Oracle® Banking Microservices Architecture

Security Guide

Release 14.7.5.0.0
G15007-01
September 2024

ORACLE

Oracle Banking Microservices Architecture Security Guide, Release 14.7.5.0.0

G15007-01

Contents

Preface

Purpose

This guide provides security-related usage and configuration recommendations for Oracle Banking Microservices Architecture. 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 Microservices Architecture 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 Microservices Architecture 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.

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 that have purchased support 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.

Critical Patches

Oracle advises customers to get all their security vulnerability information from the Oracle Critical Patch Update Advisory, which is available at Critical Patches, Security Alerts and Bulletins. All critical patches should be applied in a timely manner to make sure effective security, as strongly recommended by Oracle Software Security Assurance.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Related Resources

For more information on any related features, refer to the following documents:

- Oracle Banking Microservices Architecture Product User Guides

- Oracle Banking Microservices Architecture API Security Guide

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Acronyms and Abbreviations

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

Table 1 Acronyms

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

1
Prerequisite

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 Secuirty
  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 Microservices Architecture Application
  This topic describes about securing the Oracle Banking Microservices Architecture.

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 Secuirty

This topic describes about oracle database security.

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

This topic contains the following subtopics:

- Oracle Banking Microservices Architecture Recommended configuration
  This topic contains security recommendations for the Database used for Oracle Banking Microservices Architecture.

1.3.1 Oracle Banking Microservices Architecture Recommended configuration

This topic contains security recommendations for the Database used for Oracle Banking Microservices Architecture.

Table 1-1 Security Recommendations

Field 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 Microservices Architecture 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 Banking Microservices Architecture 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 Microservices Architecture API layer with the same controls (i.e., OAuth)

Support for SSL (Secure Transformation of Data)

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

1.5 SSL Support

This topic provides the information for SSL Support.

This topic contains the following subtopics:

- SSL Setup
  This topic provides the information for SSL Setup.
- Choice of the SSL cipher suite
  This topic describes about choice of the SSL cipher suite.
- Product configurations for SSL
  This topic provides the information for Product configurations for SSL.

1.5.1 SSL Setup

This topic provides the information for SSL Setup.

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

1.5.2 Choice of the SSL cipher suite

This topic describes about choice of the SSL cipher suite.

Oracle WebLogic Server allows SSL clients to initiate SSL connection with a null cipher suite. The null cipher suite does not employ use any bulk encryption algorithm, as a result of which all data is transmitted over the wire.

The default configuration of Oracle WebLogic Server is to disable the null cipher suite. Make sure 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.

Below is an example for config.xml that restricts the cipher suites to those supporting 128-bit symmetric keys or higher. It uses RSA for key exchange.

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

- The configuration of the WebLogic Server to support the above cipher suites requires passing an additional command line argument to the WebLogic Server so that the 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 must be done for every managed server.
- The order of cipher suites is important. The Oracle WebLogic Server selects the first cipher suite in the list, which is also has client support.
- Cipher suites with RC4 are enabled despite it being second best to AES. This is mainly for older clients that do not support AES. For example, Microsoft Internet Explorer 6, 7, and 8 on Windows XP.

1.5.3 Product configurations for SSL

This topic provides the information for Product configurations for SSL.

Refer to **Oracle Banking Microservices Architecture Deployments** section in *Oracle Banking Microservices Platform Foundation Installation Guide* for SSL Configuration.

1.6 Securing the Oracle Banking Microservices Architecture Application

This topic describes about securing the Oracle Banking Microservices Architecture.

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 Microservices Architecture application acting as the service provider
- SAML SSO Integration

JWT (JSON Web Tokens)

In addition to the authentication, the Oracle Banking Microservices Architecture online web application uses JWT to maintain the state for authenticated users.

JSON Web Tokens are an open and industry-standard RFC 7519 method that secures the claims securely between two parties. JWT is a compact and URL-safe for transferring claims between two parties. The claims in JWT are encoded as a JSON object, which is used as the payload of the JWS or as plain text of the JWE structure, allowing 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 backends.
- No cookies required, it is 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 security and better usability, every authentication/refresh request is secured by a 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:** The Platform security module hashes the JWT footer with HS512 algorithm.
- **Refresh Token:** The 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 the below claims during the process.

Table 1-2 JWT Claims set

Claim Name	Description	Mandatory	Type
iss	Issuer	Yes	Registered

Table 1-2 (Cont.) JWT Claims set

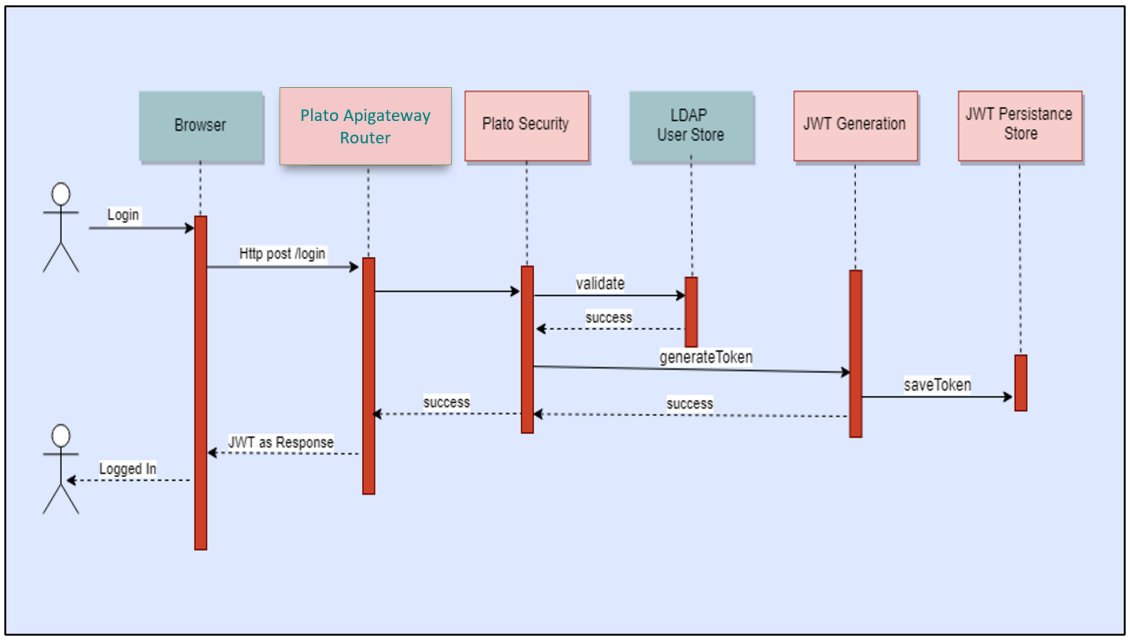
Claim Name	Description	Mandatory	Type
sub	Subject	Yes	Registered
aud	Audience	No	Registered
exp	Expiration Time	Yes	Registered
nbf	Not Before	No	Registered
jat	Issued At	Yes	Registered
jti	JWT Id	Yes	Registered
tid	Tenant Id	Yes	Private

- **Token Expiry:** The 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 will no longer be used for any purpose.

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

LDAP Authentication

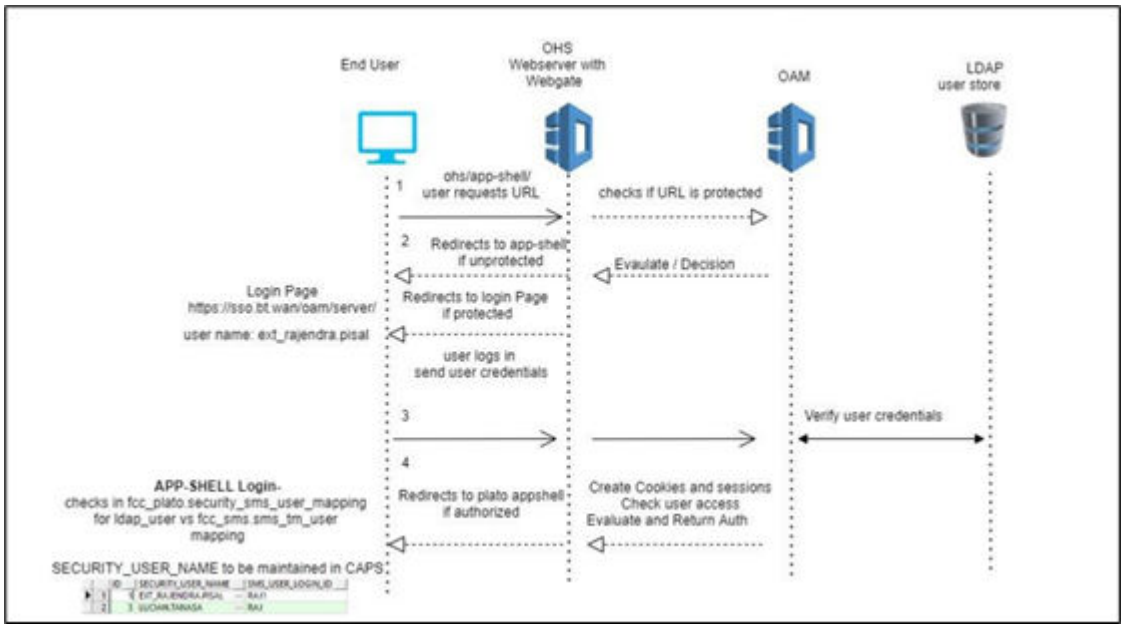
Figure 1-1 LDAP Authentication



- The user is presented the standard login page for the Oracle Banking Microservices Architecture application.
- The user enters the **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 **User Id** and **Password**. In case of success, OAM sets the corresponding user profile details in the security context.
- The request is sent to plato-apigateway-router and request will be routed to the Gateway.
- The API Gateway creates a JWT token (Using Oracle Security Developer Toolkit part of Oracle Platform Security Services), maintains 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 Microservices Architecture 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

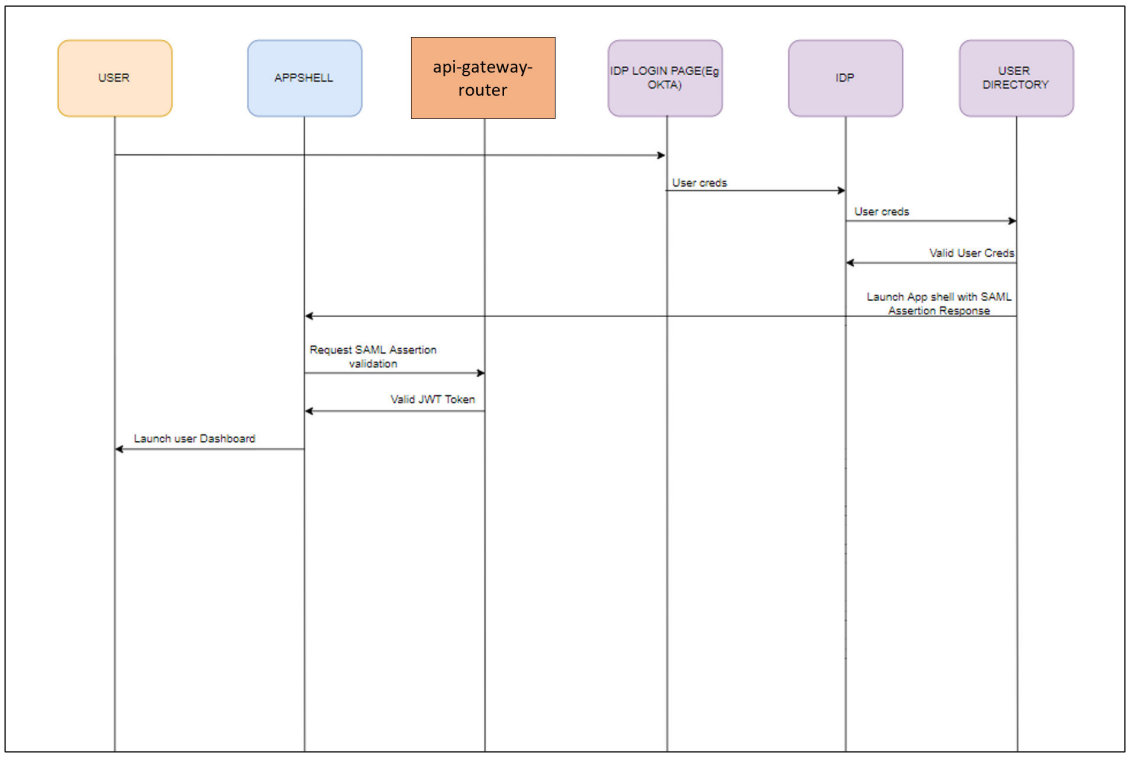
Figure 1-3 PLATO.SECURITY_SMS_USER_MAPPING table

	ID	SECURITY_USER_NAME	SMS_USER_LOGIN_ID
▶ 1	1	EXT_RAJENDRA.PISAL	RAJ1
2	3	LUCIAN.TANASA	RAJ

SAML Authentication

IDP Initiated SAML Authentication

Figure 1-4 IDP Initiated SAML Authentication

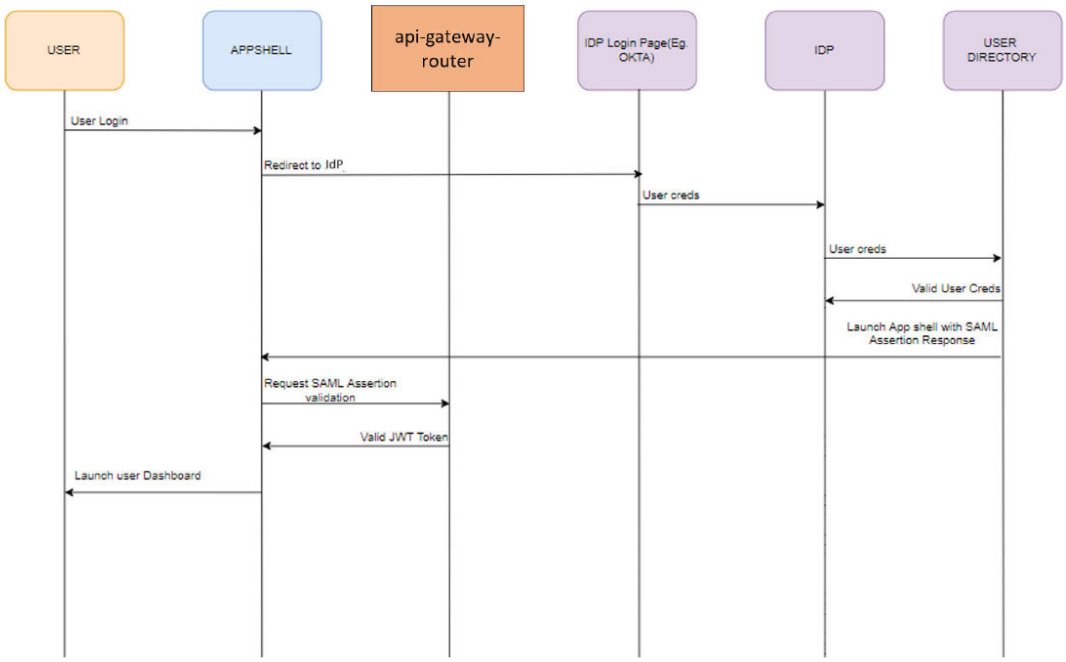


- The Identity Provider is external to the Oracle Banking Microservices Architecture application with the Oracle Banking Microservices Architecture application acting as the Service Provider. For example, OKTA.
- Client requests protected resource from Oracle Banking Microservices Architecture. The Idp 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 sent to plato-apigateway-router and request will be routed to the Gateway.
- 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.
- Configure an external service to do the SAML Verification in API Gateway with **EXTERNAL_SSO_VALIDATION_URL** parameter in the **SECURITY_CONFIG** table in **PLATO-SECURITY** schema.

- The implementation team needs to develop an external service to validate the SAML token.

SP Initiated SAML Authentication

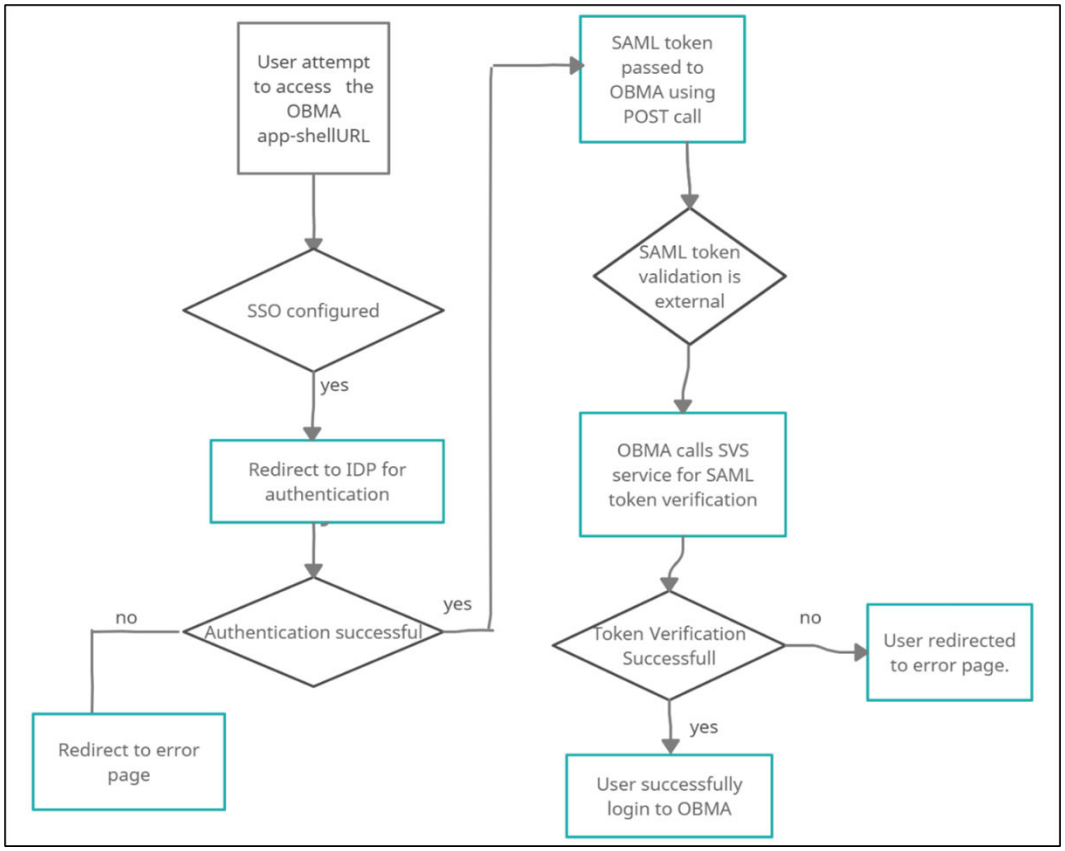
Figure 1-5 SP Initiated SAML Authentication



- The user initiates a call to the Oracle Banking Microservices Architecture application and is redirected to the federate login page of the bank.
- Oracle Banking Microservices Architecture does not have any code for the IDP Redirection. This has to be completely customized by the implementation team where a new service file has to be built and to be deployed which can perform the redirection.
- The Identity Provider is external to the Oracle Banking Microservices Architecture (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 sent to plato-apigateway-router and request will be routed to the Gateway.
- 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.
- Configure an external service to do the SAML Verification in API Gateway with **EXTERNAL_SSO_VALIDATION_URL** parameter in the **SECURITY_CONFIG** table in **PLATO-SECURITY** schema.
- The implementation team needs to develop an external service to validate the SAML token.

SAML SSO Implementation

Figure 1-6 SAML SSO Implementation



Steps to achieve SSO-SAML Authentication:

- Bank user will try to access the Oracle Banking Microservices Architecture app-shell URL.
- Oracle Banking Microservices Architecture 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.
- Oracle Banking Microservices Architecture does not have any code for the IDP Redirection. This has to be completely customized by the implementation team where a new service file has to be built and to be deployed which can perform the redirection.
- On successful authentication IDP will generate the SAML token and pass the token to the Oracle Banking Microservices Architecture assertion consumer service URL in the body of POST method through EXTERNAL_SSO_KEY parameter.
- Oracle Banking Microservices Architecture 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 Microservices Architecture 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 Microservices Architecture will pass the SAML token through EXTERNAL_SSO_TOKEN_KEY parameter in the body of the POST to SVS.
- The implementation team needs to develop an external service to validate the SAML token.

- SVS will return a response with Status as success and, Oracle Banking Microservices Architecture would generate JWT token using the user id from the SVS response and allow the user to login.
- In case of failure, Oracle Banking Microservices Architecture 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 Microservices Architecture** from UBS section in the **Oracle Banking Microservices Architecture Installation Guide**.

1.6.2 API Security

This topic describes about API Security

Refer to **Oracle Banking Microservices Architecture 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 Microservices Architecture Application

This topic describes about securing the Oracle Banking Microservices Architecture Application.

This topic contains the following subtopics:

- Desktop Security
  This topic describes about desktop security.
- Oracle Banking Microservices Architecture Controls
  This topic describes about Oracle Banking Microservices Architecture controls.

2.1 Desktop Security

This topic describes about desktop security.

Refer to the vendor specific relevant sections for securing the Desktops Operating system. Also refer to the Browser specific security settings mentioned in the vendor specific documents.

Refer the client browser setting required for Oracle Banking Microservices Architecture.

2.2 Oracle Banking Microservices Architecture Controls

This topic describes about Oracle Banking Microservices Architecture controls.

This topic contains the following subtopics:

- Overview
  This topic describes describe the various programs available within Oracle Banking Microservices Architecture, 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 describe the various programs available within Oracle Banking Microservices Architecture, 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 **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).

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 Microservices Architecture operation such as new, copy, query, unlock, and so on are enabled based on function rights available for the user. The function rights are checked for each operation performed by the user in Security Management Service module of Oracle Banking Microservices Architecture.

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 module.
- Security Management System supports ROLE BASED access of Screens and different types of operations.
- Oracle Banking Liquidity Management supports dual control methodology, in which another user must be authorized with the requisite rights for each operation performed.
- Security Management System provides an option to map multiple roles for a user in a given branch. Allowed operations are mapped to the roles and Security Management System 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 Microservices Architecture can be done in two steps.

- **Branch level:** The user cannot view even the menu list of the Oracle Banking Microservices Architecture when the user tries to login into the restricted branch. Thus, no transactions could be performed.
- **Roles wise:** Based on the user-roles mapping, the user can access different functions of Oracle Banking Microservices Architecture. For example, a bank clerk has access to customer creation, account opening, term-deposits opening, and liquidation screens, but does not have access to User Creation function activity.

2.2.9 Password Management

This topic describes about password management.

The Oracle Banking Microservices Architecture application relies on external password management and does not store any credentials. The password management and policy rules can be set on OCI IAM.

Certain user password related parameters are defined at OCI 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.

Password Policies

Password validation criteria are configurable for your identity domain in the Identity Cloud Service console. The passport policies are also customized to meet the business and security requirements of the customer.

> **Note:**
>
> Refer **Managing Password Policies** topic in **Oracle Cloud Infrastructure** documentation for the detailed explanation.

3
General Information

Cryptography

Oracle Banking Microservices Architecture uses cryptography to protect the sensitive data. For encryption, AES is used which is considered the gold standard. It produces a key size of 256 bits when it comes to symmetric key encryption.

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.

3.1 Cryptography

This topic describes about cryptography.

Oracle Banking Microservices Architecture uses cryptography to protect the sensitive data.

For encryption, AES is used which is considered the gold standard. 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 is 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 addresses the Access privilege by separating responsibilities.

Data Protection

Advance Security provides the most advanced encryption capabilities to protect sensitive information without requiring any change to the application. TDE is native database solution that is completely transparent to existing applications.

It 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 the client using encryption, Network encryption using SSL/TLS.

Monitoring and Compliance

Audit Vault (AV) transparently collects and consolidates the audit data from multiple databases across the enterprise. It provides valuable insight into all details including privileged users. The integrity of the audit data is ensured using controls including DV and Advance Security. Access to AV data is strictly controlled. It also provides graphical summaries of the activity causing alerts. The database audit settings 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/depgd/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

Index