

Berlin Open Banking Configuration Guide
Oracle Banking Digital Experience
Release 20.1.0.0.0

Part No. F30659-01

May 2020

ORACLE®

Berlin Open Banking Configuration Guide

May 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

www.oracle.com/financialservices/

Copyright © 2006, 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 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	1-1
1.1 Intended Audience	1-1
1.2 Documentation Accessibility	1-1
1.3 Access to Oracle Support	1-1
1.4 Structure	1-1
1.5 Related Information Sources	1-1
2. Objective and Scope.....	2-1
2.1 Background.....	2-1
3. Technology Stack	3-2
4. Pre-requisites	4-3
5. Headers Configuration	5-4
6. Properties	6-1
7. OAuth Configuration	7-1
7.1 UI configuration.....	7-1
7.2 Weblogic configuration	7-1
7.3 Code_challenge and Code_verifier configuration	7-4
8. Extensibility and Code Conventions	8-1

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=acc&id=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=acc&id=info> or visit

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=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:

- Introduction
- Preferences & Database
- Configuration / Installation.

1.5 Related Information Sources

For more information on Oracle Banking Digital Experience Release 20.1.0.0.0, refer to the following documents:

- Oracle Banking Digital Experience Installation Manuals

2. Objective and Scope

2.1 Background

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

Scope

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

3. Technology Stack

Software	Version
Java	Java JDK or JRE version 8
OBDX/OBAPI	20.1.0.0.0
OAuth	OBDX Internal OAuth

Abbreviations

OOTB	Out of the Box
TPP	Third Party Providers
ASPSP	Account Servicing Payment Service Provider

4. Pre-requisites

- Java JDK or JRE version 7 or higher must be installed. For installation of Java please refer **Installation Guide**.
- OAuth Setup

5. Headers Configuration

There are three types of headers configuration available for Berlin Open Banking.

- System Headers (i.e. Mandatory Headers and its respective value validation)
- Configuration Headers (i.e. Mandatory Headers).
- API Configuration Headers (i.e. Mandatory Headers of a specific API)

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

System Headers:- Both Header name and Header value are validated for System Headers.

For configuring more system headers, below script is to be executed in the OBAPI 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 ('berlin.%%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-Request-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 OBDX/OBAPI 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 ('berlin.%%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';
```

API Configuration Headers :- As of now in OOTB multiple headers have been added as mandatory. This header is required to be sent by the TPP to the ASPSP mandatorily with a corresponding suitable value.

Header name is validated if the entry is made for requested API only in case of API Configuration Headers.

For configuring more api config headers, below script is to be executed in the OBDX/OBAPI 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 ('%%API_PATH%%.%%HTTP_METHOD%%','
OpenbankingApiConfigHeaders',%%HEADER NAME%%,'N',null,'Open
Banking','ofssuser',sysdate,'ofssuser',sysdate,'Y',1);
```

Example : 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 ('accounts/{account-id}/balances.GET','OpenbankingApiConfigHeaders','Consent-ID','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 = 'OpenbankingApiConfigHeaders';
```

6. Properties

Below are the properties required to be updated in the Berlin 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.berlin.BerlinConsentHandler	Handler defines the Region specific behavior of the Open Banking framework. Berlin Consent Handler is used for Berlin Open Banking compatibility

Table: DIGX_FW_CONFIG_ALL_B

Category-Id : taskEvaluatorFactories

Property Id	Property Value (Out of the Box)	Purpose
OB_C_ACD	com.ofss.digx.app.openbanking.consent.task.evaluator.berlin.ConsentTaskEvaluatorFactory	This value is used to provide Consent TaskEvaluator Factory

Table: DIGX_FW_CONFIG_ALL_B**Category-Id :** AspectDataEvaluator

Property Id	Property Value (Out of the Box)	Purpose
OB_C_ACD	com.ofss.digx.app.openbanking .consent.task.evaluator.berlin.C onsentAuthenticationDataEvalu ator	This value is used to provide Consent AuthenticationData Evaluator

Table: DIGX_FW_CONFIG_ALL_B**Category-Id :** SecurityConstants

Token Settings

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. OAuth Configuration

7.1 UI configuration

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

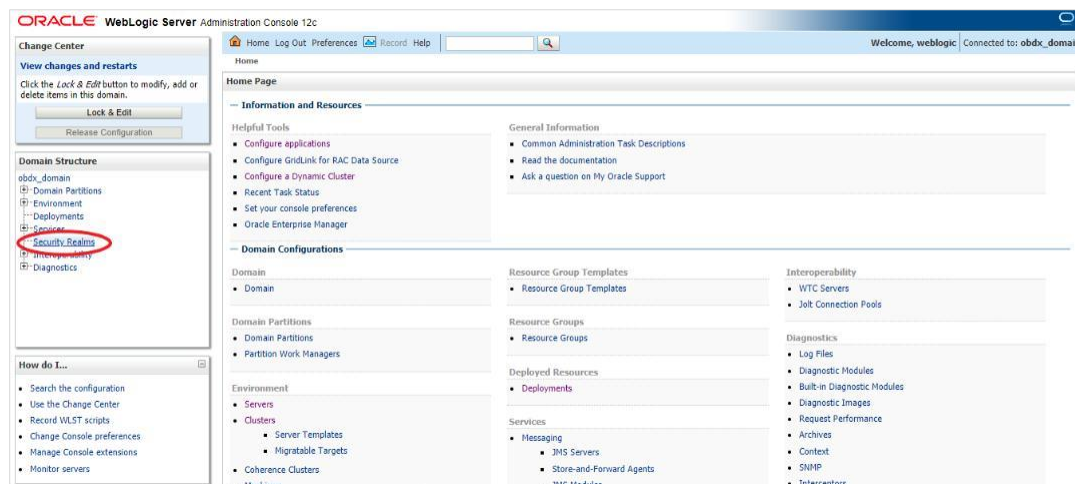
The value of Consent Page URL (Menu -> OAuth -> Identity Domain Maintenance) is configured as `http://host:port?homeComponent=authorize-consent-berlin&homeModule=open-banking&applicationType=digx-auth&menuNavigationAvailable=false`.

7.2 Weblogic configuration

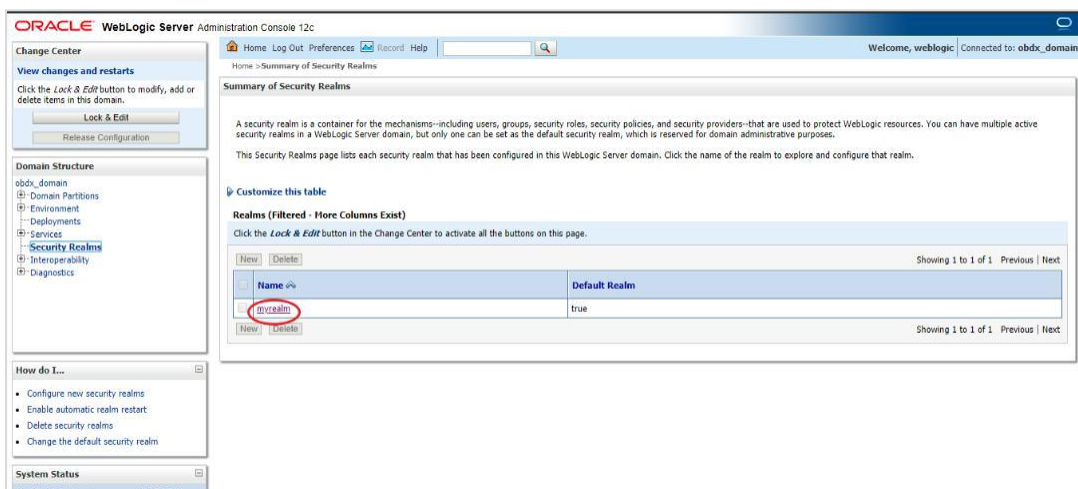
OAuth Maintenance will require below maintenance in weblogic to configure an URL.

Step 1: Login to weblogic

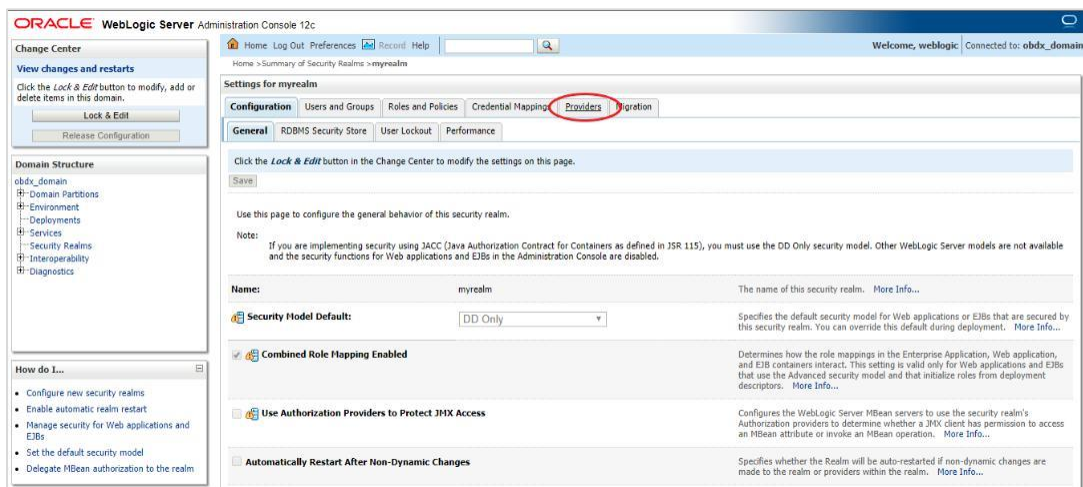
Step 2: Go to Security Realms



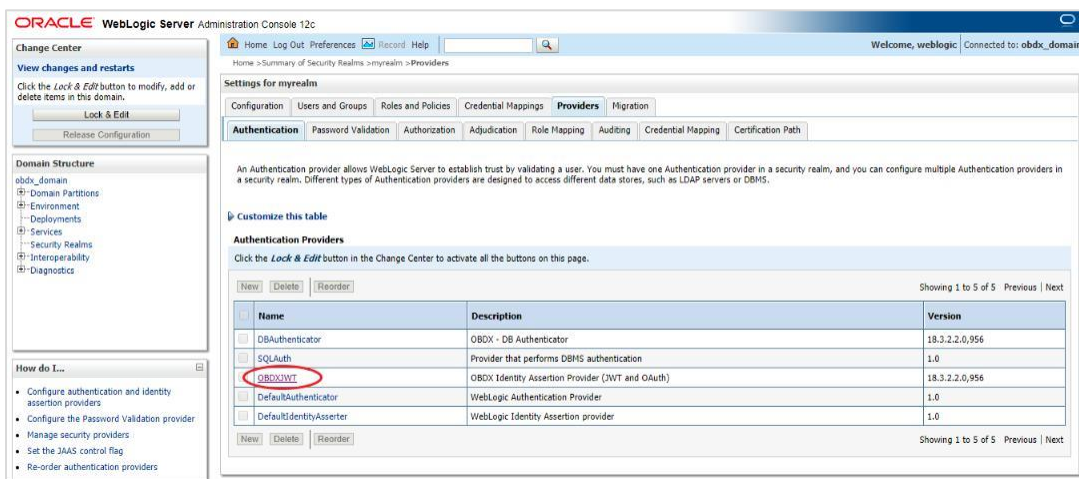
Step 3: Go to myrealm



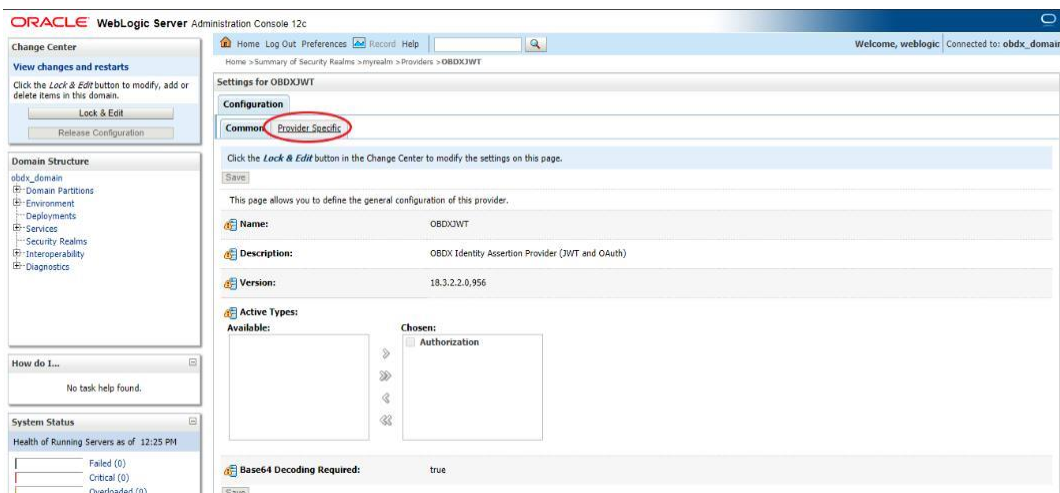
Step 4: Go to Providers



Step 5: Go to OBDXJWT

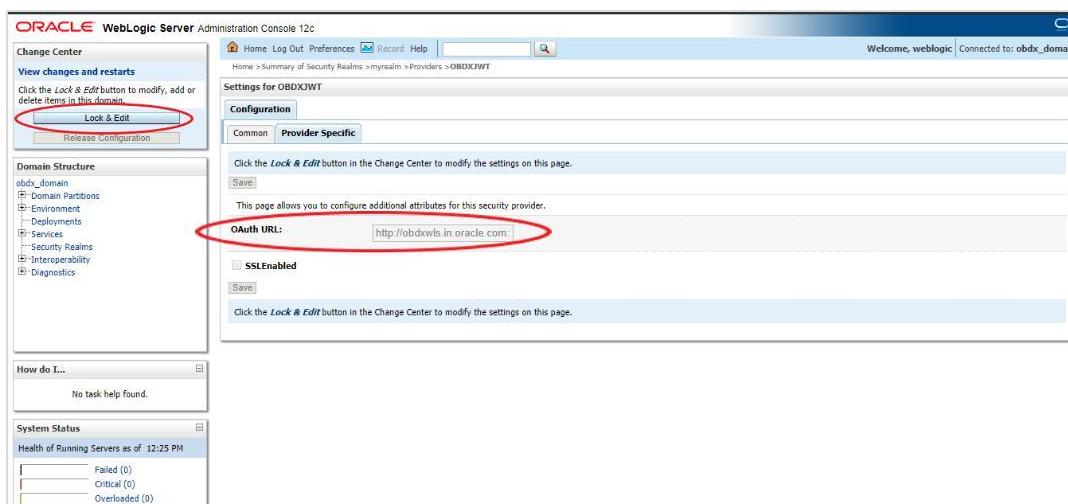


Step 6: Go to Provider Specific



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

“http://{{host}}:{{manage-server-port}}/digx-auth/v1/token/info”



7.3 Code challenge and Code verifier configuration

To enable code_challenge and code_verifier functionality in Oauth, below maintenance in the oauth property file is required.

Property File : common_rest.properties

Property Name : isCodeChallengeEnabled

Property Value : true/false

The default value for 'isCodeChallengeEnabled' is 'false'. Change it to 'true' to enable this feature.

8. Extensibility and Code Conventions

Error Message Framework

The Error Message Framework helps convert the OBAPI error response according to the BERLIN Open Banking Specifications.

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

```

{ "tppMessages" :[
  {
    "category" : "",
    "path" : "",
    "code" : "",
    "text" : ""
  }
]
}

```

The Berlin Open Banking specified error response is handled using DIGX_OB_BERLIN_OBDX_ERROR_MAP table.

The contents of the table are as follows:

Column Name	Description
DIGX_ERROR_CODE	Represents the OBAPI error codes. This is a Primary and Unique Key
BERLIN_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 OBAPI error codes with Berlin Open Banking specified codes below script can be used:

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

Below Query is used to check the OBAPI errors mapped with BERLIN Open Banking specified error codes in the system

```
select * from DIGX_OB_BERLIN_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';

[Home](#)