Oracle Banking Digital Experience

Security Guide
Release 15.1.0.0.0

Part No. E66313-01

October 2015

ORACLE®

Oracle Banking Digital Experience, Release 15.1.0.0.0

Table of Contents

1. Preface

Welcome to the Oracle Banking Digital Experience Security Guide. This guide describes how you can configure security for Oracle Banking Digital Experience.

1.1 Audience

This guide is intended for the Bank's IT staff responsible for application installation and security configuration.

1.2 Notes

- As with any other information system, do not attempt to implement any of the recommendations in this guide without first testing in a non-production environment.
- This document is only a guide containing recommendations. It is not meant to replace well-structured policy or sound judgment. Furthermore this guide does not address site-specific
  optimization, configuration concerns.
-
- Care must be taken when implementing this guide to address local operational and policy concerns.

1.3 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 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 Abbreviations

OBDX	Oracle Banking Digital Experience
Java EE / JEE	Java Enterprise Edition
Java SE / JSE	Java Standard Edition
DBA	Database Administrator
XML	Extensible Markup Language
XSL	XML Style sheets
TCP	Transmission Control Protocol
HTTP	Hyper Text Transmission Protocol
HTTPS	Secured Hyper Text Transmission Protocol
SSL	Secured Socket Layer
IDS	Intrusion Detection System

2. Overview

This chapter presents an overview of the Oracle Banking Digital Experience Platform and explains the general principles of application security.

2.1 Product Overview

Oracle Banking Digital Experience is a multi-channel banking platform. Channels like the web browser, the mobile browser and mobile apps are supported.

It integrates with a core banking host like Oracle FLEXCUBE Universal Banking or any third party Core Banking host.

2.2 General Principles

The following principles are fundamental for using any application securely.

2.2.1 Restrict Network Access to Critical Services

Keep both the Oracle Banking Digital Experience application server and the database behind a firewall. In addition, place a firewall between the application server and the database. The firewalls provide assurance that access to these systems is restricted to a known network route, which can be monitored and restricted, if necessary.
It is also recommended that a firewall be placed between the web server / load balancer and the application server.

2.2.2 Encrypt Transmitted Data Whenever Possible

Data transmitted over a network, whether via wire or wirelessly, is susceptible to passive monitoring. Whenever practical mechanisms exist for encrypting this data-in-transit, they should be applied. Even if data is expected to be transmitted only over a local network, it should still be encrypted.

2.2.3 The Principle of Least Privilege

The principle of least privilege states that users should be given the least amount of privilege to perform their jobs. User privileges should be reviewed periodically to determine relevance to current job responsibilities.

2.2.4 Monitor System Activity

The system needs to be constantly monitored to detect possible attacks.

3. Configuring OBDX Securely

3.1 Secure the Bank Admin Password

The OBDX Bank Admin password should be secured after the initial setup. The financial institution should create additional administrative users for their operational needs.

3.2 Create keys for Login Password Encryption

The login password is encrypted on the client using the RSA algorithm. The public key – private key pair for this is configurable.

To generate a new key-pair the following steps should be followed:

1. Change directory to <OBDX_INSTALL_HOME>/system/home
2. Execute the following command:
   keytool -genkeypair -alias fcdb -keystore fcdb_keystore.jks -storetype JKS -keyalg RSA -keysize 2048
3. Follow the on-screen instructions to create or update the keystore.
4. Execute the below command to extract the public certificate (for use with iOS devices):
   keytool -export -alias fcdb -keystore fcdb_keystore.jks -file fcdb_sign.cer

The keystore is to be placed in < OBDX_INSTALL_HOME>/system/home.
For the internet channel, the public key gets sent by the server in the login page response.
For the mobile applications, the public key gets distributed in the mobile application itself, as a property.

Please note that the alias for the key pair is “fcdb”. DO NOT change that.

The keystore password and the private key password are to be stored in the properties file fcat.properties using the following properties.

FCDB.KEYSTORE.PASSWORD
FCDB.KEY.fcdb.PASSWORD

We recommend that the 2 passwords be kept different.

Also, on a Unix system we recommend that the folder permissions for <OBDX _INSTALL_HOME>/system/home be set to 600 post key generation.

3.3 Configure password hashing iterations

The user login password is stored in the database after applying a one way hash function on it. The number of hashing iterations to be performed is configurable and is stored in the property HASHING.ROTATION.COUNT in < OBDX _INSTALL_HOME>/system/home /fcat.properties file. If no value is mentioned, the default value is assumed to be 1000.
In case it starts to affect performance, then a lower value can be configured.

3.4 Encrypt fcat.properties

The fcat.properties file should be encrypted using the following batch (.bat / .sh) file

securepropertiesfile <ClearText Properties File> <Output Encrypted Properties File>

The clear text properties file should be backed up and securely stored.

Note: Please set file permissions in such a way that only authorized users have access to these files on the operating system.

3.5 Branding Considerations

The WAR file should be cleaned after the branding exercise for any unwanted static content, JSP files etc. The images, JS, CSS files that are not required should be removed before the final deployment.

3.6 Remove Copyrights from Static Web Content

The JS, CSS and Static HTML files can contain copyrights and modification history information. This should be removed from the static content from within the WAR files to avoid any social engineering attacks.

No modification history and copyrights should be maintained in the static content.

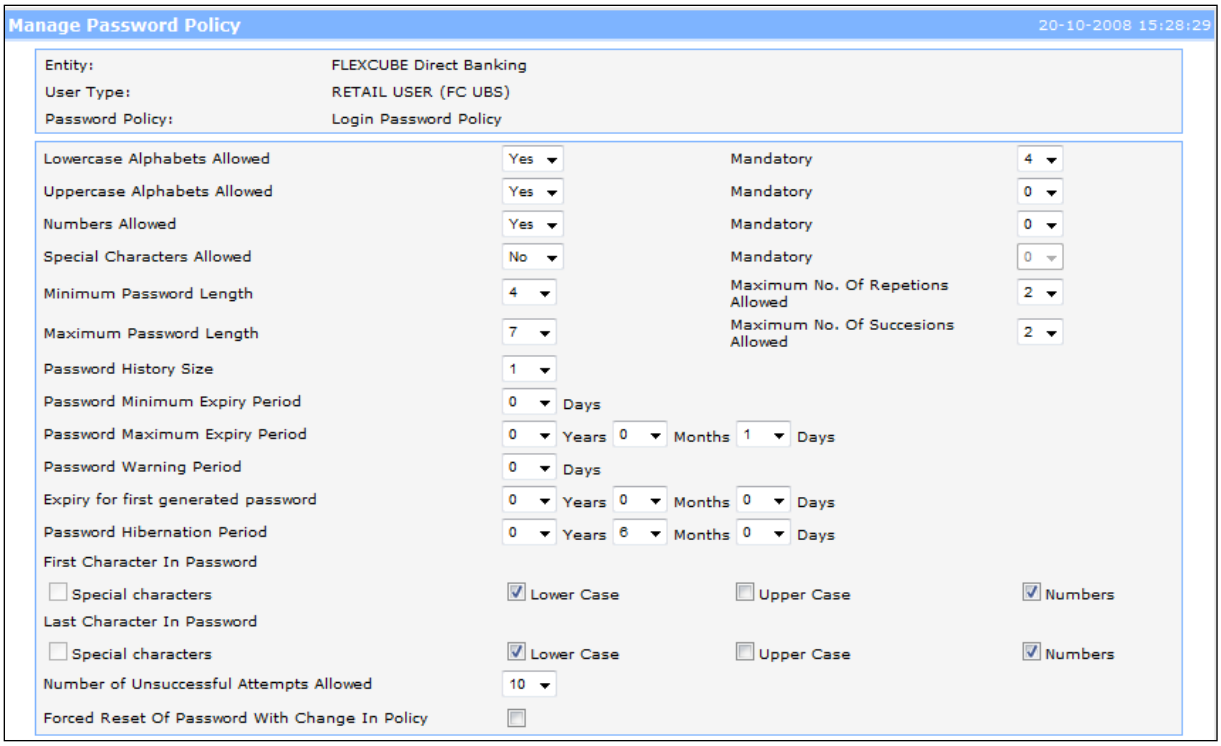
3.7 Disable Directory Browsing

Directory browsing feature should be disabled in the WAR files if newly built. This is configurable on the web server.

3.8 Password Management

Password Management refers to managing the password policies in the application.

The Manage Password Policy function should be used by the administrators to setup the appropriate password policy rules. This is available in the administration console of the Oracle Banking Digital Experience application.



The following parameters can be configured.

- Minimum and Maximum Length (We recommend a minimum of 8 characters)
- History Size
- Expiry period
- Warning period
- Min Number of Digits required (We recommend at least 1 digit to be made mandatory in the password)
- Min Number of Lower case alphabets required (We recommend at least 1 lower case alphabet to be made mandatory in the password)
- Min Number of upper case alphabets required (We recommend at least 1 upper case alphabet to be made mandatory in the password)
- Special characters allowed (We recommend at least 1 special character to be made mandatory in the password)
- Maximum unsuccessful login attempts (We recommend that this number be not more than 5)

3.9 Setup Audit Logging

To ensure that users cannot repudiate the transactions that they have done, auditing transactions is necessary. This is a critical security feature of any financial application on the internet. The audit logging feature should be enabled for all the financial transactions within Oracle Banking Digital Experience.

This is enabled using the AUDITLOGREQUIRED flag in the table MSTCHANNELATS for the required transactions.

3.10 Protect against Clickjacking

On the web server, the value of the property X-Frame-Options should be set to SAMEORIGIN to prevent clickjacking attacks.

3.11 Protect against Injections

On the web server, the value of the property Content-Security-Policy should be set to default-src 'self' to prevent content injection attacks. This ensures prevention of some of the Cross Site Scripting attacks as well.

3.12 Protect against XSS

The application provides a way to identify malicious characters (used in XSS) and filter them out of all the user input fields.

3.12.1 Single Transactions

The configuration is placed in <entity>.xml file and the properties are named as
FCAT.FILTER.EXP.TXN and
FCAT.REPLACE.EXP.TXN

The set of malicious characters are placed in FCAT.FILTER.EXP.TXN as a comma separated list of values. The corresponding set of characters that replace the above characters is placed in FCAT.REPLACE.EXP.TXN

3.12.2 Bulk Transactions

The configuration is placed in the VALUESTRING column of the table APPLDATA → DATANAME= 'DEFAULT_VALIDATOR' and DATAVALUE= 'BLACK_LIST'. It is a comma separated list of malicious characters that are to be filtered out.

The out of the box list of blacklisted characters is as follows:

<
>
(
)
&
;
/
"
'

You may add more characters to this list, but we DO NOT recommend removing any characters from the above list unless the business cannot absolutely run without it.

3.13 OBDX – Host (FCUBS) authentication

If the backend host (FCUBS only) switches on authentication at its end, then OBDX also has a configuration in place to send the password along with the user name in the request XMLs that are sent to the host.
In the database table MSTPROPERTIES (Admin Schema), 2 properties need to be set

1) <ID_ENTITY>.HAS.HOST.PASSWORD → Y
2) <ID_ENTITY>.HOST.PASSWORD → <ACTUAL PASSWORD>

Sample Database Scripts:

insert into mstproperties (IDSERVER, PROPNAME, PROPVALUE, ENABLED, ISMODIFIED, DATEFFECTIVE, DATMODIFIED, ISGUIENABLED) values ('ZZ', '<ID_ENTITY>.HAS.HOST.PASSWORD', 'Y', 'Y', 'Y', to_date('30-07-2015 18:26:50', 'dd-mm-yyyy hh24:mi:ss'), to_date('30-07-2015 18:26:50', 'dd-mm-yyyy hh24:mi:ss'), 'Y');

insert into mstproperties (IDSERVER, PROPNAME, PROPVALUE, ENABLED, ISMODIFIED, DATEFFECTIVE, DATMODIFIED, ISGUIENABLED) values ('ZZ', '<ID_ENTITY>.HOST.PASSWORD', '<FCUBSPASSWORD>', 'Y', 'Y', to_date('30-07-2015 18:26:50', 'dd-mm-yyyy hh24:mi:ss'), to_date('30-07-2015 18:26:50', 'dd-mm-yyyy hh24:mi:ss'), 'Y');

3.14 Authorization Engine

OBDX is extensible in a way that new transactions can be built using the given framework in a noninvasive way (without having to change the base code), by the implementation team.

The details given in this section are relevant when a new transaction is being developed in the noninvasive way by Consulting / Partners.

When building a new transaction, security against data spoofing attacks can be provided by using the Authorization Engine framework.

There are a few modules available to ensure security related validations. The complete list can be obtained from the table mstmodules in the Admin schema. The correct modules (based on your transaction) need to be configured in the table mstinitauthmodules in the Admin schema.

For example: The module ACM (Account Check Module) checks whether the source account sent in the request for the given transaction belongs to the logged in user or not.

For details, please refer to the document Oracle_Banking_Digital_Experience_System_Handbook_Volume_IV.pdf

3.15 Configuring secure SMTP

The product uses a framework to generate Email alerts to its users. The protocol to be used is SMTPS out of the box. The default port for secure SMTP is 465.

We recommend that you use the out of the box protocol (SMTPS). We also recommend that you DO NOT use the out of the box SMTP port. It is a good practice to not use the default ports. However if you need to change this, then it is configurable.

The port and the protocol, both are configurable in the Database in the table called mstproperties.

The properties have been named as follows:
ALERTNOTIFY_EMAIL.PROTOCOL
ALERTNOTIFY_EMAIL.PORT

4. Database Security

4.1 Listener Configuration

By default, the TNS Listener receives service requests on TCP port 1521. Configure it to listen on another port number. Although not foolproof, this makes attacks more difficult.

4.2 Database Security

It is recommended that Transparent Data Encryption (TDE) be done on the Database so as to secure the data files. Sensitive data such as Credit Card Numbers thus gets secured.

The configuration is explained in detail in the Oracle® Database Advanced Security Administrator's Guide. We recommend encrypting all the tablespaces.