PASTA Worksheet for Sneaker Company Mobile App

Stage I: Define Business and Security Objectives

Business Objectives:

- Ensure seamless connections between buyers and sellers, with secure messaging and account management.
- Maintain user data privacy and comply with industry regulations.
- Enable a smooth and legally compliant payment process with multiple payment options.

Stage II: Define the Technical Scope

Technologies Used:

- Application Programming Interface (API)
- Public Key Infrastructure (PKI)
- SHA-256
- Structured Query Language (SQL)

Prioritization of Technology: The **API** is prioritized as it acts as the primary medium for data exchange between the mobile app and backend servers. APIs can be vulnerable to unauthorized access, injection attacks, and data leaks. Proper authentication and encryption mechanisms must be enforced to secure API endpoints.

Stage III: Decompose Application

Sample Data Flow:

- User requests product details → API fetches data from the database → API sends data to the user interface.
- User submits payment → API transmits payment details securely to the payment gateway.
- User messages seller → API processes and transmits messages.

Stage IV: Threat Analysis

Potential Threats:

- **Internal Threat:** Insider threats such as unauthorized employee access to sensitive data.
- **External Threat:** SQL Injection attacks targeting the database to extract sensitive information.

Stage V: Vulnerability Analysis

Potential Vulnerabilities:

- **Codebase Weakness:** Poorly sanitized input fields allowing SQL Injection.
- **Database Weakness:** Inadequate encryption of stored credit card data, making it susceptible to theft.

Stage VI: Attack Modeling

Sample Attack Tree:

- **Goal:** Gain unauthorized access to user payment data.
    - **Method 1:** Exploit API vulnerabilities.
        - Use brute-force attacks on authentication.
        - Inject malicious scripts.
    - **Method 2:** Target database vulnerabilities.
        - Execute SQL Injection to extract sensitive information.
        - Exploit weak encryption techniques.

Stage VII: Risk Analysis and Impact

Security Controls to Mitigate Threats:

1. **API Security Best Practices:** Implement authentication using OAuth 2.0 and enforce rate limiting to prevent API abuse.
2. **Database Security Enhancements:** Use parameterized queries and stored procedures to mitigate SQL Injection risks.
3. **Encryption and Secure Storage:** Utilize AES-256 encryption for sensitive user data and enforce proper key management.
4. **Regular Security Audits:** Conduct vulnerability assessments and penetration testing to identify and remediate security gaps.