Security Guide
Oracle FLEXCUBE Universal Banking & Oracle FLEXCUBE UI Refresh
Release 14.4.0.0.0
[May] [2020]

ORACLE
FINANCIAL SERVICES

ORACLE

Table of Contents

1. About this Manual

1.1 Introduction

Purpose:

This document provides security-related usage and configuration recommendations for Oracle FLEXCUBE Universal Banking & Oracle FLEXCUBE UI Refresh. This guide may outline procedures required to implement or secure certain features, but it is also not a general-purpose configuration manual.

Audience:

This guide is primarily intended for IT department or administrators deploying FLEXCUBE, Oracle FLEXCUBE UI Refresh and third party or vendor software's. Some information may be relevant to IT decision makers and users of the application are also included. Readers are assumed to possess basic operating system, network, and system administration skills with awareness of vendor/third-party software's and knowledge of FLEXCUBE application.

1.2 Scope

1.2.1 Read Sections Completely

Each section should be read and understood completely. Instructions should never be blindly applied. Relevant discussion may occur immediately after instructions for an action, so be sure to read whole sections before beginning implementation.

1.2.2 Understand the Purpose of this Guidance

The purpose of the guidance is to provide security-relevant configuration recommendations. It does not imply the suitability or unsuitability of any product for any particular situation, which entails a risk decision.

1.2.3 Limitations

This guide is limited in its scope to security-related issues. This guide does not claim to offer comprehensive configuration guidance. For general configuration and implementation guidance refer to other sources such as Vendor specific sites.

1.2.4 Test in Non-Production Environment

To the extent possible, guidance should be tested in a non-production environment before deployment.

Ensure that any test environment simulates the configuration in which the application will be deployed as closely as possible.

2. Prerequisite

2.1 Operating Environment Security

Please refer the vendor specific documentation for making the environment more safe and secured.

2.2 Network Security

Please refer the vendor specific documentation for making the environment more safe and secured.

2.3 Oracle Database Security

Please refer the Oracle Database Security specification document for making the environment more safe and secured.

2.3.1 Oracle FLEXCUBE & UI Refresh Recommended configuration

This section contains security recommendations for the Database used for Oracle FLEXCUBE & UI Refresh Application.

Init.ora	REMOTE_OS_AUTHENT=FALSE	Authentication
Init.ora	_TRACE_FILES_PUBLIC=FALSE	Authorization
Init.ora	REMOTE_OS_ROLES=FALSE	Authorization
Init.ora	O7_DICTIONARY_ACCESSIBILITY = FALSE	Authorization
Init.ora	AUDIT_TRAIL = OS	Audit
Init.ora	AUDIT_FILE_DEST = E:\logs\db\audit	Audit

To audit sessions	SQL> audit session;	Audit
To audit schema changes	SQL> audit user;	Audit
To audit other events	SQL> AUDIT DATABASE LINK; -- Audit create or drop database links SQL> AUDIT PUBLIC DATABASE LINK; -- Audit create or drop public database links SQL> AUDIT SYSTEM AUDIT; -- Audit statements themselves SQL> AUDIT ALTER ANY ROLE by ACCESS; -- Audit alter any role statements SQL> AUDIT ALTER DATABASE by ACCESS; -- Audit alter database statements SQL> AUDIT ALTER SYSTEM by ACCESS; -- Audit alter system statements SQL> AUDIT CREATE ROLE by ACCESS; -- Audit create role statements SQL> AUDIT DROP ANY ROLE by ACCESS; -- Audit drop any role statements SQL> AUDIT PROFILE by ACCESS; -- Audit changes to profiles SQL> AUDIT PUBLIC SYNONYM by ACCESS; -- Audit public synonyms statements SQL> AUDIT SYSDBA by ACCESS; -- Audit SYSDBA privileges SQL> AUDIT SYSOPER by ACCESS; -- Audit SYSOPER privileges SQL> AUDIT SYSTEM GRANT by ACCESS; -- Audit System grant privileges	Audit

To audit the events, login through sqlplus as SYSTEM and issue the commands.

2.4 Application Server Security

Please refer the Oracle Weblogic Security specification document for making the environment more safe and secured.

Apart from the Oracle Weblogic Security specification, Oracle FLEXCUBE UBS Application recommends to implement the below security specifications.

Support for Single Sign on (SSO)

Oracle FLEXCUBE Universal Banking Solution supports Single sign-on capability with SAML (Security Assertion Markup Language) authentication. Ensure that the LDAP used for Oracle FLEXCUBE Single Sign-on deployment with SAML (if SAML validation opted) is certified to work with Oracle Access Manager.

Oracle Access Manager consists of the Access System and the Identity System. The Access System secures applications by providing centralized authentication, authorization and auditing to enable single sign-on and secure access control across enterprise resources. The Identity System manages information about individuals, groups and organizations. It enables delegated administration of users, as well as self-registration interfaces with approval workflows. These systems integrate seamlessly.

For details on configuration, refer to the document FCUBS_V.UM_OAM_Integration_Enabling_SSO.zip.

Support for LDAP (External Password Authentication)

FLEXCUBE UBS also supports authentication through LDAP/MSAD without the use of SSO.

Depending on the value of the property EXT_USERLOGIN in fcubs.properties file the length of userid field in login screen will change. If the value is “Y” then user will be able to input up to 30 characters in userid field. Otherwise userid field will allow only 12 characters.

Depending on the value PASSWORD_EXTERNAL in fcubs.properties file, the password will be validated with LDAP/MSAD or FCUBS Application.

For details on configuration of LDAP*, refer to Oracle FLEXCUBE Universal Banking Installation Guide document (Sec 1.4).*

Support for SSL (Secure Transformation of Data)

The FLEXCUBE & UI Refresh Installer allows a deployer to configure that all HTTP connections to the application are over SSL/TLS. In other words, all HTTP traffic in the clear will be prohibited; only HTTPS traffic will be allowed. It is highly recommended to enable this option is a production environment, especially when WebLogic Server acts as the SSL terminator.

For details on configuration of SSL*, refer to Oracle FLEXCUBE Universal Banking Installation Guide document (Sec 1.4.1 for Weblogic, Sec 1.4.2 for WebSphere)*

Support for SMTPS (Mail communication)

Also mail session configuration required in Application server. Sample details for creating a mail session are listed below:

Name: FCUBSMailSession

JNDI Name: mail/FCUBSMail (The same need to be maintained in property file creation.)

Java Mail Properties for SMTPS protocol:

mail.host=<HOST_MAIL_SERVER>

mail.smtps.port=<SMTPS_SERVER_PORT>

mail.transport.protocol=smtps

mail.smtps.auth=true

mail.smtps.host==<HOST_SMTPS_MAIL_SERVER>

For details on configuration of Mail Session *process, refer to the document < Resource_Creation_WL.doc for weblogic or Resource_Creation_WAS.doc for websphere >.*

Support for Securely store the credentials in CSF

Oracle FLEXCUBE supports to store encryption key (Symmetric key) store in secure credential storage area.

To support CSF, OPSS component should be available in the application server domain.

Oracle FLEXCUBE INSTALLER allows administrator to enable CSF component to the application. If CSF component enabled, then the application look into CSF to get the required properties values.

The default CSF option is enabled for the application.

2.5 Application Server Security

Please refer the Oracle Web Logic Security specification document for making the environment more safe and secured.

Oracle FLEXCUBE UI Refresh application supports the following authentication schemes for the online web application

- Standard LDAP Directory (e.g. OUD/AD)
- SSO with OAM (Oracle Access Manager – Part of the Oracle Identity Management Suite)
- SAML assertions with a Service Provider protecting the resource and an Identity Provider.

Oracle FLEXCUBE UI Refresh application supports the following authentication scheme for the API layer

- OAuth (CLIENT CREDENTIALS) with OAM

In case the customer does not have OAM, it is expected that the customer has an enterprise API Management Layer that protects Oracle FLEXCUBE UI Refresh application's API layer with the same controls (i.e. OAuth)

Support for SSL (Secure Transformation of Data)

The Oracle FLEXCUBE UI Refresh application to be configured that all HTTP connections to the application are over SSL/TLS. In other words, all HTTP traffic in the clear will be prohibited; only HTTPS traffic will be allowed. It is highly recommended to enable this option is a production environment, especially when WebLogic Server acts as the SSL terminator.

2.6 Third-party Applications

Support for OWSM (Securing Web services)

Oracle FLEXCUBE Universal Banking supports to the WebLogic Server WS-Policies for enforcing security for Web services. Customer can implement any Oracle WSM WS-Security policies and use them with WebLogic Web services.

The Oracle WSM policies are documented in the Oracle Fusion Middleware Security and Administrator's Guide for Web Services < http://docs.oracle.com/cd/E21764_01/web.1111/b32511/toc.htm>

2.7 Choice of the SSL cipher suite

Oracle WebLogic Server allows for SSL clients to initiate a SSL connection with a null cipher suite. The null cipher suite does not employ any bulk encryption algorithm thus resulting in transmission of all data in clear text, over the wire.

The default configuration of Oracle WebLogic Server is to disable the null cipher suite. Ensure that the usage of the null cipher suite is disabled, preventing any client from negotiating an insecure SSL connection.

Furthermore, for installations having regulatory requirements requiring the use of only 'high' cipher suites, Oracle WebLogic Server can be configured to support only certain cipher suites. The restriction can be done in config.xml of the WebLogic domain. Provided below is an example config.xml restricting the cipher suites to those supporting 256-bit symmetric keys or higher, and using RSA for key exchange.

```
....

<ssl>

       <enabled>true</enabled>

       <ciphersuite>TLS_RSA_WITH_AES_256_CBC_SHA</ciphersuite>

      </ssl>

....
```

- configuration of WebLogic Server to support the above defined cipher suites might also require an additional command line argument to be passed to WebLogic Server, so that a

FIPS 140-2 compliant crypto module is utilized. This is done by adding **-Dweblogic.security.SSL.nojce=true** as a JVM argument.

- The restriction on cipher suites needs to be performed for every managed server.
- The order of cipher suites is important – Oracle WebLogic Server chooses the first available cipher suite in the list, that is also supported by the client.
- Cipher suites with RC4 are enabled despite it being second best to AES. This is primarily for older clients that do not support AES (for instance, Microsoft Internet Explorer 6, 7 and 8 on Windows XP).

2.8 Securing the Oracle FLEXCUBE Universal Banking Application

The following guidelines serve to secure the Oracle FLEXCUBE Universal Banking application deployed on Oracle WebLogic Server.

2.8.1 Setting up Secure Flag for Cookies

If the secure flag is set on a cookie, then browsers will not submit the cookie in any requests that use an unencrypted HTTP connection, thereby preventing the cookie from being trivially intercepted by an attacker monitoring network traffic.

Below configuration has to be ensured in weblogic.xml within the deployed application ear.

1. Cookie secure flag set to true

```
<wls:session-descriptor>
  <wls:cookie-secure>true</wls:cookie-secure>
  <wls:url-rewriting-enabled>false</wls:url-rewriting-enabled>
</wls:session-descriptor>
```

Always make sure Cookies are set with always Auth Flag enabled by default for WebLogic server and also recommended to apply the weblogic patch 10.3.5 for versions using below weblogic 10.3.5 to reflect the above changes.

2.8.2 Credential Over mail

To enable this feature mail server details needs to be provided at the time of property file creation .Below are the required parameters

Host Server

User ID

User Password

JNDI Name

2.8.3 Session time out and Token Management

Session timeout represents the event occurring when a user do not perform any action on a web site during a interval (defined in application). The event, on server side, change the status of the user session to 'invalid' (ie. "not used anymore") and instruct the Application/web server to destroy it (deleting all data contained into it). Application allows defining the session time out.

The default value for session time out is 30 minutes.

The entire subsequent request within the session will be having the Authenticated and Cross-site request forgery tokens .Every request send to the application from the browser is validated against the IsAuthenticated attribute and Cross-site request forgery token.

2.8.4 Two-way SSL Connection

A two-way SSL is used when the server needs to authenticate the client. In a two-way SSL connection the client verifies the identity of the server and then passes its identity certificate to the server. The server then validates the identity certificate of the client before completing the SSL handshake.

In order to establish a two-way SSL connection, need to have two certificates, one for the server and the other for client.

For Oracle FLEXCUBE Universal Banking Solutions, need to configure a single connector. This connector is related to SSL/TLS communication between host or browser and the branch which uses two-way authentication.

For details on implementation of Two-way SSL process, refer to the document available for FLEXCUBE < SSL_OR_TLS_ Configuration.doc>.

2.8.5 Securely store the credentials in CSF:

Oracle FLEXCUBE application use CSF to securely store the properties values in a credentials store and the additional benefits of CSF, such as the ability to manage / operations use Oracle Fusion Middleware user interfaces / em console.

For details on implementation of OPSS CSF, refer to the document installation / configuration documents in user manuals.

2.9 Securing the Oracle FLEXCUBE UI Refresh Application

Securing the Oracle FLEXCUBE UI Refresh application Application includes securing

A) The Online Web Application and
B) The API Layer exposed to external consumers

2.9.1 Online Web Application

Access to the online web application is granted only via the following methods

- Standard LDAP Directory authentication
- SSO with OAM and
- SAML with the Oracle FLEXCUBE UI Refresh application acting as the service provider

In addition to the authentication, the Oracle FLEXCUBE UI Refresh application online web application uses JWT (JSON Web Tokens) to maintain the state for authenticated users.

JSON Web Tokens are an open, industry standard RFC 7519 method for representing claims securely between two parties. JSON Web Token (JWT) is a compact, URL-safe means of representing claims to be transferred between two parties. The claims in a JWT are encoded as a JSON object that is used as the payload of a JSON Web Signature (JWS) structure or as the plaintext of a JSON Web Encryption (JWE) structure, enabling the claims to be digitally signed.

- No Session to Manage (stateless): The JWT is a self-contained token which has authentication information, expire time information, and other user defined claims digitally signed.
- Portable: A single token can be used with multiple backend. No Cookies Required, So It's Very Mobile Friendly
- Good Performance: It reduces the network round trip time.
- Decoupled/Decentralized: The token can be generated anywhere. Authentication can happen on the resource server, or easily separated into its own server.

In addition, the following policies are followed for JWT,

- **Token Store:** To increase the security and better usability, every authentication/refresh request is secured by random unique key. The generated token and the secure key are persisted in the table, so that during the horizontal scaling of the servers, any API gateway instance can serve for the request.
- **Cipher strength:** Platform security module hashes the JWT footer with HS512 algorithm.
- **Refresh Token:** Users are allowed to get the new token any time before expiring the existing token.
- **Claims:** The JWT Claims Set represents a JSON object whose members are the claims conveyed by the JWT. Platform security module validates below claims during the process.

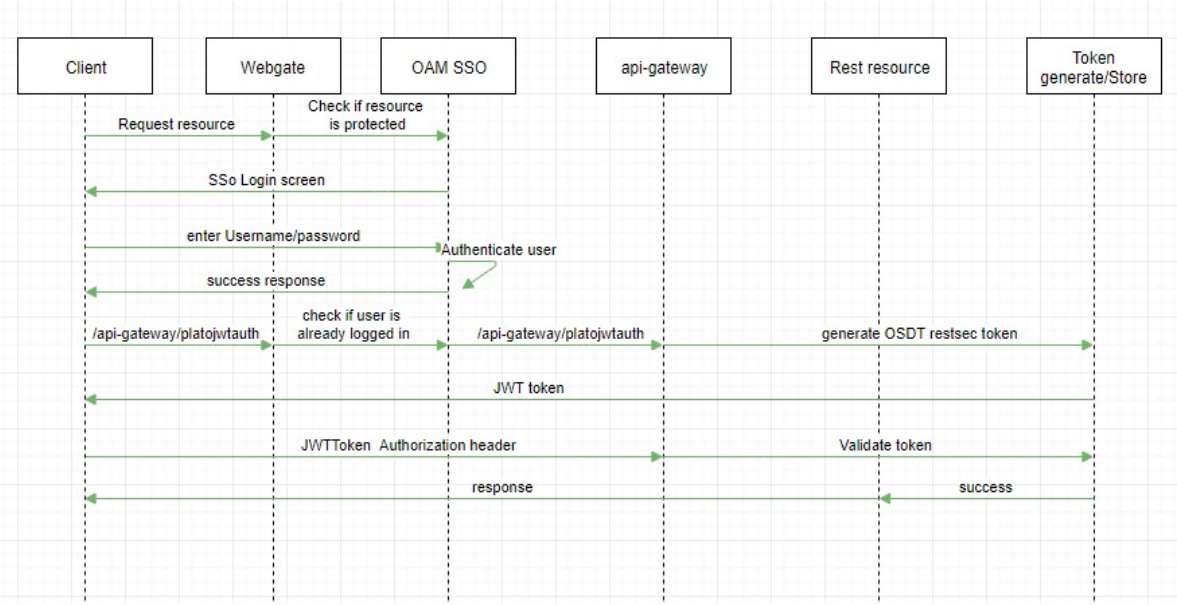
Claim Name	Description	Mandatory	Type
iss	Issuer	Yes	Registered
sub	Subject	Yes	Registered
aud	Audience	No	Registered
exp	Expiration Time	Yes	Registered

nbf	Not Before	No	Registered
iat	Issued At	Yes	Registered
jti	JWT Id	Yes	Registered
Tid	Tenant Id	Yes	Private

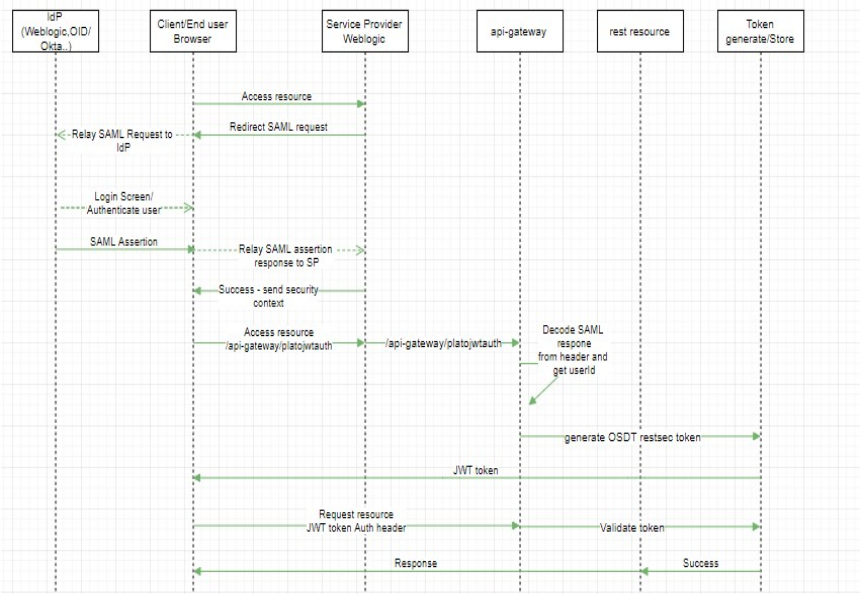
- **Token Expiry:** Platform security module invalidates the token, if the client submits after the Expiration time. In addition, token becomes invalid, if the user password changed after the token issuance.
- **Logout:** While user calls the logout operation, platform security module clears the issued token and deletes the record from the table as well. The old token no longer will be used for any purpose.

The various security flows for the online web application are depicted below

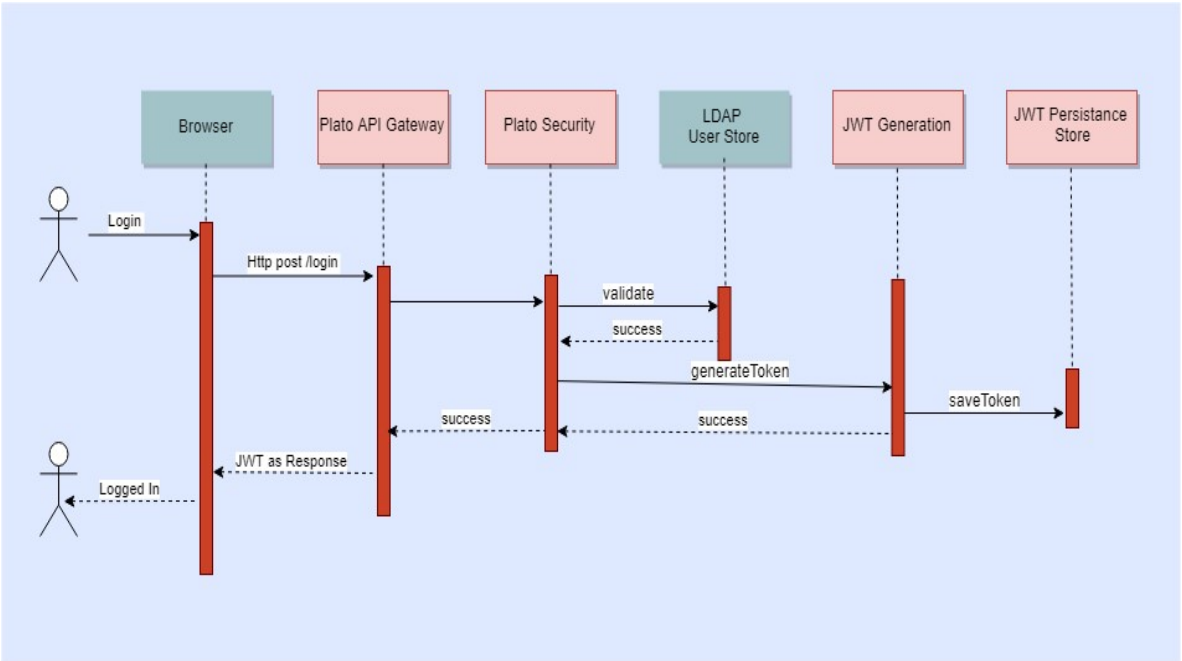
OAM Based SSO



- The online UI is protected on OAM.
- Client requests protected resource. OAM presents SSO Login screen
- Client enters a user id and password. In case of success, OAM sets the corresponding user profile details in the security context
- The request is routed to the Gateway which extracts the profile details from the security context
- The API Gateway creates a JWT token (Utilizing Oracle's Security Developer Toolkit part of Oracle's Platform Security Services), persists it in the Database and returns the same.
- The UI layer uses this token to maintain state and conduct subsequent invocations



- The Identity Provider is external to the Oracle FLEXCUBE UI Refresh application (e.g. OKTA) with the Oracle FLEXCUBE UI Refresh application acting as the Service Provider
- Client requests protected resource from ORACLE FLEXCUBE UI REFRESH APPLICATION. The Idp presents a configured login screen to the user
- Client enters a user id and password. In case of success, the Idp sets the corresponding user profile details in the security context
- The request is routed to the Gateway which extracts the profile details by decoding the SAML response
- The API Gateway creates a JWT token (Utilizing Oracle's Security Developer Toolkit part of Oracle's Platform Security Services), persists it in the Database and returns the same.



- The user is presented the standard login page for the Oracle FLEXCUBE UI Refresh application
- The user enters a user id and password. The credentials are validated against a standard

LDAP store.

- If successful, the API Gateway generates a JWT token (Utilizing Oracle's Security Developer Toolkit part of Oracle's Platform Security Services), persists it in the Database and returns the same.

2.9.2 API Layer

The Oracle FLEXCUBE UI Refresh application provides an API Layer (also known as the Service API Layer) which is used by external consumers to access Oracle FLEXCUBE UI Refresh application's functionality.

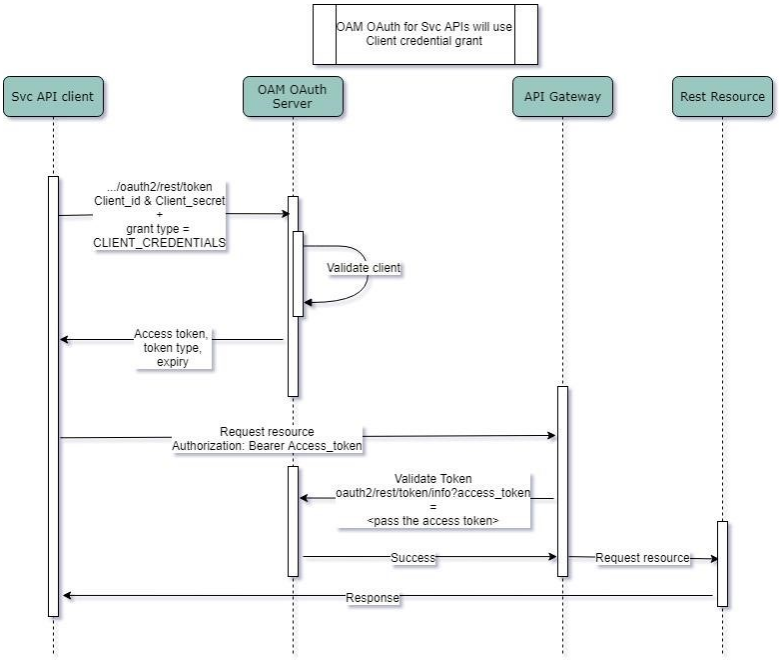
Access to this API layer is granted only via the following methods

- OAuth with OAM (Oracle Access Manager)

As stated before, in case the customer does not have OAM, an enterprise API Management layer should be implemented to protect the service API(s)

OAuth with OAM

The flow is depicted below



- API clients pass the client id & client secret and grant type as CLIENT CREDENTIALS, to get the access token , using the below endpoint o /oauth2/rest/token
- API Clients will pass the access token in the Authorization Header as Bearer token in their subsequent calls to access the Service APIs.
- API Gateway validates the client access token on OAM Authorization server If valid, it passes the request on to the Svc APIs and gets the response.
- The client can choose to get a new token (refresh) before the expiry of the current token. In case the token expires, they will pass the client Id and client secret to get a new token.

2.9.3 Two-way SSL Connection

A two-way SSL is used when the server needs to authenticate the client. In a two-way SSL connection, the client verifies the identity of the server and then passes its identity certificate to the server. The server then validates the identity certificate of the client before completing the SSL handshake.

In order to establish a two-way SSL connection, must have two certificates, one for the server and the other for client.

2.10 Securing the Switch Integration Gateway

2.10.1 Overview

Oracle FLEXCUBE Universal Banking supports communication with external channels, one of them being ATM switches. The below listed set of measures are recommended for securing the communication between the ATM switch and the Switch Integration Gateway of Oracle FLEXCUBE Universal Banking.

Refer SWITCH_INTERFACE_Installation docs for more information.

2.10.2 Securing the link to Switch Integration Gateway

The ATM Switch communicates with the Switch Integration Gateway of FLEXCUBE Universal Banking, using the ISO 8583 protocol, over a TCP/IP channel. The following measures are recommended to secure this link:

2.10.2.1 Usage of a Dedicated Channel

It is recommended to have a dedicated private link between the ATM switch and the Switch Integration Gateway of FLEXCUBE Universal Banking.

2.10.2.2 Usage of a Dedicated Server

It is recommended to have the Switch Integration Gateway deployed on a separate machine. Additionally, access to this machine is to be controlled, in accordance with the data center practices.

2.10.3 Securing the Link to the Integration Gateway

The Switch Integration Gateway communicates with the Integration Gateway of FLEXCUBE Universal Banking. Transport-level security can be employed to secure this link as described:

2.10.3.1 Usage of a secure channel

The Switch Integration Gateway can be configured to communicate with the Integration Gateway, over the T3S protocol, instead of the T3 protocol.

It is recommended to employ T3S due to the usage of TLS/SSL to encrypt the communication passing through the channel. Additional information on the same, can be obtained from the configuration document titled “Switch Interface Installation with SSL Configuration Document”.

2.11 Securing the Gateway Services

2.11.1 Overview

Different applications deployed on disparate platforms and using different infrastructure need to be able to communicate and integrate seamlessly with Oracle FLEXCUBE in order to exchange data. The Oracle FLEXCUBE Integration Gateway will cater to these integration needs.

The integration needs supported by the Gateway can be broadly categorized from the perspective of the Gateway as follows:

- Inbound application integration – used when any external system needs to add, modify or query information within Oracle FLEXCUBE
- Outbound application integration – used when any external system needs to be notified of the various events that occur within Oracle FLEXCUBE.

2.11.2 External System Maintenance

An external system needs to be defined that will communicate with the Oracle FLEXCUBE Integration Gateway. Below are the details requiring inputting while creating the external system.

External System-- Specify a name for the external system. This should be the same as the Source in an incoming message.

Description - Specify a brief description for the External System.

Request-- A way needs to be defined in which the external system should correlate its request message with the response message. Message ID can be chosen of a request message as the Correlation ID in the response message. Alternatively, user can choose Correlation ID of a request message and maintain it as the Correlation ID of the corresponding response message.

Request Message--User can choose the Request message to be 'Full Screen' or 'Input Only'. If you select 'Full Screen' as the request message, the response message will also display 'Full Screen'.

Response Message--User can choose the Response message to be 'Full Screen' or 'Record Identification Msg'.
Default Response Queue-- You can define a response queue for each of the In Queue's through which the External System will communicate with Oracle FLEXCUBE. Define a valid queue name as the Default Response Queue.

Dead Letter Queue--If the messages received are non-readable, such messages are directed to Dead Letter Queue defined for the external system.

XSD Validation Required-- Check this box to indicate if the request message should be validated against its corresponding XSD.

Register Response Queue Message ID--Check this box to indicate if the message ID provided by the Response Queue should be logged when a response message is posted into the queue.

2.11.3 Accessing Services and Operations

In a message it is mandatory to maintain a list of Service Names and Operation Codes. This information is called Gateway Operations.

A combination of every such Service Name and Operation Code is mapped to a combination of Function ID and Action. Every screen in Oracle FLEXCUBE is linked with a function ID. This information is called Gateway Functions.

User can gain access to an external system using the Gateway Functions. The Function IDs mapped in Gateway Functions should be valid Function IDs maintained in Oracle FLEXCUBE. Hence, for every new Service or Operation being introduced, it is important that you provide data in Gateway Operations and Gateway Functions.

2.11.4 Gateway Password Generation Logic for External System Authentication

As a secure configuration password authentication should be enabled for the external system maintained. The same can be verifying in External system detail screen level.

Once these features enable, system will validate for Encrypted password as part of every request sent by the External System.

The Message ID which is present as part of the header in Request XML, is considered as hash. External System generates an unique Message ID, which is functional mandatory field in the header. Create a Message Digest with SHA-512 algorithm.

The hash created from the previous step and the password in clear text together is encrypted in DESede encryption method. Apply Base64 encoding to encrypted value and send to the Oracle FLEXCUBE gateway.

3. Securing Oracle FLEXCUBE

3.1 Desktop Security

Please refer the vendor specific relevant sections for securing the DeskTops Operating system. Also do refer the Browser specific security settings mentioned in the vendor specific docs.

Refer the client browser setting required for FCUBS.

3.2 Oracle FLEXCUBE Universal Banking Controls

3.2.1 Overview

This chapter describes the various programs available within Oracle FLEXCUBE & UI Refresh, to help in the maintenance of security.

Access to the system is possible only if the user logs in with a valid ID and the correct password. The activities of the users can be reviewed by the Security Officer in the Event Log and the Violation Log reports.

3.2.2 Disable Logging

It is recommended that the debug logging facility of the application be turned off, once the system is in production. This is achieved by updating the property file of the application via the Oracle FLEXCUBE UBS & UI Refresh Installer.

The above described practice does not disable logging performed by the application in the database tier. This can be disabled by running the lockdown scripts provided. The lockdown scripts will disable logging across all modules and across all users in the system.

3.2.3 Audit Trail Report

A detailed Audit Trail is maintained by the system on all the activities performed by the user from the moment of login. This audit trail lists all the functions invoked by the user, along with the date and time. The program reports the activities, beginning with the last one. It can be displayed or printed. The records can be optionally purged once a printout is taken. This program should be allotted only to the Security Officer.

3.2.4 Security Violation Report

This program can be used to display or print the Violation Report. The report gives details of exceptional activities performed by a user during the day. The difference between the Violation Report and the Audit Trail is that the former gives details of all the activities performed by the users during the day, and the latter gives details of exceptional activities, for e.g. forced password change, unsuccessful logins, User already logged in, etc. The details given include:

- Time
- The name of the operator
- The name of the function
- The ID of the terminal
- A message giving the reason for the login

The system gives the Security reports a numerical sequence. The Security Report includes the following messages:

3.2.4.1 Sign-on Messages

Message	Explanation
User Already Logged In	The user has already logged into the system and is attempting a login through a different terminal.
User Authentication Failed	An incorrect user ID or password was entered.
User Status is Locked. Please contact your System Administrator	The user profile has been disabled due to an excessive number of attempts to login, using an incorrect user ID or password. The number of attempts could have matched either the successive or cumulative number of login failures (configured for the system).

3.2.5 Display/Print User Profile

This function provides an on-line display / print of user profiles and their access rights. The information includes:

- The type (customer / staff)
- The status of the profile - enabled or disabled or on-hold
- The time of the last login
- The date of the last password /status change
- The number of invalid login attempts
- The language code / home branch of the user

3.2.6 Clear User Profile

A user ID can get locked into the system due to various reasons like an improper logout or a system failure. The Clear User Profile function can be run by another user to reset the status of the user who got locked in. This program should be used carefully and conditionally.

3.2.7 Change User Password

Users can use this function to change their passwords. A user password should contain a minimum of six characters and a maximum of twelve characters (both parameterizable). It should be different from the current and two previous passwords. The program will prompt the user to confirm the new password when the user will have to sign-on again with the new password.

3.2.8 List of Logged-in Users

The user can run this program to see which users are in use within Oracle FLEXCUBE & UI Refresh at the time the program is being run. The information includes the following:

- The ID of the terminal
- The ID of the user
- The login time

3.2.9 Change Time Level

Time levels have to be set for both the system and the users. Ten time levels are available, 0 to 9. Restricted Access can be used to set the Users time level. The Change Time Level function can be used to do the same for the branch. A user will be allowed to sign-on to the system only if his/her time level is equal to or higher than the system time level. This concept is useful because timings for system access for a user can be manipulated by increasing the system time level. For e.g. the End of Day operators could be allotted a time level of 1, and the users could be allotted a time level of 0. If the application time-level is set at 1 during End of Day operations, only the End of Day operators will have access to the application. The other users will be denied access.

3.2.10 Authentication & Authorization

First, only authorized users can access the system with the help of a unique User ID and a password. Secondly, a user should have access rights to execute a function.
The user profile of a user contains the User ID, the password and the functions to which the user has access. FLEXCUBE & UI Refresh operation such as new, copy, query, unlock etc will be enabled based on function rights available for the user. The function rights will be checked for each operation performed by the user.

Administrator can define the maximum number of unsuccessful attempts after which a User ID should be disabled. When a User ID has been disabled, the Administrator should enable it. The password of a user can be made applicable only for a fixed period. This forces the user to change the password at regular intervals thus reducing security risks. Further, Administrator can define passwords that could be commonly used by a user as Restrictive Passwords at the user, user role and bank level. A user cannot use any password that is listed as a Restrictive Password at any of these levels.

3.2.11 Role Based Access Controls

Application level access has implemented via the Security Management System (SMS) module.
SMS supports “ROLE BASED” access of Screens and different types of operations.
FLEXCUBE Universal Banking Solutions & FLEXCUBE UI Refresh supports dual control methodology, wherein every operation performed has to be authorized by another user with the requisite rights.

3.2.12 Masking

Personally identifiable information in scoped function id’s are enhanced to display masked or unmask values depending on the user definition. Masking personally identifiable information is based on the policies created in database.

3.2.13 Granular Access

Customer and Customer Account maintenance, transaction restricted to users based on the access group restriction attached at user level for the scoped function ids. User will not be able to query, view, create or amend data based on access group restriction.

3.2.14 Right to be forgotten

Personally identifiable information of both closed Users and Customers are permanently anonymized. Once PII information is permanently anonymized corresponding Users and Customers cannot be queried from application. Right to be forgotten will be processed based on the number of days to forget customer and on customer request.

3.2.15 Access controls like branch level

User can indicate the branches from where a user can operate in the Restricted Access screen (function-ID).

3.2.16 Maker – Checker

Application supports dual control methodology, wherein every operation performed has to be authorized by another user with the requisite rights.

3.2.17 User Management

FLEXCUBE UBS enables creation of users through SMDUSRDF UI. On authorization of the newly created user, the credentials are automatically mailed to the user's email id. This reduces the risk of password been known to the administrator, who creates users for the bank.

User is forced to change the password on first login. The password supplied is hashed iteratively after being appended with a randomly generated salt value. Hashing algorithm used is of SHA-2 family and above.

User privileges are maintained by Roles. Roles definition is captured via another UI. These roles are mapped to a user in the SMDUSRDF UI. Basing on these user- roles mapping the user will have access to different modules in FLEXCUBE.

3.2.18 Access Enforcement

Access management in FLEXCUBE & UI Refresh can be done in four steps.

1. Branch level— in such a case the user cannot view even the menu list of the FCUBS when he tries to login into the restricted branch. Thus, no transactions could be performed
2. Roles wise—as described above basing on the user-roles mapping, the user can access different modules in FCUBS. For an example, a bank clerk will have access to customer creation, account opening, term-deposits opening and liquidation screens, but he will not have access to SMDUSRDF UI, which is for user creation.
3. Function-ID wise—here, the user can be restricted to launch even the UI on clicking on the menu list.
4. Product/ Account class wise— here, the user can be prevented access to certain account classes or products. This will disable him from creating any accounts or transactions using those prevented account class and product respectively.

3.2.19 Privacy controls

Tokenization mechanism is implemented in FCUBS, where the token is created for every request that hit server for avoiding forgery attacks. Also, to avoid Clickjacking and frame spoofing attack FCUBS have respective header and code configuration. Proper privacy control and content type has been placed.

3.2.20 Password Management

Certain user password related parameters should be defined at the bank level.
These parameters will apply to all the users of the system. Examples of such parameters are the number of invalid login attempts after which a user-id should be disabled, the maximum and minimum length for a password, the number of previous passwords that should not be used, the interval at which the password should be changed by every user, etc.

3.2.20.1 Invalid Logins

In FLEXCUBE & UI Refresh user should specify the allowable number of times an invalid login attempt is made by a user. Each user accesses the system through a unique User ID and password. While logging on to the system, if either the User Id or the Password is wrong, it amounts to an invalid login attempt.
By default, the allowable number of cumulative invalid attempts is six, and the allowable number of consecutive invalid attempts is three. These default values can be changed and specify the allowable number of attempts in each case. An allowable number for cumulative attempts are between 6 and 99, and for consecutive (successive) attempts are between 3 and 5.

When authentication of credentials is unsuccessful due to an incorrect user ID, then the user id will not be logged in the audit logs. In case the user id is correct and the password is wrong,
the attempt is logged in the audit log and the successive and cumulative failure count is incremented. When the user id and password are correct, this is logged into the audit logs.

3.2.20.2 Specifying Parameter

Dormancy Days

Oracle FLEXCUBE & UI Refresh allows you to automatically disable the profile of all the users who have not logged into the system for a pre-defined period of time. A user ID is considered dormant if the difference between the last login date and the current date is equal to or greater than the number of 'Dormancy Days' that has been specified. This is reckoned in calendar days i.e. inclusive of holidays. All dormant users (whose home branch is same as the current branch) are disabled during the end of day run at the current branch.

3.2.20.3 Specifying Parameters for User Passwords

Password Length (characters)

The range of length (in terms of number of characters) of a user password can be set. The number of characters in a user password is not allowed to exceed the maximum length, or fall below the minimum length that has been specified.

The minimum length defaults to 8, and the maximum length to 15. The defaults values can be changed and specify the required range. The length can specify a minimum length between 6 and 15 characters, and a maximum length between 10 and 15 characters. The minimum length that specified must not exceed the maximum length that have specified.

Force Password Change after

The password of a user can be made valid for a fixed period after which a password change should be forced. After the specified number of days has elapsed for the user's password, it is no longer valid and a password change is forced. The number of calendar days defined will be applicable for a password change of any nature - either through the 'Change Password' function initiated by the user or a forced change initiated by the system. The system defaults to a value of 30, which can be changed. The number of days can be between 15 and 180 days,

Password Repetitions

The number of previous passwords that cannot be set as the new current password can be configured, when a password change occurs. The system defaults to a value of three (i.e., when a user changes the user password, the user's previous three passwords cannot be set as the new password). The default value can be changed and it can specify a number between one and five.

Minimum Days between Password Changes

The minimum number of calendar days that must elapse between two password changes can be configured. After a user has changed the user password, it cannot be changed again until the minimum numbers of days you specify here have elapsed.

Intimate Users (before password expiry)

The number of working days before password expiry can be configured, which is used to display a warning message to the user. When the user logs into the system (the stipulated number of days before the expiry date of the password), a warning message will continue to be displayed till the password expires or till the user changes it. By default, the value for this parameter is two (i.e., two days before password expiry).

3.2.20.4 Placing Restrictions on User Passwords

Application allows placing restrictions on the number of alpha and numeric characters that can be specified for a user password.

Maximum Consecutive Repetitive Characters

The maximum number of allowable repetitive characters occurring consecutively, in a user password can be specified. This specification is validated whenever a user changes the user password, and is applicable for a password change of any nature - either through the 'Change Password' function initiated by the user or a forced change initiated by the system.

Minimum Number of Special Characters in Password

Application allows defining minimum number of special characters allowed in a user password. The system validates these specifications only when a user chooses to change the password. Following is the default value application used:

Minimum No of Special Characters = 1

Minimum Number of Numeric Characters in Password

Likewise, application allows defining the minimum number of numeric characters allowed in a user password. The system validates the password only when a user chooses to change his password. Following is the default value used:

Minimum No of Numeric Characters = 1

Minimum Number of Lower Case Characters in Password

The minimum number of lowercase characters allowed in a user password also can be configured. The allowed lower case characters are from the US-ASCII character set only. The system validates these specifications only when a user chooses to change the password

Following is the default value used:

Minimum No of Lower Case Characters = 1

Minimum Number of Upper Case Characters in Password

The minimum number of upper case characters allowed in a user password can be configured. The allowed upper case characters are from the US-ASCII character set only. The system validates these specifications only when a user chooses to change the password.

Following is the default values used:

Minimum No of Upper Case Characters = 1

3.2.20.5 Password Restrictions

Application allows defining a list of passwords that cannot be used by any user of the system in the bank. This list, called the Restrictive Passwords list can be defined at three levels:

- At the bank level (applicable to all the users of the system)
- At the user role level (applicable for all the users assigned the same role)
- At the user level (applicable for the user)

The list of Restrictive Passwords should typically contain those passwords the users are most likely to use: the name of your bank, city, country, etc. For a user role, it could contain names, or terms, that are commonly used in the department. At the user level, it could contain the names of loved ones, etc. By disallowing users from using such common passwords, you can reduce the risk of somebody other than the user knowing the password.

4.1 Cryptography

FLEXCUBE & UI Refresh uses cryptography to protect the sensitive data..
It uses Hashing algorithm while storing user passwords. SHA-2 family hashing algorithm is used for the purpose. SHA-256 algorithm produces 32 bytes hash value.

For encryption, AES, which is considered to be of gold standard, is used. It produces a key size of 128 bits when it comes to symmetric key encryption.

4.2 Security patch

Security patches needs to be applied whenever it's available for the applicable product version.

4.3 Oracle Database Security Suggestions

Access Control

Database Vault (DV) Provides enterprises with protection from the insider threats and in advantage leakage of sensitive application data. Access to application data by users and administrators is controlled using DV realms, command rules and multi factor authorization. DV also address Access privilege by separating responsibilities.

Data Protection

Advance Security provides the most advance encryption capabilities for protecting sensitive information without requiring any change to the application. TDE is native database solution that is completely transparent to the existing applications. TDE encrypts sensitive data stored in data files. To prevent unauthorized decryption, TDE stores the encryption keys in a security module external to the database.

Advance Security also provides strong protection for data in transit by using network encryption capabilities. Features like Easy to deploy, Ensure secure by default to accept communication from client using encryption, Network encryption using SSL/TLS.

Oracle Secure Backup (OSB)

OSB is tightly integrated with the Oracle database, hence provides optimal security and performance, eliminating backup of any associated database UNDO data. Supports Comprehensive tape backup solutions for Oracle database and file systems. Provides single point of control for enterprise-wide tape backup and associated encryption key.

Monitoring and Compliance

Audit Vault (AV) transparently collects and consolidate audit data from multiple databases across the enterprise, does provide valuable insight into who did what with which data & when including privilege users. The integrity of the audit data is ensured using controls including DV, Advance Security. Access to AV data is strictly controlled. It also does provide graphical summaries of activity causing alerts, in addition database audit setting are centrally managed and monitored.

4.4 Oracle Software Security Assurance - Standards

Every acquired organization must complete the Mergers and Acquisitions (M&A) Security Integration process. The issues identified during this review must be addressed according to the agreed upon M&A remediation plan. The acquired organization must complete SPOC assignments and plan integration of OSSA methodologies and processes into its SDLC.

4.5 Oracle Digital Assistant Integration

Application supports Integration of Oracle Digital Assistant (ODA) with FLEXCUBE UBS Application. The ChatServer configuration to be in secure mode or Cloud Instance of ChatServer details to be configured with application. The communication happens between application and ChatServer are using secure protocol.

enableSecureConnection: true,

To enable secure connection the above configuration should be true.

4.6 References

4.6.1 Datacenter Security considerations

Please refer to the following links to understand Datacenter Security considerations

http://docs.oracle.com/cd/B14099_19/core.1012/b13999/rectop.htm

4.6.2 Database Security considerations

Please refer the below links to understand more on Database Security considerations recommended to be followed

http://www.oracle.com/us/products/database/security/overview/index.html

http://www.oracle.com/technetwork/database/security/twp-security-checklist-database-1-132870.pdf

4.6.3 Security recommendations / practices followed for Database Environment

Please refer the below mentioned links to understand more on Security recommendations / practices followed for Database Environment

http://docs.oracle.com/cd/B28359_01/network.111/b28531/guidelines.htm

4.6.4 Common security considerations

Please refer below links to understand some of the common security considerations to be followed

http://docs.oracle.com/cd/B14099_19/core.1012/b28654.pdf

http://docs.oracle.com/cd/E14899_01/doc.9102/e14761/tuningforappserver.htm

http://docs.oracle.com/cd/E13222_01/wls/docs81b/lockdown/practices.html

http://docs.oracle.com/cd/E23943_01/web.1111/e14529/security.htm

http://www.oracle.com/us/solutions/oos/weblogic-server/overview/index.html

Security Guide
[May] [2020]
Version 14.4.0.0.0

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/