

OAuth2 based Web Services Access Authentication
Oracle Financial Services Lending and Leasing
Version 1.0
December 2020



Table of Contents

1. INTRODUCTION	1-1
1.1 BACKGROUND	1-1
1.2 PURPOSE.....	1-1
1.3 ABBREVIATIONS	1-1
2. WEB SERVICES AUTHENTICATION USING OAUTH2.....	2-1
2.1 UNDERSTANDING OAUTH SERVICES	2-1
2.1.1 Identity Domains.....	2-2
2.1.2 Clients.....	2-2
2.1.3 Resource Server.....	2-2
2.1.4 Resource Owner.....	2-2
2.1.5 Types of OAuth REST API.....	2-2
3. ENABLING OAUTH SETUP CONFIGURATIONS.....	3-3
3.1 ENABLING OAUTH SUPPORT FOR OFSLL REST APIS	3-3
3.2 IDENTITY DOMAIN CREATION	3-3
3.3 RESOURCE SERVER CREATION	3-5
3.4 CLIENT CREATION.....	3-7
3.5 GETTING ACCESS TOKEN	3-8
3.5.1 How OFSLL API works with access token?.....	3-8
3.5.2 Access Token for CLIENT_CREDENTIALS grant type.....	3-9
3.5.3 Access Token for PASSWORD grant type.....	3-10
3.5.4 Access Token for JWT_BEARER grant type	3-11
3.5.5 Access Token for REFRESH_TOKEN grant type	3-13
3.5.6 How to get access token through Basic Authentication	3-15
3.5.7 How to access the REST API using the access token.....	3-16
3.6 EMBEDDING EXTERNAL APPLICATION WITHIN OFSLL	3-16

1. Introduction

1.1 Background

Oracle Financial Services Lending and Leasing (OFSLL) suite is a comprehensive, end-to-end solution that supports full lifecycle of direct and indirect consumer lending business with Origination, Servicing and Collections modules. This enables financial institutions to make faster lending decisions, provide better customer service and minimize delinquency rates through a single integrated platform. It addresses each of the lending processes from design through execution. Its robust architecture and use of leading-edge industry standard products ensure almost limitless scalability.

To extend OFSLL SaaS, OAuth2 can be used for securing OFSLL web services user access Authentication. This document details the process of web services authentication using OAuth services and enabling OAuth setup configurations.

1.2 Purpose

The purpose of this document is to provide detailed information for consulting and partner teams to implement an OAuth2 based REST API access authentication mechanism for OFSLL customers.

1.3 Abbreviations

Abbreviation	Detailed Description
OFSLL	Oracle Financial Services Lending and Leasing
IDM	Identity Management
OAuth	Open Authorization
SaaS	Software as a service
PaaS	Product as a service
OAM	Oracle Access Management
API	Application Program Interface
URL	Uniform Resource Locator
XML	Extensible Markup Language
JWT	JSON Web Token
CSF	Critical success factor

2. Web services authentication using OAuth2

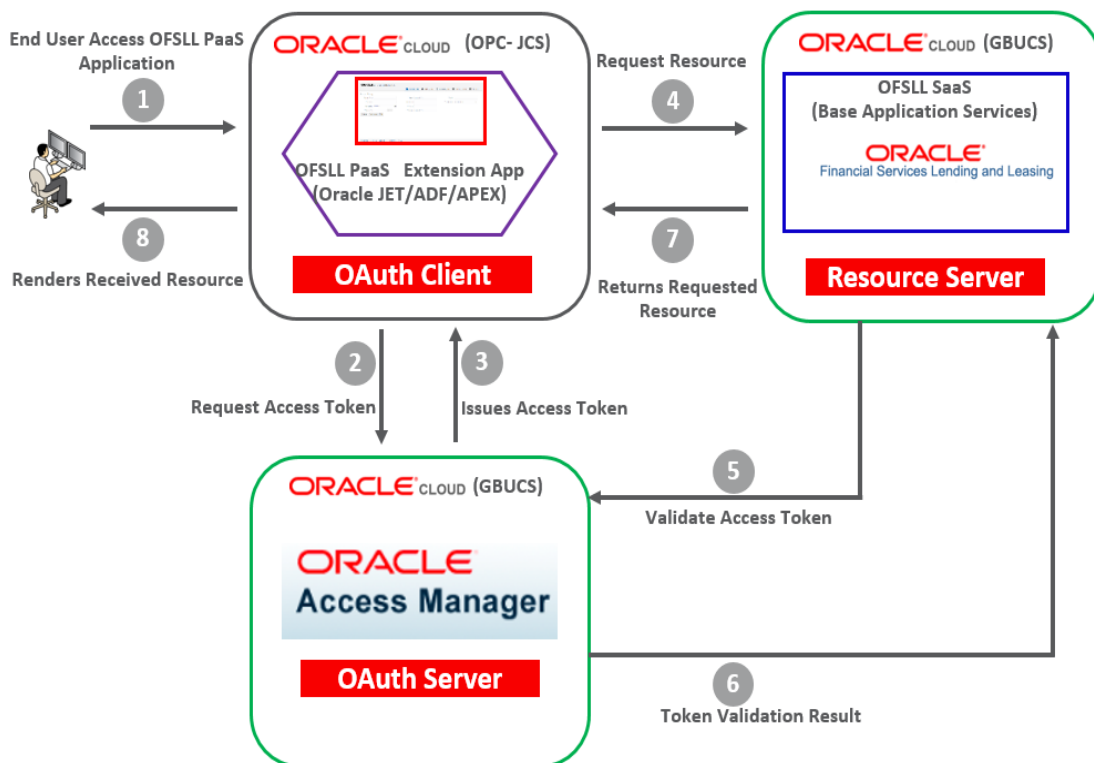
Web services authentication using OAuth2 is one of the best approach for securing user authentication to extend OFSLL SaaS. This uses Oracle / Non-Oracle PaaS to authenticate service access request from an external partner application without sharing OFSLL environment access credentials (UID / Password) and leverages the built-in support for OAuth 2.0.

OAuth 2.0 is an open standard token-exchange technology for verifying a user's identity across multiple systems and domains without risking the exposure of a password.

Third-party applications (those not hosted on Oracle Cloud PaaS) can use OAuth for making calls into OFSLL Cloud REST APIs. PaaS / On-Premise application can pass a user's authentication information and request an OAuth token from OFSLL Cloud, and then use the token to interact with an OFSLL Cloud API. PaaS or On-Premise and SaaS components can be with different ID Domains and security is managed with Shared IDM.

2.1 Understanding OAuth Services

Oracle Access Management (OAM) implemented the OAuth core 2.0 specifications to offer OAuth services. OAuth is an open standard authorization protocol that provides authentication and access control between a Client (such as Web services) and a Resource Owner (Service Provider) on the web.



2.1.1 Identity Domains

The Identity domains are entities that contain all artifacts required to provide standard OAuth services. Identity domains are independent entities and the primary use of this is to provide multi tenants deployments. Each Identity domain will correspond to a tenant. This will also be useful for cloud deployments where each Identity domain can correspond to a separate tenant or entity.

Following are some of the components configured within an OAuth services Identity domain.

- One or More Clients
- One or More Resource Servers

2.1.2 Clients

The client is an application which makes protected resource requests on behalf of the resource owner using its authorization. For example, OFSLL. The Client initiates the OAuth Protocol by invoking the OAuth services. The client may be public or confidential.

There are two types of clients:

- **Confidential Clients:** Web Applications are of confidential client types assigned with a client ID and secret key. These clients can interact with the OAuth services server by sending the Client ID and secret as part of an authorization header.
- **Public Clients:** Public Clients or untrusted clients are assigned with a client ID but no secret key. These are the type of external applications that are not capable of keeping a client password confidential.

2.1.3 Resource Server

The Resource server is the machine on which protected resource is hosted. The Resource server is deployed in a different location from OAM and Client. The Resource server needs to be capable of accepting and responding to protected resource requests using access tokens.

2.1.4 Resource Owner

This is an entity capable of granting access to a protected resource. When the resource owner is a person, it is referred as an end-user.

2.1.5 Types of OAuth REST API

OAuth services are enabled as part of OAM version 12c Installation process. OAM provides an API based approach for configuring OAuth Services. There are 2 types of API OAuth services providers namely Admin API and Runtime API.

The Admin API provides capability to create mandatory admin components like Identity domain, Resource Server and client etc. They must be configured before the client makes the token request.

Note: To Execute Admin API, you can refer to Oracle OAM OAuth REST API documentation available at <https://docs.oracle.com/en/middleware/idm/access-manager/12.2.1.3/oroau/api-admin-identity-domain.html>.

3. Enabling OAuth Setup Configurations

3.1 Enabling OAuth support for OFSLL REST APIs

The OAuth support for OFSLL REST API can be enabled with the following steps:

1. Add context Parameters in web.xml
2. Remove URL Security constraint tags in web.xml

Add the below configuration in web.xml of OfsslRestWS.ear:

```
<context-param>
  <description>This parameter will decide the jersey filter to be loaded</description>
  <param-name>OAUTH_AND_BASIC_ENABLED</param-name>
  <param-value>Y</param-value>
</context-param>
```

3. Remove Security configuration from weblogic.xml as well.

Note: If this context parameter is not set, only the existing basic authentication flow is supported.

3.2 Identity Domain Creation

To create identity domain, any valid reliable REST client application/tool can be used to invoke the REST API. For example, Postman tool

http:<AdminServerHost:Port>/oam/services/rest/ssa/api/v1/oauthpolicyadmin/oauthidentitydomain

Request JSON payload

```
{
  "name":"OFSLL_OAUTH_DOMAIN",
  "identityProvider":"OUD_LDAP",
  "description":"OFSLL_OAUTH_DOMAIN",
  "tokenSettings":[{"tokenType":"ACCESS_TOKEN",
  "tokenExpiry":3600,
  "lifeCycleEnabled":true,
  "refreshTokenEnabled":true,
  "refreshTokenExpiry":86400,
  "refreshTokenLifeCycleEnabled":true
  },
  {
  "tokenType":"AUTHZ_CODE",
  "tokenExpiry":3600,
  "lifeCycleEnabled":true,
  "refreshTokenEnabled":true,
  "refreshTokenExpiry":86400,
  "refreshTokenLifeCycleEnabled":true
  },
  {
  "tokenType":"SSO_LINK_TOKEN",
  "tokenExpiry":3600,
  "lifeCycleEnabled":true,
  "refreshTokenEnabled":true,
  "refreshTokenExpiry":86400,
  "refreshTokenLifeCycleEnabled":false
  }
  ],
  "errorPageURL":"/oam/pages/error.jsp",
  "consentPageURL":"/oam/pages/consent.jsp",
  "customAttrs":"Attribute of user in IDStore to store the encrypted secretkey for TOTP"
}
```

Response JSON payload

```
Successfully created entity - OAuthIdentityDomain, detail - OAuth Identity Domain :: Name
- OFSLL_OAUTH_DOMAIN,
Id - 37b278eb5e894085ab1656b9641cca1a, Description - OFSLL_OAUTH_DOMAIN,
TrustStore Identifiers - [OFSLL_OAUTH_DOMAIN],
Identity Provider - OUD_LDAP, TokenSettings - [{"tokenType":"ACCESS_TOKEN",
"tokenExpiry":3600,
"lifeCycleEnabled":true,
"refreshTokenEnabled":true,
"refreshTokenExpiry":86400,
"refreshTokenLifeCycleEnabled":true
```

```
},
{
  "tokenType":"AUTHZ_CODE",
  "tokenExpiry":3600,
  "lifeCycleEnabled":true,
  "refreshTokenEnabled":true,
  "refreshTokenExpiry":86400,
  "refreshTokenLifeCycleEnabled":true
},
{
  "tokenType":"SSO_LINK_TOKEN",
  "tokenExpiry":3600,
  "lifeCycleEnabled":true,
  "refreshTokenEnabled":true,
  "refreshTokenExpiry":86400,
  "refreshTokenLifeCycleEnabled":false}},
ConsentPageURL - oam/pages/consent.jsp,
ErrorPageURL - /oam/pages/error.jsp,
CustomAttrs - Attribute of user in IDStore to store the encrypted secretkey for TOTP
```

3.3 **Resource Server Creation**

Resource Server Name: OFSLL_OAUTH_SERVER

Identity Domain: OFSLL_OAUTH_DOMAIN

Request JSON payload

```
{
  "name":"OFSLL_OAUTH_SERVER",
  "description":"OFSLL_OAUTH_SERVER",
  "scopes":[{
    "scopeName":"OFSLL_REST_ALL",
    "description":"ALLOW_ALL"
  },
  {
    "scopeName":"OFSLL_REST_NONE",
    "description":"ALLOW_NONE"
  }],
  "tokenAttributes":
  [{"attrName":"sessionId",
    "attrValue":"$session.id",
    "attrType":"DYNAMIC"
  },
  {
    "attrName":"resSrvAttr",
    "attrValue":"RESOURCECONST",
    "attrType":"STATIC"
  }],
  "idDomain":"OFSLL_OAUTH_DOMAIN",
  "audienceClaim":{"subjects":["OFSLL_B2B_OAUTH_CLIENT"]}
}
```

Response JSON payload

```
Sucessfully created entity - OAuthResourceServer, detail -
IdentityDomain="OFSLL_OAUTH_DOMAIN",
Name="OFSLL_OAUTH_SERVER", Description="OFSLL_OAUTH_SERVER",
resourceServerId="99a3e782-ce6d-467c-baec-df687fe326a6",
resourceServerNameSpacePrefix="OFSLL_OAUTH_SERVER.",
audienceClaim={
  "subjects":["OFSLL_B2B_OAUTH_CLIENT"]},
resServerType="CUSTOM_RESOURCE_SERVER",
Scopes=[{
  "scopeName":"OFSLL_REST_ALL",
  "description":"ALLOW_ALL"},
{
  "scopeName":"OFSLL_REST_NONE",
  "description":"ALLOW_NONE"},
{
  "scopeName":"DefaultScope",
  "description":"DefaultScope"}],
tokenAttributes=[{
  "attrName":"sessionId",
```

```
"attrValue": "$session.id",  
"attrType": DYNAMIC},  
{ "attrName": "resSrvAttr", "attrValue": "RESOURCECONST", "attrType": STATIC}}
```

3.4 **Client Creation**

Name: OFSLL_B2B_OAUTH_CLIENT

idDomain: OFSLL_OAUTH_DOMAIN

```
{  
  "attributes": [{  
    "attrName": "customeAttr1",  
    "attrValue": "CustomValue",  
    "attrType": "static"  
  }],  
  "secret": "welcome1",  
  "id": "OFSLL_B2B_OAUTH_CLIENT",  
  "scopes": [  
    "OFSLL_OAUTH_SERVER.OFSLL_REST_ALL",  
    "OFSLL_OAUTH_SERVER.OFSLL_REST_NONE"  
  ],  
  "clientType": "CONFIDENTIAL_CLIENT",  
  "idDomain": "OFSLL_OAUTH_DOMAIN",  
  "description": "Client Description",  
  "name": "OFSLL_B2B_OAUTH_CLIENT",  
  "grantTypes": [  
    "PASSWORD", "CLIENT_CREDENTIALS",  
    "JWT_BEARER", "REFRESH_TOKEN",  
    "AUTHORIZATION_CODE"  
  ],  
  "defaultScope": "OFSLL_OAUTH_SERVER.OFSLL_REST_ALL"  
}
```

Response JSON payload

```
Successfully created entity - OAuthClient, detail - OAuth Client - uid = 236936a6-ed77-4d6a-bcee-c0282554a1a0,
name = OFSLL_B2B_OAUTH_CLIENT, id = OFSLL_B2B_OAUTH_CLIENT,
identityDomain = OFSLL_OAUTH_DOMAIN,
description = Client Description, secret = welcome1, clientType =
CONFIDENTIAL_CLIENT,
grantTypes = [PASSWORD, CLIENT_CREDENTIALS, JWT_BEARER,
REFRESH_TOKEN, AUTHORIZATION_CODE],
attributes = [{
  "attrName":"customeAttr1",
  "attrValue":"CustomValue",
  "attrType":STATIC
},
{
  "attrName":"sessionId",
  "attrValue":"$session.id",
  "attrType":DYNAMIC
},
{
  "attrName":"resSrvAttr",
  "attrValue":"RESOURCECONST",
  "attrType":STATIC
}],
scopes = [OFSLL_OAUTH_SERVER.OFSLL_REST_ALL,
OFSLL_OAUTH_SERVER.OFSLL_REST_NONE],
defaultScope = OFSLL_OAUTH_SERVER.OFSLL_REST_ALL, redirectURIs = []
```

3.5 Getting Access Token

A client application which wants to obtain an access token from OAuth server can access OFSLL Authentication API which in turn accesses the OAM OAuth API and generates token. The authentication REST service OFSLL provides a wrapper around OAM OAuth API.

3.5.1 How OFSLL API works with access token?

1. Client calls OFSLL authentication API (OFSLL REST API) with required headers along with body and obtains the token as response.
2. OFSLL REST API validates the token and retrieves the user ID from access token.
3. If the token is valid, then provides access to the protected resource.

Note: To use OAM OAuth API, update the following OFSLL system parameters with valid values.

OFSLL System Parameter Name	Default Value	Actual Value Required to configure OAuth Feature
OAM_OAUTH_ENABLED_IND	SETME	Y

OFSLL System Parameter Name	Default Value	Actual Value Required to configure OAuth Feature
OAM_OAUTH_TOKEN_URL	SETME	http://<hostname>:<port>/oauth2/rest/token
OAM_OAUTH_TOKEN_VALID_URL	SETME	http://<hostname>:<port>/oauth2/rest/token/info?access_token=<AccessToken>

While client applications are allowed to access OAM OAuth REST API directly, it is recommended for clients to access OFSLL Authentication REST API for all token generation and token validation features.

Authentication Resource URL:

http://<<hostname>>:<<port>>/<<context_path>>/service/api/resources/auth/token

3.5.2 Access Token for CLIENT_CREDENTIALS grant type

Request JSON payload

```
{
  "AuthRequest": { "GrantType": "CLIENT_CREDENTIALS" }
}
```

Mandatory Request Headers

Headers	Expected Value
X-OAUTH-IDENTITY-DOMAIN-NAME	OFSLL_OAUTH_DOMAIN
Authorization	Bearer <Base64encoded value of client credentials>

Response JSON payload

```
{
  "AuthResponse": {
    "Token":
      "eyJraWQiOiJPRINMTF9TU09fVEVTVF9ET01BSU4iLCJ4NXQiOiJjQldCa0pqV2JvdHRHczFmZFdIYzdtE0tMWsiLCJhbGciOiJSUzI1NiJ9.eyJpc3MiOiJodHRwOi8vbXVtMdBjaWUuaW4ub3JhY2xiLmNvbToxNDEwMC9vYXV0aDIiLCJhdWQiOiJleHAiOiE1NDU2NjYzMDIsImp0aSI6IjQ3WnF5Q1RNbHN1OE1yZnM3Mnlzc3ciLCJpYXQiOiE1NDU2NjI3MDIsInN1YiI6Ii9GU0xMU1VQUiIsIk9BVVRIX1RPS0VOlloiZXIKcmFXUWIPaUprWlidaaGRXeDBJaXdpZURWMElqb2Iza3c1VkrJNGJlaE1RakowY1c1eGQyZDRZMEZPUW5vdFFYWnpJaXdpWVd4bkIqb2IvE15TIRZaWZRLmV5SmxISEFpT2pFMU5EVTJOall6TURJc0ltcDBhU0k2SWpOSFZITjNOM1ZuTVhGQ2MwMXJOa1JlVGVGSWJsRWIMQ0pwVWVhRaU9qRTFORFUyTm9JM01ESXNjY2x1bWVhbnRlOUdVMHhNVTFWUVVpSXNjY2k5sYzNOcGlyNWZhV1FpT2IKM2QwSTVkbU12Y21WVSVFYQlpPRGxHZGxOeWVYVjV5QVDEtYjAxRINIUKVibGcyVGtsRmRYVXZWM2hZZWtKTIFUMDIJaXdpWkc5dFIXbHVJam9pWkdWbVIYVnNkQ0o5LkdKNIJOM19HUHNSQW0yTmGllbnQiOiJPRINMTF9CMkIjSkvVUX0NMSUVOVCIsInNjb3BlIjpbIj9GU0xMU1NTT19URVNUX1NFUIZFUI9CMkluQWNjb3VudERldGFpbHMlXSwiZG9tYWluljoiT0ZTTEU1NjY1RlF1RfRE9NQUIOlno0.YJJqj32KhCQfYfvUzAi5XAhbBL3s8E29AiJxQXGqMrkDU57YIFt5l36bHjUGFRTNXnHZ2UxP5bhZiZJcmivOTqs_j1laz0-TkHKCbHX2_-8NhelwEXKtYyqx8-9JKak1T8jsknXXKvO1FSv46siu2mBSxKul6rW7yeyC-TRiBBMj48h_u-dlSfLQc98X_5jxXQU8FpCV18Cb9l2HbGh9zuUmEP8G87leYQ7KTMBWbcQklbAVQVxbF0FVku2efjW2LlZ5XOJ_o_U-6GvudCCiQvbeVY3VbUI4hgXJGXCs5e3ubQ9wPF8fCd05MAStFd30KzpeKxRtGZDXjuDg3NSw",
    "Expires_in": 3600,
    "TokenType": "Bearer",
    "RefreshToken":
      "A79Gdo4lhOSCGmvmsRqWMg==~nHVr44Sa3QZgpl3eepIO8t323SjYEd3r6+IF24xBoct9SxybWy6PcpHDjSoLTOmW+OcqtfqTenEmoIWcyfh0cTGzcmYch1KMOMfCGns+M2KkwusUCCGWnyrhoUjevwhbK14U20B3E6orBVkZxhtmQLqkATXbvHS0tGqIKIqWrgUCjNlwsSDFgBCj4umfQMllt63pmgckNtwpQcOedxB6y2B9fl3BFY8j2D53xogK3coE40pl4f+SufnZ0WI+0DkCGHTfdaDzdcA2TwwA5VVjZaQ16A+nCx144uHaBj1eOOpiUaypL730tK2N8aES1CSDUIZPjbl3NIEY360VvLJRoxdRq2nL4SgS0wJ7XIdu39wuxoTGtjLBWHQsDEtc0eBbgFUma2q8ug29+67cl/9H6TWOEGF+T981H+7JQTckSrma7gtyMr7MKy0QtmxR4Ns6w=",
    "Result": { "Status": "SUCCESS", "StatusDetails": "Token Generated Successfully"
  }
}
```

3.5.4 Access Token for JWT BEARER grant type

This is the grant type is to achieve the seamless SSO between the different mixes of application. This grant type provides facility to link the mainstream application SSO session with OAuth token.

When the SSO session is generated, JWT User token also generated. The generated JWT user token has the SSO "session_id" as part of its claims. The consumer client application must call OFSLL Authentication API with JWT_BEARER token grant type to get access token to access the protected resource.

Note: The rules of SSO session are applied to the OAuth Access token.

Sample Request JSON

```
{
  "AuthRequest": {
    "Assertion": "eyJraWQiOiJkZWZhdWx0IiwieDV0IjoiYkw5VDI4bHhMQjJ0cW5xd2d4Y0FOQnotQXZzliwiYWxnIjoiUIMyNTYifQ.eyJleHAiOiJlNDU2NDQ0NTEsImp0aSI6IIRUS0lsSDdWR1VyWVhVbHdyZ2IuOWciLCJpYXQiOiJlNDU2NDQ0NTEsInN1Yil6Ik9GU0xMU1VQUiilsInNlc3Npb25faWQiOiJCeW90c2h6LzR3K2hhekVHcnNqWnJBPT1-bVN2eU5DaEtLa29xTk5tcUlyQkUvM3IOUTBiNENYVWITQktqWXdlY1JlazdQYXBzajN6a1pkbnJqYWVlOURPbWVlRTFBSURocG1QN0tTd1hKUDVFdzRpbmZHTes1VGlsYldDYUJWLOVmVkIxQIM5K2FaY1oxQ25oUTV0VVF3U3ciLCJkb21haW4iOiJkZWZhdWx0In0.NfLQHdh219p2NjzR44q9xgrQ9m6ky1paJ2GpHf2Re8tXjKyjZNFxjYu9Tb78RoX3-xlsXOdmrRJBmW0_z1vy-0NrnHkU2fpBrBVdauqsXadCCKFFnkYy8AAJZg2WXYUNmaAcZWPT9z3svcQBHq9OQMdrkUvq3WbD91LbS5MA5pOkU8LofMn2j8nisoLRaQ904CXillKPI8jWILXtai-8hHgZ5t62Z-B-Yis3m1xiWPJ7zEctMRoule5pyFRYHxwudBht3Y9M04uDEQaIAk3d0uiVDup4eFJBt-Vt1Jt42f5hX28GyQQNu13s-rVAraXYxHGx4hzNZZTIw9EUdDPuEg",
    "GrantType": "JWT_BEARER"
  }
}
```

Mandatory Request Headers

Headers	Expected Value
X-OAUTH-IDENTITY-DOMAIN-NAME	OFSLL_OAUTH_DOMAIN
Authorization	Bearer <Base64encoded value of client credentials>

Sample JSON Response

```
{
  "AuthResponse": {
    "Token":
      "eyJraWQiOiJPRINMTF9TU09fVEVTVF9ET01BSU4iLCJ4NXQiOiJjQldCa0pqV2JVdHRHczFmZFdIYzdteE0tMWsiLCJhbGciOiJSUzI1NiJ9.eyJpc3MiOiJodHRwOi8vbXVtMdBjaWUuaW4ub3JhY2xiLmNvbToxNDEwMC9vYXV0aDIiLCJhdWQiOiJldLCJleHAiOiE1NDU2NzQ1NzQsImp0aSI6InZTcE1LVzIheVF2VngxZU5KRUZ1ZVEiLCJpYXQiOiE1NDU2NzA5NzQsInN1YiI6Ik9GU0xMU1VQUiIsImN1c3RvbWVbdHRyMSI6IkN1c3RvbVZhbHVliiwic2Vzc2l2bklkIjoiMzExY2MxOWUtODhkZi00ZDdhLTg5YzQtODFjNmRiMDE5ZDUzfEZqN2NTcTzQcWlQUVZSNXU4TTNIRVpqbUJEZHVKQm9Hek5sMDBNQkFnS009IiwicmVzU3J2QXR0cil6IlJFU09VUkNFQ09OU1QiLCJjbGllbnQiOiJPRINMTF9CMkIjFjSkVUX0NMSUVOVCIsInNjb3BlIjpbIk9GU0xMX1NTT19URVNUX1NFUIZFUI9CMkluQWNjb3VudERldGFpbHMlXSwiZG9tYWluljoiT0ZTTExfU1NPX1RFU1RfRE9NQUIOI0.guHeG7eZilGWpJhMWllpH4K3IbGtM8buuwJPIIk6EengFTeicbfpd0E3qZwp8SYRFuzvw4FX7wCSbbBt2WM9G4L6uM0NTvZpSTcwUeOljuysMiCmPzQ-8cSijpM4G55Fb35laulC7eiCNdMtKoH34A2IScX7lamjlpC0u4SV4V-8cB4VviGtrd_sXIqOfgSadpjrXQatuaRID1at4aNoAGv1Da7E4xrMzy9m41cxHtujSNU2aDxG73-b2qOJiNZbvf-zlaaa2pu1TOOr1ynZDvbe3STsZkAKO1VKFczHmYw8Tppqovc6MNd0TPyhNFUJHDBsPH-nKV_nkFQHyy0_jw",
    "Expires_in": 3600,
    "TokenType": "Bearer",
    "Result": {
      "Status": "SUCCESS",
      "StatusDetails":
        "Token Generated Successfully"
    }
  }
}
```

3.5.5 Access Token for REFRESH_TOKEN grant type

Sample JSON Request

```
{
  "AuthRequest": {
    "GrantType": "REFRESH_TOKEN",
  }
}
```



```

"RefreshToken": "Mtn+NHd1zyCwelBiAcfy3w==~ALGXo2ieTTwvkn8tk5HOi5L4FN900wWW
PCpZgmHMaDX8PgbwQRU/HH+TGtamC5YU4Pqp3ZRM7va/Tzlc8EHEQxsfY0e3pBGI/
KD2CkfeL0MOPfqstzE1Lq7bkfQxKj9jZNI FvkEHnFVLP Gh2XR0xSjlbTK3EeP5Eyoxf7aaT
NU79CBM3Ws5mrkGKhj2l2o2LFTTAbw/thEQLK9A7vXIGj8cSAuuFsAREfAy/skriZZfMQK
xKY3Ewh1Cm1coBmG5aT3DFe9LxgPuSmght1kLKqyDhjK0nOokUO1XsHwPh0cWXge
WBIKXVzd3o06h0eJ/SaTvFIUEEictWAgyMGDISVVDfriZnHVQAO5TQsXFskoTPtrjA14C
PWoibLgYRdiPIObcj32c+CMmywicLK/+v4+xx44IBu92fvW/HzPxAWDJ3IPyM2AvgP1SEd
8LiHcrwyh8rR8JTkJpWkNMmxW4S0p/6g//dUiWD9QEGCwUnBINweUpRoA76k2Gbgq+
NJ18ohF8epeUy+ftvjjeefNuoU41KI3mprlrReHuG00I5GJScwC3w1zuEUgJsBZc4QN9Fju
Js/l/aMTPKRT683x33WLQt3NkHOCW+g8XyeiSVaic/k2DtDyPWpnCuzNbZzqvjgx93Zga
o0vBMIDB9+hZzfNMybOVrllcdvwegAubETgplEdmcu/2BpclXjLJgJ8hMd9xE5xrv/7NDg+
sUjpMkriN8OnEq0rzUlwLMgC2aI05PMgkGA2RudhM/4VUXzKEK4ItGoR2CEMPHTGrKp
uFvdvZoyjh4JMfwTgH6F1KUV4AMgE+vPzQENNgSxxseewavmJbjB8OjGvuC/G00hdY3
YYWvz2CtEWA08261unvuroUceNfGbTSK/Z3x4I8iuCfg8n7ZLyc9a3m8WTVsoodkLnZS
8mYU4dFExXvsS35gnuyzZvXR0BkJ44+2VpfdmB55qOPUziZK2UKZJ1Hg5jqTCwSMDg
+9qWVhbnpdzFtjqETF0edUI7F0QkotXkEHUTMSOTgR4d47paOJQDISW3Lr1n8+7YNA/r
KIUTTIoP6I7lcl/rZ2BDazmVgf3axZ/Oo+xbtJCAbrfGqNJAJIOcBf/hJmktSXY+osj4CairKGh
cteFPziEOeo5+sbAsTthAadaLYcPs6/4mNoK7yvlyLoxulOEY7CSXZSaPOotsV49LX3fEjH
gkvkDU3dhcPPc9DmlyDDySKO18K7wgaPnJtCSu1fq2AwVwDmwRd6BZsAlsn3dHqGDu
+XTgr7dt7ag3JyxmtuZQRGJiPbJp5gExgnS6JyjIF2co75kXvWoHm3O/p8="
}
}

```

Mandatory Request Headers

Headers	Expected Value
X-OAUTH-IDENTITY-DOMAIN-NAME	OFSLL_OAUTH_DOMAIN
Authorization	Bearer <Base64encoded value of client credentials>

Sample JSON Response

```
{
  "AuthResponse": {
    "Token":
      "eyJraWZiOiJPRINMTF9TU09fVEVTVF9ET01BSU4iLCJ4NXQiOiJjQldCa0pqV2Jv dHRHczFmZFdIYzdtE0tMWsiLCJhbGciOiJSUzI1NiJ9.eyJpc3MiOiJodHRwOi8vbXVtMDBjaWUuaW4ub3JhY2xiLmNvbToxNDEwMC9vYXV0aDIiLCJhdWQiOiJleHAiOiE1NDU2NjI4MTgslmp0aSI6InZGZ082eUIFb0k2X0xoZ3czcmczTUEiLCJpYXQiOiE1NDU2NjI3NTgslmN1YiI6Ik9GU0xMU1VQUiIsIk9BVVRlX1RPS0V0IjoiZXIKcmFXUWIPaUprWl daaGRXeDBJaXdpZURWMElqb2Iza3c1VkrJNGJlaE1RakowY1c1eGQyZDRZMEZPUW5vdFFY WnpJaXdpWVd4bkIqb2IVbE15TIRZaWZRLmV5SmxISEFpT2pFMU5EVTJORFF4T0RBc 0ltcDBhU0k2SWpGSIZWbGlibXBFWkVwVWdWQk5TbJvUkVOTmlzY2IMQ0pwWVhRa U9qRTFORFUyTkRBMU9EQXNjBk4xWWIJNklrOUdVMHhNVTFWUVVpSXNJbk5sYzN OcGlyNWZhV1FpT2IjJemNWRXplbVZ2TldWWIEzVnJhbK5qUW1aRWFfSIJQVDEtU2sx amJHMW1TMW8wUmXKUK9GTnJSVXBPVXpCdGR6MDIJAXdpWk5dFIXbHVJam9pW kdWbVIYVnNkQ0o5LpHQWdTZGR5S0szRGpIMIZsTUImbzMxTTV0cFBpaXluUVpGY2R ibEFiV2xhektPWVfZb2hqbzdrODQxQm9SMWUtWXZWLXpWbk5hamhCYy1CbzAwZINs VDdsVmNmRXA2ekxUdENHRnc1MkNZRXpfOHpYdjc1YkF2Q2FWeGRvZnlUaGFhdFox cVF5Qm5TLUJ1MGdKNjJSOThkaDFwY1FOSmFhWDZ1RkxkSGVoVUIDY1pNLUJwbU drbi01TXhHcm5fQnl0T2oxc0JnRjZ1SWY0N3d6NUINOU4zODdHQ29WZDBPR3c0QXIm VVk2T2FGSINOS1hYbnpsYUVDTKktMEJmQXhFQkISX1oxVE1wZEdSOHkwaHltMWNT SzRGYkKjYnQxZnlhYmdLMG1SQ0tFVHdMWFJrcnFMcmkxTnVtbjNKeFhmZVBsWTVJT m1ndDA2U1BVaFpYUEU2ZyIsImN1c3RvbWVbDHRyMSI6IkN1c3RvbVZhbHVliiwic2Vzc 2lhbklkIjoiQ09PS0IFX0JBU0VEliwicmVzU3J2QXR0cil6IiJFU09VUkNFQ09OU1QiLCJjbG llbnQiOiJPRINMTF9CMkKJfSkVUX0NMSUVOVCIsInNjb3BlIjpbIk9GU0xMX1NTT19URVN UX1NFUIZFUI9CMklUQWNjb3VudERldGFpbHMlXSwiZG9tYWluljoiT0ZTTExfU1NPX1R FU1RfRE9NQUIOIIn0.BKsWO1yBEmc_f0jCdG16DxzkTkkN805VmYIBbyMmmMqzNiNsync orlzHAZ0RHTDqNLjKdq--wxzTNQK4PRM9ChBeHKBCU5dzHD64ddbscyt0YxpdPnF0grMZHiploNC_- nZxyZRbLI5aQeGPXOZ4qtPEZ1ggBkgoXXa16eJ2JLZbYotcvPbLcbkfHpMCz-wOzi_o0t30KG9T1931NyMaCvYp4O-ZODTneHc9-c7cJaj2zVhkOFej796TTrEHV4jv7p2OTsawkm8vSYmRBv5K1J8M_a1PgEluqc4kS6d0op UAJOKT6C356OMdEpeO_zkXGyfodUFKojdG3PWHXG007ww",
    "Expires_in": 3600,
    "TokenType": "Bearer",
    "Result": {
      "Status": "SUCCESS",
      "StatusDetails":
        "Token Generated Successfully"
    }
  }
}
```

3.5.6 How to get access token through Basic Authentication

Mandatory CSF Key

CSF Map name	Key
ofsl.int.common	ofsl.jwt.JwtSecretKey

The Ofssl.jwt.JwtSecretKey refers to the secret that must be associated at the time of token generation. This is the key would be used to validate the token.

Mandatory Request headers

Headers	Expected Value
Content-Type	application/json
Authorization	Bearer <Base64encoded value of resource owner credentials>

Request JSON payload

```
{ "AuthRequest": {
    "GrantType" : "PASSWORD"
}}
```

Response JSON payload

```
{
  "AuthResponse": {
    "Token":
    "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJPRINMTFNVUFlpLCJpc3MiOiJPRINMTF9SRVNUX0FQSSIsImV4cCI6MTU0NTY3MjEyMSwiaWF0IjoxNTQ1NjcxMjlfQ.
    tM5JB4h6VjJ59dXteth9ZY4b0ayz9XpT5J2jYu8zIHr4uvkKan-
    yvRgU1OSXhovdyw8zMI_ajqDLdESc_Izv3w",
    "Result": {
      "Status": "SUCCESS",
      "StatusDetails": "Token Generated Successfully"
    }
  }
}
```

3.5.7 How to access the REST API using the access token

In Every OFSLL REST API request, please send the following headers with correct values

Mandatory Request headers

Headers	Expected Value
ofssl_access_token	The valid access token received from any of the above mention flow
X-OAUTH-IDENTITY-DOMAIN-NAME	Valid OAuth Identity domain associated with access token

3.6 Embedding External Application within OFSLL

As part of subsequent releases of OFSLL, to embed external application within OFSLL base application, we would provide one external link each under origination, servicing and collection modules. The associated menu links can be enabled through access screens.

Response JSON payload

```
{
  "AuthResponse": {
    "Token":
      "eyJraWQiOiJPRINMTF9TU09fVEVTVF9ET01BSU4iLCJ4NXQiOiJjQldCa0pqV2JVdHRHczFmZFdIYzdtE0tMWsiLCJhbGciOiJSUzI1NiJ9.eyJpc3MiOiJodHRwOi8vbXVtMdBjaWUuaW4ub3JhY2xiLmNvbToxNDEwMC9vYXV0aDIiLCJhdWQiOiJldiJleHAiOiE1NDU2NjI4MTgslmp0aSI6InZGZ082eUJFb0k2X0xoZ3czcmczTUEiLCJpYXQiOiE1NDU2NjI3NTgslmN1YiI6Ik9GU0xMU1VQUiIsIk9BVVRlX1RPS0V0IjojZXIKcmFXUWIPaUprWlidaaGRXeDBJaXdpZURWMElqb2Iza3c1VkrJNGJlaE1RakowY1c1eGQyZDRZMEZPUW5vdFFYWnpJaXdpWVd4bkIqb2IvE15TIRZaWZRLmV5SmxISEFpT2pFMU5EVTJORFF4TORBc0ltcDBhU0k2SWpGSIZWbGlibXBFWkVwVWdWQk5TbJvUkVOTmlzY2IMQ0pwVWhRaU9qRTFORFUyTkRBMU9EQXNjbk4xWWIjNklrOUdVMHhNVTFWUVVpSXNJbk5sYzNOcGlyNWZhV1FpT2IjJemNWRXplbVZ2TldWWIEzVnJhbk5qUW1aRWFfSIUQVDEtU2sXamJHMW1TMW8wUmXKUK9GTnJSVXBVXpCdGR6MDIJJaXdpWk5dFIXbHVJam9pWkdWbVIYVnNkQ0o5LlpHQWdTZGR5S0szRGpIMIZsTUImbzMxTTV0cFBpaXluUVpGY2RibEFiV2xhektPWVfZb2hqbzdrODQxQm9SMWUtWXZWLXpWbk5hamhCYy1CbzAwZINsVDdsVmNmRXA2ekxUdENHRnc1MkNZRXpfOHpYdjc1YkF2Q2FWeGRvZnlUaGFhdFoxcVf5Qm5TLUJ1MGdKNjJSOThkaDFwY1FOSmFhWDZ1RkxkSGVoVUIDY1pNLUJwbUdrbi01TXhHcm5fQnl0T2oxc0JnRjZ1SWY0N3d6NUINOU4zODdHQ29WZDBPR3c0QXImVVK2T2FGSINOS1hYbnpsYUVDTKktMEJmQXhFQkISX1oxVE1wZEdSOHkwaHltMWNTSzRGYkKjYnQxZnlhYmdLMG1SQ0tFVHdMWFJrcnFMcmkxTnVtbjNKeFhmZVBsWTVJTm1ndDA2U1BVaFpYUEU2ZyIsImN1c3RvbWVbdHRyMSI6IkN1c3RvbVZhbHVliiwic2Vzc2lvaWV0eU90PS0IFX0JBU0VEliwicmVzU3J2QXR0ciI6IiJFU09VUkNFQ09OU1QiLCJjbGllbnQiOiJPRINMTF9CMkKJfSkVUX0NMSUVOVCIsInNjb3BlIjpbIk9GU0xMX1NTT19URVNUX1NFUJZFU09CMkluQWNjb3VudERldGFpbHMlXSwiZG9tYWwuljoiT0ZTTExfU1NPX1RfU1RfRE9NQUIOIIn0.BKsWO1yBEmc_f0jCdG16DxzkTkkN805VmYIBbyMmmMqzNiNsyncorlzHAZ0RHTDqNLjKdq--wxzTNQK4PRM9ChBeHKBCU5dzHD64ddbscyt0YxpdPnF0grMZHiploNC_-nZxyZRbLI5aQeGPXOZ4qtPEZ1ggBkgoXXa16eJ2JLZbYotcvPbLcbkfHpMCz-wOzi_o0t30KG9T1931NyMaCvYp4O-ZODTneHc9-c7cJaj2zVhkOFej796TTrEHV4jv7p2OTsawkm8vSYmRBv5K1J8M_a1PgEluqc4kS6d0opUAJOKT6C356OMdEpeO_zkXGyfodUFKojdG3PWHXG007ww", "Expires_in": 3600,
    "TokenType": "Bearer",
    "Result": {
      "Status": "SUCCESS",
      "StatusDetails": "Token Generated Successfully"
    }
  }
}
```

ORACLE®

Financial Services

OAuth2 based Web Services Access Authentication
Oracle Financial Services Lending and Leasing
December 2020

Oracle Financial Services Software Limited
Oracle Park
Off Western Express Highway
Goregaon (East)
Mumbai, Maharashtra 400 063
India

Worldwide Inquiries:
Phone: +91 22 6718 3000
Fax: +91 22 6718 3001
<https://www.oracle.com/industries/financial-services/index.html>

Copyright © 1998, 2020, Oracle and/or its affiliates. All rights reserved.

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

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are “commercial computer software” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

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

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or recompilation of this software, unless required by law for interoperability, is prohibited. The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

This software or hardware 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. 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.