xxxx

Exercice 2

rapport

COUPEAU Laurent
12/02/2026

Table des matières

1. Introduction

This report presents a complete STRIDE threat analysis for the ShopNow ecommerce platform. It builds on the asset inventory and criticality assessment performed in Exercise 1 and prepares the foundation for:

- deriving security requirements (Exercise 3),
- designing a Zero Trust architecture (Exercise 4),
- defining security tests (Exercise 5),
- and implementing detection & response (Exercise 6).

The analysis follows the STRIDE methodology, which classifies threats into six categories:

- **S – Spoofing**
- **T – Tampering**
- **R – Repudiation**
- **I – Information Disclosure**
- **D – Denial of Service**
- **E – Elevation of Privilege**

Each asset and flow is evaluated against these threats, with justification and impact analysis.

2. STRIDE Threat Model Overview

STRIDE helps identify how attackers may compromise:

- **identities** (S),
- **data integrity** (T),
- **accountability** (R),
- **confidentiality** (I),
- **availability** (D),
- **and privilege boundaries** (E).

For ShopNow, STRIDE is particularly relevant because:

- the platform handles **PII**, **payment data**, **tokens**, and **orders**,

- it exposes **public APIs**,
- it integrates with an **external payment provider**,
- it has **administrators with high privileges**,
- and it has already suffered **bot attacks**, **credential stuffing**, and **log exposure**.

3. STRIDE Threats per Asset and Flow

Below is the **completed STRIDE table**, with threats marked and justified.

4. Completed STRIDE Table

Legend:

✓ = Threat applies (blank) = Not applicable or negligible

STRIDE Table for ShopNow

Asset / Flow	S	T	R	I	D	E
A1 – Client	**✓ Credential stuffing, session hijacking**	**✓ Manipulating client-side requests**	**✓ User denies actions**	**✓ XSS leaks PII/tokens**	**✓ Bot traffic**	**✓ Bypass client-side checks**
A2 – Administrator	**✓ Admin account takeover**	**✓ Malicious admin changes**	**✓ Deny harmful actions**	**✓ Export customer data**	**✓ Disable services**	**✓ Abuse admin role**
C1 – Front-end	**✓ Fake front-end phishing**	**✓ DOM tampering, JS injection**	**✓ No reliable client logs**	**✓ XSS leaks tokens**	**✓ CDN overload**	**✓ Bypass front-end controls**
C2 – Backend	**✓ API key/token spoofing**	**✓ SQLi, parameter tampering**	**✓ Missing logs**	**✓ Verbose errors**	**✓ API saturation**	**✓ RCE, privilege escalation**

				leak data		
C3 – Database	✓ Stolen DB credentials	✓ Data modification	✓ No audit logs	✓ Full DB dump	✓ Heavy queries lock DB	✓ Superuser escalation
C5 – Auth API	✓ Credential stuffing	✓ Tampering with auth flows	✓ No login traceability	✓ Token leakage	✓ Login endpoint DoS	✓ Bypass auth checks
C6 – Payment API	✓ Impersonation of backend	✓ Modify amount/currency	✓ No traceability	✓ Leak payment refs	✓ Payment service down	✓ Abuse refund/capture
D1 – Customer data	✓ Identity theft	✓ Modify addresses	✓ No audit trail	✓ PII breach	✓ DB unavailable	✓ Social engineering
D2 – Orders data	✓ Fake orders	✓ Modify amounts/status	✓ Deny order creation	✓ Leak order history	✓ Orders API down	✓ Abuse order privileges
D4 – Tokens	✓ Token theft	✓ Modify JWT payload	✓ No token usage logs	✓ Token leakage	✓ Token service down	✓ Use token to escalate
F1 – Auth flow	✓ Credential stuffing	✓ Modify login/refresh	✓ No login logs	✓ Intercept credentials	✓ Auth endpoint DoS	✓ Bypass auth
F2 – Payment flow	✓ Impersonate	✓ Modify amount	✓ No payment	✓ Leak payment data	✓ Payment DoS	✓ Abuse payment rights

	user/back end		**traceability**			
F4 – Orders flow	**✓ Fake user**	**✓ Modify order**	**✓ Deny order**	**✓ Leak order data**	**✓ Orders API DoS**	**✓ Access other users' orders**

5. Detailed Threat Analysis per Asset

Below is a **justified, narrative analysis** for each asset.

A1 – Client

Spoofing

Attackers may impersonate clients using credential stuffing or stolen tokens.

Tampering

Clients can modify requests (e.g., cart prices) before sending them to the backend.

Repudiation

Without proper server-side logs, users can deny actions.

Information Disclosure

XSS can leak tokens or PII stored in the browser.

Denial of Service

Bots can overload the front-end or API.

Elevation of Privilege

Weak authorization may allow clients to access admin endpoints.

A2 – Administrator

Admins are the **highest-value target**.

Spoofing

Admin account takeover leads to full system compromise.

Tampering

Admins can maliciously modify products, orders, or customer data.

Repudiation

Without immutable logs, admins can deny harmful actions.

Information Disclosure

Admins can export large volumes of PII.

Denial of Service

Admins can disable critical services.

Elevation of Privilege

Admins may escalate to infrastructure-level access.

C1 – Front-end

Spoofing

Attackers can create fake front-ends to steal credentials.

Tampering

DOM manipulation or malicious JS injection alters behavior.

Information Disclosure

XSS leaks tokens and PII.

Denial of Service

CDN overload affects availability.

Elevation of Privilege

Client-side checks can be bypassed.

C2 – Backend

Spoofing

Attackers may spoof API clients using stolen keys.

Tampering

SQL injection or parameter tampering can alter data.

Repudiation

Lack of logs prevents accountability.

Information Disclosure

Verbose errors leak sensitive data.

Denial of Service

Bots can saturate the API.

Elevation of Privilege

RCE or insecure deserialization can lead to server takeover.

C3 – Database

Spoofing

Stolen DB credentials allow unauthorized access.

Tampering

Attackers can modify customer or order data.

Information Disclosure

Full DB dump is catastrophic.

Denial of Service

Heavy queries can lock tables.

Elevation of Privilege

Superuser access compromises the entire system.

C5 – Auth API

Spoofing

Credential stuffing and token replay.

Tampering

Manipulating login or refresh flows.

Information Disclosure

Token leakage in logs or errors.

Denial of Service

Login endpoint saturation.

Elevation of Privilege

Bypassing auth checks.

C6 – Payment API

Spoofing

Impersonating backend or user.

Tampering

Changing payment amounts.

Information Disclosure

Leaking payment references.

Denial of Service

Payment unavailability = revenue loss.

Elevation of Privilege

Unauthorized refunds or captures.

D1 – Customer Data

Information Disclosure

PII breach → GDPR fines, reputation loss.

Tampering

Changing addresses enables fraud.

Spoofing

Identity theft.

D2 – Orders Data

Tampering

Changing order amounts.

Information Disclosure

Order history reveals customer behavior.

D4 – Tokens

Spoofing

Token theft = account takeover.

Tampering

Modifying JWT payload.

Information Disclosure

Tokens leaked in logs.

F1 – Auth Flow

Spoofing

Credential stuffing.

Information Disclosure

Interception of credentials.

F2 – Payment Flow

Tampering

Changing payment amounts.

Information Disclosure

Leaking payment data.

F4 – Orders Flow

Elevation of Privilege

Accessing other users' orders.

6. Impact Analysis

For each threat, students must evaluate:

Technical impact

- Data corruption
- Unauthorized access
- Service unavailability
- Lateral movement
- Token/session compromise

Business impact

- Fraud
- GDPR sanctions
- Loss of revenue
- Loss of customer trust
- Operational disruption

Example: A tampered payment request (F2) → financial fraud → direct revenue loss + legal exposure.

7. Threat Prioritization

Threats are prioritized based on:

1. Business impact

- PII breach → Very High
- Payment fraud → Very High
- Admin compromise → Critical

2. Exploitability

- Credential stuffing → High
- SQL injection → High
- XSS → Medium/High

3. Exposure

- Public APIs → High
- Internal DB → Medium

4. Dependencies

- External payment provider → Medium/High

5. Operational context

- ShopNow already suffered:
    - bot attacks,
    - credential stuffing,
    - log exposure.

Thus, **Spoofing, Tampering, and Information Disclosure** are top priorities.

8. Preparing Security Requirements

Each STRIDE threat must map to a requirement:

STRIDE	Example Requirement
S	MFA, secure token storage, anomaly detection
T	HMAC, JWT signing, input validation
R	Immutable logs, audit trails
I	TLS 1.3, encryption at rest, log masking
D	Rate limiting, WAF, circuit breakers
E	RBAC, least privilege, sandboxing

9. Limits of STRIDE

Students must understand:

- STRIDE does **not** evaluate probability.
- STRIDE does **not** calculate risk by itself.
- STRIDE must be complemented with:
    - **Abuse cases**,
    - **MITRE ATT&CK**,
    - **Business impact analysis**,
    - **Threat intelligence**.

10. Conclusion

This STRIDE analysis identifies all major threats affecting ShopNow's assets, flows, and components. It highlights the most critical risks:

- Spoofing of admin and user accounts
- Tampering of orders and payments
- Information disclosure of PII and tokens
- DoS on authentication and backend APIs

- Privilege escalation via backend or DB

This analysis directly informs:

- **Security requirements (Exercise 3)**
- **Zero Trust architecture (Exercise 4)**
- **Security testing**
- **Detection & response**

It provides a complete, justified foundation for securing the ShopNow platform.