Oracle® Revenue Management and Billing Cloud Services

Release 7

Authentication JWT Configuration Guide

Revision 1.0

F24198-01

November, 2019

ORACLE

Oracle Revenue Management and Billing Cloud Services Authentication JWT Configuration Guide

F24198-01

Hazardous Applications Notice

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

Third Party Content, Products, and Services Disclaimer

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

Preface

About This Document

This document will help you to configure REST services federated identity with OAuth2.0 for Oracle Revenue Management and Billing (ORMB) cloud service.

Intended Audience

This document is intended for customers using Oracle Revenue Management and Billing (ORMB) cloud service and assumes that you have administrative privileges on the host where you want to install the software.

Contents

1. Federated REST Services with JWT

ORMB provides out-of-the-box JWT Services, which allows a Client Application to access protected REST resources that belong to an end-user. JWT (OAuth2.0) clients can use OAuth 2.0 flows to access resources protected by WebLogic. JWT (OAuth2.0) client can be an application or a service created and controlled by the customer organization, or it can be an application or a service created and controlled by another organization that requires access to resources protected by Weblogic.

In the ORMB application server, customer should configure custom JWT Assertion to handle the JWT requests. They should also configure an Authorization server that issues JWT tokens, which are used by the ORMB application server to authenticate the service requests.

1.1 JWT (OAuth2.0) Terminology

Common terminologies used in OAuth 2.0 are:

- **Resource owner:** An entity capable of authorizing access to a protected resource. When the resource owner is a person, it is called as a user.
- **JWT client:** A third-party application that requires access to the private resources of the resource owner. The OAuth client can make protected resource requests on behalf of the resource owner after the resource owner grants it authorization. OAuth 2.0 introduces two types of clients: **confidential** and **public**. Confidential clients are registered with a client secret, while public clients are not.
- **Authorization server:** The server that gives JWT clients scoped access to a protected resource on behalf of the resource owner. The server issues an access token to the JWT client after it performs the following actions successfully:
    - Authenticate the resource owner
    - Validate a request or an authorization grant
    - Obtain resource owner authorization

  Note: An authorization server can also be a resource server.
- **Access token:** A string that represents the authorization granted to the JWT client by the resource owner. This string represents specific scope and duration of access. It is granted by the resource owner and is enforced by the JWT server.
- **Protected resource:** A restricted resource that can be accessed from the JWT client using authenticated requests
- **Resource server:** The server that hosts the protected resources. It uses access tokens to accept and respond to protected resource requests. The resource server might be the same as the authorization server.
- **Authorization grant:** A grant that represents the resource owner authorization to access its protected resources. JWT clients use an authorization grant to obtain an access token. There are four authorization grant types: Authorization Code, Implicit, Resource Owner Password Credentials, and Client Credentials.

The following diagram shows how JWT roles generally interact with each other:

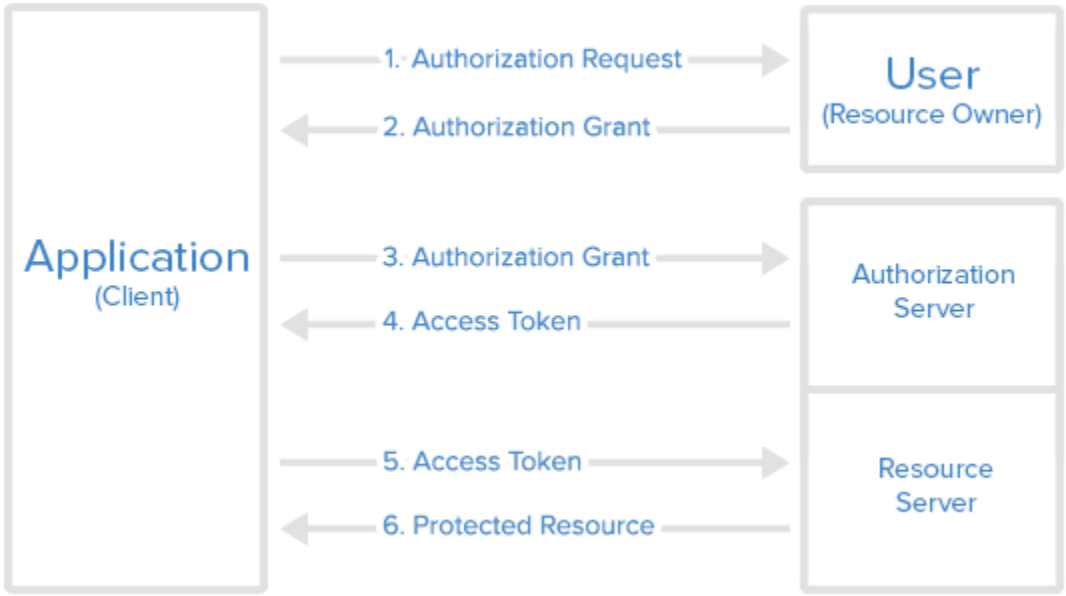

Fig 1: JWT Roles

1.2 Federated Web Service Login Overview

IDP Users (ORMB Applications login users) are expected to be configured in the ORMB application, either manually or through some provisioning process. Customer should have a provisioning process in place to add users into ORMB. ORMB provides REST services to add user into ORMB Application from super/Admin users. ORMB also provides a Webservice user to customer for first time Webservice call for user provisioning process.

Following are the steps included in the web service flow:

1. When a client application wants to access the resources of a resource owner hosted on a resource server, the client application must first obtain an authorization grant.
2. The resource owner gives authorization grant to a client application, in cooperation with the authorization server associated with the resource server.
3. The client application sends its client ID 000000 to the authorization server, so that the authorization server knows which application is trying to access the protected resources.
4. If the authorization server accepts these values, it sends back a JWT access token.
5. The client application now uses the JWT access token to request resources from the resource server. The access token serves as both authentication of client and resource owner (user), and authorization to access the resources.

1.3 JWT (OAuth2.0) Implementation/Configuration

The main steps involved in JWT implementation are:

1. End-users get JWT token from the authorization server.
2. User requests REST Services with the JWT token. (WebLogic Server acts as the REST Service endpoint.)
3. Weblogic authenticates the JWT token and returns an appropriate response to the client.

The following image shows an overview of the different ORMB components and the interaction between them:

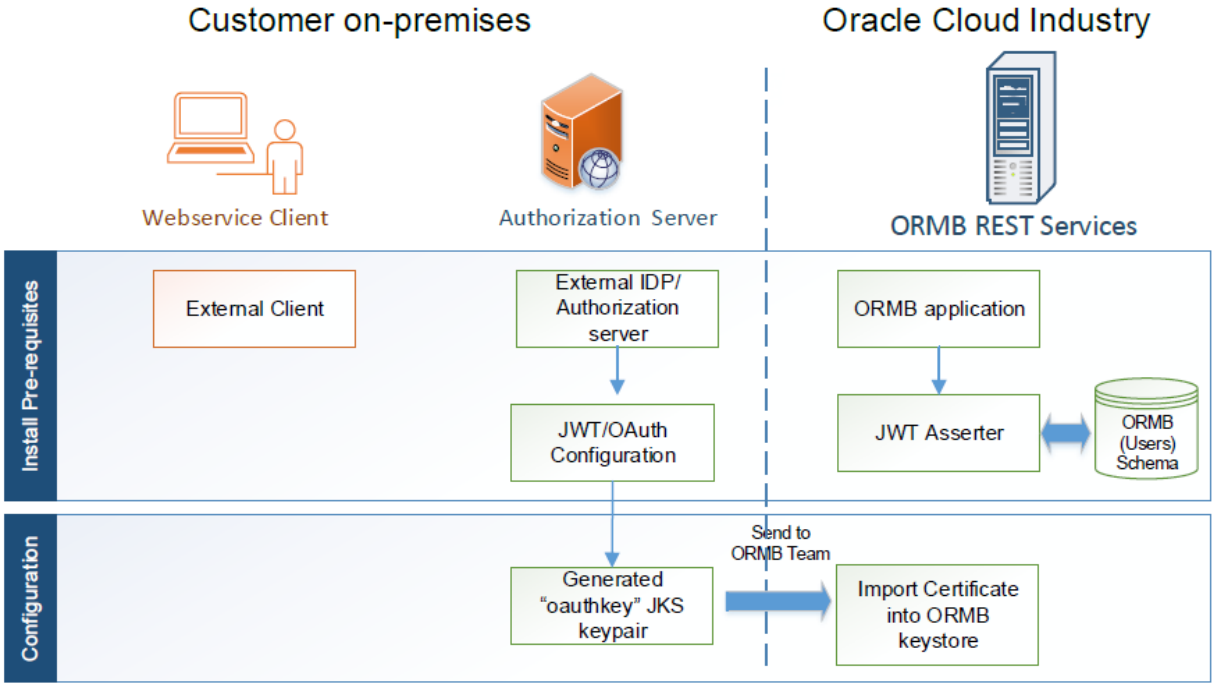

Fig 2: Overview of ORMB Components

