Oracle® Checkout

Security Guide

V1.0

09 SEP 2025

ORACLE®

Oracle Checkout Security Guide [Version]

Primary Author: Payments Security Lead

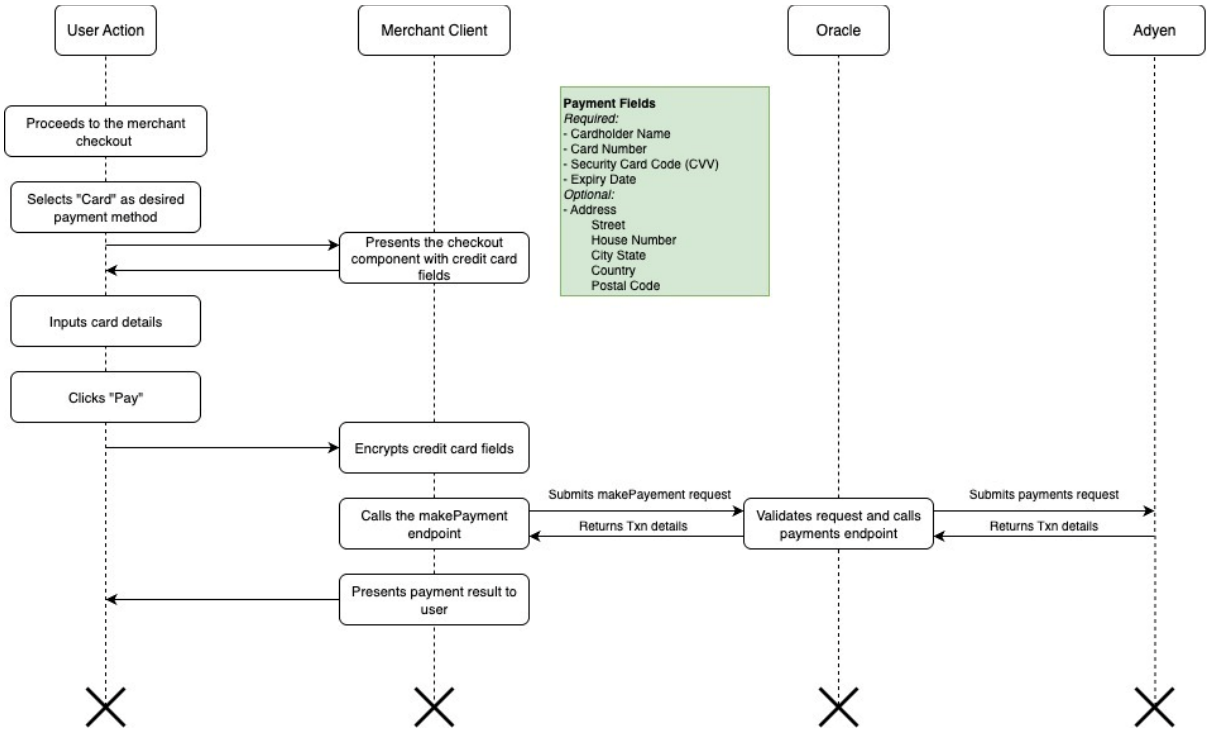
Contributors: Payments DevSecOps, Payments Team Leaders, Payments Checkout Development

Contents

Part 1: Overview

Oracle Payments Checkout is a pre-built, ready-to-use module or piece of software that allows online ordering websites or applications to easily integrate with Simphony Payment processing functionality.

Product Overview



Your Responsibilities

You are responsible for making sure that cardholder data is secure and protected before the data reaches Oracle/Adyen. Depending on your integration, you also must comply with cardholder data storage requirements.

Implement a security policy that includes an incident response plan and defines information security roles and responsibilities for all personnel.

Perform external vulnerability scans every 3 months.

Ensure all other iframes loaded into your page follow security best practices, and that the entities loading them are PCI DSS compliant.

Implement a security policy that includes an incident response plan and defines information security roles and responsibilities.

Implement Oracle Checkout in compliance with the security guidance from Adyen: https://docs.adyen.com/development-resources/integration-security-guide/

Key Technical Recommendations

Implement Subresource Integrity hashes.

Make sure that your returnUrl cannot be easily tampered with.

Do not generate the returnUrl on a system that is publicly accessible.

Automatically validate the host part of the URL before including it in your API request (to help prevent attackers from changing the destination of a payment redirect to collect a shopper's card data).

Make sure that client-side resources in your pages, such as JavaScript libraries, CSS, chatbots, and analytics, are loaded from authorized domains.

Implement Content Security Policy (CSP) (to help prevent attacks such as Cross-Site Scripting and data injection attacks).

General Security Principles

- Implement the principles of least privilege and separation of duties.

Limit privileges as much as possible. Users should be given only the access and information that's essential to perform their work. Review user privileges to both periodically to determine and verify relevance to the current work requirements.

- Implement multilayer security mechanisms.
- Protect data at rest and in transit.
- Monitor and respond to security events.

Monitor system activity. Establish who should access which system components, and how often; and monitor those components.

- Stay up to date on security alerts, patches, and software updates.

Oracle regularly issues security-related patch updates and security alerts. Install all security patches as soon as possible. See Critical Patch Updates, Security Alerts and Bulletins. **There have not been Critical Patch Updates, Security Alerts nor Bulletins for Oracle Checkout nor Oracle Payments Cloud Services.**

- Implement security-related best practices. See Security Best Practices.

Learn More

- NIST Secure Systems and Applications
- OWASP Security Verification Standard
- SANS resources

Recommended Authentication and Security Measures

User Account Management:

Replace vendor-supplied usernames and passwords, avoid sharing credentials, and ensure each user has a unique account.

HMAC Signatures for Webhooks:

Verify webhook events by using Hash-based message authentication code (HMAC) signatures to ensure the event was sent by Adyen and hasn't been tampered with.

Basic Authentication over HTTPS:

Use basic authentication with a username and password for webhook requests but ensure that it is protected by HTTPS to prevent credential compromise.

TLS Configuration:

Ensure you use the correct TLS configuration, including strong ciphersuites, to secure communication.

Content Security Policy (CSP):

Implement CSP to mitigate cross-site scripting (XSS) attacks by restricting resources to trusted sources.

3D Secure 2 Authentication:

Implement 3DS (3D Secure 2) for an added layer of verification, particularly for card-not-present transactions.

Comply with authentication regulations like PSD2 SCA.

Check if the payment terminal shows "TAMPER" or "TAMPERED":

If so, stop using the device and contact your Security Team immediately.

Don't generate Pay by Link URLs on publicly accessible systems. It is better to validate the host part of the URL automatically before sending them to your customer.