Name: Watcharapol Yotadee

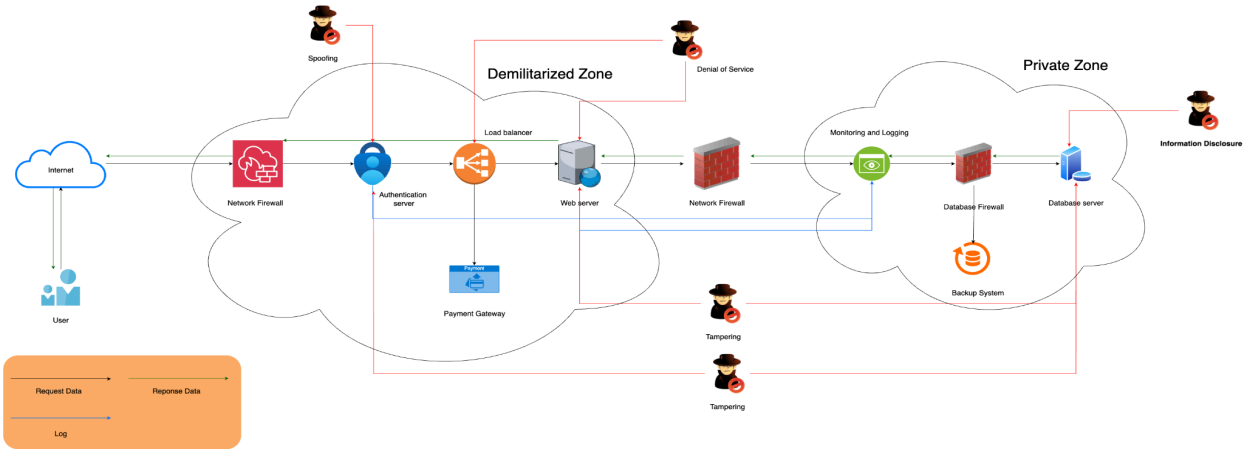
Nickname: Fluke

# Assignment Week 8

## Application Security

1. Based on the system you designed in week 4 (or any other systems), draw a threat model including mitigation controls. Use any threat modeling technique of your choice to help you identify and document all potential threats in your system.

(ระบบจองเช่ารถยนต์)




| Threat | Threat Definition | Proactive Control |
|---|---|---|
| Spoofing | - ผู้ไม่หวังดีอาจปลอมตัวเป็นผู้ใช้ที่ ได้รับอนุญาตเพื่อเข้าถึงระบบ | - ใช้ Multi-factor Authentication เพื่อเพิ่มความปลอดภัย<br>- ใช้ OAuth 2.0 หรือ JWT สำหรับ Authentication |
| Tampering | - ผู้โจมตีอาจแก้ไขข้อมูล เช่น การเปลี่ยนรายละเอียดการจอง | - เข้ารหัสข้อมูลด้วย TLS1.2+ ระหว่างการส่งข้อมูล<br>- ใช้ Digital Signatures เพื่อยืนยันความถูกต้องของข้อมูล |
| Repudiation | - ผู้ใช้สามารถปฏิเสธการกระทำ เช่น อ้างว่าไม่ได้จองรถ | - ใช้ Audit Logs ที่เก็บข้อมูลเหตุการณ์ทุกครั้งที่เกิดการจอง<br>- เก็บ Timestamp และ User ID ของแต่ละธุรกรรม |
| Information Disclosure | - ผู้ไม่หวังดีอาจเข้าถึงข้อมูลส่วนตัวลูกค้าหรือข้อมูลการจอง | - เข้ารหัสข้อมูลใน Database ด้วย AES-256<br>- ใช้ Role-based Access Control เพื่อจำกัดสิทธิ์การเข้าถึง |

Name: Watcharapol Yotadee

Nickname: Fluke

| | | |
|---|---|---|
| Denial of Service (DoS) | - การส่งคำขอปริมาณมากเพื่อ<br>ทำให้ระบบไม่สามารถให้บริการ | - ใช้ Rate Limiting และ Web Application Firewall<br>เพื่อลดผลกระทบจากการโจมตี<br>- ใช้ Load Balancer<br>เพื่อกระจายทราฟฟิก |
| Elevation of Privilege | - ผู้โจมตีสามารถยกระดับสิทธิ์<br>เช่น เปลี่ยนจากผู้ใช้ธรรมดาเป็น<br>ผู้ดูแลระบบ | - ตรวจสอบ Role และ Permission อย่างเข้มงวด<br>- ใช้ Least Privilege Principle |

Name: Watcharapol Yotadee

Nickname: Fluke

2. Select only 1 threat identified in your system, explain how you can apply OWASP's Top 10 Proactive Controls to enhance the security of your system.

**Threat ที่เลือกคือ Tampering**

**นิยามคือ ผู้ไม่หวังดีสามารถแก้ไขข้อมูลของเรา เช่น การเปลี่ยนแปลงข้อมูลการจอง หรือรายละเอียดในกรการรับ-ส่งข้อมูลระหว่าง ผู้ใช้, Web Server, และ Database โดยไม่ได้รับอนุญาต**

OWASP's Top 10 Proactive Controls 2021 มาใช้งาน

C1: Define Security Requirements

- กำหนดให้ข้อมูลทุกส่วนในระบบ เช่น ข้อมูลการจอง และการชำระเงิน มีการป้องกันผ่าน TLS และใช้โครงสร้างข้อมูลที่รองรับความปลอดภัยสูง
- กำหนด security policies ที่ชัดเจน
  เช่น ห้ามรับหรือส่งข้อมูลผ่าน HTTP (บังคับ HTTPS)

C2: Leverage Security Frameworks and Libraries

- ใช้ Framework ที่มีระบบ Validation ในตัว เช่น Spring Security หรือ Django Security
- ใช้ Cryptography Libraries เช่น libsodium หรือ bcrypt สำหรับการตรวจสอบและการยืนยันข้อมูล

C4: Encode and Escape Data

- เข้ารหัสข้อมูลที่รับ-ส่งระหว่างผู้ใช้และเซิร์ฟเวอร์ ด้วย TLS 1.2 เพื่อลดความเสี่ยงจากการดักจับข้อมูล
- ใช้ JSON Web Tokens หรือ HMAC เพื่อยืนยันว่าไม่มีการแก้ไขข้อมูลระหว่างส่ง

C6: Implement Digital Signatures

- ใช้ Digital Signatures เพื่อตรวจสอบความถูกต้องของข้อมูล เช่น การยืนยันว่าข้อมูลการจองที่ส่งมาจาก User ไม่ถูกดัดแปลงระหว่างทาง
- ใช้ Signed Tokens หรือ Certificates สำหรับเซสชันที่เกี่ยวข้องกับข้อมูลสำคัญ

C8: Protect Data Everywhere

- เข้ารหัสข้อมูลสำคัญที่จัดเก็บใน Database ด้วย AES-256
- ใช้การเข้ารหัสแบบ End-to-End ในการส่งข้อมูล เช่น ข้อมูลการจองผ่าน Payment Gateway
- ตรวจสอบให้แน่ใจว่าคีย์เข้ารหัสถูกจัดการอย่างปลอดภัย

C9: Implement Security Logging and Monitoring

- บันทึกเหตุการณ์ที่เกี่ยวข้องกับข้อมูลสำคัญ เช่น การแก้ไขข้อมูลโดยไม่ได้รับอนุญาต การเข้าถึงที่ผิดปกติจากผู้ใช้งาน
- ใช้ Monitoring Tools เช่น ELK Stack หรือ Splunk เพื่อแจ้งเตือนทันทีหากพบพฤติกรรมที่ผิดปกติ
- แจ้งเตือนผู้ดูแลระบบทันทีหากพบพฤติกรรมที่ไม่ปกติ