

Oracle Banking APIs
UK Open Banking Configuration Guide
Release 18.3.0.0.0

Part No. F12055-01

March 2019

ORACLE®

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

www.oracle.com/financialservices/

Copyright © 2018, 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 decompilation 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.

Table of Contents

1. Preface.....	4
1.1 Intended Audience	4
1.2 Documentation Accessibility	4
1.3 Access to Oracle Support	4
1.4 Structure.....	4
1.5 Related Information Sources.....	4
2. Objective and Scope	5
3. Technology Stack.....	6
4. Pre-requisites.....	7
5. Headers Configuration.....	8
6. Properties	9
7. SAML	10
7.1 SAML Setup	10
7.2 SAML Integration	10
8. OAuth Configuration	11
8.1 UI configuration	11
8.2 Weblogic configuration.....	11
9. Extensibility and Code Conventions	14

1. Preface

1.1 Intended Audience

This document is intended for the following audience:

- Customers
- Partners

1.2 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=accandid=docacc>.

1.3 Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit

<http://www.oracle.com/pls/topic/lookup?ctx=accandid=info> or visit

<http://www.oracle.com/pls/topic/lookup?ctx=accandid=trs> if you are hearing impaired.

1.4 Structure

This manual is organized into the following categories:

Preface gives information on the intended audience. It also describes the overall structure of the User Manual.

The subsequent chapters describes following details:

- Purpose
- Configuration / Installation.

1.5 Related Information Sources

For more information on Oracle Banking APIs Release 18.3.0.0.0, refer to the following documents:

- Oracle Banking APIs Licensing Guide

2. Objective and Scope

Background

Open Banking Configuration Document provides the various configurations required to enable UK Open Banking in OBAPI

Scope

- Headers Configuration
- Properties
- SAML Integration
- OAuth Configuration
- Code Convention and Extensibility

3. Technology Stack

Software	Version
Java	Java JDK or JRE version 8
OBDX/OBAPI	18.3.0.1.0
OAuth	OBAPI Internal OAuth

Abbreviations

OOTB	Out of the Box
TPP	Third Party Providers
ASPSP	Account Servicing Payment Service Provider
SAML	Security Assertion Markup Language

4. Pre-requisites

- Java JDK or JRE version 7 or higher must be installed. For installation of Java please refer [installation guide](#).
- OAuth Setup
- Weblogic Server with SAML Assertion capability

5. Headers Configuration

There are two types of headers configuration available for UK Open Banking.

- System Headers (i.e. Mandatory Headers and its respective value validation)
- Configuration Headers (i.e. Mandatory Headers).

Below are the configuration steps and Out of the box header already configured in the system.

System Headers:- As of now in OOTB one header has been added as mandatory “x-fapi-financial-id” with value as “491308330388688” (This is a random value and can be changed. This value is issued by OBIE and corresponds to the Organization Id of the ASPSP in the Open Banking Directory). This value needs to be configured by Bank or ASPSP. This header needs to be sent by the TPP to the ASPSP mandatorily with the same value. Both Header name and Header value are validated for System Headers.

For configuring more system headers, below script is to be executed in the OB API Admin schema.

```
Insert into DIGX_FW_CONFIG_ALL_B (PROP_ID, CATEGORY_ID, PROP_VALUE,
FACTORY_SHIPPED_FLAG, PROP_COMMENTS, SUMMARY_TEXT, CREATED_BY,
CREATION_DATE, LAST_UPDATED_BY, LAST_UPDATED_DATE, OBJECT_STATUS,
OBJECT_VERSION_NUMBER) values ('%%HEADER
NAME%%','OpenbankingSystemHeaders','%%HEADERVALUE%%','N',null,'Open
Banking','ofssuser',sysdate,'ofssuser',sysdate,'Y',1);
```

Below Query is used to check the System Headers in the system

```
select * from digx_fw_config_all_b where category_id = 'OpenbankingSystemHeaders';
```

Configuration Headers :- As of now in OOTB one header has been added as mandatory - “x-fapi-interaction-id”. This header is required to be sent by the TPP to the ASPSP mandatorily with any value.

Only header name is validated in case of Configuration Headers.

For configuring more config headers, below script is to be executed in the OB DX/OB API Admin schema.

```
Insert into DIGX_FW_CONFIG_ALL_B (PROP_ID, CATEGORY_ID, PROP_VALUE,
FACTORY_SHIPPED_FLAG, PROP_COMMENTS, SUMMARY_TEXT, CREATED_BY,
CREATION_DATE, LAST_UPDATED_BY, LAST_UPDATED_DATE, OBJECT_STATUS,
OBJECT_VERSION_NUMBER) values ('%%HEADER NAME%%','
OpenbankingConfigHeaders',null,'N',null,'Open
Banking','ofssuser',sysdate,'ofssuser',sysdate,'Y',1);
```

Below Query is used to check the System Headers in the system

```
select * from digx_fw_config_all_b where category_id = 'OpenbankingConfigHeaders';
```

6. Properties

Below are the properties required to be updated in the UK Open Banking. Please find the below properties, its purpose and OOTB values.

Table:- DIGX_FW_CONFIG_ALL_B

Category-Id :- OpenBankingConfig

Property Id	Property Value (Out of the Box)	Purpose
CONSENT_EXPIRYDAYS	90	This value is used to check if expiry date send by TPP for the Account Access Consent is not more than 90 days and if it is more than 90 days then ASPSP will reject this value
CONSENT_HANDLER	com.ofss.digx.app.openbanking.consent.handler.uk.UKConsentHandler	Handler defines the Region specific behavior of the Open Banking framework. By default UK Consent Handler is used for UK Open Banking compatibility

Token Settings

Category-Id :- SecurityConstants

Property Id	Property Value	Purpose
SIGNER	MAC/no row – MAC Signer X509RS256 – x509 signed token with RS256 algorithm X509PS256 - x509 signed token with PS256 algorithm	The algorithm used to generate JWT token.

7. SAML

7.1 SAML Setup

SAML Setup is required for propagating User Identification for account selection as part of consent authorization. Follow the 7th section of the document available at below location for SAML setup:

[Click Here to download SAML Setup](#)

7.2 SAML Integration

SAML Integration is required for asserting User Identification for account selection as part of consent authorization. Steps to be followed for SAML Integration are as below.

URL for SAML Account Rest should be as :- `http://<host>:<port>/ob/saml/accounts`

One default Internal Touch Point configuration will be required to handle Access to FETCH and POST Accounts through SAML.

Create a new TouchPoint for SAML services Access and configure in the web.xml of **obdx.app.rest.idm.ear** for the URL **“ob/saml/accounts”** as **“init-param :- obdx.saml.accesspoint”**. So through Role Transaction Mapping of the newly created touchpoint, the access would be provided for the SAML services of Open Banking FETCH and POST account.

As part of User Onboarding in OB API, the created touchpoint needs to be associated to the user being onboarded.

8. OAuth Configuration

8.1 UI configuration

OAuth Identity Domain Maintenance will require below maintenance to configure UI Component for Authorizing consent.

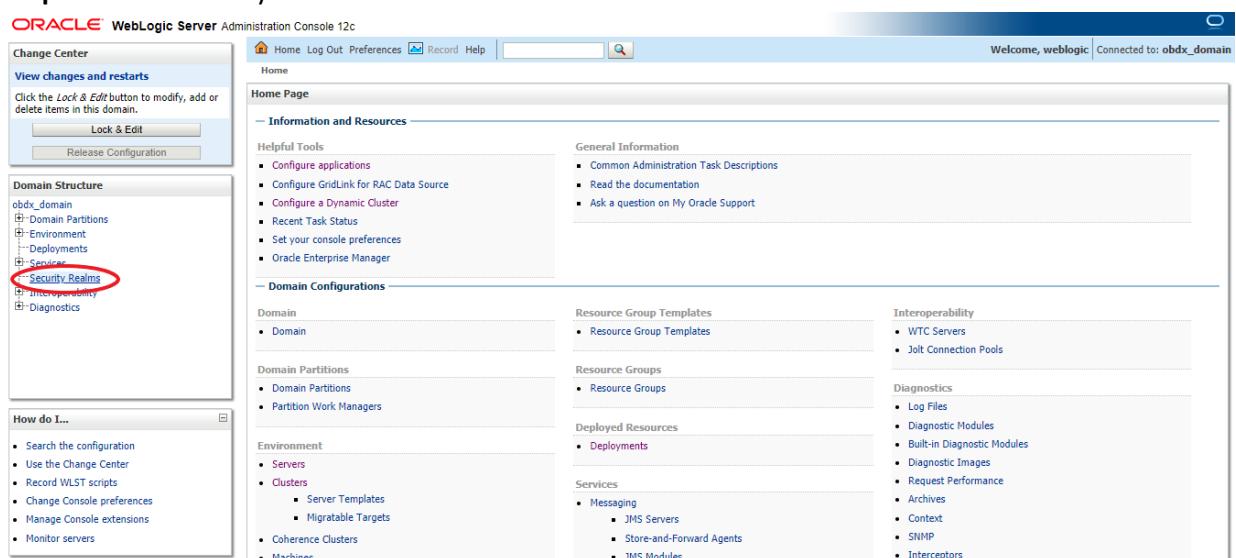
The value of Consent Page URL (Menu -> OAuth -> Identity Domain Maintenance) is configured as `http://host:port?homeComponent=authorize-consent&homeModule=open-banking&applicationType=digx-auth`.

8.2 Weblogic configuration

OAuth Maintenance will require below maintenance in weblogic to configure an URL.

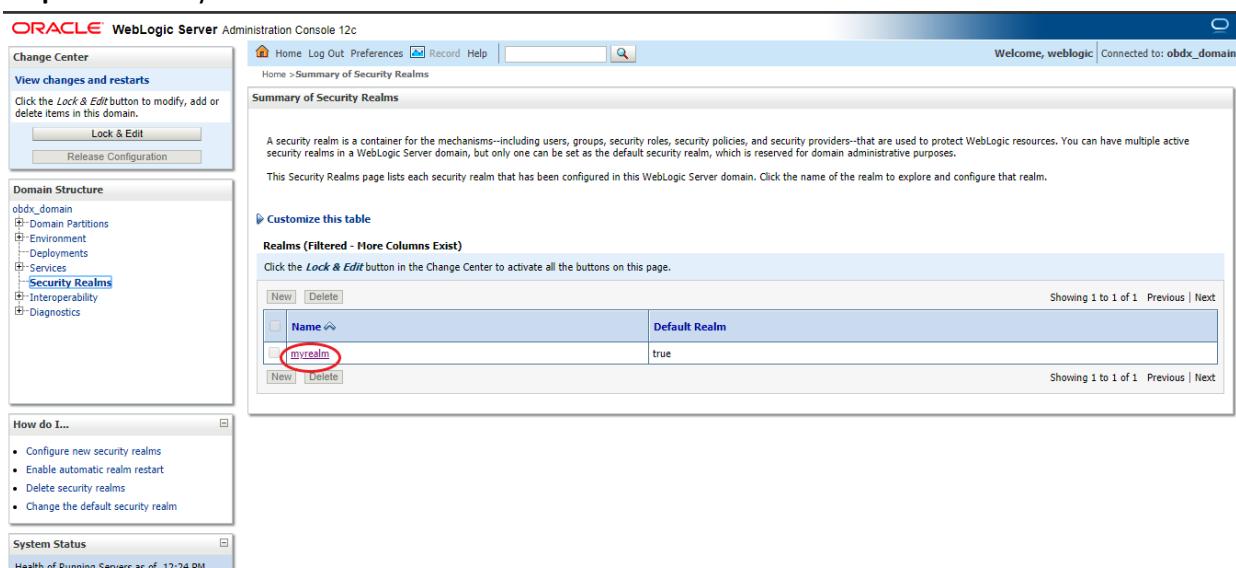
Step 1: Login to weblogic

Step 2: Go to Security Realms



The screenshot shows the Oracle WebLogic Server Administration Console 12c interface. The left sidebar has a 'Domain Structure' tree with 'Security Realms' highlighted and circled in red. The main content area displays various navigation links and configuration sections for the domain.

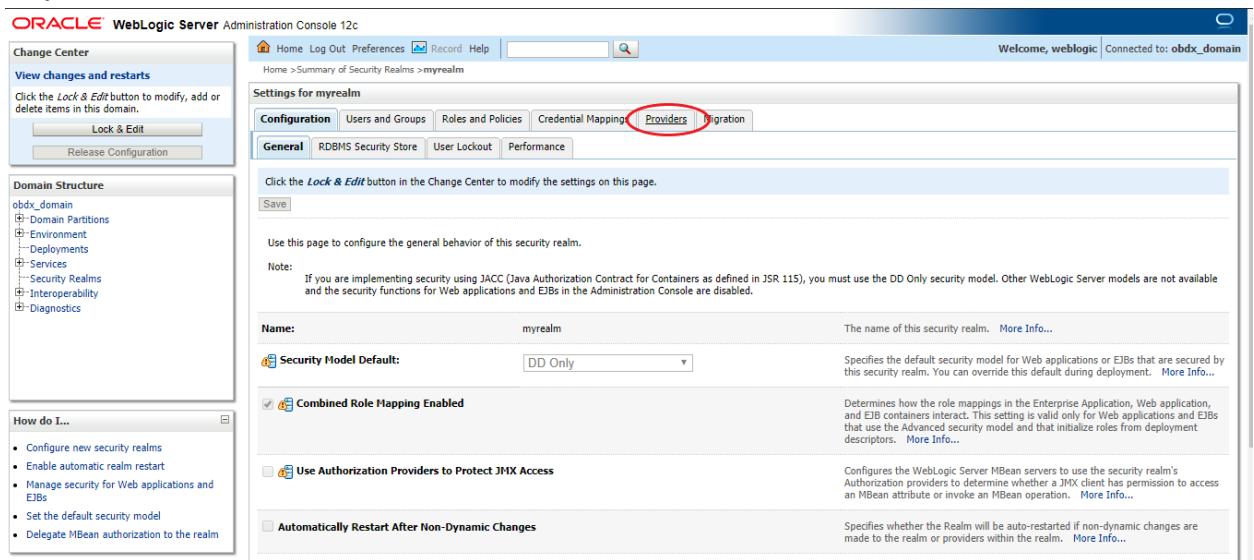
Step 3: Go to myrealm



The screenshot shows the 'Summary of Security Realms' page. It displays a table with one row for 'myrealm'. The 'Name' column shows 'myrealm' and the 'Default Realm' column shows 'true'. The 'myrealm' entry in the table is circled in red.

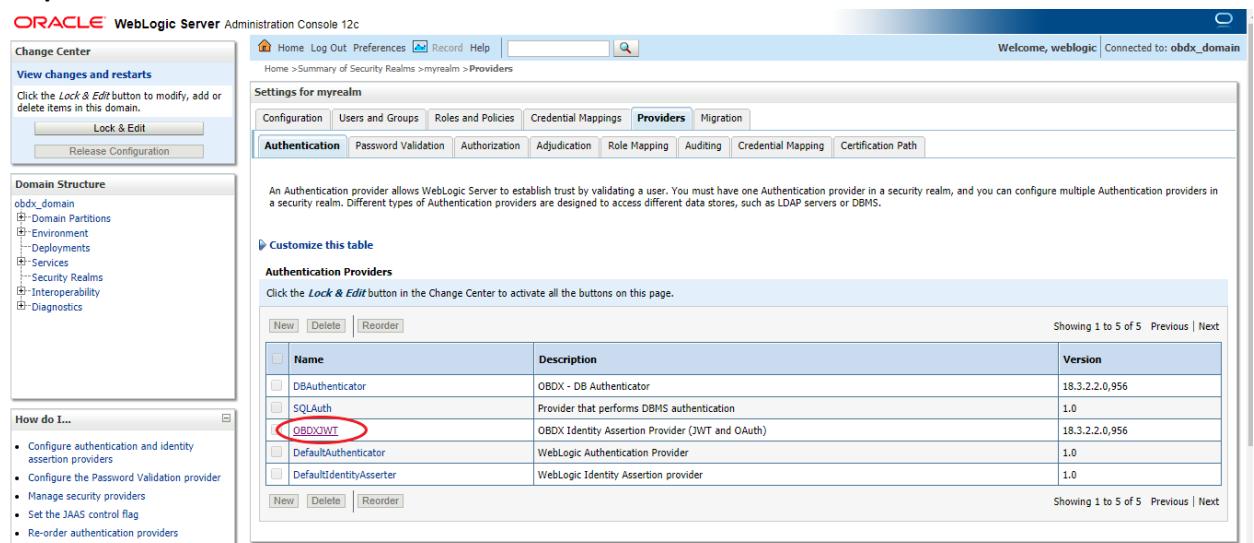
Name	Default Realm
myrealm	true

Step 4: Go to Providers



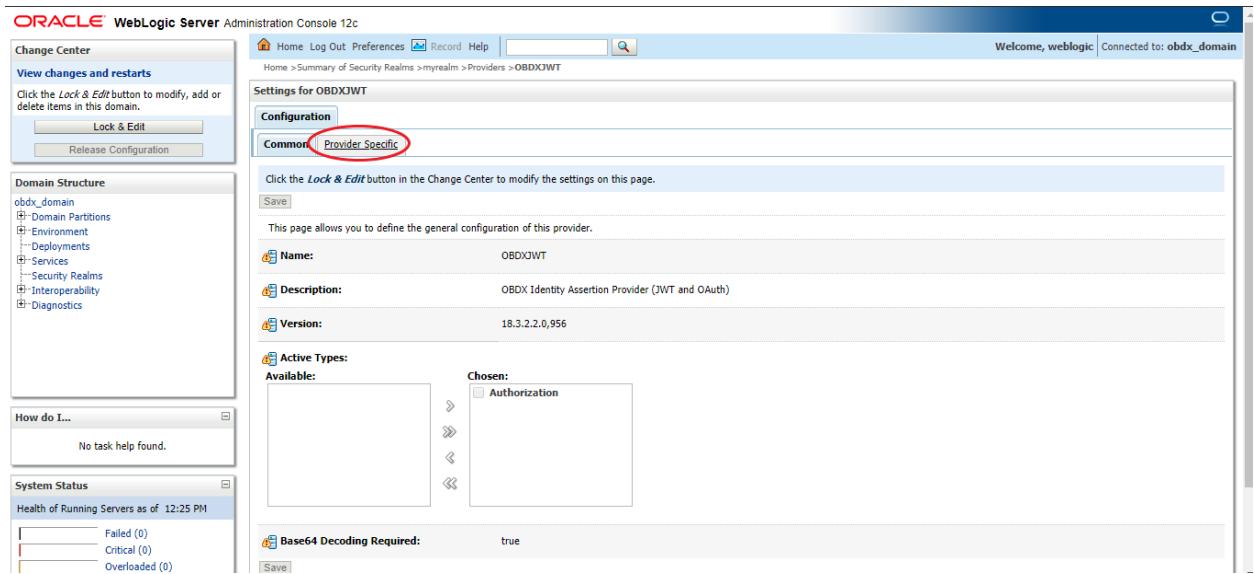
The screenshot shows the Oracle WebLogic Server Administration Console. The left sidebar shows a domain structure for 'obdx_domain' with various partitions, environments, and services. The main content area is titled 'Settings for myrealm' and shows the 'Providers' tab selected. The 'Providers' tab is highlighted with a red circle. The configuration includes a 'Security Model Default' set to 'DD Only', 'Combined Role Mapping Enabled' (checked), 'Use Authorization Providers to Protect JMX Access' (unchecked), and 'Automatically Restart After Non-Dynamic Changes' (unchecked). The 'Name' field is set to 'myrealm'.

Step 5: Go to OBDXJWT



The screenshot shows the Oracle WebLogic Server Administration Console. The left sidebar shows a domain structure for 'obdx_domain'. The main content area is titled 'Settings for myrealm' and shows the 'Providers' tab selected. The 'Providers' tab is highlighted with a red circle. The configuration includes an 'Authentication' provider table. The table shows five rows: 'DBAuthenticator' (Description: 'OBDX - DB Authenticator', Version: '18.3.2.2.0,956'), 'SQLAuth' (Description: 'Provider that performs DBMS authentication', Version: '1.0'), 'OBDXJWT' (Description: 'OBDX Identity Assertion Provider (JWT and OAuth)', Version: '18.3.2.2.0,956'), 'DefaultAuthenticator' (Description: 'WebLogic Authentication Provider', Version: '1.0'), and 'DefaultIdentityAsserter' (Description: 'WebLogic Identity Assertion provider', Version: '1.0'). The 'OBDXJWT' row is highlighted with a red circle.

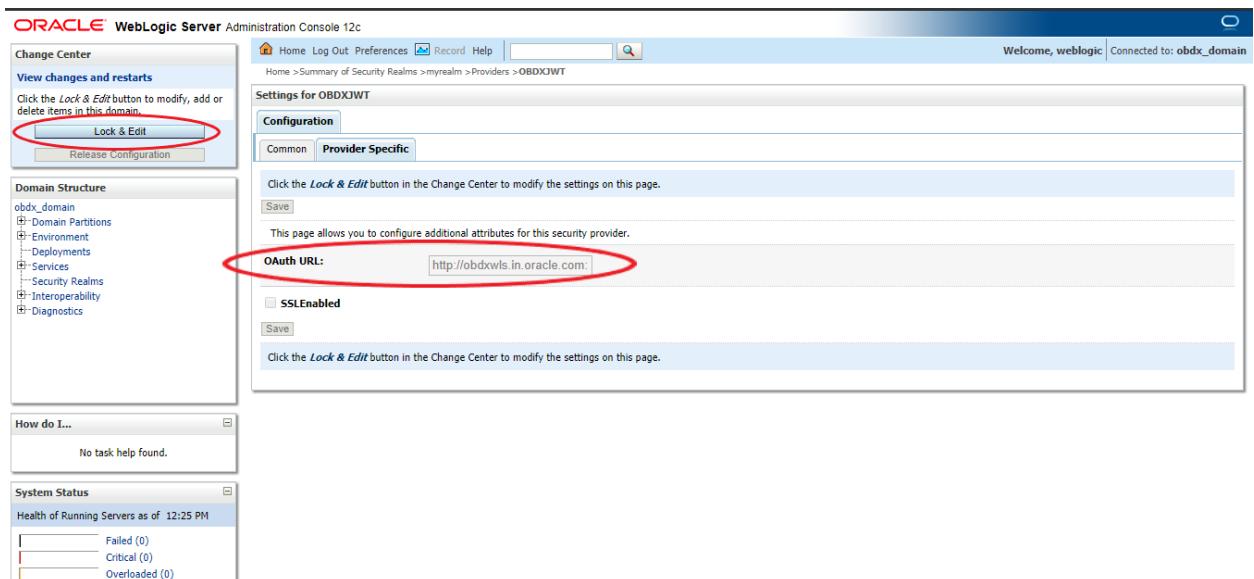
Step 6: Go to Provider Specific



The screenshot shows the Oracle WebLogic Server Administration Console. The left sidebar shows a 'Domain Structure' for 'obdx_domain' with various sub-nodes like 'Domain Partitions', 'Environment', 'Deployments', 'Services', 'Security Realms', 'Interoperability', and 'Diagnostics'. The main content area is titled 'Settings for OBDXJWT' and has a 'Configuration' tab selected. Below it is a 'Provider Specific' tab, which is circled in red. The 'Provider Specific' tab contains fields for 'Name' (OBDXJWT), 'Description' (OBDX Identity Assertion Provider (JWT and OAuth)), 'Version' (18.3.2.2.0,956), and 'Active Types'. Under 'Active Types', there are two sections: 'Available' and 'Chosen'. 'Available' contains 'Authorization', and 'Chosen' also contains 'Authorization'. There is a 'Base64 Decoding Required' field set to 'true'. A 'Save' button is at the bottom.

Step 7: Edit Oauth URL and add the following url and save.

"<http://{{host}}:{{manage-server-port}}/digx-auth/v1/token/info>"



The screenshot shows the Oracle WebLogic Server Administration Console. The left sidebar shows a 'Domain Structure' for 'obdx_domain' with various sub-nodes. The main content area is titled 'Settings for OBDXJWT' and has a 'Configuration' tab selected. Below it is a 'Provider Specific' tab, which is circled in red. The 'Provider Specific' tab contains fields for 'Name' (OBDXJWT), 'Description' (OBDX Identity Assertion Provider (JWT and OAuth)), 'Version' (18.3.2.2.0,956), and 'OAuth URL'. The 'OAuth URL' field contains the value 'http://obdxwls.in.oracle.com:'. There is a 'Save' button at the bottom. Additionally, the 'Lock & Edit' button in the 'Change Center' is circled in red.

9. Extensibility and Code Conventions

Code Convention of Account API's

Accounts related API should use below arguments and return type for working with UK Open Banking

Arguments

SessionContext sessionContext

com.ofss.digx.app.openbanking.dto.accounts.uk.AccountRequestDTO
accountRequestDTO

Return Type

BaseResponseDTO<T>

Where T extends DataTransferObject

Any service implemented with the above type of argument will be compatible with UK Open Banking.

Code Convention of Payment API's

Payment related API should use below arguments and return type for working with UK Open Banking

Arguments

Create and Read Method

SessionContext sessionContext

Any DTO Object which extends com.ofss.digx.app.openbanking.dto.consent.uk.UKPaymentDTO

Any service implemented with the above type of argument will be compatible with UK Open Banking.

Error Message Framework

The Error Message Framework helps convert the OB API error response according to the UK Open Banking Specifications.

The error response structure for Open Banking Read/Write APIs is as follows:

```
{
  "Code": "...",
  "Id": "...",
  "Message": "...",
  "Errors": [
    {
      ...
    }
  ]
}
```

```

    "ErrorCode": "...",
    "Message": "...",
    "Path": "...",
    "Url": "..."

}
]

}

```

The UK Open Banking specified error response is handled using DIGX_OB_UK_OBDX_ERROR_MAP table.

The contents of the table are as follows:

Column Name	Description
DIGX_ERROR_CODE	Represents the OB API error codes. This is a Primary and Unique Key
UK_ERROR_CODE	Represents the Open Banking specified error code
PATH	Represents the reference to the JSON Path of the field with error. Can be null.
URL	Represents the URL to help remediate the problem, or provide more information etc. Can be null.

For mapping OB API error codes with UK Open Banking specified codes below script can be used:

```
Insert into DIGX_OB_UK_OBDX_ERROR_MAP
(DIGX_ERROR_CODE,UK_ERROR_CODE,PATH,URL) values ('%%OBDX Error
Code%%','%%Open Banking specified error code%%', '%%Path%%', '%%URL%%');
```

For example –

```
Insert into DIGX_OB_UK_OBDX_ERROR_MAP
(DIGX_ERROR_CODE,UK_ERROR_CODE,PATH,URL) values
('DIGX_OB_0010','UK.OBIE.Field.Missing', 'Data.Initiation ',null);
```

Below Query is used to check the OB API errors mapped with UK Open Banking specified error codes in the system

```
select * from DIGX_OB_UK_OBDX_ERROR_MAP;
```

For configuring HTTP status codes with custom message, below script can be used:

```
Insert into DIGX_FW_CONFIG_ALL_B (PROP_ID, CATEGORY_ID, PROP_VALUE, FACTORY_SHIPPED_FLAG, PROP_COMMENTS, SUMMARY_TEXT, CREATED_BY, CREATION_DATE, LAST_UPDATED_BY, LAST_UPDATED_DATE, OBJECT_STATUS, OBJECT_VERSION_NUMBER)
values ('%%HTTP Status code%%','OpenBankingErrorConfig','%%Error Message%%','N',null,'OpenBanking Error Message','ofssuser',sysdate,'ofssuser',sysdate,'Y',1);
```

Below Query is used to check the Open Banking HTTP status codes in the system

```
select * from digx_fw_config_all_b where category_id = 'OpenBankingErrorConfig';
```

Permission Response Handler

Permissions is used in only Account API's. Based on Permissions, Response is generated based on permissions.

OB API consists of Permission Handler against each type of permissions. This configuration is available in the table **DIGX_OB_UK_PERMISSIONS_MASTER**

The contents of the table are as follows:

Column Name	Description
SERVICEID	Represents the OB API Service Id for which the permission and its handler is available
PERMISSION	Represents Permission
RESPONSEHANDLER	Represents Permission Handler

Permission Handler can be overridden or can be newly introduced. This will be required for additional fields mapping which is not available OOTB. Steps for the same are as follows

Introducing Permission Handler

New Permission Handler should implement interface IResponseHandler

New Permission Handler should have below methods

public static <T implements IResponseHandler> getInstance()

public <T extends DataTransferObject> assembleResponse(DataTransferObject object, List<String> permissions) – This method assembles response from object to the require response object which needs shown in the API response. Object is the response got from base service and T will be the response object require by API specifications. Assembling of the values will be done this method

public int getPriority() – This defines the high priority of the handler to be applied for assembling response in case of permissions and its handler has been consented by the user i.e. Basic and Detail permission will have different handlers but if the consent is both the permission the priority of the handler will decide which needs to be executed on high priority.

[Home](#)