Online Payments Processing Platform

Owner: A development team
Reviewer: A security architect
Contributors: development engineers, product managers, security architects
Date Generated: Tue Oct 07 2025

Executive Summary

High level system description

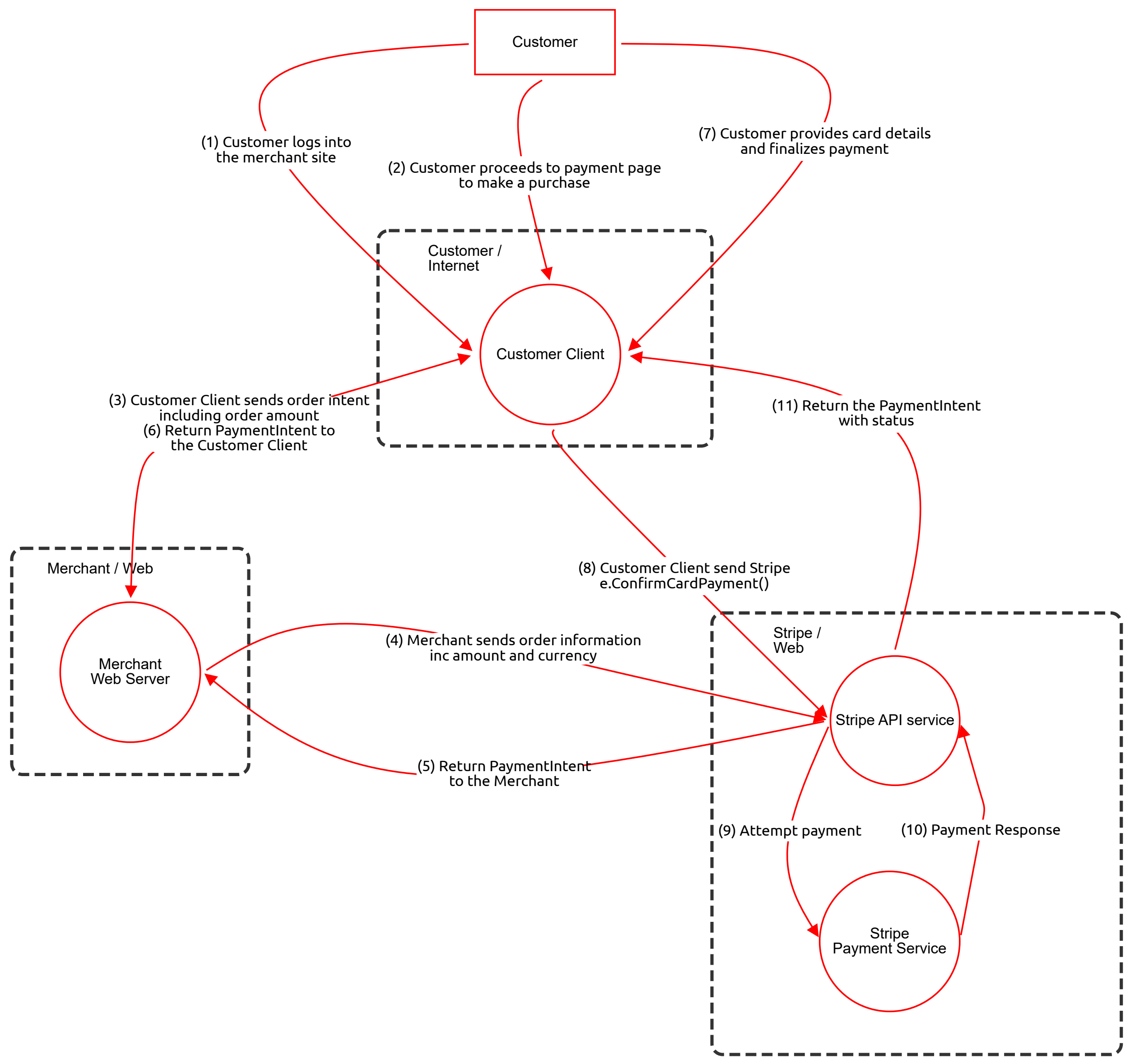
This threat model has been provided by the OWASP Threat Model Cookbook:
threat-model-cookbook/Flow Diagram/payment

Summary

Total Threats	20
Total Mitigated	0
Total Open	20
Open / Critical Severity	0
Open / High Severity	14
Open / Medium Severity	6
Open / Low Severity	0

Payment

Demo threat model for an online Payments Processing Platform
provided by the OWASP Threat Model Cookbook:
threat-model-cookbook/Flow Diagram/payment


Customer
(1) Customer logs into the merchant site
(2) Customer proceeds to payment page to make a purchase
(7) Customer provides card details and finalizes payment
Customer / Internet
Customer Client
(3) Customer Client sends order intent including order amount
(6) Return PaymentIntent to the Customer Client
(11) Return the PaymentIntent with status
Merchant / Web
Merchant Web Server
(8) Customer Client send Stripe e.ConfirmCardPayment()
(4) Merchant sends order information inc amount and currency
(5) Return PaymentIntent to the Merchant
Stripe / Web
Stripe API service
(9) Attempt payment
(10) Payment Response
Stripe Payment Service

Payment

Customer (Actor)

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Customer Identity Spoofing	Spoofing	Medium	Open		The Customer actor is located in the Customer/Internet trust boundary and initiates authentication flows. Without proper identity verification, an attacker could spoof legitimate customer identities to gain unauthorized access to the merchant system.	Implement strong multi-factor authentication and identity verification mechanisms for customer login processes.

Customer Client (Process)

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Client-Side Tampering of Payment Data	Tampering	High	Open		Customer Client process handles payment flows and is located in the Customer/Internet boundary. Client-side code could be tampered with to modify payment amounts or redirect payments.	Implement server-side validation of all payment amounts and use secure client-side libraries with integrity checks.
	Information Disclosure from Client	Information Disclosure	High	Open		Customer Client process handles sensitive payment information and communicates across trust boundaries. Sensitive data could be exposed through client-side vulnerabilities or insecure storage.	Implement proper data encryption at rest and in transit, and follow secure coding practices to prevent client-side data leaks.

(1) Customer logs into the merchant site (Data Flow)

Description: OAuth

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Authentication Flow Interception	Spoofing	High	Open		Flow (1) carries authentication data (OAuth) from Customer to Customer Client across the Customer/Internet boundary. The protocol is HTTPS but flows through public networks, making it vulnerable to interception if TLS is not properly implemented.	Enforce TLS 1.2+ with strong cipher suites and implement certificate pinning for authentication endpoints.

(2) Customer proceeds to payment page to make a purchase (Data Flow)

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Payment Flow Tampering	Tampering	High	Open		Flow (2) carries payment initiation data from Customer to Customer Client using HTTPS over public networks. The flow could be intercepted and modified to alter payment details.	Implement end-to-end encryption and digital signatures to ensure payment data integrity throughout the transaction flow.

(7) Customer provides card details and finalizes payment (Data Flow)

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Card Data Interception	Information Disclosure	High	Open		Flow (7) carries sensitive card details from Customer to Customer Client. The flow crosses trust boundaries and lacks explicit encryption specification, creating risk of card data exposure.	Implement PCI DSS compliant card data handling, including end-to-end encryption and tokenization to prevent card data exposure.

(3) Customer Client sends order intent including order amount (6) Return PaymentIntent to the Customer Client (Data Flow)

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Order Intent Manipulation	Tampering	High	Open		Bidirectional flow (3/6) carries order intent and PaymentIntent between Customer Client and Merchant Web Server across trust boundaries. The flow lacks explicit encryption and could be modified in transit.	Implement message-level encryption and digital signatures for all order and payment intent communications between systems.
	Payment Intent Repudiation	Repudiation	Medium	Open		Bidirectional flow (3/6) handles payment transactions without explicit non-repudiation mechanisms. Parties could deny involvement in payment transactions.	Implement comprehensive audit logging and digital signatures for all payment-related transactions to prevent repudiation.

(9) Attempt payment (Data Flow)

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Payment Service Denial of Service	Denial of Service	Medium	Open		Flow (9) from Stripe API service to Stripe Payment Service represents internal payment processing. This critical flow could be targeted for DoS attacks, disrupting payment processing.	Implement rate limiting, circuit breakers, and redundant payment processing services to maintain availability during attack scenarios.

(10) Payment Response (Data Flow)

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Payment Response Tampering	Tampering	High	Open		Flow (10) carries payment response data between Stripe Payment Service and Stripe API service. Tampering with payment responses could lead to incorrect transaction status reporting.	Implement secure internal communications with message authentication codes and integrity verification for all payment status messages.

(11) Return the PaymentIntent with status (Data Flow)

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Payment Status Information Disclosure	Information Disclosure	Medium	Open		Flow (11) returns PaymentIntent status from Stripe API service to Customer Client across multiple trust boundaries. Sensitive payment status information could be exposed during transmission.	Encrypt all payment status communications and implement proper access controls to ensure only authorized parties receive transaction status information.

(8) Customer Client send Stripe e.ConfirmCardPayment() (Data Flow)

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Card Payment API Abuse	Elevation of Privilege	High	Open		Flow (8) carries e.ConfirmCardPayment() from Customer Client to Stripe API service across trust boundaries. Unauthorized access to this API could allow privilege escalation for payment processing.	Implement strict API authentication, authorization checks, and rate limiting for all payment confirmation requests to prevent unauthorized payment processing.

(5) Return PaymentIntent to the Merchant (Data Flow)

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	PaymentIntent Tampering	Tampering	High	Open		Flow (5) returns PaymentIntent from Stripe API service to Merchant Web Server. Tampering with PaymentIntent data could lead to incorrect payment processing or financial losses.	Implement secure API communications with message integrity verification and digital signatures for all PaymentIntent data exchanges.

(4) Merchant sends order information inc amount and currency (Data Flow)

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Order Information Interception	Information Disclosure	Medium	Open		Flow (4) carries order information including amount and currency from Merchant Web Server to Stripe API service across trust boundaries. Sensitive order data could be exposed during transmission.	Encrypt all order information in transit and implement secure API authentication to protect sensitive business data.

Merchant Web Server (Process)

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Merchant Server Privilege Escalation	Elevation of Privilege	High	Open		Merchant Web Server process handles payment transactions and communicates with both customer-facing systems and Stripe API. Compromise could lead to unauthorized payment processing or data access.	Implement principle of least privilege, regular security updates, and strict access controls for merchant server components.
	Merchant Server Denial of Service	Denial of Service	Medium	Open		Merchant Web Server is critical for payment processing and located in the Merchant/Web boundary. DoS attacks could disrupt the entire payment flow for customers.	Implement DDoS protection, load balancing, and auto-scaling capabilities to maintain service availability during high traffic or attack scenarios.

Stripe API service (Process)

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	API Service Authorization Bypass	Elevation of Privilege	High	Open		Stripe API service process handles payment requests and is located in the Stripe/Web boundary. Weak authentication could allow unauthorized access to payment processing functions.	Implement strong API authentication with OAuth 2.0, API keys, and comprehensive authorization checks for all payment operations.
	API Service Tampering	Tampering	High	Open		Stripe API service processes sensitive payment data and coordinates between multiple systems. Compromise could allow tampering with payment transactions or data.	Implement secure coding practices, regular security testing, and integrity monitoring for API service components.

Stripe Payment Service (Process)

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Payment Service Privilege Escalation	Elevation of Privilege	High	Open		Stripe Payment Service process handles core payment processing within the Stripe/Web boundary. Unauthorized access could lead to financial fraud or data compromise.	Implement strict access controls, segregation of duties, and regular security audits for payment service components.
	Payment Service Information Disclosure	Information Disclosure	High	Open		Stripe Payment Service processes highly sensitive payment card data. Compromise could lead to massive data breach of customer payment information.	Implement PCI DSS compliance measures including encryption, tokenization, and strict access controls for payment card data processing and storage.