. For example, if your network connectivity is down and you have set the cache password duration to 2 days, then it will enable users to login to Oracle Identity Cloud Service for only 2 days. However, if Active Directory is still not reachable for longer than the specified duration, then you will not be able to login to Oracle Identity Cloud Service.

In order to guard against the possibility that someone can use brute force attacks to access your account, you can limit the number of unsuccessful password attempts during password caching in Oracle Identity Cloud Service. After several failed attempts, Oracle Identity Cloud Service locks your user account. There is a limit of 5 which is configurable.

You cannot perform the following operations while the network connectivity is down:

- A user cannot change their own password
- A user cannot reset their own password by validating the token
- A user cannot change their own email address
- An administrator cannot change a user's password to a known value
- An administrator cannot reset a user's password whose password is authenticated by Active Directory

However, if you recently changed a password in Active Directory, then you can login to Oracle Identity Cloud Service with that password while connectivity is down, provided you have already login to Oracle Identity Cloud Service while Active Directory was available.

> **Note:**
>
> Sometimes, you might encounter a system error even if you provide a correct password. This is either because the password cache is empty or because the password has expired.

Activate Local Password Caching

You must activate the local password caching functionality to enable delegated authentication users to login intoOracle Identity Cloud Service in case Microsoft Active Directory is not reachable.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Delegated Authentication**.
2. Expand the node to the left of the AD Bridge for which you want to activate password cache.
3. Turn **On** the **Do you want to activate password cache** switch.
4. Set the duration you want to cache this password in **Cache password duration (days)**.

5. Select how many unsuccessful password attempts that you want during password caching in **No of unsuccessful password attempts during Password Caching**.
6. Click **Save**.

34

Manage Passwordless Authentication

Learn how to configure Passwordless Authentication for users.

- Typical Workflow for Passwordless Authentication
- Understand Passwordless Authentication
- Configure Passwordless Authentication for User Accounts

Typical Workflow for Passwordless Authentication

Learn how to configure passwordless authentication to allow users to authenticate their sign on with an identity provider.

Task	Description	Additional Information
Understand Passwordless Authentication.	You can learn about passwordless authentication and how it can be used with email and mobile authentication.	Understand Passwordless Authentication.
Configure Passwordless Authentication for User Accounts.	You can learn how to configure passwordless authentication and the identity providers available for you to use.	Configure Passwordless Authentication for User Accounts.

Understand Passwordless Authentication

Passwordless authentication allows you to bypass the standard web-form-based authentication presented to users when using email or a mobile device.

Prerequisite

Enable Password Authentication. This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service.

The first time login is through the standard login form. During the first time while accessing the protected resource, users are redirected to the standard login form. After successful login, you can enable passwordless notification-based authentication.

The next time (and subsequently) the user accesses the protected page and is required to log in, a message is displayed (instead of the standard login page) mentioning that a push notification is sent to the user's mobile device.

> **Note:**
>
> In order to use passwordless authentication, users must configure their profiles to use the authentication policy that you have configured.

Configure Passwordless Authentication for User Accounts

You can configure passwordless authentication to allow users email and mobile authentication.

To configure passwordless authentication:

1. In the Oracle Identity Cloud Service console, expand the **Navigation Drawer**, and select **Settings**, and then click **Session Settings**.
2. In the **Session Settings** page, select **Enable User Name First**.

   See Change Session Settings.
3. Click **Save**.
4. From the **Navigation Drawer**, select **Security**, **MFA**.
5. In the **Select the factors that you want to enable** section, select the factor or factors you want to use. For example, one or both **Email** and **Mobile App Passcode**.

   You can use one or both of these factors with passwordless authentication. See Learn About Using Mobile Authenticator Apps with MFA.
6. Click **Save**, and then **OK** in the **Confirmation** dialog box.
7. From the **Navigation Drawer**, select **Security**, **IDP Policies**.
8. In the **Identity Provider Policies** page, click the identity provider policy that you want to modify.

   The policy opens and displays three tabs: Details, Identity Provider Rules, and Apps. See View Details About an Identity Provider Policy for more information about these tabs.
9. Click the **Identity Provider Rules** tab.
10. Click the **Action** menu for the rule you want to modify and select **Edit**.
11. In the **Edit...** dialog box, click in the **Assign Identity Providers** box and select the identity provider, such as Mobile App Passcode or Mobile App Notification, that you want to assign to this rule.

    > **Note:**
    >
    > If MFA factors, such as Mobile App Passcode or Mobile App Notification, were enabled before passwordless authentication has been enabled, then you have to disable the MFA factors and save, then enable the MFA factors again and save, otherwise the factors won't show up in **Assign Identity Providers**.

    Repeat this step to assign additional Identity Providers.
12. Click **Save**.

Passwordless authentication is now configured.
Users must configure their profiles to use the authentication policy that you have configured. On the user log in page, they need to click Show alternative login methods.

35

Transfer Oracle Identity Cloud Service Configuration Data

Learn how to transfer Oracle Identity Cloud Service configurations.

Topics:

Overview of Transferring Oracle Identity Cloud Service Configurations

If you have more than one Oracle Identity Cloud Service environment, you can transfer configurations from one environment to another.

Transferring configurations permits you to transfer configuration information from one Oracle Identity Cloud Service to another Oracle Identity Cloud Service. It helps reduce the downtime involved in setting up a service.

You can transfer data using the following methods:

- The Identity Cloud Service console
- SCIM-based APIs

In the following sections, you learn how to transfer data by using the Identity Cloud Service console.

For more information about how to use SCIM APIs, see REST API for Oracle Identity Cloud Service.

This table summarizes the transfer operations permitted in the UI.

Operation	Description	Administrator Role Required	Additional Information
Export users.	Export user accounts to a CSV file.	Identity Domain Administrator User Administrator	Exporting User Accounts
Export groups.	Export groups to a CSV file.	Identity Domain Administrator User Administrator	Exporting Groups
Export application role memberships.	Export users and groups for Oracle application roles	Identity Domain Administrator Application Administrator	Export Users and Groups for Oracle Application Roles

Typical Workflow for Transferring Oracle Identity Cloud Service Configurations

Use this typical workflow to get started transferring configurations.

After each import step, analyze the data recorded during the bulk load operation. After you import the file, a dialog box appears with the Job ID link for your import job, click the link. Review the details that appear on the Jobs page. This page shows how many accounts you imported, how many accounts imported successfully, and how many accounts can't be imported because of a system error. Common issues that prevent the system from importing the account include:

- Invalid email address format
- Invalid field formats
- Missing required fields
- Invalid CSV file

If there are many invalid accounts, correct the errors in the import file and then import the file again. See Viewing Jobs and Job Details.

Task	Description	Additional Information
Step 1: Export users.	Use this task to create users only.	Exporting User Accounts
Step 2: Export groups.	Use this task to create groups and user memberships.	Exporting Groups
Step 3: Export application role memberships.	Use this task to create to create application role memberships for users and groups.	Export Users and Groups for Oracle Application Roles
Step 4: (Optional) Gather diagnostic data.	If you encounter errors, you can set a diagnostics level to capture operational logs. You can then view those logs to help you to determine the cause of the problem. Use the REST API for Oracle Identity Cloud Service to capture diagnostic data.	See Diagnostic Records REST Endpoints
Step 5: (Optional) Resolving errors after an export operation.	If you encounter errors during an export operation, resolve the errors and then try the export operation again. If Oracle Identity Cloud Service can't export a user account, then it evaluates the next account in the CSV file. View the details of the export job. If the job contains errors, you can export those errors to see the cause. If you cannot resolve the errors, use the diagnostic data report to capture operational logs to see if you can determine the cause of the problem.	View Jobs and Job Details Export Job Errors

Download Exported Files

After you export configuration files from Oracle Identity Cloud Service, you must download the files, for example, if you want to import them to another Oracle Identity Cloud Service.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Jobs**.
2. Locate the specific job for which you want to export the file.
3. Click **View Details**.
4. Click **Download**.

   A comma-separated value (CSV) file downloads to your local machine. The CSV file contains a record for each error that includes the error type and the error description.

36
Use Device Fingerprints

You can use information collected about the user system to maintain a fingerprint which can be used to identify the end user system uniquely.

Topics:

About Device Fingerprints

Information about the device used by an end user is collected and a hash value is generated. This allows the end user system to be uniquely identified.

Enable Device Fingerprints. This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service.

A user's device can be identified uniquely by collecting details of the device and encrypting them to create a fingerprint hash value.

The login calls either from the default Oracle Identity Cloud Service sign-in page or from custom UI sign-in page are modified to send device detail attributes. If any of the attributes are missing from the device details sent to SSO, a value of null will be used when the fingerprint hash is generated.

Device Fingerprints and Custom Sign In Pages

Learn how to use device fingerprints with custom sign in pages.

> **Note:**
>
> This is only for the case of custom login pages. If you are using the default login page, device fingerprinting works after you enable it.

For a full description about collecting device details and adding the details in a custom log in page, see Enable the 'Access for an unknown device' Event for a Custom Sign-In Page.

The device fingerprint is stored in a cookie as part of the SSO request. If the device fingerprint value is already present in the persistent cookie, the value will be used across the flow for auditing. If the device fingerprint value is not present in the persistent cookie, a new device fingerprint hash value is created based on the device details sent in login request to SSO. The use cases below show the state of the cookie the first time a user logs in using a custom sign in page, and then on subsequent logins.

An alternative way of using device fingerprints with custom sign in pages is described in Enable the 'Access for an unknown device' Event for a Custom Sign-In Page.

Use Case: First time login using custom sign in page

The first time login use case is the scenario where there is no device fingerprint value of the end user in the cookie at end user browser or with custom UI. In this case, the request payload to */sso/v1/sdk/authenticate* authN api carries device attributes. SSO processes these device attributes in order to calculate device fingerprint hash value and it sends this value in the response payload.

Request payload:

```
{
    "op":"credSubmit",
    "credentials":
    {
        "username" : "testUser",
        "password": "Welcome@1",
        "device":
        "
            {

\"screenWidth\":null,\"screenHeight\":900,\"screenPixelDepth\":\"24\",
            \"windowPixelRatio\":2,\"language\":\"en-
GB\",\"userAgent\":\"Mozilla 5.0 (Macintosh;
            Intel Mac OS X 10_13_6) AppleWebKit 537.36 (KHTML, like Gecko)
Chrome 85.0.4183.102
            Safari 537.36\",\"plugins\":\"3\",\"timeZone\":-330
            }
        "
    },

    "requestState": "requestState"
}
```

In case of AuthN SDK, the custom UI application might have access to more end user information. In this case, the application can send these additional device details in the request payload to authN api in addition to the other device attributes.

However, it is suggested that the additional values are provided in string format in the request payload. For example, if the custom UI has access to the location of the end user, the additional detail `"location":"Hyd"` can be sent as part of the device details.

Response payload:

```
{
    "authnToken": "....",
    "status": "success",
    "ecId": "6cjO20H1000000000",
    "deviceFingerPrint":
"db68d19334304e2624ca217de1196d30e9150335e0bd7b6072c44a0841073a79"
}
```

Use Case: Logins using a custom sign in page after the first time

Logins after the first time are represented by this use case, where device fingerprint is already present in the cookie at end user browser or with custom UI. The custom UI renders this value from the cookie and sends the value in the POST request payload to `/sso/v1/sdk/authenticate` authN api. SSO service in this case will use this device fingerprint hash value sent by the custom UI and store the value in sdk context(request state) for all the subsequent processing.

Request Payload:

```
{
    "op": "credSubmit",
    "credentials":
    {
        "username": "testUser",
        "password": "Welcome@1",
        "device": "
        {....}
        "
    },
    "requestState": "requestState",
    "deviceFingerPrint": "123456789"
}
```

Response Payload:

```
{
    "status": "success",
    "ecId": "6cjO20L1000000000",
    "nextAuthFactors": [
        "SECURITY_QUESTIONS"
    ],
    "nextOp": [
        "credSubmit",
        "getBackupFactors"
    ],
    "scenario": "AUTHENTICATION",
    "requestState" : "...."
    "trustedDeviceSettings": {
        "trustDurationInDays": 15
    },
    "deviceFingerPrint": "123456789"
}
```

Enabling the Device Fingerprint

To use device fingerprint, you have to enable SSO settings.

Enable SSO settings for device fingerprinting by setting `deviceFingerPrint` to true using the REST API Replace SSO Settings `PUT` call. See Replace SSO Settings (PUT).

ssoSettings payload

```
{
    "logoutLandingPageURI": "/ui/v1/myconsole",
    "userMappingAttribute": "userName",
    "fedSsoOnly": false,
    "cookieSessionTimeout": 15,
    "ssoChooserEnabled": false,
    "sessionExpiryMinutes": 480,
    "sdkEnabled": true,
    "userNameFirst": null,
    "deviceFingerPrint": true,
    "schemas": [
        "urn:ietf:params:scim:schemas:oracle:idcs:SsoSettings"
    ]
}
```

Device Fingerprint in UserDevices

The UserDevices REST API allows you to list all the devices linked to a specific user. See REST API for Oracle Identity Cloud Service.

To list all the devices with a unique fingerprint used by a specific user, use

```
GET /admin/v1/UserDevices?filter=owner eq "userid"
```

Sample response payload

```
"Resources":
[
    {
        "deviceType": "Mac",
        "verified": true,

"deviceFingerPrint":"c2NyZWVuQ29sb3JEZXB0aD0yNCwgbWltZVR5cGVzPW51bGwsIHBsdWdpb
nM9bnVsbCwgYnJvd3Nlcj1DaHJvbWUsIGxhbmd1YWdlPWVuLUdCLW94ZW5kaWN0LCBwbGF0Zm9ybT1
NYWMgT1MgWA==",
        "dfp":
"7039602ba45600acf1234046c5d6133b327187ff9dc9af65eb6d97b6324a44d8"
        "id": "b4c9a31d67aa4c45b9f1a441ee71b0d5"
    }
]
```

The `dfp` value in the response payload is calculated and stored only when `deviceFingerPrint Sso Settings` is enabled. When it is not enabled, the `dfp` is marked as `Unknown` for new devices.

Use cases

Here, you can see when a New Device Login Detected with Your Account email is sent.

Use Case	Behaviour
Alex logs in from Chrome browser for the first time and they **do not have** any previous entry in `UserDevices` from Chrome and **Adaptive Access is enabled** and **deviceFingerPrint Sso setting disabled**.	Record the entry for Chrome in `UserDevices`. The `dfp` value is `Unknown`, and send a notification.
Alex logs in from Chrome browser for the first time and they **do not have** any previous entry in `UserDevices` from Chrome and **Adaptive Access is enabled** and **deviceFingerPrint Sso setting enabled**.	Record the entry for Chrome in `UserDevices` with a proper `dfp` value and send a notification.
Alex logs in from Chrome browser for the first time and they **do not have** any previous entry in `UserDevices` from Chrome and **Adaptive Access is disabled** and **deviceFingerPrint Sso setting disabled**.	Do not record any value and don't send a notification.
Alex logs in from Chrome browser for the first time and they **do not have** any previous entry in `UserDevices` from Chrome and **Adaptive Access is disabled** and **deviceFingerPrint Sso setting enabled**.	Record the entry for Chrome in `UserDevices` with a proper `dfp` value, but don't send a notification.
Alex logs in from Chrome browser for the first time and they **have a previous entry** in `UserDevices` from Chrome and **Adaptive Access is enabled** and **deviceFingerPrint Sso setting disabled**.	Record the `dfp` value as `Unknown` for the existing device if a `dfp` value does not already exist, but don't send a notification.
Alex logs in from Chrome browser for the first time and they **have a previous entry** in `UserDevices` from Chrome and **Adaptive Access is enabled** and **deviceFingerPrint Sso setting enabled**.	Record a proper `dfp` value for the existing device if it does not already exist, but don't send a notification.
Alex logs in from Chrome browser for the first time and they **have a previous entry** in `UserDevices` from Chrome and **Adaptive Access is disabled** and **deviceFingerPrint Sso setting disabled**.	Do not record any value and don't send a notification.
Alex logs in from Chrome browser for the first time and they **have a previous entry** in `UserDevices` from Chrome and **Adaptive Access is disabled** and **deviceFingerPrint Sso setting enabled**.	Record a proper `dfp` value but don't send a notification.

Device Fingerprint in Tokens

Device fingerprint is added as part of ID tokens and Access tokens.

Whenever device fingerprint settings are enabled, `dfp` will be added as a claim in the following use cases:

- **JWT Assertion flow**: When custom application exchanges the authnToken from sso/v1/sdk/authenticate endpoint for the access token, the device fingerprint from authnToken is added to the access token. See Understand the REST API Calls.
- **Authz code flow**: In openId connect flows, when a custom application exchanges authz code for AT/ IT, the device fingerprint from the existing Identity Cloud Service session is added to the tokens based on scopes.
- **Get token from IDCS console**: After successful login with Identity Cloud Service, the token retrieved from console to **Invokes other APIs** will also bear the device fingerprint as a claim. See Generate Personal Access Tokens.

Example 36-1 authn api response post successful authentication


```
{
    "authnToken":

"eyJ4NXQjUzI1NiI6Iks0R0hvZVdoUmFhOTd6Um0xeDIzM0pwdlB3bm1GQVJGVlE1cE5QRDhsTEUiL
CJ4NXQiOi

JUYkdPcWVUWnJpeXZNZGplTC01MjAtaGVfRUUiLCJraWQiOiJTSUdOSU5HX0tFWSIsImFsZyI6IlJT
MjU2In0.ey

J1c2VyX3R6IjoiQW1lcmljYVwvQ2hpY2FnbyIsInN1YiI6ImFkbWluQG9yYWNsZS5jb20iLCJ1c2Vy
X2xvY2FsZSI

6ImVuIiwiaWRwX25hbWUiOiJVc2VyTmFtZVBhc3N3b3JkIiwiaWRwX2d1aWQiOiJVc2VyTmFtZVBhc
3N3b3JkIwiY

W1yIjpbIlVTRVJOQU1FX1BBU1NXT1JEIl0sImlzcyI6Imh0dHBzOlwvXC9pZGVudGl0eS5vcmFjbGV
jbG91ZC5jb21
        …"
    "status": "success",
    "ecId": "9uQ251a3000000000",
    "deviceFingerPrint":
"d59c35b2baba97730fc359c1a2b592f6e773cc1eec073d0f11385edf6abd7fd4"
}
```


Example 36-2 authn token decrypted


```
{
    "user_tz": "America/Chicago",
    "sub": "admin@oracle.com",
    "user_locale": "en",
    "idp_name": "UserNamePassword",
    "idp_guid": "UserNamePassword",
    "amr": [
        "USERNAME_PASSWORD"
    ],
    "iss": "https://identity.oraclecloud.com/",
    "island_name": "Global",
    "user_tenantname": "slc1",
    "client_id": "91932046c2df4090851c7b93efe1cf1f",
    "sid": "11eb7d7b9ee020bba7e3e7cd3f20296a",
    "acs": "N4K...",
    "azp": "custom ui",
    "authn_strength": 2,
    "auth_time": 1614925546,
    "client_tenantname": "slc1",
    "session_exp": 1614954346,
    "region_name": "slc12kps",
    "user_lang": "en",
    "exp": 1614954346,
    "iat": 1614925546,
    "client_name": "custom ui",
    "client_guid": "c513ad464a664ca091452937bf4030cf",
    "tenant": "slc1",
```

```
    "idp_type": "LOCAL",
    "jti": "11eb7d7b9ee2439ca7e3a131d1100b2a",
    "user_displayname": "admin opc",
    "sub_mappingattr": "userName",
    "primTenant": false,
    "tok_type": "IT",
    "dfp": "d59c35b2baba97730fc359c1a2b592f6e773cc1eec073d0f11385edf6abd7fd4",
    "aud": [
        "https://identity.oraclecloud.com/",
        "91932046c2df4090851c7b93efe1cf1f",
        "custom ui"
    ],
    "user_id": "2003002e92e4492e960fde36c34e45c9",
    "tenant_iss": "https://slc1.identity.internal.oracle.com:8943"
}
```


Device Fingerprint in Audit Logs

The audit event attribute `ssoDeviceFingerPrint` maintains the device fingerprint hash value in all the SSO events, and it is added to audit records.

Example of an SSO Audit Event


```
{
    "ssoSessionExpiryTime": "2020-12-22T19:01:39Z",
    "idcsCreatedBy": {
        "type": "User",
        "display": "admin opc",
        "value": "16b5ae8e9dfd4ec4a68f449252159e5b",
        "$ref": "https://tenant1.identity.internal.oracle.com:8943/admin/v1/
Users/16b5ae8e9dfd4ec4a68f449252159e5b"
    },
    "actorName": "admin@oracle.com",
    "id": "cc50265b-9448-4f19-8981-0d14703ace2a"
    .......
    "clientIp": "10.191.242.183",
    "ssoComments": "Session create success",
    "ssoBrowser": "Chrome",
    "ssoMatchedSignOnRule": "DefaultSignOnRule",
    "ssoDeviceFingerPrint":
"db68d19334304e2624ca217de1196d30e9150335e0bd7b6072c44a0841073a79",
    "eventId": "sso.session.create.success",
    .......
}
```


`ssoDeviceFingerPrint` is not directly searchable in audit events.

Instead you can search for the application ID, a searchable attribute to which `ssoDeviceFingerPrint` is appended. See Audit Log Report

The `sso.application.id` will be "`application.Id -ssoDeviceFingerPrint`"

For example:

```
ssoApplicationId": "ec0cc4b24bfc4b87a3945c6f7e251882 -
ca6c3e77d75dc3e3186084934957261503 71a4910c5c8a2994007c1f7e07875b


ssoApplicationId": "adminconsole -
ca6c3e77d75dc3e3186084934957261503 71a4910c5c8a2994007c1f7e07875b


 ssoApplicationId": "NA -
ca6c3e77d75dc3e3186084934957261503 71a4910c5c8a2994007c1f7e07875b
```

Part V
Support

Learn about frequently asked questions, troubleshooting, and supported languages.

Chapters

37

Frequently Asked Questions for Oracle Identity Cloud Service

To see a list of frequently asked questions for Oracle Identity Cloud Service, see the FAQ page at cloud.oracle.com.

38

Troubleshoot Oracle Identity Cloud Service

Learn about common problems that you might encounter when using Oracle Identity Cloud Service and learn how to solve them.

Topics

System Settings and Profile Information

Learn about common problems that you might encounter when specifying system settings and profile information and learn how to solve them.

Topics

I'm an administrator. On the Default Settings page, I changed the language of the identity domain from English to French, and clicked Save. However, everything still appears in English. Why is that?

Sign out of the Identity Cloud Service console and log back in to see language-related changes for the Identity Cloud Service console.

I am trying to change my password but the Submit button is not enabled. How can I change my password?

Two things could be wrong:

- Make sure the password that you are setting meets all the requirements. If your password conforms to the password policy, then each criterion displays a green check mark.
- Make sure the password that you are entering in the **New Password** field and the password you are entering in the **Confirm New Password** field are the same.

I'm a user and I want to provide a separate email address for password recovery. However, when I go to the Email Options tab of the My Profile console, I don't see a way to do this. Why is that?

If your administrator did not enable the **Password Recovery Email** option for your Oracle Identity Cloud Service identity domain, then you can't specify a password recovery email address that is different than your primary email address.

I logged out of the Identity Cloud Service console, logged back in, and am redirected to the My Profile console. Why is that?

On the **Session Settings** page, the **Logout URL** field contains the URL that Oracle Identity Cloud Service uses to redirect the user after a successful login attempt. To redirect the user to the Identity Cloud Service console, change the value in the **Logout URL** field from `/ui/v1/myconsole` to `/ui/v1/adminconsole`.

How do I change the domain name of the email address that appears in notifications to my company's domain name?

As an example, in the **Forgot your password?** page, if a user enters an incorrect user name (the user name doesn't exist in the Oracle Identity Cloud Service identity store) and clicks **Submit**, then a message appears, stating that an email will be sent to `****@oracle.com`.

How do I change the domain name of this email address to my company's domain name?

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, and then click **Default Settings**.
2. In the **Email Addresses** text area, replace `oracle.com` with your company's domain name.
3. Click **Save**.
4. In the **Confirmation** window, click **Yes**.

Users

Learn about common problems that you might encounter when managing users and learn how to solve them.

Topics

I logged in successfully to Oracle Identity Cloud Service as an administrator. However, on the Home page, I see only a small subset of functionality. As an example, I don't see the Applications tab. Also, the dashboard doesn't display the Applications pane. Why is that?

The content that you see on the Identity Cloud Service console reflects the administration roles assigned to you. So, if you are not assigned to either the identity domain administrator or application administrator administration roles, you won't see any application-based functionality because you don't have permissions to do so. See Understanding Administrator Roles for a listing of the types of Oracle Identity Cloud Service administrator roles and the privileges for each role.

I am locked out of my account? I know I did not change my password. What happened? How do I access the service?

For security reasons, your administrator might have had to reset passwords for all users.

Check your inbox for an email from your administrator. If you do not have an email from your administrator, contact your administrator directly.

When importing users from a CSV file, I get an error that says “Invalid UTF-8...”

The problem is that the CSV file was not saved in an UTF-8 format. If you do not save the file in a CSV format with UTF-8 encoding, the import fails. Ensure that you have saved the CSV file in UTF-8 format and try to import the file again.

See Import User Accounts and Import Groups.

While importing users, I am getting the following error, “Unable to determine ID for : [MANAGER NAME]”. How do I fix this?

Validate that the user with the ID given in the “Manager Name” column already exists in the system or that the user is being created as a new user in the same CSV import.

When activating a new user or resetting a user password, the administrator profile displays instead of the profile page for the newly activated user. Why?

This behavior occurs if the administrator and the user are sharing the same browser session or window at the same time.

To work around this issue, ensure that the administrator and the user are not sharing the same browser session or window at the same time.

Close or Cancel button not available when the user forgets their password

There are a couple of situations where the user wants to reset their password and the pages they see do not have a Cancel button or a Close button. This happens when the user is directly accessing the **Forgot your password?** page, such as in these cases:

- You have a custom page with a link to the **Forgot your password?** page and the user uses it.
- The user directly enters the URL for the **Forgot your password?** page in the browser.

Usually when the user cancels the reset password procedure or closes the page after submitting the request, the login page is shown. But in these cases, the user has accessed the **Forgot your password?** page directly. This means that the SSO Cookie that should call the SSO endpoint and redirect to the login page hasn't been generated.

1. The **Cancel** button is not be displayed on the **Forgot your password?** page.
2. The **Close** button is not displayed on the following page when the user has entered their user name and clicked **Next**.

Groups

Learn about common problems that you might encounter when using groups and learn how to solve them.

Topics

While importing groups, I am getting the following error, “Unable to determine ID for : <userId>” What does this mean?

This error usually occurs because the user does not exist in the system. Make sure the user with the ID listed in the error exists in the system. New users are not created while importing groups.

Import Users and Groups

Learn about common problems that you might encounter when importing users and groups and learn how to solve them.

Topics

While importing users or groups, I am getting one of the following errors.

- “Failure Reason: Mandatory CSV Header not found : <Header Name>.”
- “Failure Reason: Invalid CSV headers found….”
- “Reading of CSV file unsuccessful : Import CSV records count exceeds records limit”

“Failure Reason: Unable to parse CSV file for upload….”

This error occurs when the CSV import file is invalid. Check the format of CSV file by opening the file in a standard spreadsheet application, such as Microsoft Excel or Google Sheets to make sure that each row contains same number of columns.

> **Tip:**
>
> This error also occurs if the CSV file that you are uploading is empty.

“Failure Reason: Mandatory CSV Header not found : <Header Name>.”

There are minimum set of headers that must be provided in the CSV file while importing users or groups.

- For users, the CSV import file must contain “User ID”, “Last Name”, “First Name”, “Work Email” columns.
- For groups, the CSV import file must contain "Display Name", "Description", and "User Members".

“Failure Reason: Invalid CSV headers found….”

This error means that the CSV file contains an extra CSV header, which is not permitted as a part of the CSV file.

1. Open the file in a standard spreadsheet application, such as Microsoft Excel or Google Sheets, and then delete the column identified in the error message.
2. Reimport the CSV file.

“Reading of CSV file unsuccessful : Import CSV records count exceeds records limit”

This error means that the CSV import file contains more records than the permissible limit. The maximum number of user accounts that can be imported in a single job must not exceed 100,000 user accounts. For optimal performance, Oracle recommends that you import your users in batches of 25,000. The maximum number of groups that can be imported in a single job must not exceed 100,000 groups. The maximum number of user members per group row in your CSV file must not exceed seven. For optimal performance, Oracle recommends that you import your groups in batches of 10,000.

“No data to process for import”.

This error means that the uploaded CSV import file contains no data in it. Check the file to make sure that it contains data.

Applications

Learn about common problems that you might encounter when using applications and learn how to solve them.

Topics

I added a custom mobile application, but the Client Secret doesn't appear in the Application Added window. Why is that?

Because a custom browser or mobile device application runs on an unauthenticated browser, machine, or mobile device, Oracle Identity Cloud Service doesn't generate a **Client Secret** for this type of application.

I'm trying to add an Oracle application to Oracle Identity Cloud Service, but I can't do this. Why is that?

When you use an Oracle application as part of a subscription-based service, your application is cloud-ready (and ready for you). Therefore, you don’t have to add it to Oracle Identity Cloud Service.

Unable to Obtain Access Token with Special Characters in the Client ID or Client Secret.

When requesting an access token, the client id and client secret cannot contain special characters.

I deleted an App Link associated with my SAML App, but it is still appearing on the My Apps page.

You need to wait (typically a few seconds) for the asynchronous task to remove the App Link before the App no longer appears on the **My Apps** page.

I'm an end user and I do not see an application that has been granted to me on the My Apps page. Why is that?

Contact your administrator with the details such as the application name. A likely cause could be that the administrator did not select the **Display in My Apps** option for that application.

Unable to synchronize a deleted user when that user is created again in the authoritative application.

After deleting a user from the authoritative application, you need to perform full synchronization before you create the user again in the authoritative application.

When you delete a synchronized user from Oracle Identity Cloud Service, then the user is also deleted from the authoritative application.

There is no workaround for this at the moment.

Identity Providers

Learn about common problems that you might encounter when using identity providers and learn how to solve them.

Topics

I am getting invalid signature errors for my Identity Provider. My certificates look correct in the metadata. What could be wrong?

If an Identity Provider partner is created using metadata and the metadata contains two certificates with use=”signing” specified, the runtime verifies that the messages from the Identity Provider are signed with the first certificate. If you see invalid signature errors, your Identity Provider is probably signing with the second certificate.

To remove the second signing certificate that is not being used by the Identity Provider to sign the messages, update the metadata.

I am trying to import the Oracle Identity Cloud Service metadata. However, it fails because the certificates are not considered as valid. Why is that?

Unlike many SAML 2.0 Identity or Service Providers, Oracle Identity Cloud Service does not use self-signed certificates for signing and encrypting of SAML 2.0 requests and responses. However, the metadata file only includes the signing and encryption certificates. To get the missing root certificate from Oracle Identity Cloud Service, see Obtaining the Root CA Certificate from Oracle Identity Cloud Service.

Password Policies

Learn about common problems that you might encounter when using password policies and learn how to solve them.

Topics

I'm trying to customize the Simple and Standard password policies, but I can't do this. Why is that?

The Simple password policy is used for your developer services and demos when you don't want to customize a policy for them. You can't modify this type of password policy.

The Standard password policy is used when you don't want to use the Oracle-recommended password policy for your enterprise applications. You can't modify this type of password policy.

Use the Custom password policy to tailor the strength of your password policy to meet the business and security requirements for your enterprise applications.

The Microsoft Active Directory (AD) Bridge

Learn about common problems that you might encounter when using the Microsoft Active Directory (AD) Bridge and learn how to solve them.

Topics

I can’t use the client for the AD Bridge to connect to Oracle Identity Cloud Service. What’s wrong?

If you receive the following error message when you’re creating an AD Bridge:

```
The underlying connection was closed: Could not establish trust relationship
for the SSL/TLS secure channel.
```

then select the **Use SSL** check box because your AD server is using an SSL connection to communicate with the bridge.

I’m trying to use the client for the AD Bridge to connect to my AD server. All of my connection details appear to be correct. However, when I click Test, the client can’t recognize the URL. Why is that?

Make sure that the Identity Cloud Service URL matches the URL that's shown on the **Install a Bridge for the Microsoft Active Directory Domain** page. To access this page, launch the Identity Cloud Service console, expand the **Navigation Drawer**, click **Settings**, click **Directory**

Integrations, and then click **Add**. In addition to the Identity Cloud Service URL, the page also displays the Client ID and Client Secret.

My AD Bridge now has a status of Unreachable, even though previously, it had a status of Active. Why is that?

Your AD Bridge can have an **Unreachable** status because:

1. The Oracle Identity Cloud Service administrator uninstalled the client associated with your AD Bridge, but the bridge couldn't be removed from the **Directory Integrations** page of the Identity Cloud Service console because the client can't connect to the Oracle Identity Cloud Service server. Oracle Identity Cloud Service can't use the bridge to communicate with AD. See Remove a Microsoft Active Directory (AD) Bridge.
2. The administrator regenerated the Client Secret for your AD Bridge, and then uninstalled the client for the bridge.
3. Your AD Bridge is installed and configured. However, the back-end service (or agent) used to establish communication between Oracle Identity Cloud Service and AD is stopped.

   To restart this agent:

   a. Click **Start**.
   b. In the **Search programs and files** text box, enter **Services**, and then press **Enter**.
   c. In the **Services** window, click **Services (Local)**, **Identity Cloud Service Microsoft Active Directory Bridge Service**, and then click **Start**.
   d. Verify that **Started** appears as the status for the service.

I used the AD Bridge to import a group into Oracle Identity Cloud Service, and then deleted the group in Oracle Identity Cloud Service. How can I re-establish a link between the group in AD and the group in Oracle Identity Cloud Service?

1. In the Identity Cloud Service console, click **Settings**.
2. In the side navigation bar, click **Directory Integrations**.
3. Click the AD Bridge that you want to configure.
4. Click the **Configuration** tab.
5. In the **Select organizational units (OUs) for groups** pane, clear the check box for the designated group, and then click **Save**.
6. Select the check box for the group, and then click **Save** again.
7. Run the AD Bridge to synchronize the group between Oracle Identity Cloud Service and AD immediately.

I regenerated the Client Secret for my AD Bridge, and now my bridge isn't working. Why is that?

If you're using the 17.2.6 version of the client for the AD Bridge, then you must upgrade your client to the latest version. See Create a Microsoft Active Directory (AD) Bridge to install the updated client for the bridge.

I'm trying to use the AD Bridge to import AD users into Oracle Identity Cloud Service, but I'm not able to do this. Why is that?

The AD Bridge must be able to access the AD organizational units (OUs) and the parent OUs that contain the users you want to import into Oracle Identity Cloud Service. To ensure that the bridge can access the OUs:

1. Launch Active Directory Users and Computers.
2. Right-click the OU that contains the users you want to import into Oracle Identity Cloud Service, and select **Properties** from the drop-down menu.
3. In the **Properties** window, click the **Security** tab.
4. In the **Advanced Security Settings** window, click the **Security** tab, and click **Advanced**.
5. Click **Add**.
6. In the **Permission Entry** window, click the **Select a Principal** link.
7. In the **Select User, Computer, Service Account, or Group** window, search for the user with which the AD Bridge is configured, and click **OK**.
8. In the **Permission Entry** window:
    a. From the **Type** drop-down menu, select **Allow**.
    b. From the **Applies to** drop-down menu, select **This Object and all descendant objects**.
    c. From the **Permissions** pane, select the **List contents**, **Read all properties**, and **Read permissions** check boxes.
    d. Click **OK**.
9. In the **Advanced Security Settings** window, click **OK**.
10. In the **Properties** window, click **OK**.
11. Close Active Directory Users and Computers.

Reports

Learn about common problems that you might encounter when using reports and learn how to solve them.

Topics

I'm an audit administrator and I'm running the Successful Login Attempts report. Is there a way for me to see which users logged into Oracle Identity Cloud Service successfully by using an identity provider, and which users logged in successfully directly through Oracle Identity Cloud Service?

When you open this report, click the **Provider** column. Oracle Identity Cloud Service sorts users who logged in successfully by the provider. If an external identity provider is not used, **localIDP** appears in the **Provider** column, signifying users logged in successfully directly through Oracle Identity Cloud Service. Otherwise, you'll see the name of the provider sorted in ascending or descending order.

Customize the Interface

Learn about common problems that you might encounter when customizing the interface and learn how to solve them.

Topics

I'm having trouble uploading images and icons in Oracle Identity Cloud Service. Are there standards I'm supposed to follow?

See the following table for guidelines.

If there's no max width allowed and no max height allowed listed, then you can uoploaad any width or height as long as the file size is less than or equal to the Maximum File Size listed. Upload the image or icon by using the recommended dimensions and then make adjustments as needed.

Allowed file types are: GIF, JPEG, JPG or PNG.

All measurements below are in pixels.

Image or Icon	Max Width and Height Allowed	Recommended Dimensions Width X Height	Maximum File Size	Ratio	Notes	Reference
SAML Identity Provider Icons	None	48W X 48H	300 KB	1:1	If you use larger images, be aware that larger images will be resized to maintain a 1:1 ratio. The icon should have a transparent background. In addition to the other supported file types, FIG is also allowed for SAML Identity Provider icons.	Add a SAML Identity Provider

Image or Icon	Max Width and Height Allowed	Recommended Dimensions Width X Height	Maximum File Size	Ratio	Notes	Reference
Social Identity Provider Icons	None	95W X 95H	300KB	1:1	If you use larger images, be aware that larger images will be resized to maintain a 1:1 ratio. The icon should have a transparent background. In addition to the other supported file types, FIG is also allowed for Social Identity Provider icons.	Add a SAML Identity Provider
Sign-in Page Logo	250W X 50H	250W X 50H	300 KB	5:1	If you use larger images, be aware that larger images will be resized to maintain a 5:1 ratio.	Brand the Consoles Customize the Sign In Page
Sign-in Page Background Image	None	None	300 KB	Not applicable	If the image is too small, the image is repeated horizontally and vertically.	Brand the Consoles
Admin Console and My Console Logo	64H No restriction in width.	250W x 50H	300 KB	5:1	If you use an image that is 250 pixels wide or larger, the text on My Console might be displaced.	Brand the Consoles
Email Notification Templates - Header Logo	None	160W X 40H	300 KB	4:1	None	Brand Notification Templates

Web Browser

Learn about common problems that you might encounter when using a web browser with Oracle Identity Cloud Service and learn how to solve them.

Topics

When I try to launch Oracle Identity Cloud Service using Firefox on Linux, the web browser crashes. What now?

The problem is that you need an updated version of the Linux Firefox browser.

Install a newer version of Linux Firefox browser from https://ftp.mozilla.org/pub/firefox/releases/.

When I try to configure the Microsoft Active Directory (AD) Bridge after downloading it by using Safari, it doesn't work. Why?

The AD Bridge runs only on a Windows environment. If you must download it to another environment, then you still have to install it on a Windows environment.

The User Name and Password Fields are Pre-Populating in the UI. How do I prevent this?

This is a common issue across browsers.

To resolve this issue, when the browser prompts you to remember the user name or password, for example, when logging in to Oracle Identity Cloud Service, select the option **Never save password for this site**.

Troubleshoot App Gateway

Learn about common problems that you might encounter when setting up App Gateway and how to solve them:

I Made Changes in Oracle Identity Cloud Service but the App Gateway Server Doesn't Reflect the Changes

Changes you make to enterprise applications and App Gateway definitions in Oracle Identity Cloud Service may not be reflected immediately on App Gateway because App Gateway

caches Oracle Identity Cloud Service information, such as resources, authentication policies, and header values of enterprise applications.

Explanation: App Gateway contacts Oracle Identity Cloud Service using agents to collect host and port information. When you start App Gateway, its NGINX server is automatically configured with this information. Any changes to Oracle Identity Cloud Service is periodically polled by the agents.

By default the policy and headers refresh time are 3600 seconds (1 hour) each. To change these values, log in to the App Gateway server, and edit the `/usr/local/nginx/conf/cloudgate.config` file. Change the `ttl` value for `policy` and `headers` in the `caching` section as per the following example, and then restart both App Gateway server and the agent.

```
"caching" : {
  "minimumTtl"            : 300,
  "headers"               : { "ttl": 3600 },
  "discovery"             : { "ttl": 3600 },
  "policy"                : { "ttl": 3600},
  "tenantKeys"            : { "ttl": 86400 }
}
```

You can also change the poll interval of the agents. By default, the agent's refresh time to get new App Gateway configuration from Oracle Identity Cloud Service is 60 seconds, which is the minimum amount of time supported. In the `/usr/local/nginx/conf/cloudgate.config` file, change the `pollIntervalSecs` value in the `agentConfig` section as per the example:

```
"agentConfig": {
    "pollIntervalSecs"    : 60,
    "daemon"          : true,
    "logLevel"         : "warn",
    "logFolder"         : ""
}
```

If you want the changes in the Enterprise Application configuration to be reflected immediately, stop the App Gateway server and then start the server.

```
/scratch/oracle/cloudgate/home/bin/cg-stop
/scratch/oracle/cloudgate/home/bin/cg-start
```

If you want the changes in the App Gateway configuration to be reflected immediately, stop the agent and then start the agent.

```
/scratch/oracle/cloudgate/home/bin/agent-stop
/scratch/oracle/cloudgate/home/bin/agent-start
```

See Start and Stop App Gateway

Error Log Files Contain Invalid_session Message

When App Gateway can't communicate correctly with Oracle Identity Cloud Service, you'll find `invalid_session` messages in the App Gateway error log files.

The following is an example of an `invalid_session` messages in `error.log` file:

```
www-authenticate: Bearer error="invalid_session",
error_description="Authentication Failure
```

This can be because of the way App Gateway processes a client request to a protected resource. App Gateway uses `NGINX` sub requests to make requests to Oracle Identity Cloud Service, and then App Gateway requires Linux `NGINX` resolver to be configured appropriately to allow these sub requests to function correctly.

1. Verify that the resolver setting in the file `/usr/local/nginx/conf/nginx-cg-sub.conf` is set to the correct IP.
2. Verify that the tenant name in `/usr/local/nginx/conf/cloudgate.config` file is configured correctly.

Error Log Files Contain GET 127.0.0.1:53 Command Responding Error Number 500

Because App Gateway makes sub requests to an internal servlet, App Gateway requires your virtual machine to listen to port `53`.

The App Gateway server must communicate to itself through IP address `127.0.0.1` and port `53`.

If you're running App Gateway in a virtual machine software, configure port forward for this port from the host to the guest. See Configure Port Forwarding Rules

App Gateway Server Can't Communicate With Oracle Identity Cloud Service

Use a SSH client such as `PuTTY` and the following credentials to log in to the App Gateway server:

1. Execute the `sudo su -` command to login as `root`, and when prompted provide the oracle password.
2. Install telnet by running the following command:

   ```
   yum install telnet
   ```

3. Run the following telnet command and try to establish a connection to your Oracle Identity Cloud Service instance and your application from the App Gateway server.

   ```
   telnet <idcs-tenant>.identity.oraclecloud.com 443
   ```

   If telnet can't connect to your Oracle Identity Cloud Service, then contact your network administrator to apply any other network configuration to enable the App Gateway server to establish connection with your Oracle Identity Cloud Service instance.

4. Execute the `exit` command, to log out from root account.

39
Supported Languages

Oracle Identity Cloud Service offers a localized user experience for its web interface.

By default, the web interface language is set to match the web browser locale, but users can override this setting in their profile details. If users change their language setting, the change won't take effect until the next time they sign in.

The following languages are available:

Chinese – Simplified	Italian
Chinese – Traditional	Japanese
English	Korean
Finnish	Norwegian
French	Portuguese – Brazilian
French – Canadian	Spanish
German	

Part VI

Complete Oracle Identity Cloud Service Scenarios

Learn about Oracle Identity Cloud Service scenarios such as enabling multi-factor authentication and migrating from traditional cloud accounts to cloud accounts with Oracle Identity Cloud Service.

Topics:

40

Enable Multi-Factor Authentication Security for Oracle Cloud

This scenario is applicable to customers who have recently signed up for the Oracle Cloud Service or those who have migrated to a new Oracle Cloud account. In this scenario, you add one more layer of security to the Oracle Cloud sign in process by configuring Multi-Factor Authentication (MFA).

Topics:

Scenario Description

In this scenario, the Oracle Cloud customer has hired a third-party company to work in the customer's cloud environment. These partner employees work remotely and need to manage Plataform (PaaS) and Infrastructure (IaaS) instances in the customer's cloud environment.

The customer's security office requires that each partner employee provide a second verification factor on top of the traditional user name and password for authentication.

This scenario’s requirements are as follows:

- A second verification factor is required each time that a user signs in. Users can't sign in using just their user name and password.
- Enable two factors for 2-Step Verification: The Oracle Mobile Authenticator (OMA) app on each user's own mobile device and the one-time passcode (OTP) sent to the user's registered email address.
- During the authentication process, users can enable their device as a trusted device for a maximum time frame of one day and can register only one trusted device.
- The email passcode must not be valid for more than 10 minutes.

Understand MFA Options in Oracle Identity Cloud Service

The MFA feature in Oracle Identity Cloud Service enables the customer to add an extra security step to the authentication process. There are four possible factors and a backup method that can be enabled.

- **Security Questions:**

  Users are prompted during the sign-in process to correctly answer a defined number of security questions to verify their identity.

- **Mobile Authenticator Application:**

  Users generate an OTP on the OMA App on their device that must be used during log in.

- **Text Message (SMS):**

  Users receive a temporary passcode as a text message (SMS) on their device that must be used during log in.

- **Email:**

  Users receive an email message that contains a temporary passcode that must be used during log in.

- **Bypass Code:**

  Oracle Identity Cloud Service also enables users to generate a bypass code, which can be used as a backup method when users have forgotten the answers for the security questions, don't have a mobile phone, or can't access their email.

This scenario requires you to enable **Mobile Authenticator Application** and **Email**.

Create the Partners Group

Before setting up the MFA factors, you need to create a Partners group in Oracle Identity Cloud Service console.

Each of the partner's user need to be assigned to the Partners group.

Enable the Factors

In this scenario, you select **Mobile App Passcode** and **Email** as the MFA factors available to the users, and configure a sign-on policy rule for the Partners group.

The following are high level steps to enable these authentication factors:

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **MFA**.
2. Select **Mobile App Passcode** and **Email** from the available factors, and then click **Save**.
3. Click **Security** in the **Navigation Drawer**, and then click **Sign-On Policies**.
4. Select the **Default Sign-On Policy**, click **Sign-On Rules** tab, and then click **Add** to add a new rule.
5. Enter a **Rule Name**
6. In the **Conditions** section, **And is a member of these groups** field, select **Partners** from the list that appears.

7. In the **Actions** section:
   a. Select **Prompt for an additional factor**.
   b. Set **Enrollment** as **Required** to force the user to enroll in MFA.
8. Click **Save**.
9. After you save the rule, drag the new rule to the position above the **Default Sign-On Rule**.
10. Click **Save** to save the default sign-on policy.

Configure MFA for Email

Select Email, so that Oracle Identity Cloud Service sends a one-time passcode to the user's primary email address that was provided while setting up the user's account.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Security**, and then click **Factors**.
2. Click the **Email** tab, and update the settings according to your requirements.

Configure MFA for the Mobile Authenticator App

Select OMA, so that Oracle Identity Cloud Service you can use Mobile Authentication App to allow or deny access to Oracle Identity Cloud Service.

- In Oracle Identity Cloud Service console, expand the **Navigation Drawer**  , click **Security**, and then **Factors**.

  The **Factors** page opens, and the **Mobile App** tab is selected by default.

The default values for the **Passcode Policy** section are the industry-recommended settings. There is no need to change any of these values.

Although there are no specific requirements about using rooted devices or an older operating system version, the **Compliance Policy** section allows you to define such verification:

- Mobile authenticator app version check: Block users from using an outdated app.
- Minimum OS version check: Block users from using the app on a device that has an outdated operating system. Users won't receive push notification requests and won't be able to generate passcodes.
- Rooted devices check (iOS and Android only): Block users from using the app on a device that is rooted or where rooted status is unknown. Users won't receive push notification requests and won't be able to generate passcodes.

The **Compliance Policy** section also allows you to define device screen lock verification. Device screen lock check can be used to prevent users from using the app on a device that doesn't have a screen lock or where the screen lock status is unknown. Users won't receive push/pull notification requests and won't be able to generate passcodes.

Create Users

Use the Oracle Identity Cloud Service console to create a user for each partner employee, and then assign all of them to the **Partners** group.

The users need to be registered in Oracle Identity Cloud Service with their third-party company's email address, to receive the Welcome email.

Verify that Users Can Access Oracle Cloud

After you enable and configure **Email** or **Mobile App** factors and create users in Oracle Identity Cloud Service, the first time that a user logs in to Oracle Cloud, the user is prompted to enroll for MFA.

1. The user must select **Enable**, and then select **Email** or **Mobile App**. The user generates an OTP using the OMA App or receives an OTP via email, and then enters the code as the second verification method.
2. The user must then select an additional method as a backup factor. The user selects either **Mobile App** or **Email,** and then follows the on-screen instructions.
3. After the user successfully enrolls for both factors and closes all browser windows, the user should be able to open a new browser window and log in to the customer's Oracle Cloud environment.

Generate and Use the Bypass Code

If you enabled the Bypass Code factor in Oracle Identity Cloud Service, users can choose to generate a bypass code after they enroll in 2-Step Verification.

1. After the user successfully signs in and provides a second authentication factor, they can access the **My Profile** page in Oracle Identity Cloud Service by clicking their user initials in the upper-right corner, and then selecting **My Profile** from the drop-down list.
2. The user needs to then select the **2-step Verification** tab, and then click **Manage**.
3. On the **2-Step Verification** page, the user clicks **Generate Bypass Code**. In the confirmation dialog box, the user should copy the bypass code and store it in a safe place for future usage as a backup verification method.
4. The user then clicks **Done**. The bypass code appears in a table that displays the number of uses allowed and when the code expires.
5. Instruct the user to sign out and then access Oracle Identity Cloud Service's **My Profile** page again. Have the user sign in, but instead of providing the 2-Step Verification factor, click **Use backup verification method**, select **Use a bypass code**, and then provide the previously generated bypass code. The user should be able to successfully access the Oracle Identity Cloud Service **My Profile** page.

The generated Bypass Code works can only be used once. The user should generate a new bypass code and store it for future use.

41

Migrate from Traditional Cloud Accounts to Cloud Accounts with Identity Cloud Service

This section describes how to migrate users and role memberships for Oracle Cloud services from traditional cloud accounts to cloud accounts with Oracle Identity Cloud Service.

Topics:

Typical Workflow for Migrating from Traditional Cloud Accounts to Cloud Accounts with Identity Cloud Service

You can migrate traditional cloud accounts to cloud accounts with Identity Cloud Service by migrating users, role memberships, and **identity domain administrator** roles, provisioning and synchronizing users between traditional cloud accounts and cloud accounts with Identity Cloud Service, and migrating service-specific data and artifacts.

Task	Description	Additional Information
Understand traditional cloud accounts and cloud accounts with Identity Cloud Service.	You can learn about how traditional cloud accounts differ from cloud accounts with Identity Cloud Service.	About Traditional Cloud Accounts and Cloud Accounts with Identity Cloud Service

Task	Description	Additional Information
Understand migrating services from traditional cloud accounts to cloud accounts with Identity Cloud Service.	You can learn why you should migrate your traditional cloud accounts to cloud accounts with Identity Cloud Service. You can also learn about services in traditional cloud accounts that contain users and role memberships you can migrate, along with the corresponding services for cloud accounts with Identity Cloud Service to which you can import the users and application role memberships.	About Migrating Services from a Traditional Cloud Account to a Cloud Account with Identity Cloud Service
Understand your prerequisites before migrating services.	You can learn about what you should have before migrating services from traditional cloud accounts to cloud accounts with Identity Cloud Service.	Before You Begin
Migrate users.	You can migrate users from traditional cloud accounts to cloud accounts with Identity Cloud Service.	Migrate Users
Migrate role memberships.	You can migrate role memberships from traditional cloud accounts to cloud accounts with Identity Cloud Service.	Migrate Role Memberships
Migrate **identity domain administrator** roles.	You can assign the **identity domain administrator** role to the users whom you migrated from traditional cloud accounts to cloud accounts with Identity Cloud Service.	Migrate Identity Domain Administrator Roles
Provision and synchronize users between traditional cloud accounts and cloud accounts with Identity Cloud Service.	You can configure a traditional cloud account to be a service provider and Oracle Identity Cloud Service to be an identity provider. As a result, a user can use their federated SSO credentials to log in to the traditional cloud account through their cloud account with Identity Cloud Service. You can also create an Oracle Cloud application in Oracle Identity Cloud Service that's used to provision and synchronize users between traditional cloud accounts and cloud accounts with Identity Cloud Service.	Provision and Synchronize Users Between Traditional Cloud Accounts and Cloud Accounts with Identity Cloud Service

Task	Description	Additional Information
Understand the mapping between traditional cloud roles and application roles in Oracle Identity Cloud Service.	View a reference table that lists the services in traditional cloud accounts that contain roles you want to export. It also lists the services for cloud accounts with Identity Cloud Service and the application roles to which you want to assign users. Use this table when migrating role memberships.	Map Between Traditional Cloud Roles and Application Roles in Oracle Identity Cloud Service
Migrate service-specific data and artifacts.	You can migrate service-specific data and artifacts for some cloud services.	Migrate Service-Specific Data and Artifacts

About Traditional Cloud Accounts and Cloud Accounts with Identity Cloud Service

You use an Oracle Cloud account to access your cloud services and log into the Oracle Cloud Infrastructure Classic Console, which is where you manage your account and your services.

When you go to sign in to your Oracle Cloud account, you can choose to sign in to two different types of cloud accounts: a traditional cloud account (also known as a cloud service account) and a cloud account with Identity Cloud Service. Traditional cloud accounts use one identity management system which is different from the identity management system associated with cloud accounts with Identity Cloud Service.

About Migrating Services from a Traditional Cloud Account to a Cloud Account with Identity Cloud Service

You can migrate users and role memberships for Oracle Cloud services from traditional cloud accounts to cloud accounts with Identity Cloud Service.

Each Oracle Cloud service has a corresponding cloud account with Identity Cloud Service to which you can import the users and application role memberships. By migrating services from a traditional cloud account to a cloud account with Identity Cloud Service, the services can use Oracle Identity Cloud Service to manage users and to control access to the services. For this reason, you want to migrate your traditional cloud accounts to cloud accounts with Identity Cloud Service.

See Map Between Traditional Cloud Roles and Application Roles in Oracle Identity Cloud Service to view a table that lists the services in traditional cloud accounts and the role memberships that you can export along with the services for cloud accounts with Identity Cloud Service and the application roles to which you can assign users. Also, refer to Migrate Role Memberships to learn how to migrate role memberships from traditional cloud accounts to cloud accounts with Identity Cloud Service.

Before You Begin

Before migrating services from traditional cloud accounts to cloud accounts with Identity Cloud Service, ensure that you have the following access and administrative privileges:

- Access to the traditional cloud account that contains the users and role memberships you want to export
- Administrative privileges to export users and role memberships from this traditional cloud account
- Access to a cloud account with Identity Cloud Service
- Administrative privileges in Oracle Identity Cloud Service to import users and application role memberships

In the cloud account with Identity Cloud Service, create instances for each service that you plan to migrate. For example, if you plan to migrate two instances of Oracle Integration Cloud Service, then use the cloud account with Identity Cloud Service to create two instances of this Oracle Cloud service. Also, if there are multiple instances of a cloud service, then migrate each instance, one by one. So, there should be a one-to-one mapping between an old instance and a new service instance.

Migrate Users

To migrate users, export them from the traditional cloud accounts to a CSV file, modify the heading row in the CSV file, and import the users from the CSV file into Oracle Identity Cloud Service.

When you're migrating users to cloud accounts with Identity Cloud Service, you may want users to use the passwords from their traditional cloud accounts. To do this, change the minimum length of the custom password policy in Oracle Identity Cloud Service to eight characters.

> **Note:**
>
> Before importing user accounts into Oracle Identity Cloud Service, the account administrator or service administrator should notify the users about the migration of their traditional cloud accounts to cloud accounts with Identity Cloud Service.
>
> After all user accounts are imported, each user will receive a Welcome email notification. The user can use the link in this notification to access their account and set a password for it.

Export Users from Traditional Cloud Accounts

1. Sign in to the Oracle Cloud Infrastructure Classic Console of the traditional cloud account that contains the users that you want to export.
2. Expand the **Navigation Drawer**  in the top left corner, and then click **Users**.
3. In the **User Management** page, select the users that you want to export.
4. Click **Export**. The users will be exported into a CSV file.
5. In the dialog box that appears, save the CSV file to your machine.

Modify the CSV File

> **Note:**
>
> At a minimum, the file must have these exact column headings and the fields in these columns must be unique.
>
> - User ID
> - Last Name
> - First Name
> - Work Email
> - Primary Email
> - Primary Email Type

1. Open the CSV file for editing.

   You can use a standard spreadsheet application, such as Microsoft Excel or Google Sheets, or you can use a text editor, such as Notepad or TextPad.
2. Change the **User Login** column heading to **User ID** and the **Email** column heading to **Work Email**.
3. Add a **Primary Email Type** column.

   When importing users, the attribute `Recovery` cannot be specified as one of valid values for Primary Email Type. The valid values for Primary Email Type are `home`, `work`, or `other`.
4. Save your changes to the CSV file.

Import Users into Oracle Identity Cloud Service

1. Sign in to Oracle Cloud.
2. In the Oracle Cloud Infrastructure Classic Console, expand the **Navigation Drawer**  in the top left corner, and then click **Users**.
3. In the **User Management** page, click **Identity Console**.

   The Identity Cloud Service console opens.
4. Expand the **Navigation Drawer**, and then click **Users**.
5. Click **Import**.
6. In the **Import Users** dialog box, click **Browse** to locate and select your CSV file.
7. Verify that the path and name of the CSV file you selected appear in the **Select a file to import** field.
8. Click **Import**.
9. After Oracle Identity Cloud Service evaluates all users, review the job results.
   - If the job can be processed immediately, then a dialog box appears with the **Job ID** link for your import job. Click the link and review the details that appear on the **Jobs** page.

- If the job can't be processed immediately, then a message appears with a Schedule ID in it. Copy that Schedule ID, and use it to search for the job on the **Jobs** page. The job will appear when processing completes. Go to step 9.

10. In the **Jobs** page, locate the job that you want to view, and then click **View Details**.

    A table displays the first names, last names, email addresses, user names, and statuses of the users that you imported into Oracle Identity Cloud Service.

11. Review the details that appear on the **Jobs** page.

    This page shows how many users you imported, how many users you imported successfully, and how many users can't be imported because of a system error.

Migrate Role Memberships

To migrate role memberships, first, export them from the traditional cloud accounts. Then, modify the CSV file that contains the role memberships you exported so that you can import them into Oracle Identity Cloud Service. Next, import the role memberships into Oracle Identity Cloud Service.

You migrate role memberships individually. So, if you want to migrate 10 roles from a traditional cloud account to a cloud account with Identity Cloud Service, then you need to migrate one role at a time.

To see the mappings between roles in traditional cloud accounts and application roles in cloud accounts with Identity Cloud Service, refer to the table in Map Between Traditional Cloud Roles and Application Roles in Oracle Identity Cloud Service.

Export Role Memberships from Traditional Cloud Accounts

1. Sign in to the Oracle Cloud Infrastructure Classic Console of the traditional cloud account that contains the role memberships that you want to export.
2. Expand the **Navigation Drawer** in the top left corner, and then click **Users**.
3. Navigate to the **Roles** tab.
4. Select the role that contains memberships you want to export.
5. Click **Export**. The role memberships will be exported into a CSV file.
6. In the dialog box that appears, save the CSV file to your machine.

Modify the CSV File

1. Locate the exact name of the application role to which you want the users to belong.
2. Open the CSV file that you exported (because you want to modify it).
3. Modify the CSV file as follows:
   a. Remove the **First Name**, **Last Name**, and **Email** column headings.
   b. Change the **User Login** column heading to **Grantee Name**.
   c. Add the **Entitlement Value** column heading. The value for all rows of this column should be the name of the application role you noted in step 1 of this procedure.
   d. Add the **Grantee Type** column heading. The value for all rows of this column should be **User**.
4. Save your changes to the CSV file.

Import Role Memberships into Oracle Identity Cloud Service

1. In the **Applications** page of the Identity Cloud Service console, click the application that has a role to which you want to assign users.
2. Click **Application Roles**.
3. Click **Import**.
4. In the **Import Application Roles** dialog box, click **Browse** to locate and select your CSV file.
5. Verify that the path and name of the CSV file you selected appear in the **Select a file to import** field.
6. Click **Import**.
7. After Oracle Identity Cloud Service evaluates all users that are to belong to the application role, review the job results.
   - If the job can be processed immediately, then a dialog box appears with the **Job ID** link for your import job. Click the link and review the details that appear on the **Jobs** page.
   - If the job can't be processed immediately, then a message appears with a Schedule ID in it. Copy that Schedule ID, and use it to search for the job on the **Jobs** page. The job will appear when processing completes. Go to step 8.
8. In the **Jobs** page, locate the job that you want to view, and then click **View Details**.

   A table displays the user names, classification type (User), and statuses of the users that you imported and assigned to the application role in Oracle Identity Cloud Service.

Migrate Identity Domain Administrator Roles

If a user is an identity domain administrator for their traditional cloud account, then they should also be an identity domain administrator for their cloud account with Identity Cloud Service.

If the user set up their cloud account with Identity Cloud Service, then they will have the **identity domain administrator** role. However, if this role is not assigned to the user, you must assign it.

In this topic, you assign the **identity domain administrator** role to the users that you imported into Oracle Identity Cloud Service.

1. Sign in to the Oracle Cloud Infrastructure Classic Console of the cloud account with Identity Cloud Service.
2. Expand the **Navigation Drawer**  in the top left corner, and then click **Users**.
3. In the **User Management** page, click **Identity Console**.

   The Identity Cloud Service console appears.
4. Expand the **Navigation Drawer**, click **Security**, and then click **Administrators**.
5. Expand the **Identity Domain Administrator** node.
6. Click **Add**, select the check boxes only for those users whom you imported into Oracle Identity Cloud Service and who are identity domain administrators for their traditional cloud accounts, and then click **OK**.

Provision and Synchronize Users Between Traditional Cloud Accounts and Cloud Accounts with Identity Cloud Service

User provisioning and synchronization are important aspects of application management. Provisioning allows you to manage the lifecycle of accounts in applications like creating and deleting accounts using Oracle Identity Cloud Service.

For example, when you grant the user access to an Oracle Cloud application that's used to provision users with traditional cloud accounts, then this user is provisioned with the traditional cloud account automatically. This allows you to quickly add new users to traditional cloud accounts and de-provision users from these accounts instantly when they change roles or leave your organization.

After enabling provisioning, synchronization allows you to control how operations like creating and deleting traditional cloud accounts are reflected in Oracle Identity Cloud Service.

For provisioning and synchronization to occur for users between traditional cloud accounts and cloud accounts with Identity Cloud Service, you configure a traditional cloud account to be a service provider and Oracle Identity Cloud Service to be an identity provider. As a result, a user can use their federated SSO credentials to log in to the traditional cloud account through their cloud account with Identity Cloud Service.

A user must be authenticated only once. For this example, the user obtains a security token. This security token is then validated by Oracle Identity Cloud Service so that the user can access the traditional cloud account. This method is known as federated single sign-on (SSO), where a single token for the user is trusted across multiple IT systems. The same token can be used to authenticate the user against both the identity provider and the service provider (for this example, the cloud account with Identity Cloud Service and the traditional cloud account).

Get Information from the Traditional Cloud Account

In this topic, you get the identity domain, domain name, metadata, and signing certificate from the traditional cloud account. You need this information to set up an Oracle Cloud application in Oracle Identity Cloud Service so that provisioning and synchronization can occur for users between traditional cloud accounts and cloud accounts with Identity Cloud Service.

1. Sign in to the Oracle Cloud Infrastructure Classic Console of the traditional cloud account that contains the identity domain, domain name, metadata, and signing certificate that you want to get.
2. Expand the **Navigation Drawer**  in the top left corner, and then click **Users**.
3. Navigate to the **SSO Configuration** tab.
4. In the **Configure your Identity Provider Information** pane:
    a. Find the value associated with the **Provider id** field.
    b. Copy the identity domain and the domain name to a text editor, such as Notepad or TextPad.

> **Note:**
>
> If the value of the **Provider Id** field is `https://login.dc.migrationsample.<YOUR-DOMAIN-NAME>.com:443/oam/fed/cloud/migration_id_domain`, then the domain name is `dc.migrationsample.<YOUR-DOMAIN-NAME>.com` and the identity domain is `migration_id_domain`.

c. Click **Export Metadata**, and then select **Provider Metadata (SAML 2.0)** from the menu that appears.

d. Click **Export Metadata** again, and then select **Signing Certificate** from the menu that appears.

Get the entityID Attribute Value from the Metadata File

In this topic, you get the `entityID` attribute value from the metadata file that you exported. You need this information to set up the Oracle Cloud application in Oracle Identity Cloud Service.

1. Open the metadata file that you exported in Get Information from the Traditional Cloud Account.
2. Locate the `entityID` attribute in this file.
3. Copy the value associated with this attribute to a text editor, such as Notepad or TextPad.

Create an Oracle Cloud Application in Oracle Identity Cloud Service

In this topic, you create an Oracle Cloud application in Oracle Identity Cloud Service that's used to provision and synchronize users between traditional cloud accounts and cloud accounts with Identity Cloud Service.

Rather than build this application from scratch, use the App Catalog to create this application. The App Catalog contains pre-configured application templates. Using the templates, you can define the application, configure SSO, and configure provisioning and synchronization for the application.

1. Sign in to the Oracle Cloud Infrastructure Classic Console of the cloud account with Identity Cloud Service.
2. Expand the **Navigation Drawer**  in the top left corner, and then click **Users**.
3. In the **User Management** page, click **Identity Console**. The Identity Cloud Service console appears.
4. Expand the **Navigation Drawer**, and then click **Applications**.
5. Click **Add**.
6. In the **Add Application** window, click **App Catalog**.
7. In the **Type of Integration** area of the **App Catalog** page, click **Provisioning**.
8. In the search field, enter `Oracle Cloud`. The Oracle Cloud application appears.
9. Click **Add**.
10. Populate the **Identity Domain**, **Domain Name**, and **SSO Domain Name** fields of the **Details** tab with the values that you retrieved in Get Information from the Traditional Cloud Account, and then click **Next**.

> **Note:**
>
> Use the same value for both the **Domain Name** and **SSO Domain Name** fields.

11. Populate the **Entity ID** field of the **SSO Configuration** tab with the `entityID` attribute value that you retrieved in Get the entityID Attribute Value from the Metadata File.
12. Click **Upload** to the right of the **Signing Certificate** field, and then import the signing certificate that you exported in Get Information from the Traditional Cloud Account.
13. Click **Download Signing Certificate** to import the Oracle Identity Cloud Service signing certificate into the traditional cloud account.
14. Click **Download Identity Provider Metadata** to import the Oracle Identity Cloud Service identity provider metadata into the traditional cloud account. The traditional cloud account needs this information so that it can trust and process the assertion that is generated by Oracle Identity Cloud Service as part of the federation process. This information includes, for example, profile and binding support, connection endpoints, and certificate information.
15. Click **Next**.
16. In the **Provisioning** tab, click **Continue** in the **Grant Consent** window that appears.
17. Turn on the **Enable Provisioning** switch.
18. In the **Configure Connectivity** pane, configure connectivity for your application by providing values in the respective fields and by testing connectivity.
19. Turn on the **Enable Synchronization** switch.
20. In the **Configure Synchronization** section, modify the attributes for your application.
21. Click **Add**.

Import the Identity Provider Metadata into the Traditional Cloud Account

In this topic, you import metadata from Oracle Identity Cloud Service (the identity provider) into the traditional cloud account. The account needs this data so that provisioning and synchronization can occur for a user between the traditional cloud account and a cloud account with Identity Cloud Service.

1. In the traditional cloud account, click the **Users** menu, and then navigate to the **SSO Configuration** tab.
2. In the **Configure SSO** pane, click **Edit**.
3. In the **Edit Single Sign-On Configuration** window, select the **Import identity provider metadata** option, and then click **Choose File** to the right of the **Load Provider Metadata** field.
4. Import the Oracle Identity Cloud Service identity provider metadata that you downloaded in Create an Oracle Cloud Application in Oracle Identity Cloud Service.
5. Select the **Enter identity provider metadata manually** option, and then click **Choose File** to the right of the **Load Signing Certificate** field.
6. Import the Oracle Identity Cloud Service signing certificate that you downloaded in Create an Oracle Cloud Application in Oracle Identity Cloud Service.
7. Click **Save**.

Configure Single Sign-On for the Traditional Cloud Account

In this topic, you configure single sign-on for the traditional cloud account. As a result, a user can use their federated SSO credentials to log in to the traditional cloud account through their cloud account with Identity Cloud Service.

1. In the **SSO Configuration** tab, click **Enable SSO** in the **Enable SSO** pane.
2. In the **Enable Sign In to Oracle Cloud Services with Identity Domain credentials** pane, click **Enable**.

Provision a User with a Traditional Cloud Account

In this topic, you use Oracle Identity Cloud Service to create a cloud account with Identity Cloud Service for a user. Oracle Identity Cloud Service will provision the user with a traditional cloud account automatically.

1. In the Identity Cloud Service console, open the Oracle Cloud application.
2. Click the **Groups** tab, and then click **Assign**.
3. In the **Assign Groups** window, assign the **All Tenant Users** group to this application.

> **Note:**
>
> The **All Tenant Users** group is a default group that's created by Oracle Identity Cloud Service. All Oracle Identity Cloud Service users are assigned to this group, by default. By assigning this group to the Oracle Cloud application, all users are assigned to this application indirectly.

4. Click **Users**, and then click **Add**.
5. In the **First Name** and **Last Name** fields of the **Add User** window, enter the user's first and last name.
6. In the **User Name / Email** field, enter the user's email address.
7. Click **Next** (because you want to assign the user to the **All Tenant Users** group).
8. Select the check box for the **All Tenant Users** group, and then click **Finish**.

> **Note:**
>
> Because the user is assigned to the **All Tenant Users** group, Oracle Identity Cloud Service will provision the user with a traditional cloud account automatically.

See Verifying the Integration to confirm the integration between the traditional cloud account as the service provider and Oracle Identity Cloud Service as the identity provider.

Map Between Traditional Cloud Roles and Application Roles in Oracle Identity Cloud Service

The following table lists the services in traditional cloud accounts that contain roles you want to export. It also lists the services for cloud accounts with Identity Cloud Service and the application roles to which you want to assign users.

Service in a Traditional Cloud Account	Role in a Traditional Cloud Account	Service in a Cloud Account with Identity Cloud Service	Application Role in a Cloud Account with Identity Cloud Service
Oracle Developer Cloud Service (Traditional)	DEVELOPER_USER DEVELOPER_ADMINISTRATOR	Oracle Developer Cloud Service	DEVELOPER_USER DEVELOPER_ADMINISTRATOR
Oracle Integration Cloud Service	Oracle Integration Cloud Administrator Oracle Integration Cloud Service Runtime Oracle Integration Cloud Service Developer Oracle Integration Cloud Service Monitor Oracle Integration Cloud Service Agent Role	Oracle Autonomous Integration Cloud	ServiceAdministrator ServiceUser ServiceDeveloper ServiceMonitor ServiceDeployer
Oracle Mobile Cloud Service	MobileEnvironment_TeamMgmt MobileEnvironment_System MobileEnvironment_DbMgmt MobileEnvironment_LocationMgmt MobileEnvironment_Develop MobilePortal_TeamMember MobileEnvironment_Notifications MobileEnvironment_MobileUserMgmt MobileEnvironment_MobileUserConfig MobileEnvironment_Monitor MobileEnvironment_Deploy MobileEnvironment_MAXApplicationDeploy MobileEnvironment_Analytics MobileEnvironment_BusinessUser	Oracle Autonomous Mobile Cloud Enterprise	ServiceAdministrator ServiceAdministrator ServiceDeveloper ServiceDeveloper ServiceDeveloper ServiceDeveloper ServiceDeveloper ServiceDeveloper ServiceDeployer ServiceDeployer ServiceDeployer ServiceDeployer ServiceAnalyst ServiceBusinessUser

Service in a Traditional Cloud Account	Role in a Traditional Cloud Account	Service in a Cloud Account with Identity Cloud Service	Application Role in a Cloud Account with Identity Cloud Service
Oracle Process Cloud Service	ProcessServiceAdministrator ProcessServiceDeveloper ProcessServiceUser	Oracle Autonomous Integration Cloud	ServiceAdministrator ServiceDeveloper ServiceUser
Oracle Visual Builder Cloud Service	Application Builder Developer Application Builder Administrator	Oracle Visual Builder Cloud Service	ServiceDeveloper ServiceAdministrator

Migrate Service-Specific Data and Artifacts

You can migrate data and artifacts for the following cloud services:

- Oracle Developer Cloud Service: Migrate content from Oracle Developer Cloud Service (Traditional) to Oracle Cloud Infrastructure.
- Oracle Business Intelligence Cloud Service: Migrate content from Oracle Business Intelligence Cloud Service to Oracle Analytics Cloud.
- Oracle Mobile Cloud Service: Migrate content from Oracle Mobile Cloud Service to Oracle Mobile Hub.
- Oracle Process Cloud Service: Migrate content from Oracle Process Cloud Service to Oracle Autonomous Integration Cloud.
- Oracle Visual Builder Cloud Service: Migrate Oracle Visual Builder Traditional Cloud Account instances to Oracle Cloud Infrastructure.

Part VII

Manage Oracle Identity Cloud Service Components

Learn how to use additional Oracle Identity Cloud Service components.

Topics:

42

Manage Linux Authentication using the Linux-PAM Module

This scenario describes how you can use the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) to integrate your Linux environment with Oracle Identity Cloud Service to perform end user authentication with first and second factor authentication.

Topics:

Typical Workflow for Managing the Linux-PAM

With the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) , you can install and configure the module to allow end users to authenticate in Linux environments with Oracle Identity Cloud Service, using first and second factor authentication.

Task	Description	Additional Information
Understand the PAM.	You can receive an overview of the Oracle Identity Cloud Service Linux PAM.	About the Linux-PAM
Install and Configure the PAM.	You can download, install and configure the PAM.	Install and Configure the Linux-PAM
Configure PAM-enabled Groups and Users.	You can configure a PAM-enabled group and add users to the group using REST API's.	Configure Groups and Users for the Linux-PAM
Test Authentication into Linux using Oracle Identity Cloud Service.	You can authenticate on Linux using Oracle Identity Cloud Service first and second factor authentication.	Test Authentication into Linux Using Oracle Identity Cloud Service

About the Linux-PAM

The Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) allows you to integrate your Linux environment with Oracle Identity Cloud Service to perform end user authentication with first and second factor authentication.

Topics:

What is the Linux-PAM?

The Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) is an authentication module for Linux that performs end user authentication with Oracle Identity Cloud Service.

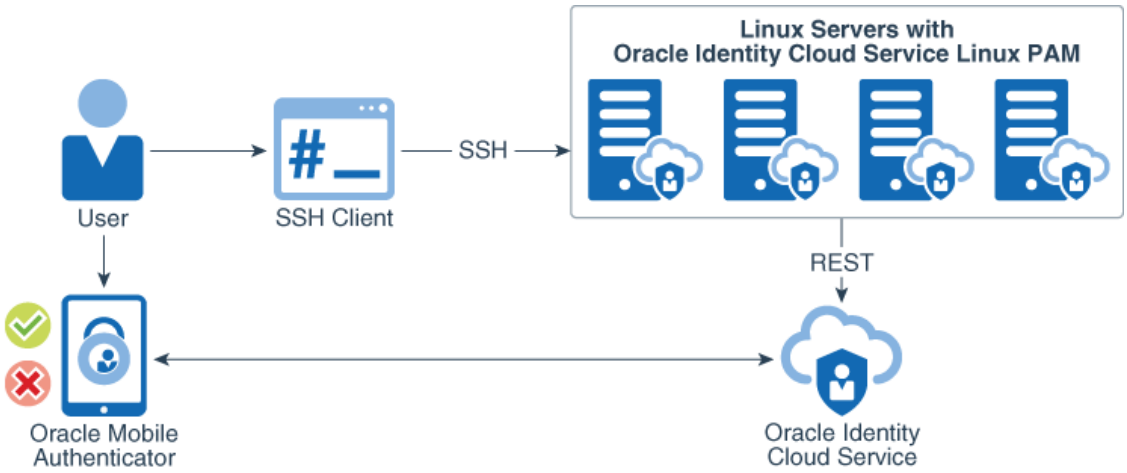
The PAM also allows Linux administrators, or end users, to query information about users and groups stored in Oracle Identity Cloud Service using standard Linux commands that utilize NSS such as `id`, `group`, and `getent`.

Why use the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM)

Use the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) when you want to authenticate users in Linux using Oracle Identity Cloud Service.

An organization may have large numbers of Linux servers, making management of users, for example creating, modifying, or deleting users, a time intensive and costly activity. With the Linux PAM you can manage Linux users centrally in Oracle Identity Cloud Service, providing cost and time savings.

Linux administrators can utilize Oracle Identity Cloud Service to authenticate end users. End users can log in to a Linux server, for example with SSH, and authenticate with their Oracle Identity Cloud Service user credentials. In addition, the multi-factor authentication offerings of Oracle Identity Cloud Service can be utilized so end users are prompted to authenticate with a second factor such as a One Time Password code sent via Email, SMS, a Mobile Authenticator application, or authenticate via security questions. As well as authenticating with single or multiple factors, administrators and end users can use NSS and standard Linux commands to query user and group information.



Certified Components

The following table lists the certified releases for Oracle Identity Cloud Service and your operating system (which is required for the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) to run).

> **Note:**
>
> Every PAM download includes all certified components.

Oracle Identity Cloud Service Release	64-Bit	Operating System
22.4.92 and earlier releases	Yes. (x86_64)	Oracle Enterprise Linux 6 Oracle Enterprise Linux 7 Oracle Enterprise Linux 8

Install and Configure the Linux-PAM

Learn how to download, install and configure the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM).

Topics:

Download the Linux-PAM

To download the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) see :

Download Oracle Identity Cloud Service SDKs and Applications

Install the Linux-PAM

To install the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) on your Linux environment, you install the PAM `rpm`s along with some dependencies:

1. Extract the downloaded zip file to a directory of your choice. This will extract the `pam_cloud.rpm` and `authn_oracle_cloud.rpm`.
2. Check the curl and json-c Linux dependencies are installed:
   - As the root user, run the following commands:
     - `yum list installed | grep curl.x86_64`
     - `yum list installed | grep json-c.x86_64`
   - If they are not installed, run the following commands:
     - `yum install json-c`
     - `yum install curl`
3. Change to the directory where you extracted the zip file:
   - `cd <folder_where_pam_oracle-cloud.pam_resides>`

4. Install the PAM rpm's as the root user.
   - If using yum:
     - `yum install pam_oracle-cloud.rpm authn-oracle-cloud.rpm`
   - If using rpm:
     - `rpm -Uvh pam_oracle-cloud.rpm authn-oracle-cloud.rpm`

   A successful installation will install the following files:
   - `/etc/opc.conf`
   - `/lib64/libnss_oracle_cloud.so.2`
   - `/lib64/security/pam_oracle_cloud.so`
   - `/lib64/libauthn_api.so`
   - `/lib64/libclntsh.so.12.1`
   - `/lib64/libclntshcore.so.12.1`
   - `/lib64/libipc1.so`
   - `/lib64/libmql1.so`
   - `/lib64/libnnz12.so`
   - `/lib64/libons.so`
   - `/usr/bin/walletMgr`

Configure a Confidential Application

To register the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) as a client application in Oracle Identity Cloud Service, you create a confidential application with the POSIX Viewer role.

1. In the Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. Click **Add**.
3. On the **Add Application** page, click **Confidential Application**.
4. In the **Add Confidential Application** wizard's **Details** page, in the **App Details Section**.
   a. Enter a **Name** for the application.
   b. Click **Next**.

   A confirmation message indicates that the application has been added in a deactivated state.
5. On the **Add Confidential Application** wizard's **Client** page, click **Configure this application as a client now**.
6. In the **Authorization** section that opens, select these two **Allowed Grant Types**:
   - **Client Credentials**
   - **JWT Assertion**
7. In the **Grant the client access to Identity Cloud Service Admin APIs** section at the bottom, click **Add**.
8. In the **Add App Role** dialog box, select these roles:

- **Me**
- **POSIX Viewer**
- **Signin**

9. Click **Add** to close the **Add App Role** dialog box.
10. At the top of the **Add Confidential Application** wizard's **Client** page, click **Next**.
11. Continue to click **Next** through the remaining wizard pages, then click **Finish**.
12. Record the **Client ID** and **Client Secret** that appear in the **Application Added** dialog box.

    To integrate with your confidential application, use this ID and secret as part of your connection settings. The **Client ID** and **Client Secret** are equivalent to a credential (for example, an ID and password) that your application uses to communicate with Oracle Identity Cloud Service.
13. Click **Close**.

    The new application's details page is displayed.
14. At the top of the page, to the right of the application name, click **Activate**.
15. In the **Activate Application?** dialog box, click **Activate Application**.

Create a Wallet

Configure a wallet on your Linux environment to store the client_id and client_secret of the confidential application with the POSIX Viewer role. This enables the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) to communicate securely with the confidential application.

- On the Linux environment, run the following commands as the root user:
  - `walletMgr add <wallet_location> client_id <client_id>`
  - `walletMgr add <wallet_location> client_secret <client_secret>`

  For example:

```
$ walletMgr add /etc/opc-wallet/ client_id b6d001f65da542c38ceb284ea8a05926

wallet initialized successfully.
key client_id is added successfully in wallet.


$ walletMgr add /etc/opc-wallet/ client_secret fea39433-5115-4050-
b486-138cce381fb2

wallet initialized successfully.
key client_secret is added successfully in wallet.
```

Configure the Linux-PAM

Configure the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) on your Linux environment.

The PAM is configured using **either** the SSSD **or** NSCD service on Linux.

> **Note:**
>
> The PAM can't be configured using both SSSD and NSCD simultaneously. Choose one configuration only. Choosing whether to use SSSD or NSCD is dependent on how your Linux environment is currently configured. Contact your Linux Administrator for details.

Topics:

Configure the Linux-PAM using SSSD

Configure the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) on Linux using the SSSD service.

> **Note:**
>
> The following prerequisites must be met before proceeding with the configuration.

- The SSSD service should be installed. If it is not installed, install via `sudo yum install sssd`.
- The service must be configured to start when the system reboots. You can perform this configuration via `sudo chkconfig sssd on`.
- The property `SELINUX` must be set as `permissive` or `disabled` in file `/etc/selinux/config`. If it is not set, then set `SELINUX=permissive` or `SELINUX=disabled`.
- Restart Linux to incorporate the above changes.

1. Verify the `/etc/sssd/sssd.conf` file exists, has 600 permission, and is owned by the root user. If the file does not exist create it as follows and run `chmod 600 /etc/sssd/sssd.conf`.

   `/etc/sssd/sssd.conf`

```
[sssd]
config_file_version = 2
services = nss, pam
domains = proxy_proxy
[nss]
fallback_homedir = /home/%u
default_shell = /bin/sh

[pam]
[domain/proxy_proxy]
auth_provider =  proxy
id_provider = proxy
proxy_lib_name = oracle_cloud
proxy_pam_target = sssd_proxy_oracle_cloud
enumerate =  false
cache_credentials = true
```

```
debug_level = 5
min_id = 500
```

Optionally, you can configure email addresses as the SSO usernames. To do this, add the line in bold (below) to the `/etc/sssd/sssd.conf` file to specify the regular expression.

```
...
[pam]
[domain/proxy_proxy]
re_expression = (?P<domain>[^\\]*?)\\?(?P<name>[^\\]+$)
auth_provider =  proxy
id_provider = proxy
...
```

2. Verify the `/etc/pam.d/sssd_proxy_oracle_cloud` file exists and is owned by the root user. If the file does not exist then create it as the root user and add the following:

   /etc/pam.d/sssd_proxy_oracle_cloud file

   ```
   auth          required      pam_oracle_cloud.so
   account       required      pam_oracle_cloud.so
   password      required      pam_oracle_cloud.so
   session       required      pam_oracle_cloud.so
   ```

3. Edit the `/etc/pam.d/sshd` and add the `pam_oracle_cloud` module:

   /etc/pam.d/sshd

   ```
   auth sufficient pam_oracle_cloud.so
   ```

   > **Note:**
   >
   > Add this either after the line `auth include password-auth`, or before the line `auth substack password-auth*`.

4. Edit the `/etc/ssh/sshd_config` to configure sshd to allow the use of Multi-Factor Authentication:

   /etc/ssh/sshd_config

   Search for the `ChallengeResponseAuthentication` property and set it to `yes`. If the property is not in the configuration file, add it.

5. Edit the `/etc/opc.conf` to allow the plugin to interact with Oracle Identity Cloud Service:

   /etc/opc.conf

   ```
   #This is sample format of opc.conf file, please use the correct
   information to configure this file.
   #Enter the Oracle Identity Cloud Service tenancy base url.
   base_url = https://identity-cloud-service-instance-url
   #There is no need to change value of scope.
   scope = urn:opc:idm:__myscopes__
   #Enter the location of the wallet.
   wallet_location = /etc/opc-wallet
   #Enter the log level, this is optional and the default is 0, which means
   ```

```
no log. 0 - None, 1 - Error, 2 - Info, 3 - Debug.
log_level = 0
#Enter the log file path, this is optional and defaults to /var/log/opc/
pam_nss.log
log_file_path = /var/log/opc/pam_nss.log
#Enter the value for proxy usage to connect to Oracle Identity Cloud
Service. Set the value to 1 to use a proxy and 0 to not use a proxy.
use_proxy=1
#Enter the information below if you set: use_proxy=1
#Enter the proxy url
proxy_url=http://proxy.example.com
#Enter the proxy port
proxy_port=80
#Enter the username to connect to the proxy url.
proxy_username=username_example
#Enter the password of username to connect proxy url.
proxy_pwd=pwd_example
```

6. Restart sssd and sshd:
   - `authconfig --enablesssd --enablesssdauth --enablemkhomedir --enablepamaccess --update`
   - `service sshd restart`
   - `service sssd restart`

Configure the Linux-PAM using NSCD

Configure the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) on Linux using the NSCD service.

> **Note:**
>
> The following prerequisites must be met before proceeding with the configuration.

- The NSCD service should be installed. If it is not installed, install via `sudo yum install nscd`.
- The service must be configured to start when the system reboots. You can perform this configuration via `sudo chkconfig nscd on`.
- The property `SELINUX` must be set as `permissive` or `disabled` in file `/etc/selinux/config`. If it is not set, then set `SELINUX=permissive` or `SELINUX=disabled`.
- Restart Linux to incorporate the above changes.

1. Edit the `/etc/nsswitch.conf` and add oracle_cloud as follows:

   `/etc/nsswitch.conf`

   ```
   passwd:     files oracle_cloud
   group:      files oracle_cloud
   ```

2. Edit the `/etc/nscd.conf` and enable caching in the nscd service:

/etc/nscd.conf

```
enable-cache        passwd      yes
enable-cache        group       yes
```

3. Edit the `/etc/pam.d/sshd` and add the pam_oracle_cloud module:

   /etc/pam.d/sshd

   ```
   auth sufficient pam_oracle_cloud.so
   #Note: the above has to be added before the following line:
   auth include password-auth
   ```

4. Edit the `/etc/ssh/sshd_config` to configure sshd to allow the use of Multi-Factor Authentication:

   /etc/ssh/sshd_config

   ```
   #Search for the ChallengeResponseAuthentication property and set it to yes
   ChallengeResponseAuthentication  yes
   ```

5. Edit the `/etc/opc.confg` to allow the plugin to interact with Oracle Identity Cloud Service:

   /etc/opc.conf

   ```
   #This is sample format of opc.conf file, please use the correct
   information to configure this file.
   #Enter the Oracle Identity Cloud Service tenancy base url.
   base_url = https://identity-cloud-service-instance-url
   #There is no need to change value of scope.
   scope = urn:opc:idm:__myscopes__
   #Enter the location of the wallet.
   wallet_location = /etc/opc-wallet
   #Enter the log level, this is optional and the default is 0, which means
   no log. 0 - None, 1 - Error, 2 - Info, 3 - Debug.
   log_level = 0
   #Enter the log file path, this is optional and defaults to /var/log/opc/
   pam_nss.log
   log_file_path = /var/log/opc/pam_nss.log
   #Enter the value for proxy usage to connect to Oracle Identity Cloud
   Service. Set the value to 1 to use a proxy and 0 to not use a proxy.
   use_proxy=1
   #Enter the information below if use_proxy=1
   #Enter the proxy url
   proxy_url=http://proxy.example.com
   #Enter the proxy port
   proxy_port=80
   #Enter the username to connect to the proxy url.
   proxy_username=username_example
   #Enter the password of username to connect proxy url.
   proxy_pwd=pwd_example
   ```

6. Restart sshd and nscd:

   - `authconfig --enablemkhomedir --enablepamaccess --update`
   - `service sshd restart`

- `service nscd restart`

Enforcing SELinux

Create a policy and ensure that PAM works when SELinux is set to enforcing:

Check that the following packages are installed on Oracle Linux:

```
rpm -q selinux-policy-targeted policycoreutils libselinux-utils libselinux-python
libselinux
```

> **Note:**
>
> When you change the SELinux mode from Permissive or Disabled to Enforcing, then you must reboot.

1. If necessary, install these packages on Oracle Linux:

   ```
   rpm -q selinux-policy-targeted policycoreutils libselinux-utils libselinux-
   python libselinux
   ```

2. Allow outbound communication on 443:

   ```
   $ sudo setsebool -P nis_enabled 1
   ++
   ```

3. Create a local policy so that `sssd_t` can create opc dir to create, and read and write to the `pam_nss.log` file (which is mentioned in `/etc/opc.conf`). It doesn't need to located in a specific location because it is compiled by the SELinux utilities.

   - Create the policy file:

     ```
     $cat my-sssdbe.te
     module my-sssdbe 1.0;
     require

     { type sssd_t; type var_log_t; type cert_t; type user_home_dir_t; class
     file
     { open read write }

     ;
     class dir { create write };
     } #============= sssd_t ============== #
     !!!! This avc is allowed in the current policy allow sssd_t cert_t:file
     write;
     allow sssd_t user_home_dir_t:dir write;
     allow sssd_t var_log_t:dir create;
     allow sssd_t var_log_t:file { open read };
     ```

4. Run:

   ```
   $ semodule -i my-sssdbe.pp
   ```

5. Run:

```
$ls my-sssdbe.pp my-sssdbe.te
```

6. Finally, authenticate the PAM user again.

   The `/opc` dir and `/opc/pam_nss.log` file are created.

Configure Groups and Users for the Linux-PAM

Learn how to create new groups and users with POSIX Viewer role attributes, or add POSIX Viewer role attributes to existing groups and users, to allow end users on Linux to authenticate with Oracle Identity Cloud Service using the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM).

Topics:

Obtain an Access Token

Obtain an admin access token to allow you to create groups and users with POSIX attributes, or add POSIX attributes to existing groups and users.

- In the Linux environment run the following command:

  ```
  curl -k -X POST -u "client-id:client-secret" -d
  "grant_type=client_credentials&scope=urn:opc:idm:__myscopes__" "https://
  identity-cloud-service-instance-url/oauth2/v1/token"
  ```

  where:

  - `client-id` is the client ID of a confidential application with administrative privileges
  - `client-secret` is the client secret of a confidential application with administrative privileges
  - `identity-cloud-service-instance-url` is your Oracle Identity Cloud Service Instance URL

> **Note:**
>
> The PAM confidential application `client-id` and `client-secret` are used by the PAM client library to create both groups or POSIX groups.
>
> However, to create a POSIX group, use the following endpoint with an admin access token.
>
> `/ui/v1/groups`

Create a Group with POSIX Attributes

Create a group with POSIX attributes.

1. Create a `group.json` file with the following request body:

   `group.json`

   ```
   { "schemas":
   [ "urn:ietf:params:scim:schemas:core:2.0:Group",
   "urn:ietf:params:scim:schemas:oracle:idcs:extension:group:Group",
   "urn:ietf:params:scim:schemas:oracle:idcs:extension:posix:Group" ],
   "displayName": "posix group",
   "urn:ietf:params:scim:schemas:oracle:idcs:extension:group:Group": {
   "description": "", "creationMechanism": "idcsui" },
   "urn:ietf:params:scim:schemas:oracle:idcs:extension:posix:Group": {
   "gidNumber": 11010 },
   "members": [] }
   ```

   where:

   - `displayName` is set to the name of the group you wish to create
   - `gidNumber` must be set to a unique group id (gid) number. Use the `getent group` command on Linux to see the existing group gid's.

2. Run the following curl command to create the group:

   ```
   curl -k -X POST -H "Content-Type: application/json" -H "Authorization:
   Bearer <token-string>" "https://identity-cloud-service-instance-url/
   admin/v1/Groups" -d '@group.json'
   ```

   where:

   - `token-string` is the OAuth access token that you obtained
   - `identity-cloud-service-instance-url` is your Oracle Identity Cloud Service Instance URL

   > **Note:**
   >
   > It is not possible to create a group with POSIX attributes using the Oracle Identity Cloud Service Administration Console.

Create a User with POSIX Attributes and Add to Group

Create a user with POSIX attributes and add the user to the group previously created.

1. Create a `user.json` file with the following request body:

   `user.json`

   ```
   {
   "password": "Securepasswd@1",
   "userName": "userPosix",
   ```

```
"Name.givenName": "user",
"Name.familyName": "Posix",
"userType": "Employee",
"emails": [
{
"value": "user.posix@example.com",
"type": "work",
"primary": true
},
{
"value": "posix@example.com",
"type": "home"
}
],
"addresses": [
{
"type": "work",
"primary": true,
"streetAddress": "401 Island Parkway",
"locality": "Redwood Shores",
"region": "California",
"postalCode": "94065",
"country": "US",
"formatted": "userPosix"
}
],
"urn:ietf:params:scim:schemas:oracle:idcs:extension:posix:User": {
"homeDirectory": "/home/userPosix",
"loginShell": "/bin/bash",
"gecos": "userPosix 24855",
"uidNumber": 12001,
"gidNumber": 11010
},
"meta": {
"resourceType": "User"
},
"schemas": [
"urn:ietf:params:scim:schemas:core:2.0:User",
"urn:ietf:params:scim:schemas:oracle:idcs:extension:posix:User"
]
}
```

where:

- `userName` is set to the username of the user you wish to create
- `homeDirectory` is set to the location of the user's home directory
- `loginShell` is set to the default shell
- `gecos` is set to general information about the user, for example the user's username and phone number
- `uidNumber` **must** be set to a unique user id (uid) number in Linux. Use the `getent passwd` command on Linux to see existing users and their uid's
- `gidNumber` **must** be set to the group id (gid) number created previously

2. Run the following curl command to create the user and add it to the group:

user.json

```
curl -k -X POST -H "Content-Type: application/json" -H "Authorization:
Bearer <token-string>" "https://identity-cloud-service-instance-url/
admin/v1/Users" -d '@user.json'
```

where:

- `token-string` is the OAuth access token that you obtained
- `identity-cloud-service-instance-url` is your Oracle Identity Cloud Service Instance URL

> **Note:**
>
> It is not possible to create a user with POSIX attributes using the Oracle Identity Cloud Service Administration Console.

Once the user is created, the user will be sent a notification email to activate their account and set a new password. The user must activate their account before testing authentication in Linux.

Add POSIX Attributes to Existing Groups

Add POSIX attributes to existing groups.

1. Create a `group_update.json` file with the following request body:

   group_update.json

   ```
   {
     "schemas": [
       "urn:ietf:params:scim:api:messages:2.0:PatchOp"
     ],
     "Operations": [
       {
         "op": "add",
         "path":
   "urn:ietf:params:scim:schemas:oracle:idcs:extension:posix:Group:gidNumber",
         "value": 11020
       }
     ]
   }
   ```

   where:

   - `gidNumber` **must** be set to a **unique** group id (gid) number. Use the `getent group` command on Linux to see the existing group gid's.

2. Run the following curl command to retrieve the group id's:

   ```
   curl -k -X GET -H "Content-Type: application/json" -H "Authorization:
   Bearer <token-string>" "https://identity-cloud-service-instance-url/
   admin/v1/Groups"
   ```

where:

- `token-string` is the OAuth access token that you obtained
- `identity-cloud-service-instance-url` is your Oracle Identity Cloud Service Instance URL

In the response, note the `id` of the group you want to update with POSIX attributes. For example, in the response below, the Marketing group `id` is `8c1f45fee6354e20aa9e57079082d6a2`:

```
.....
    {
      "displayName": "Marketing",
      "idcsLastModifiedBy": {
        "type": "User",
        "value": "f142a5ce639643c2befe8deb0ca5bcec",
        "display": "admin example",
        "$ref": "https://identity-cloud-service-instance-url/admin/v1/
Users/f142a5chjky3c2befe8deb0ca5bcec"
      },
      "idcsCreatedBy": {
        "type": "User",
        "display": "admin example",
        "value": "f142a5ce639643c2befe8deb0ca5bcec",
        "$ref": "https://identity-cloud-service-instance-url/admin/v1/
Users/f142a5chjky3c2befe8deb0ca5bcec"
      },
      "id": "8c1f45fee6354e20aa9e57079082d6a2",
      "meta": {
        "created": "2019-06-10T13:23:59.451Z",
        "lastModified": "2019-06-10T13:23:59.451Z",
        "resourceType": "Group",
        "location": "https://identity-cloud-service-instance-url/admin/v1/
Groups/8c1f45fee6354e20aa9e57079082d6a2"
      },
      "schemas": [
        "urn:ietf:params:scim:schemas:core:2.0:Group"
      ]
    },
    .....
```

3. Run the following curl command to update the group:

```
curl -k -X PATCH -H "Content-Type: application/json" -H "Authorization:
Bearer <token-string>" "https://identity-cloud-service-instance-url/
admin/v1/Groups/<id>" -d '@group_update.json'
```

where:

- `token-string` is the OAuth access token that you obtained
- `identity-cloud-service-instance-url` is your Oracle Identity Cloud Service Instance URL
- `id` is the id for the group you want to update with POSIX attributes

> **Note:**
>
> It is not possible to update a group with POSIX attributes using the Oracle Identity Cloud Service Administration Console.

Add POSIX Attributes to Existing Users

Add POSIX attributes to existing users.

> **Note:**
>
> In order to add POSIX attributes to an existing user, that user must first be part of a group, and that group must have POSIX attributes.

1. Create a `user_update.json` file with the following request body:

user_update.json

```
{
  "schemas": [
    "urn:ietf:params:scim:api:messages:2.0:PatchOp"
  ],
  "Operations": [
    {
      "op": "add",
      "path":
"urn:ietf:params:scim:schemas:oracle:idcs:extension:posix:User:homeDirector
y",
      "value": "/home/msmith"
    },
    {
      "op": "add",
      "path":
"urn:ietf:params:scim:schemas:oracle:idcs:extension:posix:User:gecos",
      "value": "msmith 25895"
    },
    {
      "op": "add",
      "path":
"urn:ietf:params:scim:schemas:oracle:idcs:extension:posix:User:uidNumber",
      "value": 12002
    },
    {
      "op": "add",
      "path":
"urn:ietf:params:scim:schemas:oracle:idcs:extension:posix:User:gidNumber",
      "value": 11020
    },
    {
      "op": "add",
      "path":
"urn:ietf:params:scim:schemas:oracle:idcs:extension:posix:User:loginShell",
```

```
      "value": "/bin/bash"
    }
  ]
}
```

where:

- `homeDirectory` is set to the location of the user's home directory
- `gecos` is set to general information about the user, for example the user's username and phone number
- `uidNumber` **must** be set to a unique user id (uid) number in Linux. Use the `getent passwd` command on Linux to see existing users and their uid's
- `gidNumber` **must** be set to the group id (gid) number updated previously
- `loginShell` is set to the default shell

2. Run the following curl command to retrieve the user id's:

```
curl -k -X GET -H "Content-Type: application/json" -H "Authorization:
Bearer <token-string>" "https://identity-cloud-service-instance-url/
admin/v1/Users"
```

where:

- `token-string` is the OAuth access token that you obtained
- `identity-cloud-service-instance-url` is your Oracle Identity Cloud Service Instance URL

In the response, note the `id` of the user you want to update with POSIX attributes. For example, in the response below, the msmith user `id` is `e5438fce80374d539b8638c289036ecd`:

```
....
{
  "idcsCreatedBy": {
        "type": "User",
        "display": "admin example",
        "value": "f142a5ce639643c2befe8deb0ca5bcec",
        "$ref": "https://identity-cloud-service-instance-url/admin/v1/
Users/f142a5chjky3c2befe8deb0ca5bcec"
      },
      "id": "e5438fce80374d539b8638c289036ecd",
      "meta": {
        "created": "2019-06-10T13:24:38.184Z",
        "lastModified": "2019-06-10T13:28:50.096Z",
        "resourceType": "User",
        "location": "https://identity-cloud-service-instance-url/admin/v1/
Users/e5438fce80374d539b8638c289036ecd"
      },
      "active": true,
      "displayName": "Mark Smith",
...
```

3. Run the following curl command to update the user:

```
curl -k -X PATCH -H "Content-Type: application/json" -H "Authorization:
Bearer <token-string>" "https://identity-cloud-service-instance-url/
admin/v1/Users/<id>" -d '@user_update.json'
```

where:

- `token-string` is the OAuth access token that you obtained
- `identity-cloud-service-instance-url` is your Oracle Identity Cloud Service Instance URL
- `id` is the id for the user you want to update with POSIX attributes

> **Note:**
>
> It is not possible to update a user with POSIX attributes using the Oracle Identity Cloud Service Administration Console.

Verify Endpoints

Verify that you can view users and groups and their POSIX attributes.

1. Obtain a POSIX access token by running the following curl command:

```
curl -k -X POST -u "client-id:client-secret" -d
"grant_type=client_credentials&scope=urn:opc:idm:__myscopes__" "https://
identity-cloud-service-instance-url/oauth2/v1/token"
```

where:

- `client-id` is the client ID for the POSIX confidential application
- `client-secret` is the client secret for the POSIX confidential application
- `identity-cloud-service-instance-url` is your Oracle Identity Cloud Service Instance URL

2. Run the following curl command to view users with POSIX attributes:

```
curl -k -X GET -H "Authorization: Bearer <token-string>" "https://identity-
cloud-service-instance-url/admin/v1/Users"
```

where:

- `token-string` is the OAuth POSIX access token that you obtained
- `identity-cloud-service-instance-url` is your Oracle Identity Cloud Service Instance URL

An example response is as follows:

`GET HOST/admin/v1/Users`

```
{
  "schemas": [
```

```
    "urn:ietf:params:scim:api:messages:2.0:ListResponse"
  ],
  "totalResults": 3,
  "Resources": [
    {
      "id": "af79f523f0f8416fb4407ed80a3bdbcb",
      "userName": "userPosix",
      "urn:ietf:params:scim:schemas:oracle:idcs:extension:posix:User": {
        "homeDirectory": "/home/userPosix",
        "loginShell": "/bin/bash",
        "gidNumber": 12001,
        "gecos": "userPosix 24855",
        "uidNumber": 11010
      }
    },
    {
      "id": "e5438fce80374d539b8638c289036ecd",
      "userName": "msmith",
      "urn:ietf:params:scim:schemas:oracle:idcs:extension:posix:User": {
        "homeDirectory": "/home/msmith",
        "loginShell": "/bin/bash",
        "gidNumber": 11020,
        "gecos": "msmith 25895",
        "uidNumber": 12002
      }
    },
    {
      "id": "f142a5ce639643c2befe8deb0ca5bcec",
      "userName": "admin@example.com"
    }
  ],
  "startIndex": 1,
  "itemsPerPage": 50
}
```


3. Run the following curl command to view groups with POSIX attributes:

```
curl -k -X GET -H "Authorization: Bearer <token-string>" "https://identity-cloud-service-instance-url/admin/v1/Groups"
```

where:

- `token-string` is the OAuth POSIX access token that you obtained
- `identity-cloud-service-instance-url` is your Oracle Identity Cloud Service URL

An example response is as follows:

`GET HOST/admin/v1/Groups`


```
{
  "schemas": [
    "urn:ietf:params:scim:api:messages:2.0:ListResponse"
  ],
  "totalResults": 3,
  "Resources": [
```

```
    {
      "displayName": "posix group",
      "id": "afb20ea78e84421aaba7009adf212ecf",
      "urn:ietf:params:scim:schemas:oracle:idcs:extension:posix:Group": {
        "gidNumber": 11010
      },
      "members": [
        {
          "value": "af79f523f0f8416fb4407ed80a3bdbcb",
          "type": "User",
          "display": "user Posix",
          "name": "userPosix",
          "$ref": "https://identity-cloud-service-instance-url/admin/v1/
Users/af79f523f0f8416fb4407ed80a3bdbcb"
        }
      ]
    },
    {
      "displayName": "Marketing",
      "id": "8c1f45fee6354e20aa9e57079082d6a2",
      "urn:ietf:params:scim:schemas:oracle:idcs:extension:posix:Group": {
        "gidNumber": 11020
      },
      "members": [
        {
          "value": "e5438fce80374d539b8638c289036ecd",
          "type": "User",
          "display": "Mark Smith",
          "name": "msmith",
          "$ref": "https://identity-cloud-service-instance-url/admin/v1/
Users/e5438fce80374d539b8638c289036ecd"
        }
      ]
    },
    {
      "displayName": "All Tenant Users",
      "id": "AllUsersId"
    }
  ],
  "startIndex": 1,
  "itemsPerPage": 50
}
```


Test Authentication into Linux Using Oracle Identity Cloud Service

Test authentication on Linux using a user in Oracle Identity Cloud Service.

1. SSH into your Linux environment where the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) is installed.
2. When prompted enter the password for the Oracle Identity Cloud Service user:

For example:

```
# ssh userPosix@host.example.com
password:

Last login: Thur Mar 28th 12:14:04 2019 from host.example.com
[userPosix@host ~]$
```

You should be logged in successfully.

Enable Multi-Factor Authentication to Authenticate into Linux

Learn how to set up Multi-Factor Authentication (MFA) so Linux users can authenticate via multiple factors.

1. Enable the MFA factors for your requirements. See Configure Multi-Factor Authentication Settings and Configure Authentication Factors
2. Create a group for MFA, and add the POSIX Users to this group.
   a. Navigate to **Groups** > **Add.**
   b. Enter the **Name** of the group and click **Next**.
   c. Search for the POSIX users you want to enable for MFA.
   d. Select the users and click **Finish**.
3. Create a Sign-On rule.
   a. Navigate to **Security** > **Sign-On Policies** and click **Default Sign-On Policy**.
   b. Click **Sign-On Rules** and then **Add**.
   c. Enter a **Rule Name**, and under **Conditions** in the field **And is a member of these groups** type and select the group that you created above. Under **Actions** make sure **Access** is set to **Allowed** and check the **Prompt for an additional factor** checkbox. Change the **Enrollment** to **Optional** and click **Save**.

   > **Note:**
   >
   > At present the only sign on policy that the Oracle Identity Cloud Service Linux Pluggable Authentication Module (PAM) supports, is the **Default Sign-On Policy**.

4. Move the newly created sign-on rule to the top by clicking on the sign-on rule and dragging it to the top of the list. Click **Save**. This will ensure that this rule gets evaluated first so that users belonging to the chosen group are prompted for MFA when they sign in.
5. Login to Oracle Identity Cloud Service as a user in the MFA Group, for example via `https://identity-cloud-service-instance-url/ui/v1/myconsole`
6. Enroll the user in MFA and select the factors to enroll in.

> **Note:**
>
> Backup factors are not currently supported with the Oracle Identity Cloud Service Linux PAM .

7. Once the user is enrolled in MFA, test authentication on Linux:
   a. SSH into your Linux environment where the Oracle Identity Cloud Service Linux PAM is installed.
   b. When prompted enter the password for the Oracle Identity Cloud Service user.
   c. Enter the second factor with which to authenticate.

   For example, for a user who has configured SMS as their second factor:

```
# ssh userPosix@host.example.com
password:
Complete 2-Step Verification

An SMS that contains a passcode was sent to +1XXXXXXX455. Enter the
passcode or use the following option, and then press Enter:
r - Resend passcode
Enter the passcode or an option (r):

Last login: Thu Mar 28 16:18:52 2019 from localhost
[userPosix@host ~]$
```

43

Use the E-Business Suite Asserter to Enable SSO for Oracle E-Business Suite with Oracle Identity Cloud Service

Use the Identity Cloud Service E-Business Suite Asserter component from Oracle Identity Cloud Service to integrate your Oracle E-Business Suite environment with other cloud and non-cloud services using Oracle Identity Cloud Service Single Sign-On (SSO).

Topics:

Typical Workflow for Using Identity Cloud Service E-Business Suite Asserter to Authenticate Oracle E-Business Suite with Oracle Identity Cloud Service

With the Identity Cloud Service E-Business Suite Asserter component that you download from the Identity Cloud Service console, you integrate your Oracle E-Business Suite with Oracle Identity Cloud Service to allow end users to authenticate in Oracle E-Business Suite environments and to Oracle E-Business Suite mobile applications using their Oracle Identity Cloud Service credentials.

Task	Description	Additional Information
Understand the Identity Cloud Service E-Business Suite Asserter	Learn what Identity Cloud Service E-Business Suite Asserter is, why you should use it to integrate your Oracle E-Business Suite environment with Oracle Identity Cloud Service, and the certified components of the architecture.	What is Identity Cloud Service E-Business Suite Asserter

Task	Description	Additional Information
What do You Need to Use the Asserter	Understand the required services and roles, how to download the asserter, and the information you need from your environment.	What do You Need to Use the E-Business Suite Asserter
Configure the Integration	Configure Oracle E-Business Suite, register E-Business Suite Asserter in Oracle Identity Cloud Service, and deploy the asserter.	Configure E-Business Suite Asserter Integration
Validate the Integration	Test the single sign-on scenarios.	Validate the Integration
Set up E-Business Suite Mobile Applications	Integrate E-Business Suite mobile applications with Oracle Identity Cloud Service for single sign-on purposes.	Set up E-Business Suite Mobile Applications
Collect Diagnostic Data	Enable and collect diagnostic data from E-Business Suite Asserter.	Collect Diagnostic Data
Monitor the E-Business Suite Asserter	Monitor the E-Business Suite Asserter to determine the status and in turn its availability.	Monitor the E-Business Suite Asserter
Deploy the Oracle App Gateway Docker Container	Deploy the Oracle App Gateway Docker container.	Deploy the Oracle App Gateway Docker Container
Troubleshoot Common Issues	List of common issues found during the configuration of this integration.	Troubleshoot Common Issues

What is Identity Cloud Service E-Business Suite Asserter

The E-Business Suite Asserter is a lightweight web application that enables single sign on (SSO) for E-Business Suite using IDCS. The asserter enables users to access E-Business Mobile Apps and the E-Business Suite Web interfaces. Users can also access other applications that are secured using Oracle Identity Cloud Service.

To enhance security for the sign-in process, you can set up sign-on and identity provider policies, and configure multi-factor authentication. You can also enable adaptive security to provide strong authentication capabilities and risk analysis for your users across applications and Oracle E-Business Suite in Oracle Identity Cloud Service.

Why You Should Use Identity Cloud Service E-Business Suite Asserter

The Identity Cloud Service E-Business Suite Asserter is a lightweight Java application. It helps to simplify the deployment topology for Oracle E-Business Suite single sign-on (SSO) by replacing Oracle Access Manager and Oracle Internet Directory with Oracle Identity Cloud Service.

You can use the asserter when you want to:

- Have your Oracle E-Business Suite integrated with other applications for single sign-on.
- Enhance security to access your Oracle E-Business Suite by enabling Oracle Identity Cloud Service security features such as multi-factor authentication, sign-on policies, account recovery, and adaptive security.

The E-Business Suite Asserter provides the following benefits:

- Multiple access modes for SSO with Oracle E-Business Suite. You can access Oracle E-Business Suite by using one of the following modes:
  - The asserter direct URL (You can bookmark this URL.).
  - The Oracle Identity Cloud Service **My Apps** page.
  - The asserter direct URL with a redirect parameter.
  - Previously bookmarked Oracle E-Business Suite URLs.
- Supports log out from multiple points including Oracle E-Business Suite, E-Business Suite Asserter, and Oracle Identity Cloud Service.
- Allows single sign-on between Oracle E-Business Suite and Oracle E-Business Suite mobile application.

Certified Components for Identity Cloud Service E-Business Suite Asserter

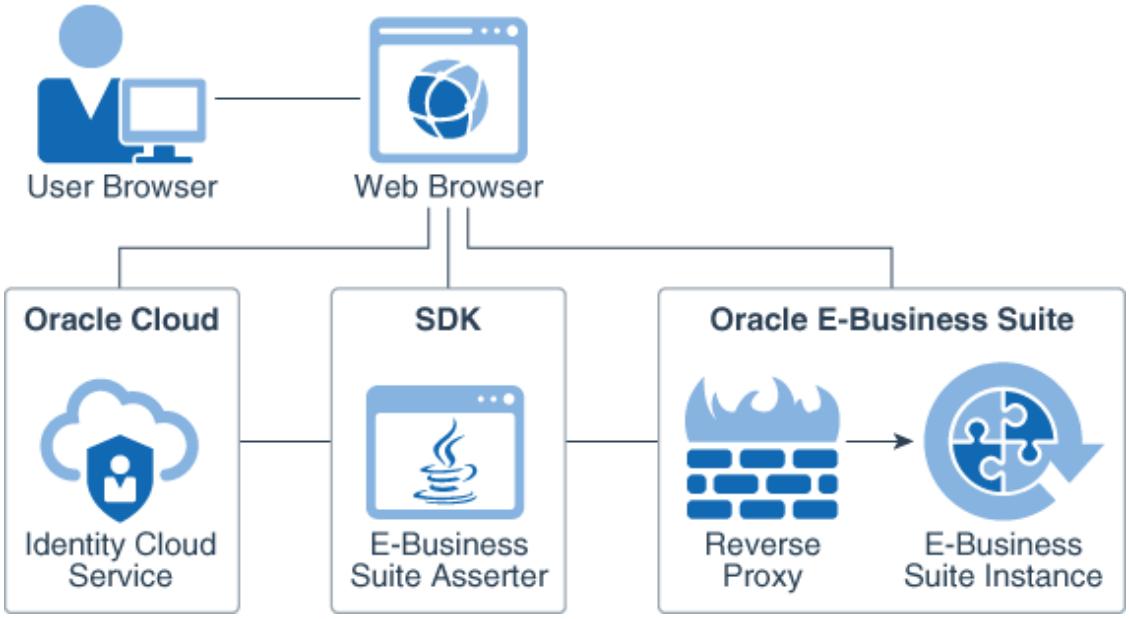
The following table lists the certified components and their versions for Oracle Identity Cloud Service, Oracle E-Business Suite, WebLogic Server, Java JDK, and the Identity Cloud Service E-Business Suite Asserter to use for integration.

Oracle Identity Cloud Service	Oracle E-Business Suite (EBS)	WebLogic Server	JDK	E-Business Suite Asserter
19.2.1+	The following versions with latest patches applied: • Oracle EBS Release 11i (11.5.10) • Oracle EBS Release 12 (12.1.3, 12.2 or greater), with latest patch applied.	Oracle WebLogic Server 12c (12.1.3 and 12.2) Oracle WebLogic Server 14c (14.1.1)	• Java SE Development Kit 8 • Java EE 8	19.1.4-1.2.2+

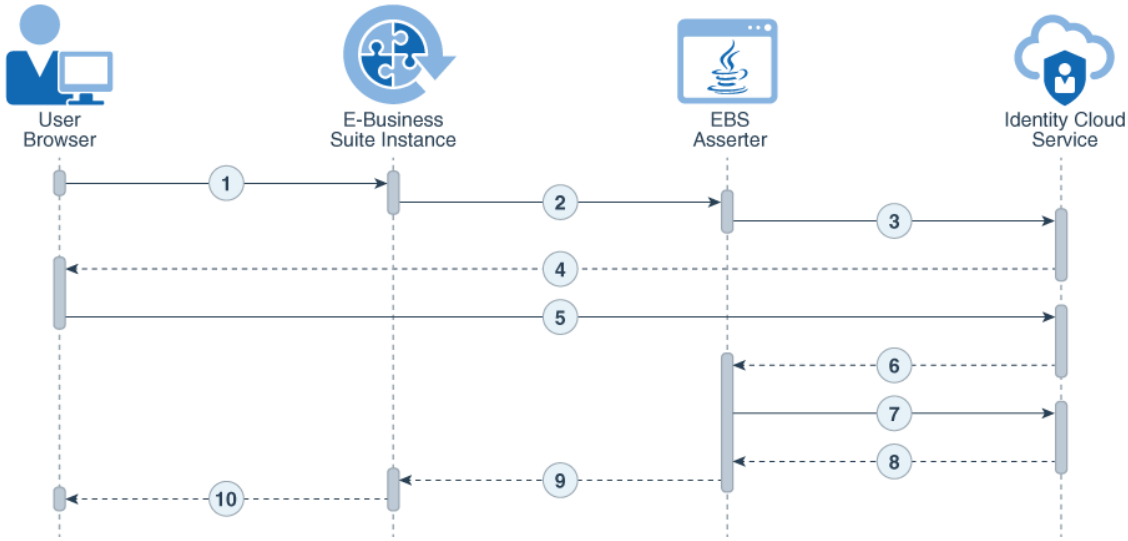
Architecture

The Identity Cloud Service E-Business Suite Asserter is deployed to a separate Oracle WebLogic Server instance. The E-Business Suite Asserter interacts with Oracle Identity Cloud Service through Oracle Identity Cloud Service REST API and redirects the user's web browser to Oracle Identity Cloud Service and to Oracle E-Business Suite.

This architectural diagram shows how the E-Business Suite Asserter, Oracle E-Business Suite, and Oracle Identity Cloud Service interact.



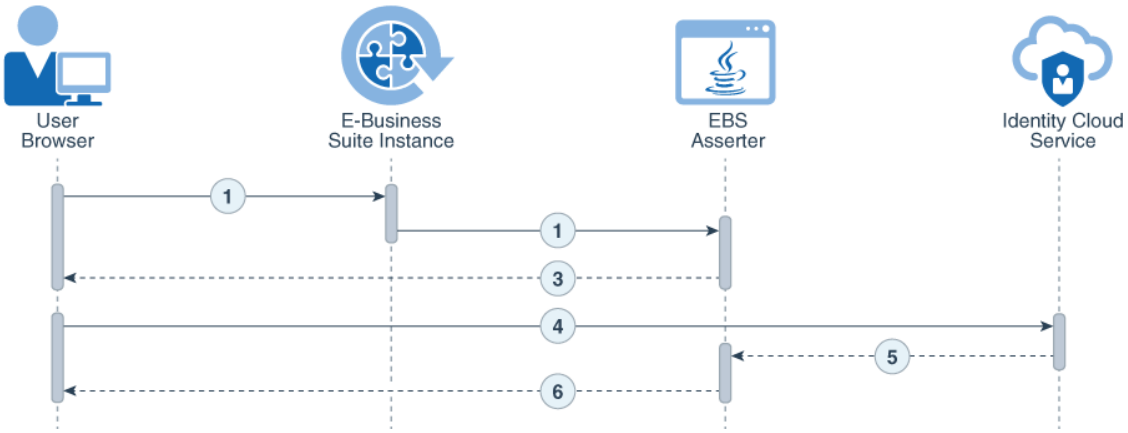
The following diagrams show the login and logout flow when using the E-Business Suite Asserter to integrate Oracle E-Business Suite with Oracle Identity Cloud Service. These flow diagrams show the login and logout process starting with Oracle E-Business Suite, but the E-Business Suite Asserter approach also supports E-Business Suite Asserter and Oracle Identity Cloud Service initiated flow.



1. The user requests access to an Oracle E-Business Suite protected resource.
2. Oracle E-Business Suite redirects the user browser to the E-Business Suite Asserter application.
3. The E-Business Suite Asserter uses an Oracle Identity Cloud Service SDK to generate the authorization URL and then redirects the browser to Oracle Identity Cloud Service.
4. Oracle Identity Cloud Service presents its sign in page to the user.
5. The user submits credentials to Oracle Identity Cloud Service.
6. Oracle Identity Cloud Service issues an authorization code and redirects the user's browser to the E-Business Suite Asserter.

7. The E-Business Suite Asserter uses an Oracle Identity Cloud Service SDK to communicate with Oracle Identity Cloud Service to exchange the authorization code for an access token.
8. Oracle Identity Cloud Service issues an access token and an ID token to the E-Business Suite Asserter.
9. The E-Business Suite Asserter creates an Oracle E-Business Suite cookie and redirects the user's browser to Oracle E-Business Suite.
10. Oracle E-Business Suite presents the user requested protected resource.

The logout process described below refers to a user invoking logout from Oracle E-Business Suite. If the logout process is initiated in Oracle Identity Cloud Service, then only step 5 and 6 are executed.



1. The user selects to logout from Oracle E-Business Suite, requesting the `/ebslogout` URL.
2. Oracle E-Business Suite logs the user out and then redirects the user's browser to the E-Business Suite Asserter application.
3. The E-Business Suite Asserter uses an Oracle Identity Cloud Service SDK to obtain the Oracle Identity Cloud Service logout URL, and then redirects the user's browser to this URL
4. The user browser invokes the Oracle Identity Cloud Service logout URL.
5. Oracle Identity Cloud Service removes the user session and then redirects the user's browser to the E-Business Suite Asserter logout URL, which is defined in the application configuration.
6. The E-Business Suite Asserter logs the user out and redirects the user's browser to the Post Logout Redirect URL, which is defined in the application configuration.

Considerations for Using the E-Business Suite Asserter

To use the E-Business Suite Asserter, you should understand the following considerations for installation and configuration.

- The host names for the EBS Asserter's WebLogic server and Oracle E-Business Suite's application server must have exactly same domain for SSO to work.
- The E-Business Suite Asserter must be accessed over SSL, since Oracle Identity Cloud Service can only be accessed over SSL. Failure to do so may cause SSO between Oracle Identity Cloud Service and the E-Business Suite Asserter to fail.
- Synchronize the server clock where the E-Business Suite Asserter runs, and the server clock where Oracle E-Business Suite runs.

- You can deploy the asserter in Oracle WebLogic Server 12c by using secure communications such as Secure Sockets Layer (SSL) and Transport Layer Security (TLS).

How to Use the Asserter With Multiple Instances of Oracle E-Business Suite

You can use the same WebLogic Server installation with multiple managed servers or from a different WebLogic Server installation, each with one managed server. In both case, each Identity Cloud Service E-Business Suite Asserter URL will have its own domain name and port number pair.

For each Oracle E-Business Suite (EBS) instance, you configure and deploy one instance of the E-Business Suite Asserter (EBS Asserter) Java application. Usually you deploy each EBS Asserter Java application to a specific WebLogic managed server.

Starting from EBS Asserter version 19.2.1-1.5.0, if you don't want to create multiple managed servers and deploy one EBS Asserter Java application to each of them, you can deploy multiple EBS Asserter Java applications to the same WebLogic managed server.

To accomplish this scenario, you need to perform the following tasks:

- Rename each EBS Asserter Java application's Web Application Resource (WAR) file before you deploy the file to the same WebLogic managed server. In this case, the domain name and port number of all EBS Asserter's URLs will be same, but the URL's context will change.
- Extract the contents of each `ebs.war` file to a folder, find the `weblogic.xml` file, edit this file, update the value of the `<cookie-path>` tag to match the EBS Asserter's URL, and then rebuild the `ebs.war`.

  For example, if you want EBS Asserter to respond to URL context `/app/ebs`, then the update the tag within `weblogic.xml` with the value `<cookie-path>/app/ebs</cookie-path>`.

For example: If you have two EBS instances named **Development 1** and **Development 2**, you want to integrate these EBS instances with Oracle Identity Cloud Service using the EBS Asserter, but you only have one WebLogic managed server for the two EBS Asserter Java applications, you need to execute the procedures in this tutorial for each EBS instance. You configure the WebLogic Server only once, and configure and deploy the EBS Asserter Java Application for each EBS instance:

- For EBS instance **Development 1**:
  - Make a copy of the `ebs.war` file and name the new file `ebsdev1.war`.
  - Udate the `weblogic.xml` file contained in the `ebsdev1.war` file, by replacing the `cookie-path` tag with the following: `<cookie-path>/ebsdev1</cookie-path>`.
  - Update the `bridge.properties` file contained in the `ebsdev1.war` file.
  - Deploy the `ebsdev1.war` file to the WebLogic managed server.
- For EBS instance **Development 2**:
  - Make a copy of the `ebs.war` file and name the new file `ebsdev2.war`.
  - Udate the `weblogic.xml` file contained in the `ebsdev2.war` file, by replacing the `cookie-path` tag with the following: `<cookie-path>/ebsdev2</cookie-path>`.
  - Update the `bridge.properties` file contained in the `ebsdev2.war` file.
  - Deploy the `ebsdev2.war` file to the WebLogic managed server.

You deploy both `ebsdev1.war` and `ebsdev2.war` files in to the same WebLogic managed server. The EBS Asserter's URL for EBS instance Development 1 will be similar to the following example: `https://ebsasserter.example.com:7002/ebsdev1`.

The EBS Asserter's URL for EBS instance Development 2 will be similar to the following example: `https://ebsasserter.example.com:7002/ebsdev2`.

What do You Need to Use the E-Business Suite Asserter

Verify which services, roles, components, and information are required to perform the configurations to integrate your Oracle E-Business Suite environment with Oracle Identity Cloud Service using the Identity Cloud Service E-Business Asserter.

Before You Begin

Before you begin using E-Business Suite Asserter, understand how to enable it, and how it works with other components.

- **Enable E-Business Suite Asserter.** This is Standard License feature. To learn about these features, see Standard License Tier Features for Oracle Identity Cloud Service.
- If your Oracle E-Business Suite is integrated with Oracle Access Manager, Oracle Internet Directory, E-Business Suite AccessGate, or uses any other SSO profile, then remove the integration between these components and Oracle E-Business Suite, and then restart the servers before using the Identity Cloud Service E-Business Suite Asserter.
- Know what's supported. All Oracle E-Business modules which use browser-based login will work with E-Business Suite Asserter for SSO. Excel-based login of Web ADI is supported. Mobile Apps for EBS, such as approvals and expenses, are supported. Modules which do not use browser-based login, such as Mobile Web Applications (MWA) and E-Signature, are not supported.

About Required Services and Roles

An Oracle Identity Cloud Service administrator must be able to access the Oracle Identity Cloud Service console to download E-Business Suite Asserter and configure and activate applications.

You must have access to the following services and products:

- Oracle Identity Cloud Service
- Oracle E-Business Suite

You must have the following roles:

Role	Required to...
Oracle Identity Cloud Service: Security administrator	Access the **Downloads** page of the Oracle Identity Cloud Service console. From this page, you can download the Identity Cloud Service E-Business Suite Asserter.
Oracle Identity Cloud Service: Application administrator	Manage applications in Oracle Identity Cloud Service, which includes registering the sample mobile app with Oracle Identity Cloud Service.
Oracle E-Business Suite: Server administrator	Access the Oracle E-Business Suite installation folder, the Oracle WebLogic Server where you deploy the E-Business Suite Asserter, and the E-Business Suite Asserter machine as an operating system user.

See *Learn how to get Oracle Cloud services for Oracle Solutions*.

Download the E-Business Suite Asserter from the Oracle Identity Cloud Service Console

You can access the Downloads page from the Oracle Identity Cloud Service console. From this page, you can download the Identity Cloud Service E-Business Suite Asserter.

1. In the Oracle Identity Cloud Service console, expand the **Navigation Drawer,** click **Settings,** and then click **Downloads.**
2. On the **Downloads** page, click the **Identity Cloud Service E-Business Suite Asserter** download button.

   For instances of Oracle EBS Release 20.1.3 onwards, the feature for EBS Asserter has to be enabled.

   - If the EBS Asserter is disabled and you try to use it you will see an error message.
   - Check the feature state by going to `tenant base url`/admin/v1/FeatureInfos. `oracle.idaas.ebs.asserter` should be marked as enabled.
   - If the feature is not enabled, contact Oracle Support and create a service request to enable `oracle.idaas.ebs.asserter`.
   - If `ebs.war` was already deployed on the WebLogic server before the EBS Asserter was enabled, then redeploy `ebs.war` after enabling the feature.
3. Save the `.zip` file to a temporary folder on your local machine (for example, `c:\temp` for Windows or `/temp` for Unix).
4. Extract the contents of the `.zip` file and find the location of the `ebs.war` and `idcs-wallet-<version>.jar` files.

   The name of the files may vary accordingly to the version.
5. Copy the `ebs.war` and `idcs-wallet-<version>.jar` files to a working folder into the E-Business Suite Asserter's WebLogic Server machine. For example, `/opt/ebssdk` (create this folder if it doesn't exist).

Provide Environment Information

Record the environment information that you'll need when you configure the E-Business Suite Asserter configuration file.

- Oracle WebLogic Server host name where the E-Business Suite Asserter is deployed. For example, `ebsasserter.example.com`
- Oracle WebLogic Server HTTPS address (including port number if not default one) where the E-Business Suite Asserter is deployed. For example, `https://ebsasserter.example.com:7002`
- Oracle E-Business Suite host name. For example, `ebs.example.com`
- Oracle E-Business Suite HTTPS address. For example, `https://ebs.example.com:8001/`
- Oracle Identity Cloud Service HTTPS address (including port number if not using the default one). For example, `https://idcs-example.identity.oraclecloud.com`
- Oracle E-Business Suite Database name. For example, `EBSDB`
- Oracle E-Business Suite Database host. For example, `ebs.example.com`

- Oracle E-Business Suite Database port. For example, `1521`
- Oracle E-Business Suite APPS user password. For example, `apps`

Configure E-Business Suite Asserter Integration

Set up Oracle E-Business Suite, register the asserter in Oracle Identity Cloud Service, and deploy the asserter.

Topics:

Create Users and Update the Administrator's Email in Oracle E-Business Suite

Create a user for the E-Business Suite Asserter to communicate with Oracle E-Business Suite, a user in Oracle Identity Cloud Service that correspond to the System Administrator in your Oracle E-Business Suite, and then update the email address of the `SYSADMIN` user in Oracle E-Business Suite.

Topics:

Create an Application User on Oracle E-Business Suite

You must create a specific application user that will be authorized to connect to the Oracle E-Business Suite database. The Apps Schema Connect role determines the authorization to connect to the Oracle E-Business Suite database. A user that has this role will be authorized to connect to the Oracle E-Business Suite database.

1. Log in to the Oracle E-Business Suite as an administrator. For example, `sysadmin`.
2. In the **Oracle E-Business Suite Home** page, scroll down the **Navigator**, expand **User Management**, and then click **Users**.
3. In the **User Management** page, select **User Account** from the **Register** drop-down menu, and then click **Go**.
4. In the **Create User Account** page, enter the following details to create a new user, and then click **Submit**.
   - **Username:** Provide a username
   - **Password:** Provide a password
   - **Description:** `E-Business Suite Asserter Service User`
   - **Password Expire:** `None`

Provide a temporary password for this user because the user needs to reset the password after first log in.

5. After the **A new user account has been created.** message appears, click **Assign Roles**, and then click **Assign Roles** in the **Update User** page.
6. In the **Search and Select: Assign Roles** window, search for **Code** `UMX|APPS_SCHEMA_CONNECT`.
7. Select **Apps Schema Connect Role**, and then click **Select**.
8. In the **Update User** page, enter the justification as `EBS Asserter Service User`, and then click **Save**.

   You can ignore the warning message regarding the Workflow Background Engine.
9. After the user is created, log off Oracle E-Business Suite application,and then log in using the username and password you provided in step 4 to reset the user password.

Create Oracle E-Business Suite's System Administrator in Oracle Identity Cloud Service

Create a user in Oracle Identity Cloud Service that correspond to the System Administrator in your Oracle E-Business Suite.

This user is necessary otherwise the system administrator won't be able to login to the Oracle E-Business Suite console after Oracle E-Business Suite is configured to use Oracle Identity Cloud Service for authentication.

1. Sign in to Oracle Identity Cloud Service to access the Identity Cloud Service console.
2. In the Identity Cloud Service console, expand the **Navigation Drawer**, click **Users**, and then click **Add** in the **Users** page.
3. In the **Add User** window, provide the following values, and then click **Finish**:
   - **First Name**: `EBS`
   - **Last Name**: `Sysadmin`
   - Uncheck `Use the email address as the username`.
   - **Username**: `sysadmin`
   - **Email**: If the SYSADMIN user in your Oracle E-Business Suite is configured with an email address, provide this email address. Otherwise, provide an email address and then update Oracle E-Business Suite's System Administrator email address with this same value.

Oracle Identity Cloud Service sends a **Welcome** notification email to the email you provided with the procedures to reset the password of the `sysadmin` user.

Update Oracle E-Business Suite's System Administrator Email Address

Update the email address of the `SYSADMIN` user in Oracle E-Business Suite to match the email address you provided to the corresponding user in Oracle Identity Cloud Service.

1. Login as administrator (for example, `sysadmin`) to the Oracle E-Business Suite application.
2. In the **Oracle E-Business Suite Home** page, scroll down to the **Navigator**, expand **User Management**, and then click **Users**.

3. In the **User Maintenance** page, search for the **Username** `SYSADMIN`, and click the update icon for the `SYSADMIN` user.
4. If the **Email** field hasn't been set, update this field value with the same email address you provided during the creation of the system administrator user in Oracle Identity Cloud Service, and then click **Apply**.
5. Close Oracle E-Business Suite application.

Configure the E-Business Suite Asserter in Oracle E-Business Suite

Register the E-Business Suite Asserter application server with Oracle E-Business Suite.

Register the E-Business Suite Asserter with Oracle E-Business Suite

To establish communication with Oracle E-Business Suite, the E-Business Suite Asserter uses the application server ID in the database connection file. The database connection file is generated while registering the E-Business Suite Asserter application server with Oracle E-Business Suite.

1. Log in to the Oracle E-Business Suite application server machine. Don't use the root user. Use the user that installed and ran the WebLogic Server.
2. Run the commands `echo $JAVA_HOME` and `echo $WL_HOME`, and then make note of the value that is set for each:
   - `JAVA_HOME`: `/usr/java/jdk1.7.0_201`
   - `WL_HOME`: `/u01/oracle/wlserver`

   If the values of the commands `$JAVA_HOME` and `$WL_HOME` aren't set, request that the WebLogic administrator set them. The `$WL_HOME` value is only needed if you use a version of Oracle E-Business Suite greater than 12.2.

   The values for the `$JAVA_HOME` and `$WL_HOME` may differ from your environment. Update the fields with the correct values for your environment.
3. Run the following command to create a working folder:

   ```
   cd /opt
   mkdir ebssdk
   cd ebssdk
   ```

4. Extract the `fndext.jar` file, which is located in the `WEB-INF/lib` folder inside the `ebs.war` file that you have downloaded from the Oracle Identity Cloud Service console.
5. Copy the `fndext.jar` file to the working folder you created in the previous step and also to the E-Business Suite Asserter WebLogic `$DOMAIN_HOME/lib` folder.

   The name of the `fndext.jar` file may vary depending on the current version.
6. Locate your Oracle E-Business Suite environment file (in this example, `/u01/install/VISION/EBSapps.env`) and run the following command:

   ```
   source /u01/install/VISION/EBSapps.env
   ```

   The path to the `.env` file may vary depending on your environment.
7. Locate the `.dbc` file that is associated with your Oracle E-Business Suite instance in the following folder: `$FND_SECURE/EBSDB.dbc`.

If your database instance name is `EBSDB`, the file should have a name like `EBSDB.dbc`. Make a note of the full path of the `.dbc` file (including the file name itself): `/u01/install/VISION/fs1/inst/apps/EBSDB_ebs/appl/fnd/12.0.0/secure/EBSDB.dbc`.

8. Run the following command to register the E-Business Suite Asserter application server with Oracle E-Business Suite:

```
cd /opt/ebssdk
java oracle.apps.fnd.security.AdminDesktop apps/apps CREATE
NODE_NAME=ebsasserter.example.com DBC=/u01/install/VISION/fs1/inst/apps/
EBSDB_ebs/appl/fnd/12.0.0/secure/EBSDB.dbc
```

9. Run the following command:

```
cat EBSDB_ebsasserter.example.com.dbc
```

The resulting file name may be in all uppercase letters. Make a note of the `APPL_SERVER_ID` value.

10. Copy the *`EBSDB_ebsasserter.example.com`*`.dbc` file to the EBS Asserter's WebLogic Server machine under the `/opt/ebssdk` folder. Create the folder if the folder doesn't exists.

Register and Activate the E-Business Suite Asserter in Oracle Identity Cloud Service

To establish communication with Oracle Identity Cloud Service, the E-Business Suite Asserter uses the client ID and the client secret of an Oracle Identity Cloud Service registered application.

1. In the Oracle Identity Cloud Service console, expand the **Navigation Drawer**, and then click **Applications**.
2. On the **Applications** page, click **Add**.
3. In the **Add Application** dialog box, click **Trusted Application**.
4. In the **Details** pane, enter the following information, and then click **Next**.
   - **Name:** `EBS Asserter`
   - **Description:** `E-Business Suite Asserter Application`
   - **Application URL:** `https://ebsasserter.example.com:7002/ebs`
   - **Display in My Apps:** Select this check box
5. In the **Client** pane, select **Configure this application as a client now**, and then enter or select the following values:
   - **Allowed Grant Types**: **Client Credentials** and **Authorization Code**
   - **Redirect URL**: `https://ebsasserter.example.com:7002/ebs/response`
   - **Logout URL**: `https://ebsasserter.example.com:7002/ebs/logout`
   - **Post Logout Redirect URL**: `https://ebs.example.com:8001/OA_HTML/OA.jsp?OAFunc=OANEWHOMEPAGE`
6. In the **Client** pane, scroll down, and click **Add** below **Grant the client access to Identity Cloud Service Admin APIs**.

7. In the **Add App Role** dialog box, select **Authenticator Client** and **Me** from the list, and then click **Add**.
8. In the **Client** pane, click **Next**.
9. Click **Next** until you reach the last pane, and then click **Finish**.
10. In the **Application Added** dialog box, make a note of the **Client ID** and **Client Secret** values, and then click **Close**.

    E-Business Suite Asserter needs these values to integrate with Oracle Identity Cloud Service.
11. Click **Activate**.
12. In the **Activate Application** dialog box, click **Activate Application**.

Configure and Deploy the E-Business Suite Asserter

After registering the E-Business Suite Asserter in Oracle Identity Cloud Service, you must configure and deploy the E-Business Suite Asserter that will act as an interface between an identity token issued by Oracle Identity Cloud Service and a user session created in Oracle E-Business Suite.

Create a Wallet for the E-Business Suite Asserter

For security purposes, the E-Business Suite Asserter component uses a wallet to register the client ID, client secret, and Oracle Identity Cloud Service URL as parameters.

1. Log in to the E-Business Suite Asserter application server machine, and navigate to the `/opt/ebssdk` folder.

   Make sure the user has enough privileges to perform the following actions.
2. Access the folder where the `idcs-wallet-<version>.jar` file is located.
3. Run the command `java -jar idcs-wallet-<version>.jar`, and then provide the following values when prompted:
   - `Enter Client ID`: Enter the client ID generated while registering and activating the E-Business Suite Asserter in Oracle Identity Cloud Service.
   - `Enter Client Secret`: Enter the client secret for the client ID.
   - `Enter IDCS base URL`: Enter Oracle Identity Cloud Service base URL. For example: `https://MYTENANT.identity.oraclecloud.com`.

   The command line creates a wallet file named `cwallet.sso` in the provided path.

Make note of the path of the `cwallet.sso` file.

Update the E-Business Suite Asserter Configuration File

After you register the Identity Cloud Service E-Business Suite Asserter (EBS Asserter), you can configure the asserter configuration file to connect with Oracle Identity Cloud Service during authentication.

Starting from Identity Cloud Service E-Business Suite Asserter version 19.1.4-1.4.0 onward, the asserter contains a properties file called `bridge.properties`. This file is located inside the `ebs.war` file. You need to update the information in the `bridge.properties` file, and then regenerate the `ebs.war` file, before deploying it to a WebLogic Server.

> **Note:**
>
> For E-Business Suite Asserter versions before 19.1.4-1.4.0 release, the war file may not contain the `bridge.properties` file inside. You need to create this file in a folder of the E-Business Suite Asserter's WebLogic server, update its content as per step 3, save the file, and then set an environment variable, as per the following example:
> `export ebs_property_file="/opt/ebssdk/bridge.properties"`

1. In the server where you downloaded the E-Business Suite Asserter zip file, navigate to the location where you decompressed the `ebs.war` file.
2. Using a zip utility, decompress the `ebs.war` file, locate the `bridge.properties` file, and open the file for editing.
3. Uncomment the following properties by removing the # from the beginning of each line, and update their values as follows:

```
#################################################################
## SSO Bridge for E-Business Suite
#################################################################
# Properties File
app.url=https://ebsasserter.example.com:7002/ebs
app.serverid=APPL_SERVER_ID_value
ebs.url.homepage=https://ebs.example.com:8001/OA_HTML/OA.jsp?
OAFunc=OANEWHOMEPAGE
ebs.ds.name=visionDS
ebs.user.identifier=username
idcs.iss.url=https://identity.oraclecloud.com
idcs.aud.url=https://idcs-example.identity.oraclecloud.com
#post.logout.url=https://ebs.example.com:8001/OA_HTML/OA.jsp?
OAFunc=OANEWHOMEPAGE
wallet.path=[FULL_PATH_OF_THE_WALLET_FILE]
whitelist.urls=https://ebs.example.com:8001/OA_HTML/RF.jsp,https://
ebs.example.com:8001/OA_HTML/OA.jsp,https://ebs.example.com:8001/OA_HTML/
BneApplicationService,https://ebs.example.com:8001/OA_HTML/jsp/fnd/
close.jsp
ebs.renew.session=true
proxy.mode=true
proxy.home.url=https://ebs.example.com:8001/OA_HTML/RF.jsp?
function_id=1031198&resp_id=-1&resp_appl_id=0&security_group_id=0&lang_code
=US
#istore.pages=ibeCZzdMinisites.jsp,ibeCAcpSSOLoginR.jsp
#idcs.user.identifier=email/username
#################################################################
```

The following table provide the description for each `bridge.properties` parameter and optional parameters supported by each EBS Asserter version.

Parameter	Description	EBS Asserter Version
`app.url`	The URL and port number for the E-Business Suite Asserter application.	19.1.4 onward

Parameter	Description	EBS Asserter Version
`app.serverid`	Corresponds to the APPL_SERVER_ID value in the `.dbc` file generated while registering the E-Business Suite Asserter.	19.1.4 onward
`ebs.url.homepage`	The URL address for the Oracle E-Business Suite home page.	19.1.4 onward
`ebs.ds.name`	The data source name to be created in the Oracle WebLogic Server where the E-Business Suite Asserter is deployed.	19.1.4 onward
`ebs.user.identifier`	Oracle E-Business Suite field used to match the Oracle Identity Cloud Service username. Allowed values are username (representing the FND_USERS.USER_NAME column) or email (representing the FND_USERS.EMAIL_ADDRESS column). Ensure that the attribute chosen here has unique values in FND_USERS otherwise the login will fail.	19.1.4 onward
`idcs.iss.url`	Oracle Identity Cloud Service issuer URL. This value can be found in the Oracle Identity Cloud Service Discovery Doc endpoint. The default value is `https://identity.oraclecloud.com`. This value must match the Issuer value set in the Oracle Identity Cloud Service OAuth settings. See Configure OAuth Settings.	19.1.4 onward
`post.logout.url`	This is an optional parameter. Uncomment this parameter so that E-Business Asserter redirects to this URL after logging the user out from the Single Sign-On. This value must match the value of the **Post Logout Redirect URL** parameter in Oracle Identity Cloud Service.	19.1.4 onward
`wallet.path`	The full path of the wallet file, including the file name.	19.1.4 onward
`whitelist.urls`	Lists the URL E-Business Suite Asserter can accept as the `requestUrl` parameter value. If the `requestUrl` value doesn't match one of the `whitelist.urls` values, then the test scenario for **SSO Using the E-Business Suite Asserter Direct URL with a Redirect Parameter** will fail.	19.1.4 onward

Parameter	Description	EBS Asserter Version
`ebs.renew.session`	This is an optional parameter. Use this parameter to control how the E-Business Suite Asserter manages the Oracle E-Business Suite session when the Oracle E-Business Suite cookie has expired. If you add this parameter to the `bridge.properties` file, and set the value to `true`, then the asserter refreshes the Oracle E-Business Suite Forms session after having reach the configured limit (ICX:Session Timeout). If the parameter is set to `false`, then after reaching the configured limit, the Forms session is invalidated closing all active Forms, however the Oracle E-Business Suite session in the browser will be active, allowing the user to reopen a new Forms session.	19.2.1 onward
`proxy.mode`	This is an optional parameter. Add this parameter to the `bridge.properties` file, and set the value to `true` to enable Oracle E-Business Suite Proxy User feature. Users trying to log in as a proxy user, are redirected to the URL you provide in the `proxy.home.url` parameter.	19.3.3-1.7.0 onward
`proxy.home.url`	This attribute is mandatory if `proxy.mode=true`. After the user signs in to Oracle Identity Cloud Service, the EBS Asserter redirects the proxy user to this URL. Typically this URL is Oracle E-Business Suite's **Switch User** page. For example: `https://ebs.example.com:8001/OA_HTML/RF.jsp?function_id=1031198&resp_id=-1&resp_appl_id=0&security_group_id=0&lang_code=US`	19.3.3-1.7.0 onward

Parameter	Description	EBS Asserter Version
`istore.pages`	Lists the comma separated value of iStore pages E-Business Suite Asserter will accept. If the requestUrl matches one of the `istore.pages` values, then user will be redirected to the requested iStore page post login. Add the iStore pages to the existing list of `istore.pages`.	19.3.3-1912170009 onward
`idcs.user.identifier`	This is an optional parameter. The Oracle Identity Cloud Service user attribute used to match with `ebs.user.identifier`. Allowed values are username (representing the username attribute in Oracle Identity Cloud Service), email (representing the email attribute in Oracle Identity Cloud Service), custom attribute name (representing the custom attribute of a user in Oracle identity Cloud Service e.g: `employee_no`). If this value is not provided in `bridge.properties`, then it will be defaulted to the value of `ebs.user.identifier`. Ensure that there is one-to-one mapping between the `idcs.user.identifier` attribute in Oracle Identity Cloud Service to the `ebs.user.attribute` attribute in `FND_USERS` otherwise the login will fail. **Note:** Ensure that the custom attribute used in `idcs.user.identifier` is added to the user schema in IDCS. The custom attribute feature is available in EBS Asserter version 20.1.3 onwards.	19.3.3-1912170009 onward

Parameter	Description	EBS Asserter Version
base.lang	The Oracle Identify Could Service EBS Asserter supports the user's language configuration provided in EBS. If the FND_OVERRIDE_SSO_LANG profile option is enabled for a user in EBS, Asserter creates an EBS session based on the value of the ICX_LANGUAGE profile option of this user. If no language configuration is present for the users in EBS and the browser language needs to be overwritten across all the users of the application, the base.lang property can be set in the bridge.properties file. For example, if base.lang is set to US and user does not possess any language specific configuration in EBS, irrespective of the browser(with local languages) from which the user tries to login into EBS with Asserter, the EBS session is created in American English.**Note:** The base.lang configuration is relevant in case the EBS is enabled with multiple languages. If there's only one language enabled in EBS, the asserter creates the EBS session with the base installed language even without a base.lang configuration.	Oracle Identity Cloud Service release version

4. Rebuild the ebs.war file and make sure it contains the updated version of the bridge.properties file. The structure of the ebs.war file is as follows:

```
META-INF/
   MANIFEST.MF
WEB-INF/
   classes/
   lib/
   bridge.properties
   web.xml
   weblogic.xml
```

Configure Hostname Verification in WebLogic Console

You can configure the hostname verification in Oracle WebLogic Server Administration Console.

1. Start the Oracle WebLogic Server Administration Console by entering http://wls_host:wls_port/console in the URL line of a web browser. For example, https://ebsasserter.example.com:7002/console.

2. Log in to WebLogic console as an administrator.
3. In the left panel, click **Lock & Edit**, expand **Environment**, select **Servers**.
4. Click the name of the target server where you want to deploy the EBS Asserter. In this example, **AdminServer**.
5. Click the **SSL** tab. Scroll down and expand the **Advanced** section.
6. Update the **Hostname Verification** parameter with the value **None**, and then click **Save**.
7. Click **Activate Changes**.
8. Restart the servers.

Configure Keystores in WebLogic Console

If you are using Custom Trust Store in WebLogic for asserter deployment, instead of using **Custom Identity and Custom Trust Store** with WebLogic server, use **Custom Identity and Java Trust Store**. With this configuration, you do not need to import Oracle Identity Cloud Service certificate.

1. Start the Oracle WebLogic Server Administration Console by entering `http://wls_host:wls_port/console` in the URL line of a web browser. For example, `https://ebsasserter.example.com:7002/console`.
2. Log in to WebLogic console as an administrator.
3. In the left panel, click **Lock & Edit**, expand **Environment**, select **Servers**.
4. Click the name of the target server where you want to configure the keystore.
5. Click **Keystores** under the **Configuration** tab.
6. In the left panel, click **Lock & Edit** to make the changes.
7. Select **Custom Identity and Java Trust Store**.
8. Click **Save** and **Activate Changes**.
9. Restart the WebLogic server.

Define the Data Source

In the Oracle WebLogic Server where E-Business Suite Asserter is deployed, you must configure database connectivity by adding data sources to your WebLogic domain. WebLogic Java Database Connectivity (JDBC) data sources provide database access and database connection management.

1. Enter the following URL in a web browser, replacing `host:port` with the host name and port for the WebLogic Administration Console:

   ```
   http://wls_host:wls_port/console
   ```

   For example, `https://ebsasserter.example.com:7002/console`.
2. Log in to WebLogic console as an administrator.
3. In the administration console under **Domain Structure**, expand **Services** and then click **Data Sources**.
4. Under the **Data Sources** table heading, click the **New** drop-down list, and then select **Generic Data Source**.

5. In the **JDBC Data Source Properties** section, specify the following values, and then click **Next**:
   - **Name**: `visionDS`
   - **JNDI Name**: `visionDS`
   - **Database Type**: `oracle`

   The value of the **Name** parameter must match the `ebs.ds.name` parameter in the E-Business Suite Asserter configuration file.
6. Select a database driver, and then click **Next**.
   - If you are using an XA data source, select `*Oracle's Driver (Thin XA) for Instance connections; Versions:any`.
   - If you are using a non-XA data source, select `*Oracle's Driver (Thin) for Instance connections; Versions:Any`.
7. In the **Transaction Options** section, perform one of the following, and click **Next**:
   - For a non-XA data source, uncheck the **Supports Global Transactions** check box.
   - For an XA data source, leave the check box checked.
8. In the **Connection Properties** section, specify the following appropriate values and then click **Next**.
   - **Database Name**: `EBSDB`
   - **Host Name**: `ebs.example.com`
   - **Port**: `1521`
   - **Database Username**: Enter the username you created earlier.
   - **Password**: Enter the password for the username.
9. In the **Driver Class Name** field, enter one of the following:
   - `oracle.apps.fnd.ext.jdbc.datasource.AppsDataSource` if you use a non-XA data source.
   - `oracle.apps.fnd.ext.jdbc.datasource.AppsXADataSource` if you are using an XA data source.

   Optionally, you can use the `oracle.jdbc.OracleDriver` driver instead, but you need to provide administrative database credentials during this value. If you don't want to expose administrative database credentials to WebLogic administrators, use one of the two values provided for **Driver Class Name** in this task.
10. In the **Properties** text box, keep the current value for `user`, add a new line, and enter the path to the `dbc` file as per the example below:

```
user=IDETITYADMIN
dbcFile=/opt/ebssdk/EBSDB_ebsasserter.example.com.dbc
```

> **Note:**
>
> This field is case sensitive. Make sure the name of the file is correctly written with the correct uppercase and lowercase letters.

11. Review the data source properties values, confirm that the database is running, and click **Test Configuration**.

    Make sure your network doesn't block communication between the E-Business Suite Asserter's WebLogic server machine and the Oracle E-Business Suite database through the port number you provided in the datasource.

12. When you see the **Connection test succeeded** message, then click **Next**.
13. In the **Select Targets** section, select the target server (for example, **EBSAsserter_server**), and click **Finish**.
14. In the **Change Center**, click the **Activate Changes** button.

Deploy the E-Business Suite Asserter on Oracle WebLogic Server

You must deploy the E-Business Suite Asserter to the Administration Server instance of Oracle WebLogic Server for the purpose of performing end-to-end testing of the integration.

1. Copy the E-Business Suite Asserter war file (`ebs.war`) to the working folder in the Oracle WebLogic Server `/opt/ebssdk`.
2. Enter the following URL in a web browser, replacing `host:port` with the host name and port for the Oracle WebLogic Server Administration Console:

   ```
   http://wls_host:wls_port/console
   ```

   For example, `https://ebsasserter.example.com:7002/console`.
3. Log in to the WebLogic console as an administrator.
4. In the **Change Center**, click the **Lock & Edit** button.
5. Under Domain Structure, click **Deployments**.
6. On the right, under **Deployments**, click the **Install** button.
7. Enter the path for the E-Business Suite Asserter war file as `/opt/ebssdk`.
8. Select the `ebs.war` file and click **Next**.
9. Select **Install this deployment as an application**, and then click **Next**.
10. Select the target server (for example, **EBSAsserter_server**) and then click **Next**.
11. Accept the default values and click **Finish**.
12. Click **Activate Changes**.

Update Oracle E-Business Suite Profiles

Configure the URL whitelist property to prevent access to the Oracle E-Business Suite local login and direct all requests to the E-Business Suite Asserter login instead.

1. Log in to Oracle E-Business Suite console as a user that is assigned the Functional Administrator responsibility (typically `sysadmin`).
2. Use the drawer icon (E-Business Suite version 12.2.8) or navigator icon (E-Business Suite version 12.1/12.2), click **Functional Administrator**.
3. In the **Oracle Applications Administration** page, click the **Core Services** tab, and then click **Profiles**.
4. Enter `APPS_AUTH_AGENT` in the **Code** field and then click **Go**.

5. On the list of profiles, select **Application Authenticate Agent**, and then click **Define Profile Values**.
6. On the **Define Profile Values: Application Authenticate Agent** page, enter the E-Business Suite Asserter URL in the **Site Value** field, and then click **Update**.
7. Click **Profiles** under the **Core Services** tab, enter `APPS_SSO` in the **Code** field, and then click **Go**.
8. On the list of profiles, select **Applications SSO Type**, click **Define Profile Values**, change the **Site Value** field from **SSWA** to **SSWA w/SSO**, and then click **Update**.
9. Click **Profiles** under the **Core Services** tab, enter `ICX_SESSION_COOKIE_DOMAIN` in the **Code** field, and then click **Go**.
10. On the list of profiles, select **Oracle Applications Session Cookie Domain**, click **Define Profile Values**, replace the **Site Value** field from **HOST** to **DOMAIN**, and then click **Update**.
11. Restart the Oracle E-Business Suite servers.

Validate the Integration

In order to successfully test SSO with Oracle E-Business Suite, make sure that you create a user in Oracle Identity Cloud Service whose username is identical to an Oracle E-Business Suite user name. Then, assign the E-Business Suite Asserter application to this user. These sample configurations and testing apply to an Oracle E-Business Suite demo distribution (VISION) environment. Check the values of the encoded URLs before applying them to your environment.

Test the SSO Using the E-Business Suite Asserter Direct URL

You can use the E-Business Suite Asserter direct URL to verify the integration and ensure that the SSO works.

1. Open a browser and enter the URL `https://ebsasserter.example.com:7002/ebs` for the E-Business Suite Asserter.
2. The Oracle Identity Cloud Service **Sign In** page appears. Use the username and password of the previously created user to sign in.
3. Upon successful authentication, the user is redirected to the Oracle E-Business Suite home page without having to enter Oracle E-Business Suite credentials.
4. If the Oracle E-Business Suite home page appears, verify the logged-in username.
5. Log out from Oracle E-Business Suite. The browser is redirected to the Oracle Identity Cloud Service **Sign In** page.

Test the SSO Using the E-Business Suite Asserter Icon in Oracle Identity Cloud Service

You can use the E-Business Suite Asserter icon within the **My Apps** page in Oracle Identity Cloud Service to verify the integration and ensure that the SSO works.

1. Open a browser and enter the Oracle Identity Cloud Service **My Console** URL: `https://idcs-example.identity.oraclecloud.com/ui/v1/myconsole`.
2. Sign in using the credentials of the previously created user.

3. In the **My Apps** page, click the E-Business Suite Asserter icon to access the **Oracle E-Business Suite Home** page.
4. If the **Oracle E-Business Suite Home** page appears, verify the logged-in user name.
5. Log out from Oracle E-Business Suite. The browser is redirected to the Oracle Identity Cloud Service **Sign In** page.

Test the SSO Using the E-Business Suite Asserter Direct URL with a Redirect Parameter

You can use the URL for the E-Business Suite Asserter with a redirect parameter to verify the integration and ensure that the SSO works.

1. Open a browser and enter the URL for the E-Business Suite Asserter along with the `requestUrl` parameter. In the following example, the parameter value points to one of the Oracle E-Business Suite pages (for example, Self Service Reports page - P11D Reports).

   ```
   https://ebsasserter.example.com:7002/ebs?
   requestUrl=http%3A%2F%2Febs.example.com%3A8000%2FOA_HTML%2FRF.jsp%3Ffunctio
   n_id%3D1023615%26resp_id%3D54745%26resp_appl_id%3D800%26security_group_id%3
   D0%26lang_code%3DUS%26oas%3DZGSSqTllSAVkI4tpzTqoZw..%26params%3DYQiYllX3TGJ
   Smdkebayqm4plh8uddwPMseD54DE-G-c
   ```

   The `requestUrl` parameter value must match one of the `whitelist.urls` and must be URL encoded.
2. The Oracle Identity Cloud Service **Sign In** page appears. Use the username and password of the previously created user to sign in.
3. Upon successful authentication, the user is redirected to the page passed as a parameter to the E-Business Suite Asserter URL in the `requestUrl` parameter.
4. Log out from Oracle E-Business Suite. The browser is redirected to the Oracle Identity Cloud Service Sign In page.

Test the SSO Using a Previously Oracle E-Business Suite Bookmarked URL

You can use the Oracle E-Business Suite URL that you have bookmarked to verify the integration and ensure that the SSO works.

1. Open a browser and enter one of the Oracle E-Business Suite URLs that you have bookmarked (for example, the Self Service Reports page - P11D Reports):

   ```
   https://ebsasserter.example.com:7002/ebs?
   requestUrl=http%3A%2F%2Febs.example.com%3A8000%2FOA_HTML%2FRF.jsp%3Ffunctio
   n_id%3D1023615%26resp_id%3D54745%26resp_appl_id%3D800%26security_group_id%3
   D0%26lang_code%3DUS%26oas%3DZGSSqTllSAVkI4tpzTqoZw..%26params%3DYQiYllX3TGJ
   Smdkebayqm4plh8uddwPMseD54DE-G-c
   ```

2. The Oracle Identity Cloud Service **Sign In** page appears. Use the user name and password of the previously created user to sign in.
3. Upon successful authentication, the user is redirected to the Oracle E-Business Suite page passed as a parameter to the E-Business Suite Asserter URL in the `requestUrl` parameter.

4. Log out from Oracle E-Business Suite. The browser is redirected to the Oracle Identity Cloud Service **Sign In** page.

Validate the Service

You can validate the Asserter configuration, E-Business Suite configuration and IDCS Application setup using the Validation Service.

Access the validation service using the endpoint `app.url/validate` for example, *`https://ebsasserter.example.com:7002`*`/ebs/validate`.
The validation result of each configuration will be one of `Success`, `Failure`, `Undetermined`.

If the result is `Failure` or `Undetermined`, the response body will suggest the expected configuration. Re-configure EBS Asserter or the Oracle E-Business Suite or IDCS Application as appropriate.

Login with Non-US English Language

The latest Oracle Identity Cloud Service EBS Asserter will support the language configuration of a user provided in EBS. If the `FND_OVERRIDE_SSO_LANG` profile option is enabled for a user in EBS, Asserter creates an EBS session based on the value of the `ICX_LANGUAGE` profile option of this user.

For a given EBS deployment, if there is no such configuration for a user in EBS, the language can be controlled using the `langCode` parameter as mentioned below or `base.lang` can be set in the bridge.properties.

You can login in to the EBS Home Page in a different language by specifying the code for the language in the login URL:

```
https://apps.example.com/OA_HTML/AppsLogin?langCode=language
            code>
```

Set up E-Business Suite Mobile Applications

You can authenticate Oracle E-Business Suite mobile applications such as **Approvals for EBS** with Oracle Identity Cloud Service, when your Oracle E-Business Suite is making use of **Identity Cloud Service E-Business Suite Asserter**.

By doing so, when users open the mobile application and try to access Oracle E-Business Suite information, the asserter uses Oracle Identity Cloud Service to authenticate these users through the mobile application.

This set up is valid for the following Oracle E-Business Suite mobile applications:

- Custom mobile applications based on **Oracle Mobile Application Framework** (Oracle MAF).
- **Approvals for EBS** from **Oracle America, Inc**.
- **Oracle Fusion Expenses** from **Oracle America, Inc**.

Before You Begin

You need to make sure you have configured the E-Business Suite mobile application to work with your Oracle E-Business Suite.

- For Oracle E-Business Suite version 12.1.3 and version 12.2, apply or validate if mobile application patches have been applied as per Oracle E-Business Suite documentation. If you use Oracle E-Business Suite 12.2.8, you can skip applying the patches as those are already included.
- To execute the following configurations, you need to log in to Oracle E-Business Suite with any user (excluding sysadmin) assigned to **Mobile Applications Manager** role.

You don't need to configure Oracle E-Business Suite with Oracle Access Manager (OAM) Access Gate. **Identity Cloud Service E-Business Suite Asserter** replaces OAM Access Gate as the authentication mechanism for your Oracle E-Business Suite.

Configure E-Business Suite for Mobile Applications

Configure Oracle E-Business Suite to enable E-Business Suite mobile applications to authenticate with Oracle Identity Cloud Service.

1. Access the drawer icon (E-Business Suite version 12.2.8) or navigator icon (E-Business Suite version 12.1/12.2), select **Mobile Applications Manager**, and then select **Applications**.
2. Search for **Application Name**. For example, `EBS Approvals`.
3. In the results list, click the **Configure** icon for the application. For example, **EBS Approvals**.
4. In the **Configure Mobile Application** page, expand the **Connection Settings**.
5. Select **Sub Category** as `AppsSSO Login`.
6. Expand the **Connection Settings** category, and then update the parameters as follows:
   - **LoginURL**: `%APPS_AUTH_AGENT%/login/sso`
   - **LogoutURL**: `%APPS_AUTH_AGENT%/logout/sso`
   - **LoginSuccessURL**: `%APPS_AUTH_AGENT%/login/sso`
   - **APPS_SESSION_SERVICE**: `%APPS_AUTH_AGENT%/login/apps`
7. Click **Apply**.

After you save the changes, restart Oracle E-Business Suite.

Test Authentication for E-Business Suite Mobile Applications

Test the authentication of the Oracle E-Business mobile applications in scenarios in which E-Business Suite uses Identity Cloud Service E-Business Suite Asserter.

If you configured E-Business Suite Mobile Applications to make use of Oracle Identity Cloud Service EBS Asserter, then you perform the following test:

1. Use your mobile device to open the Oracle E-Business mobile application and access a protected feature.

   The mobile device opens Oracle Identity Cloud Service **Sign In** page.
2. Sign in to Oracle Identity Cloud Service.

   After successful authentication, the mobile application completes the login process and activates the requested feature.

The mobile application completes the login flow and shows the protected feature only if it detects a successful return that redirects to the URL configured in the **Login Success URL** parameter.

Collect Diagnostic Data

Configuration and log files help you diagnose issues relating to the Oracle Identity Cloud Service and Oracle E-Business Suite integration using the E-Business Suite Asserter.

You must collect the following files:

- The E-Business Suite Asserter configuration file (`bridge.properties`).
- The E-Business Suite Asserter diagnostic logs.
- The domain log file for the WebLogic domain where the E-Business Suite Asserter is deployed.
- The HTTP header trace.

Enable the E-Business Suite Asserter Debug Log

To send logs to a file, add `FileHandler` to the `handlers` property in the `logger.properties` file. This will enable file logging globally.

1. Create a `logger.properties` file with entries as follows:

```
handlers = java.util.logging.FileHandler, java.util.logging.ConsoleHandler
java.util.logging.FileHandler.pattern = %h/ebsasserter.log
java.util.logging.FileHandler.formatter = java.util.logging.SimpleFormatter
java.util.logging.FileHandler.level=ALL
java.util.logging.ConsoleHandler.formatter =
java.util.logging.SimpleFormatter
java.util.logging.ConsoleHandler.level=ALL
com.oracle.ebs.sso.level=ALL
oracle.apps.fnd.ext.level=ALL
oracle.security.jps.idcsbinding.level=ALL
```

2. Add the option `-Djava.util.logging.config.file=<logger.properties file created above>` in Oracle WebLogic Server:
   a. Using a browser, access the Oracle WebLogic Server Administration Console.
   b. In the Oracle WebLogic Server Administration Console, click **servers** under **Environment** in the **Domain Structure**.
   c. In the **Servers** table, click the name of the server instance where the E-Business Suite Asserter is deployed.
   d. From the **WebLogic Server** menu, select **Administration**, then select **Server Start**.
   e. From the **Server Start** page, you can add the option `-Djava.util.logging.config.file=<logger.properties file created above>` in the **Arguments** field.
   f. Click **Save**.
3. Restart the Oracle WebLogic Server where the E-Business Suite Asserter is deployed.

   The E-Business Suite Asserter debug log file is located in `<HOME DIR>/ebsasserter.log`.

Use Fiddler to Capture HTTP Traffic

You can use Fiddler to view and debug HTTP traffic between a client and a host computer.

1. Download the Fiddler installer.
2. Run the Fiddler installer and follow the wizard to install Fiddler on your client machine.
3. Stop all other programs and services that might access the internet or use HTTP. This helps to obtain a clean and uncluttered trace.
4. Select the Fiddler icon in the workstation's Start menu to run Fiddler.

   Fiddler starts capturing events as soon as it launches. Fiddler logs all network requests instantly and these requests are summarized in the left hand pane of the tool.

The individual sections of the Fiddler trace are color coded. Each color is meaningful, but failures are shown in red. The result column contains the HTTP code that the section returned. For example, if a section returned an HTTP 404 or Not Found error message, the section would be red. Since we are typically looking for errors or failures, the sections we want to concentrate on are the red sections in the trace. Fiddler creates a file containing the trace with a file extension of `.saz`.

Monitor the E-Business Suite Asserter

You can monitor the E-Business Suite Asserter to determine the status and in turn its availability.

Use the *`app.url`*`/about` URL to monitor the availability of the EBS Asserter. For example, `https://ebsasserter.example.com:7002/ebs/about`.

Troubleshoot Common Issues

Here are a few errors that you might encounter while integrating Oracle E-Business Suite with Oracle Identity Cloud Service using the E-Business Suite Asserter.

Resolve an Insufficient Privileges Error

After Oracle Identity Cloud Service authentication, instead of getting access to Oracle E-Business Suite, the user gets redirected back to Oracle E-Business Suite with the error message `"You have insufficient privileges for the current operation."` and prompts the user to sign in again.

Generally, when the Oracle E-Business Suite application throws this error, it means that the cookie is set with an incorrect domain. To confirm this, check the E-Business Suite Asserter debug log (`<HOME DIR>/ebsasserter.log`). The E-Business Suite Asserter debug log shows that the `sessionCookieDomain` has an incorrect value. The `CookieDomain` was set to `.oracle.com`.

```
Aug 22, 2018 2:26:34 PM oracle.apps.fnd.ext.common.EBiz init
FINE: Ebiz init(): sessionCookieDomain =.oracle.com ; protocol=https:;
ssoCookieName= ORASSO_AUTH_HINT
```

The ICX_PARAMETERS.SESSION_COOKIE_DOMAIN must not be set to a value of any kind. You must update the SESSION_COOKIE_DOMAIN setting in ICX_PARAMETERS.

1. Update the SESSION_COOKIE_DOMAIN value in ICX_PARAMETERS:

```
SQL> select SESSION_COOKIE_DOMAIN from ICX_PARAMETERS;

SESSION_COOKIE_DOMAIN
------------------------------
.oracle.com
```

2. Set `session_cookie_domain` to `NULL` in the `ICX_PARAMETERS`:

```
update ICX_PARAMETERS set SESSION_COOKIE_DOMAIN = NULL;
commit;
```

3. Restart all services.
4. Retest the issue.

Resolve an Internal Server Error While Logging Out

When you are logging out from Oracle E-Business Suite, the browser throws an error message `"Internal Server Error"`.

This issue was due to an older version of `AppsLogoutRedirect.java` in the Oracle E-Business Suite side.

Check the header for `AppsLogoutRedirect.java` in the Oracle E-Business Suite side:

```
adident Header $JAVA_TOP/oracle/apps/fnd/sso/AppsLogoutRedirect.class
$Header AppsLogoutRedirect.java 120.10.12010000.7 2010/01/19 20:18:52 rsantis
ship $
```

You must apply the latest Oracle E-Business Suite Release 12 Critical Patch Update Jan 2013 or above to fix this issue. This Critical Patch Update enables `AppsLogoutRedirect.java` to leverage the `APPS_SSO` and `APPS_AUTH_AGENT` profiles. Check the Knowledge Document (July 2018) (Doc ID 2379675.1) for all the details to apply this patch.

Fix a Time Sync Issue

While you are accessing the E-Business Suite Asserter application URL, the Oracle E-Business Suite application login flow resulted in an internal server error.

The HTTP header trace looks like this:

```
GET https://xxxxxxxxxxxxxxxxx.oracle.com:7002/ebs/response?
code=AQIDBAVcZbun_M5qU4-t9LUCYDjAOgWYiDOrf1Kb5ndbWAEYd05C-uxDfSwP8Ejfn51WT-
gTuYj6bLFFYAFHQEqgYy26MTEgRU5DUllQZZIIFFVElPTl9LRVkxNCB7djF9NCAFFFABCDEF=
HTTP/1.1

Error 500--Internal Server Error
From RFC 2068 Hypertext Transfer Protocol -- HTTP/1.1:
10.5.1 500 Internal Server Error
The server encountered an unexpected condition which prevented it from
fulfilling the request
```

The E-Business Suite Asserter domain log looks like this:

```
####<Sep 23, 2018 6:53:31,380 PM AST> <Error> <HTTP> <ebshost01.oracle.com>
<AdminServer> <[ACTIVE] ExecuteThread: '6' for queue:
'weblogic.kernel.Default (self-
tuning)'> <<WLS Kernel>> <> <0b38f1ae-a3cb-48f6-80d9-00e3f3bdb263-000000a0>
<1537718011380> <[severity-value: 8] [rif: 0] [partition-id: 0] [partition-
name:
DOMAIN]> <BEA-101020> <[ServletContext@44159983[app:ebs module:ebs.war
path:null spec-version:3.1]] Servelet failed with an Exception
```

The E-Business Suite Asserter log looks like this:

```
FINE: validateToken return with result {"user_result":"America\/New_York",
"at_hash":"1A3gT4BT0WoWCTLE3IFa5A","sub":"john.doe@oracle.com","user_locale":"
en",
"idp_name":"localIDP","idp_guid":"localIDP","a mr":["USERNAME_PASSWORD"],
"iss":"https: \/\/identity.oraclecloud.com\/","user_tenantname":"idcs-
a61feab148e248508205cd98cdea4232",
"client_id":"67179f2609ab46309a75e5ca1f582a53","sid":"18ee87ea-04cf-4469-
a565-48ccc763caf9",
"authn_strength":"2","azp":"67179f2609ab46309a75e5ca1f582a53","auth_time":"153
6180435",
"session_exp":1537715029,"user_lang":"en","exp":1536209235,"iat":1536180437"id
p_type":"LOCAL",
"tenant":"idcs-a61feab148e248508205cd98cdea4232","jti":"ed7be32b-
d4e1-4e72-9868-6df142f07c6b",
"user_displayname":"John Doe","sub_mappingattr":"userName","tok_type":"IT",
"aud":["https:\/\/
identity.oraclecloud.com\/","67179f2609ab46309a75e5ca1f582a53"],
"user_id":"63bf3d3f96094a66a6b7714218338116"}
```

The `session_exp` is set to `1537715029`. Use `EpochConverter` to convert the current Unix epoch time to a human readable date and time. Hence, the expiry time in the token is set to `Sunday, September 23, 2018 3:03:49 PM GMT`. However, the time in the E-Business Suite Asserter domain log is `Sep 23, 2018 6:53:31,380 PM AST`. Note that Greenwich Mean Time is 4 hours ahead of Atlantic Standard Time. Hence the time set is `Sep 23, 2018 10:53:31 PM GMT`. The system where the E-Business Suite Asserter is deployed, is not in time sync with Oracle Identity Cloud Service, as a result the token passed by Oracle Identity Cloud Service is effectively out of the validity period and hence the error `"Token Expired"`.

Ensure the date and time on the system where the E-Business Suite Asserter is deployed is in time sync with NTP servers and hence the Oracle Identity Cloud Service host.

Handle Java Error ExceptionInInitializerError

While you are accessing the E-Business Suite Asserter application URL, the Oracle E-Business Suite application throws the `java.lang.ExceptionInInitializerError` error.

The E-Business Suite Asserter debug log shows the following Java error:

```
<Feb 26, 2019 2:17:16,884 PM PST> <Error> <HTTP> <BEA-101020>
<[ServletContext@2100554246[app:ebs module:ebs.war path:null spec-
version:3.1]] Servlet failed with an Exception
```

```
java.lang.ExceptionInInitializerError
at
com.oracle.ebs.sso.ConnectionProvider.getConnection(ConnectionProvider.java:36
)
at
com.oracle.ebs.sso.RequestWrapperFilter.doFilter(RequestWrapperFilter.java:34)
at weblogic.servlet.internal.FilterChainImpl.doFilter(FilterChainImpl.java:78)
```

This occurs due to incorrect settings in the `bridge.properties` file. Verify the `bridge.properties` file and check that it has the required configuration. Also, check that the path specified in `wallet.path` in the `bridge.properties` file is valid.

Handle Java Error RuntimeException

While you are accessing the E-Business Suite Asserter application URL, the Oracle E-Business Suite application throws `java.lang.RuntimeException`.

The E-Business Suite Asserter debug log shows the following Java error:

```
<Feb 26, 2019 2:01:33,454 PM PST> <Error> <HTTP> <BEA-101020>
<[ServletContext@1207779454[app:ebs module:ebs.war path:null spec-
version:3.1]] Servlet failed with an Exception
java.lang.RuntimeException: javax.naming.NameNotFoundException: Unable to
resolve 'visionDS1'. Resolved ''; remaining name 'visionDS1'
at
com.oracle.ebs.sso.ConnectionProvider.getConnection(ConnectionProvider.java:42
)
at
com.oracle.ebs.sso.RequestWrapperFilter.doFilter(RequestWrapperFilter.java:34)
```

Check that the `ebs.ds.name` value set corresponds to the datasource name created in WebLogic.

Fix a Deep Link Issue

After Oracle Identity Cloud Service authentication, instead of getting access to Oracle E-Business Suite, the user gets redirected back to Oracle E-Business Suite and prompts the user to sign in again.

This occurs because the deep link is not working.

Check that the `whitelist.urls` bridge property is configured. If the issue persists, specify the port numbers explicitly in the `whitelist.urls` configuration. For example, `whitelist.urls=http://ebs.oracle.com:80/OA_HTML...`. You can also check the JSESSION ID Cookie Name of the E-Business Suite Asserter App in the `weblogic.xml` file. If there is any other web app in the WebLogic with the same cookie name, it will conflict.

Issues During Log Out

If you find issues during the logout process verify the **Post Logout Redirect URL** parameter value in Oracle Identity Cloud Service and the **post.logout.url** parameter value in the `bridge.properties` file.

The **post.logout.url** in the `bridge.properties` file is an optional parameter and by default you don't need to provide a value. You use this parameter to make the E-Business Suite Asserter application redirect the user browser to the specified URL after E-Business Suite Asserter finishes the logout process.

If enabled, the value of the **post.logout.url** in the `bridge.properties` file must match the value of the **Post Logout Redirect URL** parameter for E-Business Suite Asserter application in Oracle Identity Cloud Service.

1. Open the E-Business Suite Asserter application in Oracle Identity Cloud Service and update the **Post Logout Redirect URL** value.
2. Open the `ebs.war` file, update the `bridge.properties` file, regenerate the war file, and redeploy the file to the WebLogic server. Make sure the value of this parameter matches the **Post Logout Redirect URL** parameter in Oracle Identity Cloud Service.

44

Integrate Oracle Identity Cloud Service SSO with Oracle PeopleSoft HCM

Oracle Identity Cloud Service integration with Oracle PeopleSoft Human Capital Management (HCM) provides Single Sign on (SSO) using a User ID and a Password.

Topics:

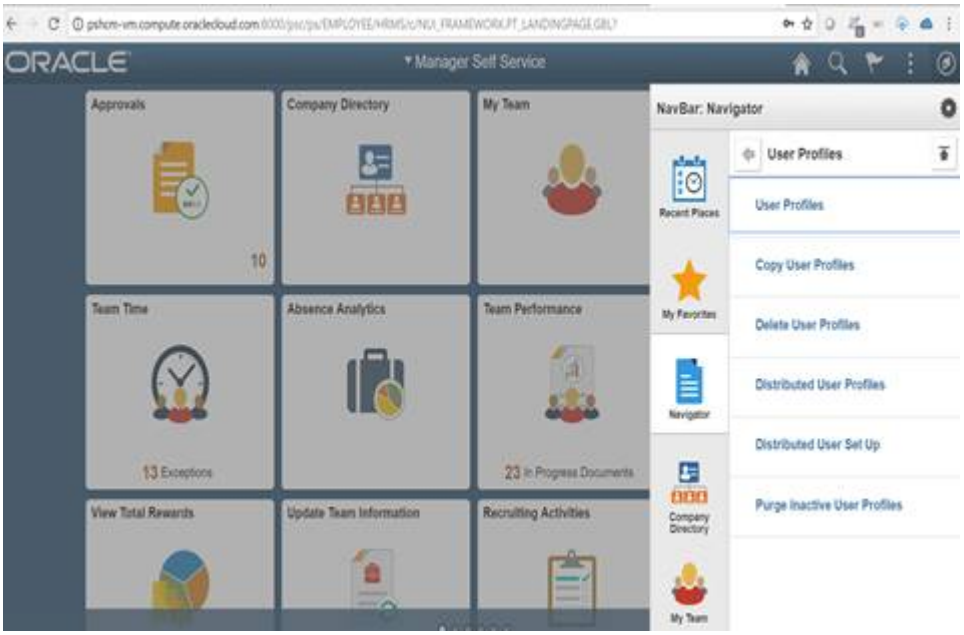
Configure Oracle Identity Cloud Service for PeopleSoft

1. Access the Identity Cloud Service console.
2. Expand the **Navigation Drawer**, click **Applications**, and then click **Add**.
3. In the Add Application page, click **Enterprise Application**.
4. Create an Enterprise application. *See* Add an Enterprise Application.
    a. On SSO Configuration, create a Resource.
    b. On SSO Configuration, create an Authentication Policy and add the headers: `OAM_REMOTE_USER` and `PS_SSO_UID`. The header values should be UserName.
    c. Click **Finish**.
5. On the Identity Cloud Service console, click **Security**, **App Gateways**, and configure a host.
    a. On Add App Gateway, click **Add**.
    b. Enter a name and description and click **Next**.
    c. Complete the host information and click **Save**.
    d. On the Apps tab, add the PeopleSoft application to the App Gateway.

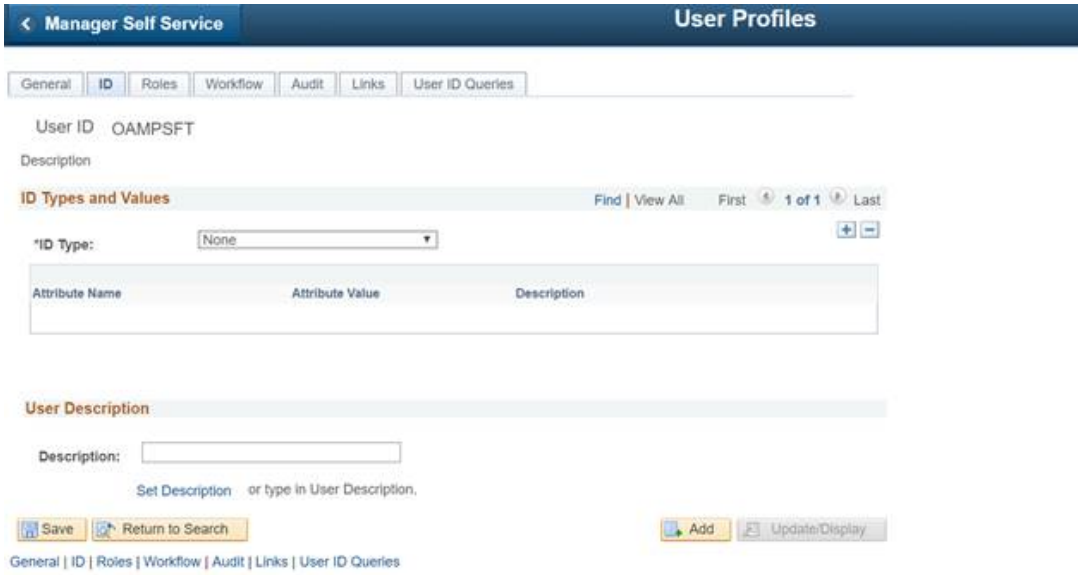
Configure Oracle PeopleSoft HCM

1. Log in to the PeopleSoft console using administrator credentials and then click the **NavBar**.

2. Navigate to **PeopleTools**, **Security**, **User Profiles**, **User Profiles**.



3. Add a new profile named **OAMPSFT**. Provide the password on the first tab. Make sure all user profiles created are in upper case. Click the ID tab and select ID as **none**.

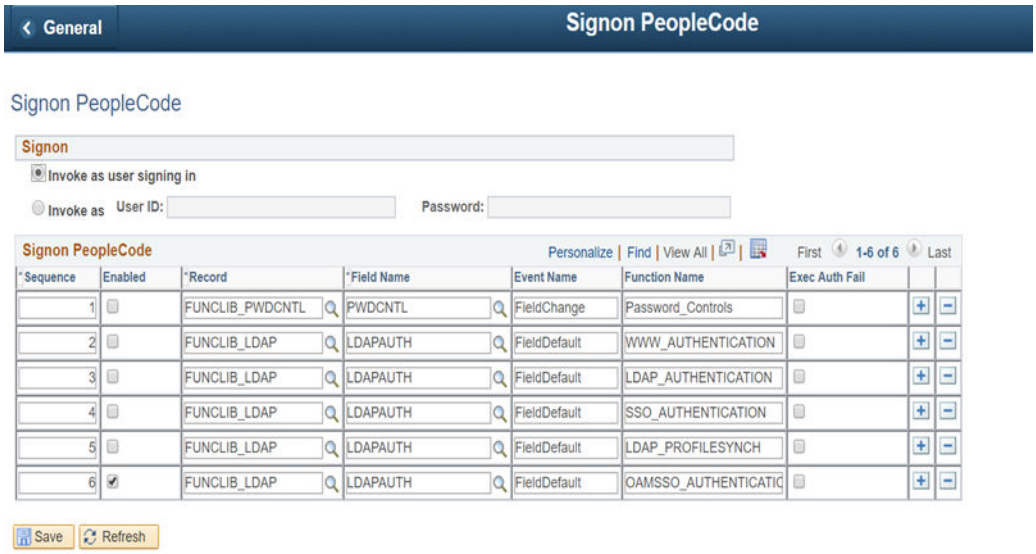


4. Click the Roles tab and add the **PeopleSoft User**. Click **Save**.

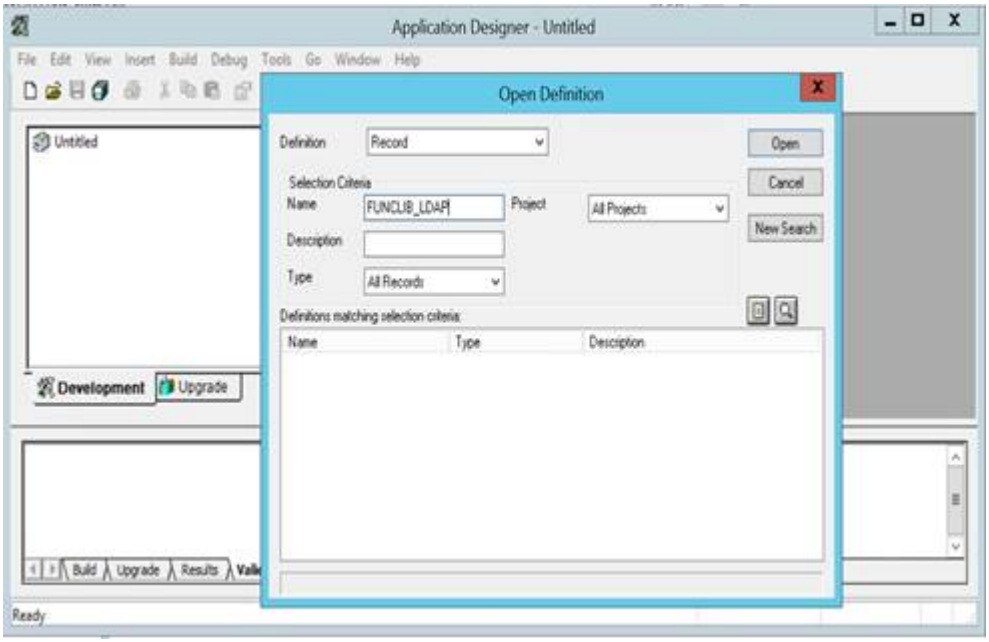
5. Navigate to **PeopleTools**, **Web Profile**, **Web Profile Configuration**. Search for the **PROD** profile and then click the Security tab. Check **Allow Public Access** and add the

OAMPSFT User ID and Password.

6. Navigate to **PeopleTools**, **Security**, **Security Objects**, **Signon PeopleCode**. Enable **OAMSSO_AUTHENTICATION**. Make sure that no other functions are enabled.



7. Log in to Application Designer. Click **Open** and select **Record** from the drop down. For the **Selection Criteria Name**, enter **FUNCLIB_LDAP** and then click **Open**.



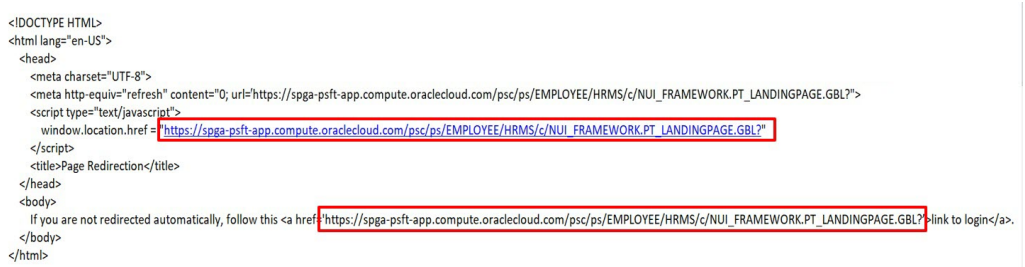
8. Right click **LDAPAUTH** and then **View PeopleCode**. Search for function `getWWWAuthConfig()` and provide **OAMPSFT** as the default `userid`.

9. Search for the `OAMSSO_AUTHENTICATION()` function and provide `PS_SSO_UID` as the Header value.

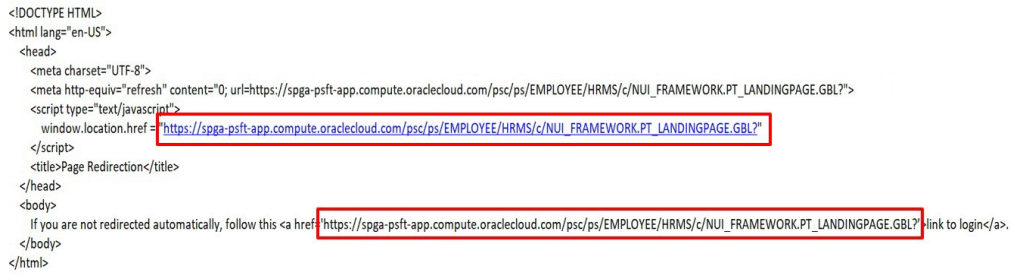
10. You have completed PeopleSoft configuration. Restart PeopleSoft.

11. Replace the `window.location.href` and the *redirect page* URLs with the Application URL in the index.html, signon.html and signin.html pages.

    - Update signon.html: `/home/psadm2/psft/pt/8.56/webserv/peoplesoft/applications/peoplesoft/PORTAL.war/ps/signon.html`



    - Update signin.html: `/home/psadm2/psft/pt/8.56/webserv/peoplesoft/applications/peoplesoft/PORTAL.war/WEB-INF/psftdocs/ps/signin.html`



    - Update signin.html: `/home/psadm2/psft/pt/8.56/webserv/peoplesoft/applications/peoplesoft/PORTAL.war/WEB-INF/psftdocs/ps/signin.html`

Configure App Gateway for PeopleSoft High Availability

1. Go to `cd /usr/local/nginx/conf/origin_conf`.
2. Create **psft_servers.conf**.
3. Add the below values in **psft_servers.conf**.

```
upstream psft-upstream {
     sticky;
        server <PeopleSoft IP1>:8000;
        server <PeopleSoft IP2>:8000;
}
```

4. Restart Ngnix server.

Update the PeopleSoft URL in the App Gateway

1. Access the Identity Cloud Service console.
2. Expand the Navigation Drawer, click **Security**, **App Gateway**, **Apps**, and select the application.

3. In the **Edit Assigned App** dialog, assign the origin server value as the name of the upstream value.