Oracle Hospitality Integration Platform

Secure Development Guide

Release 22.3

F27481-09

June 2022

ORACLE

Oracle Hospitality Integration Platform Secure Development Guide, Release 22.3

F27481-09

Contents

Preface

Oracle Hospitality Integration Cloud Service and OPERA Cloud Foundation users are authorized to access the following modules and features:

- Oracle Hospitality Integration Platform including Oracle Hospitality Developer Portal and Hospitality REST APIs.

Purpose

This document provides security reference and guidance for the Oracle Hospitality Integration Platform.

Audience

The Oracle Hospitality Integration Platform Guide is intended for customers and partners who develop applications with the Oracle Hospitality Integration Platform.

Customer Support

To contact Oracle Customer Support, access My Oracle Support at the following URL:

https://support.oracle.com

When contacting Customer Support, please provide the following:

- Product version and program/module name
- Functional and technical description of the problem (include business impact)
- Detailed step-by-step instructions to re-create
- Exact error message received
- Screen shots of each step you took

Documentation

Oracle Hospitality product documentation is available on the Oracle Help Center at http://docs.oracle.com/en/industries/hospitality/.

Revision History

Date	Description of Change
June 2022	Initial publication

1

Direct Access to Oracle Hospitality Integration Platform APIs

The Oracle Hospitality APIs have been designed to use directly by servers. It is important that the Oracle Hospitality APIs are not exposed to an end user, such as in a web browser or mobile application. All communication through the Oracle Hospitality Integration Platform needs to be made from a back-end system.

2
Securing Applications

Build your applications with secure coding practices in mind. Oracle recommends any software built to connect to Oracle Hospitality APIs be assessed to avoid the security flaws described by the OWASP Top Ten Project and the OWASP API Security Top Ten:

https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project

https://owasp.org/www-project-api-security/

While some of the advice may not seem relevant to a server that is calling the Oracle Hospitality APIs, the advice also applies to any devices that connect to your server.

3
Securing Client Environments

Ensure the operating system for your application is patched with the latest security patches and the latest versions of tools and software to prevent potential exploits within the operating system and environment itself. The Center for Internet Security (CIS) has benchmarks on operating system hardening at:

https://learn.cisecurity.org/benchmarks

Inadvertent Capture of PAN includes additional guidance on securing your operating system to avoid inadvertently capturing cardholder details.

Network Segmentation

In accordance with the Payment Card Industry (PCI) Data Security Standard, Oracle Corporation mandates every site, including wireless environments, install and maintain a firewall configuration to protect data. Configure your network so databases and wireless access points always reside behind a firewall and have no direct access to the Internet.

Personal firewall software must be installed on any mobile and employee owned computers with direct connectivity to the Internet, such as laptops used by employees, which are used to access the organization's network. The firewall software configuration settings must not be alterable by employees.

Because of the PCI Data Security Standard, Oracle Corporation mandates each site ensure that servers, databases, wireless access points, and any medium containing sensitive data are behind a firewall. The firewall configuration must restrict connections between publicly accessible servers and any system component storing cardholder data, including any connections from wireless networks.

The firewall configuration must also place the database in an internal network zone, segregated from the demilitarized zone (DMZ) with the web server. A DMZ can be used to separate the Internet from systems storing cardholder data.

4

Payment Card Industry (PCI) standards

Some of the Oracle Hospitality Integration Platform APIs let you send cardholder data, so the Oracle Hospitality Integration Platform is in scope of the PCI DSS. Client systems are also in scope of PCI DSS, so follow these guidelines:

Payment Card Industry Payment Applications - Data Security Standard (PCI PA-DSS) https://www.pcisecuritystandards.org/security_standards/index.php

Payment Card Industry Data Security Standard (PCI DSS) https://www.pcisecuritystandards.org/security_standards/index.php

PCI Requirements

The Oracle Hospitality Integration Platform uses the following standards:

- Build and maintain a secure network and systems
  - Install and maintain a firewall configuration to protect cardholder data
  - Do not use vendor-supplied defaults for system passwords and other security parameters
- Protect cardholder data
  - Protect stored cardholder data
  - Encrypt transmission of cardholder data across open, public networks
- Maintain a vulnerability management program
  - Protect all systems against malware and regularly update anti-virus software or programs
  - Develop and maintain secure systems and applications
- Implement strong access control measures
  - Restrict access to cardholder data by business need-to-know
  - Identify and authenticate access to system components
  - Restrict physical access to cardholder data
- Regularly monitor and test networks
  - Track and monitor all access to network resources and cardholder data
  - Regularly test security systems and processes
- Maintain an information security policy
  - Maintain a policy that addresses information security

Handling of Sensitive Authentication Data (PA-DSS 1.1.5)

Oracle Hospitality Integration Platform does not store sensitive authentication data, and we strongly recommend you do not store this type of sensitive data as well. However, if for any reason you need do so, the following guidelines must be followed when dealing with sensitive

authentication data used for pre-authorization (swipe data, validation values or codes, PIN or PIN block data):

- Collect sensitive authentication data only when needed to solve a specific problem
- Store such data only in specific, known locations with limited access
- Collect only the limited amount of data needed to solve a specific problem
- Encrypt sensitive authentication data while stored
- Securely delete such data immediately after use

Secure Deletion of Cardholder Data (PA-DSS 2.1)

Oracle Hospitality Integration Platform does not store cardholder data and therefore there is no data to be purged by the application as required by PA-DSS v3.2.

Any cardholder data you store outside of the application must be documented and you must define a retention period at which time you will securely delete (render irretrievable) the stored cardholder data. When defining a retention period you must take into account legal, regulatory, or business purpose.

All underlying software (this includes operating systems and/or database systems) must be configured to prevent the inadvertent capture of PAN. Instructions for configuring the underlying operating systems or databases can be found in Inadvertent Capture of PAN .

PCI-Compliant Wireless Settings (PA-DSS 6.1.a and 6.2.b)

Oracle Hospitality Integration Platform must not be accessed using wireless technologies. However, should any systems downstream of the client system implement wireless access to the client system, the following guidelines for secure wireless settings must be followed to ensure cardholder data is secure end to end, per PCI Data Security Standards 1.2.3, 2.1.1 and 4.1.1:

PCI DSS section 1.2.3: Perimeter firewalls must be installed between any wireless networks and systems that store cardholder data, and these firewalls must deny or control (if such traffic is necessary for business purposes) any traffic from the wireless environment into the cardholder data environment.

PCI DSS section 2.1.1: Change wireless vendor defaults as follows:

- Encryption keys must be changed from default at installation, and must be changed anytime anyone with knowledge of the keys leaves the company or changes positions
- Default SNMP community strings on wireless devices must be changed
- Default passwords or passphrases on access points must be changed
- Firmware on wireless devices must be updated to support strong encryption for authentication and transmission over wireless networks
- Other security-related wireless vendor defaults, if applicable, must be changed

PCI DSS section 4.1.1: Industry best practices (for example, IEEE 802.11.i) must be used to implement strong encryption for authentication and transmission of cardholder data.

Never Store Cardholder Data on Internet-accessible Systems (PA-DSS 9.1.c)

Never store cardholder data on Internet-accessible systems. For example, a web server and a database server must not be on same server.

Maintain an Information Security Program

In addition to the security recommendations included in this document, a comprehensive approach to assessing and maintaining the security compliance of the payment application environment is necessary to protect the organization and sensitive cardholder data.

The following is a very basic plan every owner of a client system provider should adopt in developing and implementing a security policy and program:

- Read the PCI DSS in full and perform a security gap analysis. Identify any gaps between existing practices in your organization and those outlined by the PCI requirements.
- Once the gaps are identified, determine the steps to close the gaps and protect cardholder data. Changes could mean adding new technologies to shore up firewall and perimeter controls, or increasing the logging and archiving procedures associated with transaction data.
- Create an action plan for on-going compliance and assessment.
- Implement, monitor and maintain the plan. Compliance is not a one-time event. Regardless of merchant or service provider level, all entities should complete annual self-assessments using the PCI Self-Assessment Questionnaire.
- Call in outside experts as needed.

5

Properly Train and Monitor Administrators

It is the responsibility of the owner of the client system (which accesses the Oracle Hospitality Integration Platform APIs) to institute proper personnel management techniques for allowing admin user access to cardholder data, site data, and so on. The client system owner can control whether each individual admin user can, for example, see full credit card PAN, or only the last 4 digits of the PAN.

In most systems, a security breach is often the result of unethical personnel. So pay special attention to whom you trust with admin access and who you allow to view full decrypted and unmasked payment information.

When administering the Oracle Hospitality Integration Platform Oracle Cloud Operations always use multi-factor authentication (MFA) using physical tokens to access production instances of Oracle Hospitality Integration Platform.

To enable Multi Factor Authentication (MFA) for IDCS accounts, refer to https://docs.oracle.com

After you purchase IDCS Standard edition, refer to Understand the User Per Month Pricing Model

6
Transport Layer Security

The Oracle Hospitality Integration Platform provides access to HTTP-based APIs, which are secured through Transport Layer Security (TLS) version 1.2 or above using only strong ciphers. The Oracle Hospitality Integration Platform does not support unsecured access to Oracle Hospitality APIs.

On connecting to Oracle Hospitality APIs, applications must validate that the TLS certificate is legitimate and not a forgery. This prevents fraudulent attacks resulting in compromised passwords and extraction of customer data.

7

Personal Information

Full details of the secure handling of personal information by the Oracle Hospitality Integration Platform is covered in the following My Oracle Support article.

To ensure Oracle can contact you about your integrations, Oracle requires you to supply some basic contact details. Name, Company Name, Email Address, and Phone Number.

To receive a copy of or correct the personal information you supplied to the Oracle Hospitality Integration Platform – or to change the appointed contact for your integrations – please log in to the Oracle Hospitality Integration Platform Developer Portal.

If you wish to remove your contact details, remember to include the Name, Company Name, Email Address, and Phone Number of the new contact nominated for that integration.

If you would like a copy of these contact details in a machine-readable format you can request this via My Oracle Support.

These contact details are permanently deleted when access to the Oracle Hospitality Integration Platform is terminated.

> **Note:**
>
> All Oracle Hospitality Developer portal users can see the contact details on all the applications created by anyone in their company.

Personal Information in Log Files

The Oracle Hospitality Integration Platform and Oracle Hospitality API logs are written in a standard format, and stored in standard locations, with timestamps.

Operational logs are sent to shared log stores in Oracle Cloud, which is subject to role-based access control.

In line with the Service Privacy Policy (https://www.oracle.com/legal/privacy/services-privacy-policy.html) Oracle is legally obligated to retain operational log file information for 90 days after which logs are automatically purged.

Assigning a Unique ID to Each Person with Computer Access

Oracle Corporation recognizes the importance of establishing unique IDs for each person with computer access. No two Oracle Hospitality Integration Platform users can have the same ID, and each person's activities can be traced, provided the customer maintains proper configuration and adheres to privilege level restrictions based on a need-to-know basis.

While Oracle Corporation makes every possible effort to conform to Requirement 8 of the PCI Data Security Standard, certain parameters, including proper user authentication, remote

network access, and password management for non-consumer users and administrators, for all system components, depend on customer / partner specific protocol and practices.

To ensure strict access control of the Oracle Hospitality Integration Platform always assign unique user names and complex passwords to each account. Oracle Corporation mandates applying these guidelines to not only Oracle Corporation passwords, but to passwords for systems accessing Oracle Hospitality Integration Platform APIs and downstream of there, including server operating system passwords and end user Windows® passwords. Furthermore, Oracle Corporation advises users to control access, through unique user names and PCI-compliant complex passwords, to any PCs, servers, and databases with payment applications and cardholder data.

8
Credentials

All calls that applications make to Oracle Hospitality APIs must include credentials.

Client systems should not store these credentials in plaintext and should instead encrypt and store the values to a secure standard.

Do not share credentials with other organizations, including Oracle.

> **Note:**
>
> Never include either a password or a clientSecret in My Oracle Support tickets.

Changing Your OPERA Integration User

To change your OPERA integration user, see the “Changing Your Integration User Password” topic in the *Oracle Hospitality Integration Platform User Guide*.

Reissuing an oAuth clientSecret

To change your clientSecret, contact My Oracle Support and include the following details in your message:

- **Client ID**
- **Environment Name** — This is the environment for which the clientSecret needs changed.
- **Production or Non Production Environment** — Indicate if the environment is production or non production.
- **Environment Description** — Include the description of the environment as it appears on the Environments page.

Multi-Factor Authentication (MFA)

To enable multi-factor authentication for your developer portal user, follow the links below:

- As a customer, have a chain administrator follow this resource if using the Oracle Hospitality Shared Security Domain (SSD): SSD MOS article (Doc ID 2329730.1)
- As a partner, follow this guide: How To Enable Multi-Factor Authentication in OCI and IDCS (Doc ID 2800621.1)

9 Application Key

All calls that applications make to Oracle Hospitality APIs are required to include an Application Key.

Ideally, client systems should not store the Oracle Hospitality Integration Platform Application Key in plaintext and should instead encrypt and store the value to a secure standard. Although the Application Key is not a security artifact, encrypting the value makes it more difficult for an attacker to impersonate a client system.

Do not share Application Keys with other organizations.

Re-issuing an Application Key

If you have access to the Oracle Hospitality Integration Platform Developer Portal you can re-issue your application key following the instructions in the Oracle Hospitality Integration Platform User Guide.

This process immediately stops all existing uses of the old application key.

10 Encryption Best Practices

When encrypting passwords and other artifacts consider the following:

Encryption becomes fallible because:

- Applications use broken implementations or use known algorithms improperly.
- Data is insecure because of easily defeated cryptography.

In addition, Base-64 encoding, obfuscation, and serialization are not encryption, and should not be mistaken for encryption.

To encrypt data successfully:

- Use the platform-specific file encryption API or another trusted source. Do not create your own cryptography.
- You must restrict access to encryption keys to the fewest number of custodians necessary.
- You must store encryption keys securely in the fewest possible locations and forms.
- Do not store the key with the encrypted data.

11

Sending Information to Oracle Hospitality APIs

Oracle Hospitality APIs will always respond with content in `application/json` format. They also expect `application/json` format in request bodies. If requests are not made in `application/json` format Oracle Hospitality APIs may return plaintext error responses.

Full details of the authentication information needed to make successful requests to Oracle Hospitality APIs is documented in the user guide. For more information, refer Oracle Hospitality Integration Platform User Guide.

12

Receiving Information from Oracle Hospitality APIs

Applications should ensure any data returned from Oracle Hospitality APIs is managed securely and safely. The system should:

- Encrypt any sensitive data. Data returned from Oracle Hospitality APIs may contain personal and other sensitive information. If your client system needs to store this information it should be stored securely and encrypted within the data store. Access to these data should be restricted and accounts should have minimum access rights to the data, to prevent elevated access.
- Any data output to users from an Oracle Hospitality APIs response should be sanitized to ensure there is no possibility of cross-site scripting.
- Validate all data received from the Oracle Hospitality APIs.

13

Overview Diagram of the Oracle Hospitality Integration Platform

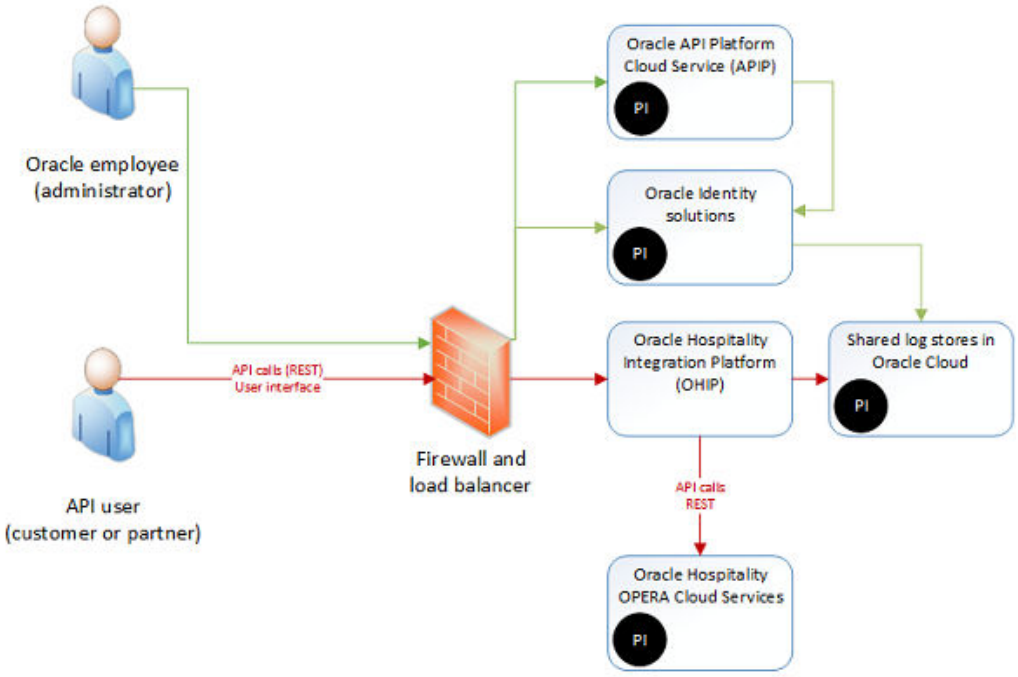
Oracle customers have access to the Oracle Hospitality Integration Platform via Oracle Hospitality APIs and via the Oracle Hospitality Integration Platform Developer Portal (a user interface).

API calls can contain personal information, and can access personal information held in Oracle Hospitality OPERA. These data are not stored in the Oracle Hospitality Integration Platform, except for usernames sent to security logs in the shared log stores in Oracle Cloud.

Customers, Partners, and Oracle employees employing user interfaces do so via corporate Oracle Identity solutions and role-based access control; when access is removed, access to both the Oracle Hospitality Integration Platform, its data, and Oracle Hospitality APIs is also removed.

Personal information (limited to Customer or Partner contact information) is provided by the customer to Oracle and can be maintained by them in the Oracle Hospitality Integration Platform Developer Portal, subject to the guidance in Chapter 7, **Personal Information**.

The following diagram summarizes how personally identifiable information is used within the Oracle Hospitality APIs:



Abbreviations	
APIP	Oracle API Platform Cloud Service
OHIP	Oracle Hospitality Integration Platform

Key

- Personal information over TLS>=1.2
- Personal information over TLS>=1.2 following Multi Factor Authentication
- No personal information
- PI — Personal information stored here

14
Inadvertent Capture of PAN

This appendix explains how to address the inadvertent capture of PAN on a Linux system.

Clear Swap Space on Your System

Perform the following commands:

1. Run `free` (to review swap usage). The following information appears:

```
root@Orthanc:~ # free
             total       used       free     shared    buffers     cached
Mem:       1034368     693128     341240          0     215448     235824
-/+ buffers/cache:     241856     792512
Swap:       524280      49576     474704
```

2. Run `swapoff -a` (requires elevated privs).
3. Run `swapon -a` (requires elevated privs).
4. Run `free` (should show that swap has been cleaned out). The following figure shows an example:

```
coalfire@ubuntu:~$ sudo swapoff -a
[sudo] password for coalfire:
coalfire@ubuntu:~$ sudo swapon -a
coalfire@ubuntu:~$ free
             total       used       free     shared    buffers     cached
Mem:       2050976     269184    1781792          0      10632     194308
-/+ buffers/cache:      64244    1986732
Swap:      2097148          0    2097148
```

   You can disable the swap space as described in the following section.

Disable Swap Space

Disable swap space can be risky to the operation of your system. With no swap space, the Linux or Unix operating system will automatically kill processes if the amount of physical RAM needed for all running processes is exceeded:

Comment out the swap entry in `/etc/fstab`.

Encrypt the Swap Space on Your System

This section explains encrypting swap through the use of dm-crypt. This requires you to run a 2.6 kernel. For example, the swap partition will be /dev/VolGroup00/LogVol01. This is the default swap partition for RedHat systems. The swap partition does not need to be part of an Logical Volume Manager (LVM). As noted previously, dm-crypt can encrypt disk partitions (/dev/hda2) or whole disks (/dev/hda). Be sure to change the commands to fit your swap partition accordingly.)

When encrypting your swap partition, you will need to temporarily turn off swap. This means you need to shut all unnecessary applications to free up memory. If this memory is not freed, you will be unable to turn off the swap space. The best way to handle this is to boot the system into single user mode. This shuts down most services with the exception of a single root shell. To boot the system into single user mode, run the following command:

```
# /sbin/telinit s
```

Turn off the swap space by running the following command:

```
# swapoff -a
```

To ensure a completely clean and sterile swap space, overwrite the swap partition with random data. This will help prevent the recovery of any data written to swap before the encryption process. The shred command overwrites the specified file or device with random data:

```
# shred -v /dev/VolGroup00/LogVol01
```

Create a file named /etc/crypttab. The main page for crypttab covers the particulars of crypttab. The below example

1. Creates an encrypted block device named swap at /dev/mapper (first field).
2. Specifies /dev/VolGroup00/LogVol01 as the underlying block device (second field).
3. Specifies /dev/random as the encryption password (third field).
4. Specifies the encrypted device as a swap device with an encryption cypher with AES encryption and unpredictable IV values (fourth field).

swap /dev/VolGroup00/LogVol01 /dev/random swap,cipher=aes-cbc-essiv:sha256

Edit /etc/fstab to point to the encrypted block device, /dev/mapper/swap as opposed to /dev/VolGroup00/LogVol01. The file should look like the following example:

/dev/VolGroup00/LogVol01 swap swap defaults 0 0

Change the file to look like this:

/dev/mapper/swap swap swap defaults 0 0

Reboot your system to create the encrypted swap space with the following command:

```
# reboot -n
```

If you do not want to reboot, you can create the encrypted swap partition with the following commands:

`# cryptsetup -d /dev/random create swap /dev/VolGroup00/LogVol01` (the `-d` part of the command specifies cryptsetup to use /dev/random as the key file and the `create` part of the command creates a mapping with the name, swap backed by the device, /dev/VolGroup00/LogVol01)

```
# mkswap /dev/mapper/swap
```

```
# swapon -a
```