

# Oracle® Banking Cash Management Security Guide



Release 14.7.0.0.0

F75202-01

November 2022

The Oracle logo, consisting of a solid red square with the word "ORACLE" in white, uppercase, sans-serif font centered within it.

ORACLE®

Copyright © 2020, 2022, Oracle and/or its affiliates.

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.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. 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 fail-safe, 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.

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

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about 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 unless otherwise set forth in an applicable agreement between you and Oracle. 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, except as set forth in an applicable agreement between you and Oracle.

# Contents

## Preface

---

## 1 Prerequisites

---

1.1	Operating Environment Security	1-1
1.2	Network Security	1-1
1.3	Oracle Database Security	1-1
1.4	Application Server Security	1-2
1.5	SSL Support	1-3
1.6	Securing the Oracle Banking Cash Management Application	1-4
1.6.1	Online Web Application	1-4
1.6.2	API Security	1-11
1.6.3	Two-way SSL Connection	1-11

## 2 Securing the Oracle Banking Cash Management Application

---

2.1	Desktop Security	2-1
2.2	Oracle Banking Cash Management Controls	2-1
2.2.1	Overview	2-2
2.2.2	Disable Logging	2-2
2.2.3	Sign-on Messages	2-2
2.2.4	Authentication and Authorization	2-2
2.2.5	Role Based Access Controls	2-3
2.2.6	Access Controls - Branch Level	2-3
2.2.7	Maker – Checker	2-3
2.2.8	Access Enforcement	2-3
2.2.9	Password Management	2-3

## 3 General Information

---

3.1	Cryptography	3-1
3.2	Security Patch	3-1
3.3	Oracle Database Security Suggestions	3-1

## 4 References

---

# Preface

## Introduction

This guide provides security-related usage and configuration recommendations for Oracle Banking Cash Management. It also describes the procedures required to implement or secure certain features, but it is not a general-purpose configuration manual.

## Audience

This guide is primarily intended for IT department or administrators deploying Oracle Banking Cash Management and Third-party or vendor software's. It includes the information related to IT decision makers and users of the application.



### Note:

Readers are expected to have basic operating system, network, and system administration skills with an awareness of vendor/third-party software's and knowledge of Oracle Banking Cash Management application.

## Scope

### 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.

### 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.

### Limitations

The 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.

### 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.

**Acronyms and Abbreviations**

The list of acronyms and abbreviations used in this guide are as follows:

**Table 1 Acronyms**

Abbreviation	Description
<b>JWE</b>	JSON Web Encryption
<b>JWS</b>	JSON Web Signature
<b>JWT</b>	JSON Web Token
<b>OAM</b>	Oracle Access Manager
<b>OSSA</b>	Oracle Software Security Assurance
<b>SAML</b>	Security Assertion Mark-up Language
<b>SSO</b>	Single Sign-On
<b>SSL</b>	Secure Sockets Layer

**List of Topics**

This guide is organized as follows:

**Table 2 List of Topics**

Topics	Description
<a href="#">Prerequisites</a>	This topic provides the information about prerequisites.
<a href="#">Securing the Oracle Banking Cash Management Application</a>	This topic provides the information about securing Oracle Banking Cash Management.
<a href="#">General Information</a>	This topic provides the information about general information related to security.

# 1

## Prerequisites

This topic describes about the Prerequisite.

This topic contains the following subtopics:

- [Operating Environment Security](#)  
This topic describes about operating environment security.
- [Network Security](#)  
This topic describes about network security.
- [Oracle Database Security](#)  
This topic describes about oracle database security.
- [Application Server Security](#)  
This topic describes about application server security.
- [SSL Support](#)  
This topic provides the information for SSL Support.
- [Securing the Oracle Banking Cash Management Application](#)  
This topic describes about securing the Oracle Banking Cash Management.

### 1.1 Operating Environment Security

This topic describes about operating environment security.

Refer to the vendor specific document, to make the environment safe and secured.

### 1.2 Network Security

This topic describes about network security.

Refer to the vendor specific document to make the environment safe and secured.

### 1.3 Oracle Database Security

This topic describes about oracle database security.

Refer to the Oracle Database Security specification document to make the environment more safe and secured.

#### **Oracle Banking Cash Management Recommended configuration**

This contains security recommendations for the Database used for Oracle Banking Cash Management.

**Table 1-1 Security Recommendations**

File Name	Property	Feature
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.

## 1.4 Application Server Security

This topic describes about application server security.

Refer to the Oracle Web Logic Security specification document for make the environment more safer and secure.

Oracle Banking Cash Management support the following authentication schemes for the online web application.



- Standard LDAP Directory (e.g., OUD/AD/Embedded Weblogic)
- 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 Banking Cash Management Solution supports the following authentication scheme for the API layer.

- OAuth (CLIENT CREDENTIALS) with OAM
- OAuth (CLIENT CREDENTIALS) without OAM

In case the customer does not have OAM, they can use OAUTH without OAM or it is expected that the customer has an enterprise API Management Layer that protects Oracle Banking Cash Management API layer with the same controls (i.e., OAuth)

### Support for SSL (Secure Transformation of Data)

The Oracle Banking Cash Management should 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 in a production environment, especially when WebLogic Server acts as the SSL terminator.

## 1.5 SSL Support

This topic provides the information for SSL Support.

### SSL Setup

Refer to **SSL Setup Guide** for the setup details.

### 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 128-bit symmetric keys or higher and using RSA for key exchange.

```
....  
<ssl>  
    <enabled>true</enabled>  
    <ciphersuite>TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256</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).

#### Product configurations for SSL

Refer to **Oracle Banking Microservices Architecture Deployments** section in *Oracle Banking Microservices Architecture Installer Guide* for SSL Configuration.

## 1.6 Securing the Oracle Banking Cash Management Application

This topic describes about securing the Oracle Banking Cash Management.

This topic contains the following subtopics:

- [Online Web Application](#)  
This topic describes about the online web application.
- [API Security](#)  
This topic describes about API Security.
- [Two-way SSL Connection](#)  
This topic describes about the two-way SSL connection.

### 1.6.1 Online Web Application

This topic describes about the online web application.

Authentication and authorization to requests to access the Online Web Application (*appshell*) are controlled using the below industry standard approaches:

- Standard LDAP Directory authentication
- SSO with OAM
- SSO with other External SSO Agents
- SAML with the Oracle Banking Cash Management application acting as the service provider
- SAML SSO Integration

#### JWT (JSON Web Tokens)

In addition to the authentication, the Oracle Banking Cash Management 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 plain text 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.

The policies for JWT are as follows:

- **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.

**Table 1-2 JWT Claims set**

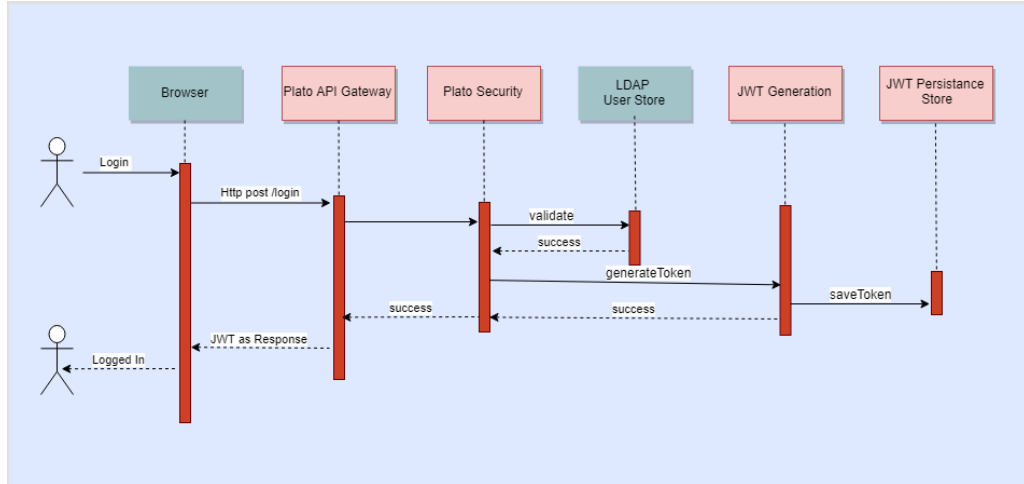
Claim Name	Description	Mandatory	Type
Iss	Issuer	Yes	Registered
Sub	Subject	Yes	Registered
Aud	Audience	No	Registered
Exp	Expiration Time	Yes	Registered
Nbf	Not Before	No	Registered
Iat	Issued At	Yes	Registered
Jti	JWT Id	Yes	Registered
Tid	Tenant Id	Yes	Private

- **Token Expiry:** Platform security module invalidates the token, if the client submits after the Expiration time.
- **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 are as follows:

## LDAP Authentication

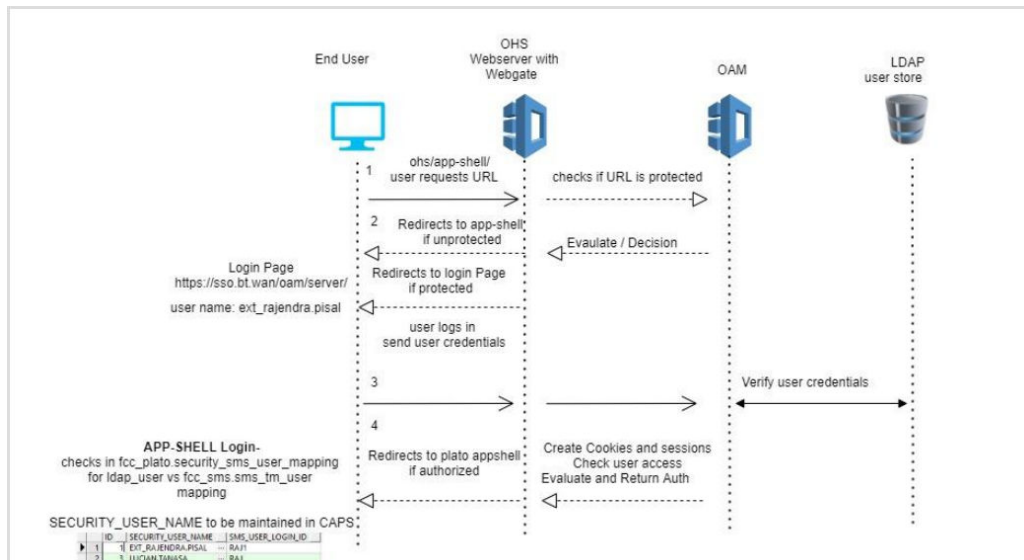
Figure 1-1 LDAP Authentication



- The user is presented the standard login page for the Oracle Banking Microservices Architecture products
- 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.

## OAM Based SSO

Figure 1-2 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

**Product configuration**

The following parameters need to be set to enable a successful integration with OAM as SSO in Oracle Banking Cash Management products.

PLATO.SECURITY\_CONFIG table

USER\_HEADER\_ATTRIBUTE\_KEY,IS\_SSO\_CONFIGURED USER\_MAPPING\_REQUIRED to be set as true.

ID	KEY	VALUE
1	USER_HEADER_ATTRIBUTE_KEY	userId
2	USER_HEADER_ATTRIBUTE_REQUIRED	Y
3	IS_SSO_CONFIGURED	true
4	USER_MAPPING_REQUIRED	true

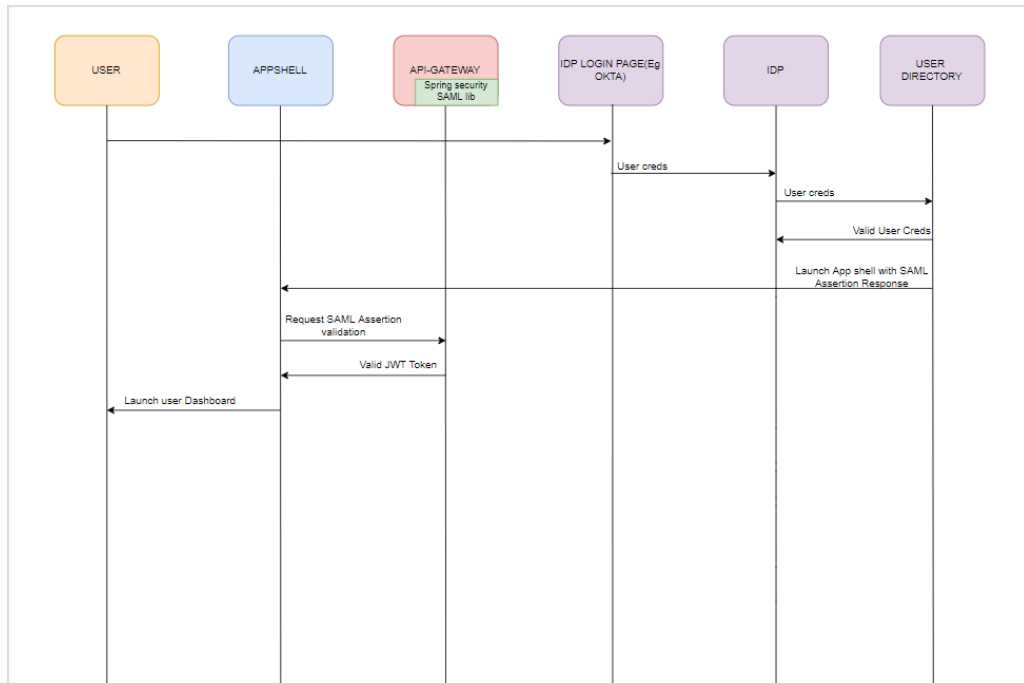
**Figure 1-3 PLATO.SECURITY\_SMS\_USER\_MAPPING table**

	ID	SECURITY_USER_NAME	SMS_USER_LOGIN_ID
▶	1	EXT_RAJENDRA.PISAL ...	RAJ1
	2	LUCIAN.TANASA ...	RAJ

**SAML Authentication**

**IDP Initiated SAML Authentication**

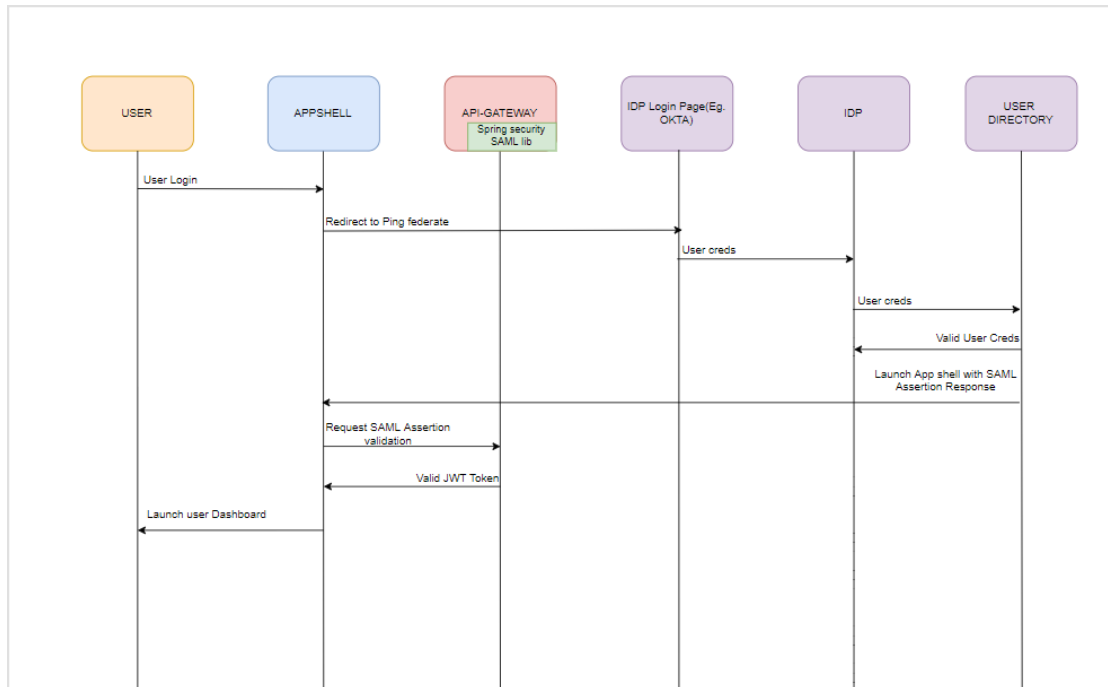
**Figure 1-4 IDP Initiated SAML Authentication**



- The Identity Provider is external to the Oracle Banking Cash Management (e.g. OKTA) with the Oracle Banking Microservices Architecture products acting as the Service Provider.
- Client requests protected resource from Oracle Banking Cash Management. 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.
- It is possible to configure an external service to do the SAML Verification instead of the API Gateway using the **EXTERNAL\_SSO\_VALIDATION\_URL** parameter in the **SECURITY\_CONFIG** table in **PLATO-SECURITY** schema.

**SP Initiated SAML Authentication**

**Figure 1-5 SP Initiated SAML Authentication**

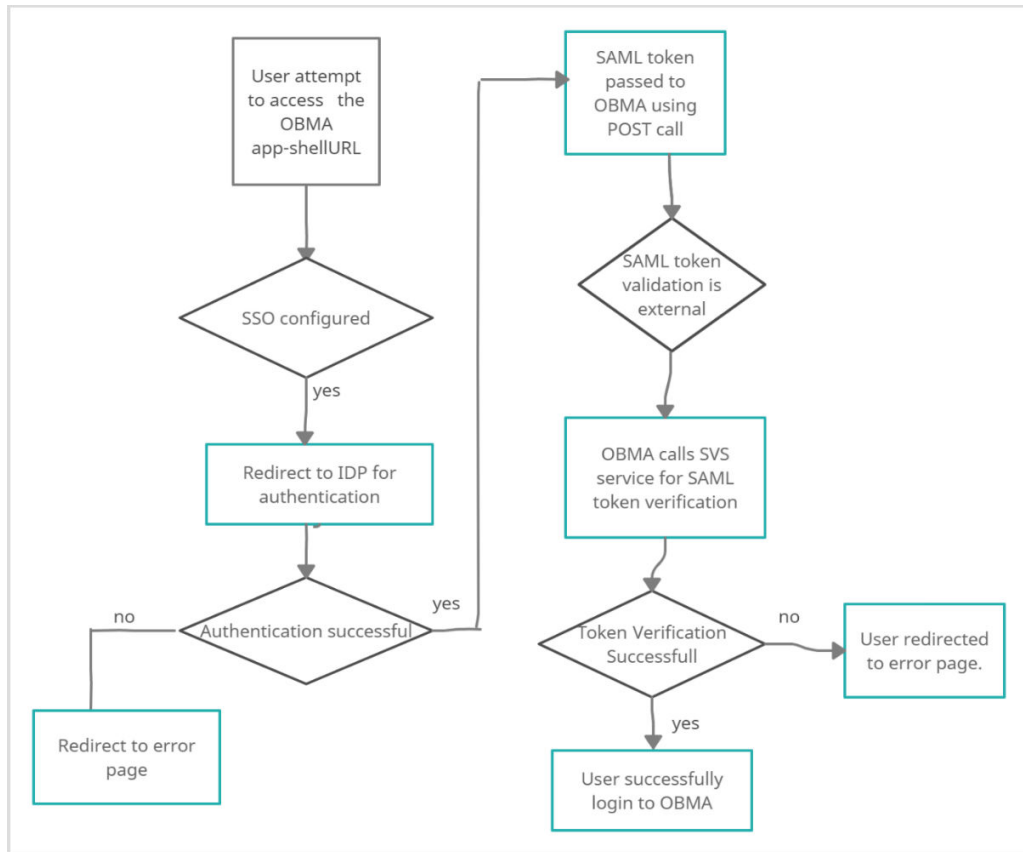


- The user initiates a call to the Oracle Banking Cash Management application and is redirected to the federate login page of the bank.
- The Identity Provider is external to the Oracle Banking Cash Management (e.g. OKTA) with the Oracle Banking Microservices Architecture products acting as the Service Provider
- 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.

### SAML SSO Implementation

It is possible to configure an external service to do the SAML Verification instead of the API Gateway.

**Figure 1-6 SAML SSO Implementation**



**Steps to achieve SSO-SAML Authentication:**

- Bank user will try to access the Oracle Banking Cash Management app-shell URL.
- Oracle Banking Cash Management will check if the IS\_SSO\_CONFIGURED parameter is set to true in the SECURITY\_CONFIG table.
- If the IS\_SSO\_CONFIGURED parameter is true the user will be redirected to the IDP for authentication.
- On successful authentication IDP will generate the SAML token and pass the token to the Oracle Banking Cash Management assertion consumer service URL in the body of POST method through EXTERNAL\_SSO\_KEY parameter.
- Oracle Banking Cash Management will receive the token and check if the SSO\_SERVICE\_PROVIDER is set to EXTERNAL in the SECURITY\_CONFIG table.
- If SSO\_SERVICE\_PROVIDER is EXTERNAL, Oracle Banking Cash Management would make a HTTP Post call to SVS using the EXTERNAL\_SSO\_VALIDATION\_URL configured in the SECURITY\_CONFIG table for SAML token validation. Oracle Banking Cash Management will pass the SAML token through EXTERNAL\_SSO\_TOKEN\_KEY parameter in the body of the POST to SVS.
- SVS will return a XML response with IsValid tag as TRUE or FALSE. If the tag value is TRUE, Oracle Banking Cash Management would generate JWT token



using the user id from the <subject> </subject> tag of SVS response and allow the user to login.

- In case of failure, Oracle Banking Cash Management would give login error to the user.

**Product Configurations required:**

The following parameters needs to be configured in the SECURITY\_CONFIG table in the PLATO-SECURITY schema to enable SAML SSO.

KEY	VALUE
IS_SSO_CONFIGURED	True
JWT_EXP_SECONDS	JWT expiry time
JWT_ALGORITHM	HS512
EXTERNAL_SSO_VALIDATION_URL	SVS URL
EXTERNAL_SSO_KEY	Parameter in which the SAML token will be passed to Oracle Banking Microservices Architecture from IDP after user authentication.
SSO_SERVICE_PROVIDER	EXTERNAL
EXTERNAL_SSO_TOKEN_KEY	Parameter in which the SAML token will be passed to SVS URL for token validation.
HEADERS	Request headers for making HTTP call to SVS URL

**FCUBS integration with Oracle Banking Microservices Architecture as SSO Provider**

Refer to Launching **Oracle Banking Cash Management** from UBS section in the **Oracle Banking Cash Management Installation Guide**.

## 1.6.2 API Security

This topic describes about API Security.

Refer to **Oracle Banking Cash Management API Security Guide** for the detailed explanation.

## 1.6.3 Two-way SSL Connection

This topic describes about the 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.

To establish a Two-way SSL connection, user must have two certificates as follows:

- Server
- Client

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

- Cookies are set with Http only as true
- Cookie secure flag set to true

- Cookie path to refer to deployed application

```
<wls:session-descriptor>  
  <wls: cookie-http-only>true</wls: cookie-http-only>  
</wls: session-descriptor>
```

```
<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.

# 2

## Securing the Oracle Banking Cash Management Application

This topic describes about securing the Oracle Banking Cash Management Application.

This topic contains the following subtopics:

- [Desktop Security](#)  
This topic describes about desktop security.
- [Oracle Banking Cash Management Controls](#)  
This topic describes about Oracle Banking Cash Management controls.

### 2.1 Desktop Security

This topic describes about desktop security.

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 Oracle Banking Cash Management.

### 2.2 Oracle Banking Cash Management Controls

This topic describes about Oracle Banking Cash Management controls.

This topic contains the following subtopics:

- [Overview](#)  
This topic describes describes the various programs available within Oracle Banking Cash Management, to help in the maintenance of security.
- [Disable Logging](#)  
This topic describes about disabling the logging.
- [Sign-on Messages](#)  
This topic lists the sign-on messages and its explanations.
- [Authentication and Authorization](#)  
This topic describes about the authentication and authorization to have the access to the system.
- [Role Based Access Controls](#)  
This topic describes about role based access controls.
- [Access Controls - Branch Level](#)  
This topic describes about access controls at branch levels.
- [Maker – Checker](#)  
This topic describes about maker and checker.
- [Access Enforcement](#)  
This topic describes about access enforcement.

- [Password Management](#)  
This topic describes about password management.

## 2.2.1 Overview

This topic describes describes the various programs available within Oracle Banking Cash Management, 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

## 2.2.2 Disable Logging

This topic describes about disabling the 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 logback.xml file of the application.

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

## 2.2.3 Sign-on Messages

This topic lists the sign-on messages and its explanations.

**Table 2-1 Sign on Messages**

Message	Explanation
User Authentication Failed/Invalid Login	An incorrect <b>user ID</b> or <b>password</b> 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 <b>user ID</b> or <b>password</b> . The number of attempts could have matched either the successive number of login failures (configured for the system).

## 2.2.4 Authentication and Authorization

This topic describes about the authentication and authorization to have the access to the system.

Only authenticated users can access the system.

A user must have access rights to execute a function. The user profile of a user contains the User ID and the functions to which the user has access. Oracle Banking Cash Management operation such as new, copy, query, unlock and so on will be enabled based on function rights available for the user. The function rights will be checked for each operation performed by the user, in Security Management Service module of Oracle Banking Cash Management.

## 2.2.5 Role Based Access Controls

This topic describes about 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.
- Oracle Banking Cash Management supports dual control methodology, wherein every operation performed has to be authorized by another user with the requisite rights.
- SMS provides an option to map multiple roles for a user in a given branch. Allowed operations are mapped to the roles and SMS authorizes the user based on it.

## 2.2.6 Access Controls - Branch Level

This topic describes about access controls at branch levels.

Security Management System provides the branch level access through the roles provided for the user at a particular branch.

## 2.2.7 Maker – Checker

This topic describes about maker and checker.

The application supports dual control methodology, in which another user must have the necessary rights for each operation performed.

## 2.2.8 Access Enforcement

This topic describes about access enforcement.

Access management in Oracle Banking Cash Management can be done in two steps.

- **Branch level:** In such a case the user cannot view even the menu list of the Oracle Banking Cash Management when he tries to login into the restricted branch. Thus, no transactions could be performed.
- **Roles wise:** As described above based on the user-roles mapping, the user can access different functions of Oracle Banking Cash Management. 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 User Creation function activity.

## 2.2.9 Password Management

This topic describes about password management.

The Oracle Banking Cash Management relies on external password management and does not store any credentials. If an external LDAP is used, password management and policy rules can be set on that (For ex. For Weblogic Embedded-LDAP, the user and password rules can be configured via the admin console of the weblogic). If OIM/OAM is configured, password management and policy rules can be set on OIM. The IdP (Identity Provider) in case of SAML takes care of the password policies.

Certain user password related parameters should be defined at the system 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

### Password Policies

To enable password validation, there is a flag given in SECURITY\_CONFIG table called:

PASSWORD\_VALIDATION\_FLAG – It has to be set as Y to enable

Password validation criteria are configurable through the table created called SECURITY\_PASSWORD\_VAL\_CONFIG. Each property in that is being explained through the following table:

Property	Value	Description
MIN_PSWD_LEN	Any integer	Minimum password length required
MAX_PSWD_LEN	Any integer	Maximum password length allowed
MIN_PSWD_AGE	Any integer	Not used currently
MAX_PSWD_AGE	Any integer	Not used currently
FLAG_UPPER_CHAR	Y/N	Y- UpperCase characters required
NUM_MAND_UPPER	Integer	Minimum uppercase characters required Checked only if FLAG_UPPER_CHAR is set to Y
FLAG_LOWER_CHAR	Y/N	Y- LowerCase characters required
NUM_MAND_LOWER	Integer	Minimum lowercase characters required Checked only if FLAG_LOWER_CHAR is set to Y
FLAG_SPECIAL_CHAR	Y/N	Y- Special characters required
NUM_MAND_SPECIAL	Integer	Minimum special characters required Checked only if FLAG_SPECIAL_CHAR is set to Y
FLAG_NUMERIC_CHAR	Y/N	Y- Numeric characters required
NUM_MAND_NUMERIC	Integer	Minimum numeric characters required Checked only if FLAG_NUMERIC_CHAR is set to Y

# 3

## General Information

This topic describes about the General information.

This topic contains the following subtopics:

- [Cryptography](#)  
This topic describes about cryptography.
- [Security Patch](#)  
This topic describes about security patch.
- [Oracle Database Security Suggestions](#)  
This topic describes provides suggestions about Oracle database security.
- [Oracle Software Security Assurance - Standards](#)  
This topic describes about standards of Oracle Software Security Assurance.

### 3.1 Cryptography

This topic describes about cryptography.

Oracle Banking Cash Management uses cryptography to protect the sensitive data.

For encryption, AES, which is considered to be of gold standard, is used. It produces a key size of 256 bits when it comes to symmetric key encryption.

### 3.2 Security Patch

This topic describes about security patch.

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

### 3.3 Oracle Database Security Suggestions

This topic describes provides suggestions about Oracle database security.

#### **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.

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.

### **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.

## **3.4 Oracle Software Security Assurance - Standards**

This topic describes about standards of Oracle Software Security Assurance.

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

## References

This topic provides the references for different types of considerations.

### **Datacenter Security considerations**

Refer to the following links to understand Datacenter Security considerations

<https://docs.oracle.com/en/middleware/fusion-middleware/weblogic-server/12.2.1.4/depdgd/understanding.html#GUID-F6E8BF0B-FBCF-44D2-A33F-13C4EF2E0031>

### **Database Security considerations**

Refer the below links to understand more on Database Security considerations recommended to be followed:

<https://www.oracle.com/security/database-security/>

<https://docs.oracle.com/en/database/oracle/oracle-database/19/dbseg/database-security-guide.pdf>

### **Security recommendations and practices followed for Database Environment**

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

<https://docs.oracle.com/en/database/oracle/oracle-database/19/security.html>

<https://docs.oracle.com/en/database/oracle/oracle-database/19/dbseg/index.html>

### **Common security considerations**

Refer the below links to understand some of the common security considerations to be followed:

<https://www.oracle.com/database/technologies/high-availability/fusion-middleware-maa.html>

<https://www.oracle.com/a/tech/docs/tip4847-maa-best-practices-for-database.pdf>

<https://docs.oracle.com/en/middleware/fusion-middleware/weblogic-server/12.2.1.4/perfm/basics.html#GUID-178B107B-10E9-4563-BCA4-E06E14F5D3FF>

<https://docs.oracle.com/en/middleware/fusion-middleware/weblogic-server/12.2.1.4/lockd/securing-production-environment-oracle-weblogic-server.pdf>

<https://docs.oracle.com/en/middleware/fusion-middleware/weblogic-server/12.2.1.4/secmg/index.html>

<https://docs.oracle.com/en/middleware/fusion-middleware/weblogic-server/12.2.1.4/index.html>