Oracle® Banking APIs

Security Guide

Patchset Release 22.2.4.0.0
F99665-01
June 2024

Oracle Banking APIs Security Guide, Patchset Release 22.2.4.0.0

F99665-01

Contents

Index

Preface

Purpose

This guide is designed to help acquaint you with the Oracle Banking APIs application. This guide provides answers to specific features and procedures that the user need to be aware of the module to function successfully.

Audience

This document is intended for the following audience:

- Customers
- Partners

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 customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

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.

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.

Related Resources

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

- Oracle Banking APIs Installation Manuals

Screenshot Disclaimer

Personal information used in the interface or documents is dummy and does not exist in the real world. It is only for reference purposes.

Acronyms and Abbreviations

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

Table 1 Acronyms and Abbreviations

Abbreviation	Description
OBAPI	Oracle Banking APIs

1

General Security Principles

The following principles are fundamental for using any application securely.

- Restrict Network Access to Critical Services
- Follow the Principle of Least Privilege
- Monitor System Activity
- Keep Up To Date on Latest Security Information

1.1 Restrict Network Access to Critical Services

Keep both the Oracle Banking Digital Experience middle-tier and the database behind a firewall. In addition, place a firewall between the middle-tier and the database. The firewalls provide assurance that access to these systems is restricted to a known network route, which can be monitored and restricted, if necessary. As an alternative, a firewall router substitutes for multiple, independent firewalls.

If firewalls cannot be used, be certain to configure the TNS Listener Valid Node Checking feature which restricts access based upon IP address. Restricting database access by IP address often causes application client or server programs to fail for DHCP clients. To resolve this, consider using static IP addresses, a software or a hardware VPN or Windows Terminal Services or its equivalent.

1.2 Follow the Principle of Least Privilege

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

1.3 Monitor System Activity

System security largely depends on the following practices:

- Good security protocols
- Proper system configuration
- System monitoring

The system needs to be constantly monitored from a monitoring tool.

1.4 Keep Up To Date on Latest Security Information

Oracle continually improves its software and documentation. It is recommended to keep your software updated.

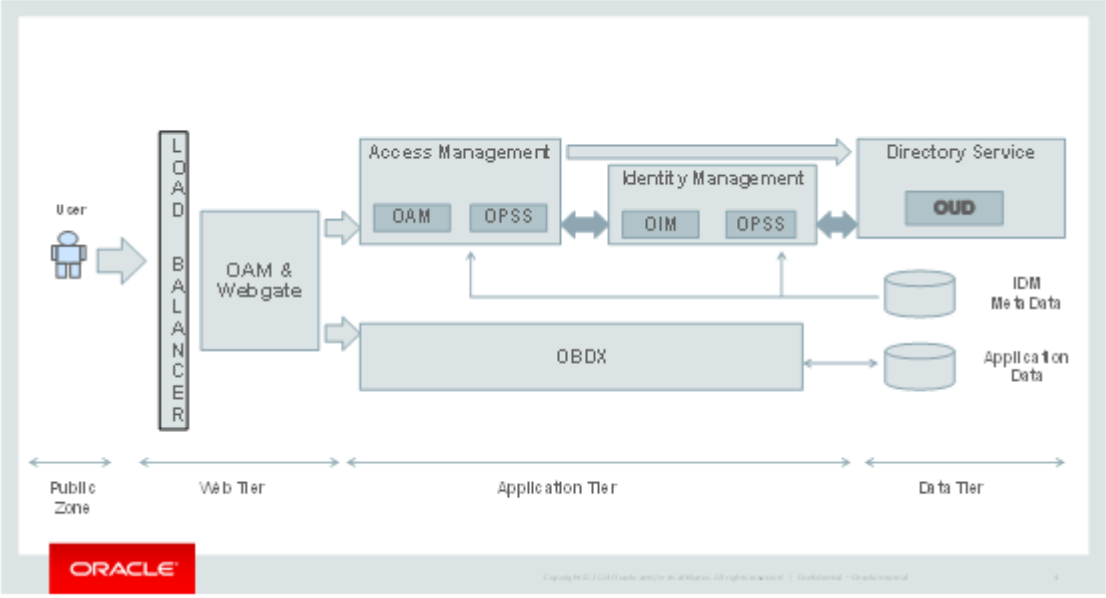
2

Secure Installation and Configuration

This chapter provides an overview of the architecture of the deployment and describes the installation and configuration procedure for Oracle Banking APIs.

Please note that this is only a guide to securing the Oracle Banking APIs application and does not replace periodic reviews of the security architecture of the entire ecosystem of multiple applications maintained by the customer. The guidance provided in this document must always be augmented by specific understanding of the security considerations of the specific deployment architecture.

2.1 Architecture Diagram



2.2 Installing WebLogic

Installation of WebLogic Server can be done by referring to the documentation published at https://docs.oracle.com/cd/E24329_01/doc.1211/e24492/toc.htm

.

2.3 Configuring SSL

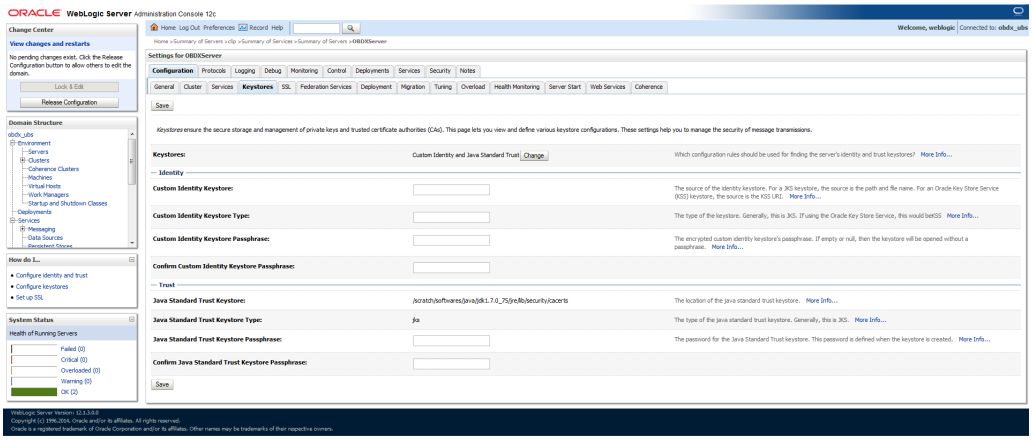
One way SSL between the presentation tier and the application on WebLogic server is supported. The detailed configuration is explained below:

> **Note:**
>
> Procure an external CA signed certificate before proceeding further. Follow the instructions below to install the certificate once the certificate is available.

1. Import the Certificate into a Java Trust Keystore.
   Execute the following command:

   ```
   keytool -import -trustcacerts -alias sampletrustself -keystore
   SampleTrust.jks
   -file SampleSelfCA.cer.der -keyalg RSAkeytool -import -alias `hostname -f`
   -file
   `hostname -f`.cer -keystore <JAVA_HOME>/jre/lib/security/cacerts -
   storepass changeit -noprompt
   ```

2. Configure Application Domain's WebLogic with Custom Identity and Trust Keystores.
   a. Open the WebLogic admin console and navigate to Home → Summary of Servers → AdminServer.
   b. Click the **Keystores** tab.



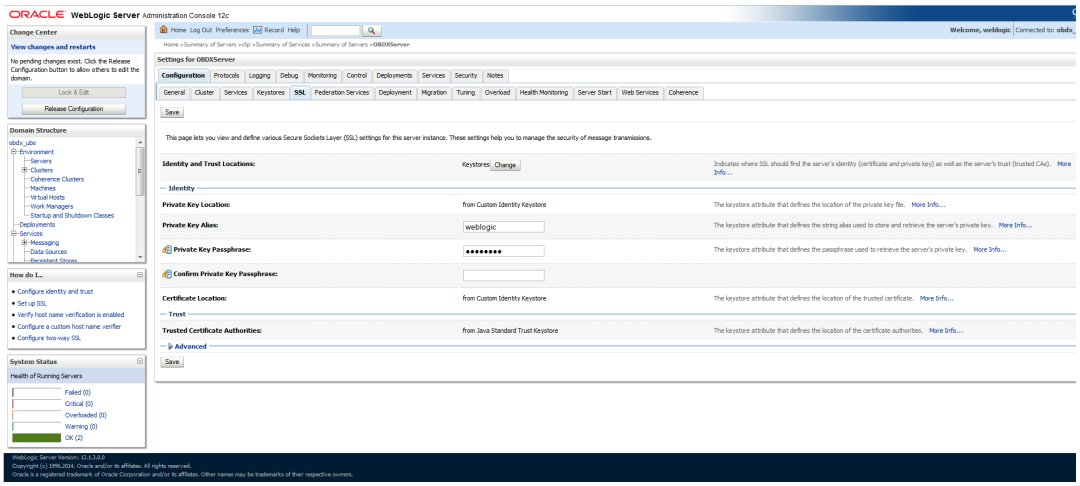
- Click the **Change** button.
- Select Custom Identity and Java Standard Trust option from the list.

- Click the **Save** button.
- Enter the following details in the **Identity** and **Trust** sections:
  Details in the **Identity** and **Trust** sections

Field	Value
Custom Identity Keystore	Absolute path of the custom keystore
Custom Identity Keystore Type	JCEKS
Custom Identity Keystore Passphrase	<Passphrase>
Confirm Custom Identity KeyStore Passphrase	<Re-enter the same Passphrase>

Enter the passphrases that were used while creating the custom Identity Keystore and certificate.

1. Click the **Save** button.
2. Click the **SSL** Tab.

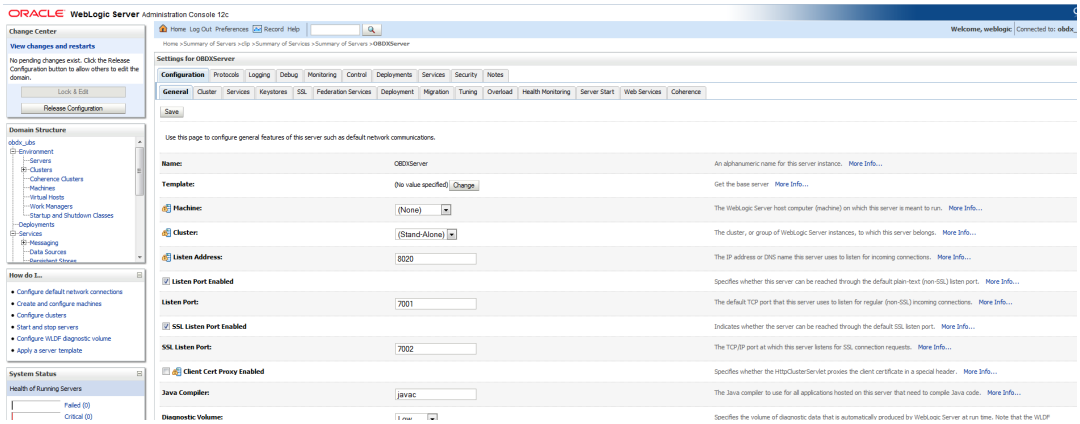


Enter the following details in the **Identity** section:

Field	Value
Private Key Alias	<Alias>
Private Key Passphrase	<Passphrase>
Confirm Private Key Passphrase	<Re-enter passphrase>

a. Enter the passphrases that were used while creating the certificate.
b. Click the **Save** button.
c. Click the **Advanced** link.
d. Ensure that **Two Way Client Cert Behavior** is set to **Client Certs Not Requested**.

3. Click the **General** tab.
4. Select the **SSL Listen Port Enabled** check box.



5. Click the **Save** button.

2.4 Disable SSLv3

By default, SSLv3 should be disabled.

Specifying the `weblogic.security.SSL.protocolVersion` system property in a command-line argument that starts the WebLogic Server lets you specify the protocol that is used for SSL connections.

The following command-line arguments can be specified so that WebLogic Server supports only TLS connections:

```
- Dweblogic.security.SSL.protocolVersion=TLS1
```

> **Note:**
>
> If you don't specify the above property, WebLogic assumes SSLv3 by default.

2.5 HTTP Response Header Configurations

The following are some HTTP Response Headers that mitigate certain vulnerabilities.

Vulnerability	HTTP Response Header
Clickjacking	X-Frame-Options
XSS	Content-Security-Policy
	X-XSS-Protection
Cookie hijacking Protocol Downgrade attacks	Strict-Transport-Security
Retrieving Sensitive data from browser cache	Cache-Control

The sections below specify how to configure these response headers in the httpd.conf file of the web server.

i. X-Frame-Options

```
Header always append X-Frame-Options SAMEORIGIN
```

ii. Content-Security-Policy

```
Header set Content-Security-Policy "default-src 'none'; img-src 'self';
script-src 'self'
'unsafe-inline' 'unsafe-eval'; style-src 'self' https://fonts.googleapis.com
'unsafe-inline';
object-src 'none'; frame-src 'none'; font-src 'self' https://
fonts.gstatic.com; connect-src 'self'
http://<OAM Server>:<OAM Port>/; child-src 'self'"
```

Please note that the policy mentioned here is for the base product. If the product gets customized and content from different URLs needs to be allowed to be executed by the browser, then this policy will have to be modified accordingly.

iii. X-XSS-Protection

```
Header set X-XSS-Protection "1; mode=block"
```

iv. Strict-Transport-Security

Set this for your top level domain. The header directive needs to be included inside the VirtualHost directive

```
<VirtualHost *:443>
Header always set Strict-Transport-Security
"max-age=31540000; includeSubDomains" </VirtualHost>
```

Consider submitting your website to be included in the HSTS preload list of websites maintained by Google Chrome at https://hstspreload.appspot.com/. Other browsers like MS IE 11, MS Edge, Firefox and Opera also refer to this list maintained by Google and therefore the security offered by this mechanism will extend to other browsers too.

v. Cache-Control

```
Header set Cache-Control "max-age=0, no-cache, no-store, must-revalidate
"Header set Pragma "no-cache"
Header set Expires 0
```

2.6 Cookie Attributes

Cookie contains sensitive information like session ID which is stored on the client. The cookie is sent with every request from client to server to maintain a valid authenticated session. Cookies can be secured by properly setting cookie attributes. The following two attributes must be set to secure a cookie.

1. **Secure:** This attribute tells the browser to only send the cookie if the request is being sent over a secure channel such as HTTPS.
2. **HttpOnly:** This attribute is used to help prevent attacks such as cross-site scripting, since it does not allow the cookie to be accessed via a client side script such as JavaScript.

Set these attributes in the WebLogic deployment descriptor file (weblogic.xml). The following attributes need to be included in

```
<wls:session-descriptor>

<wls:cookie-secure>true</wls:cookie-secure>

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

2.7 Password Policy Guidelines

Our recommendations for setting a password policy are in line with the latest recommendations from NIST as of June 2018.

1. The minimum length of a password must be at least 8 characters. You can choose to increase this number to 10 or 12.
2. The maximum length of a password must be at least 64 characters. You can choose to increase this number to 80 or 100.
3. Do not cause passwords to expire without reason. A password must be expired only when the user has forgotten it and has requested a reset.
4. Allow all printable ASCII characters, including spaces, and accept all UNICODE characters too.
5. Do not force the user to use a combination of upper case characters, lower case characters, numbers and special characters.
   Instead recommend to him that he uses “passphrases” instead of passwords, and that’s the reason why the recommended minimum length must be at least 8 and the maximum length must be at least 64.

   Passphrases are sentences like “*Wow, I like the freedom to choose this password!!*” (yes, with spaces, a comma and exclamation marks in it)

3
Guidance for Implementation Teams

4

List of Topics

This user manual is organized as follows:

Table 4-1 List of Topics

Topics	Description
Preface	This topic provides information on the introduction, intended audience, list of topics, and acronyms covered in this guide.
General Security Principles	This topic explains principles are fundamental for using any application securely.
Secure Installation and Configuration	This topic explains the overview of the architecture of the deployment and describes the installation and configuration procedure for Oracle Banking Digital Experience.
Guidance for Implementation Teams	This topic provides information about the guidance forimplementation teams such as CSRF Mitigation – Generating Nonces, Indirect Object Reference ImplementationOutput Encoding etc.

Index