PASTA worksheet

Maria Audu

Stages	**Sneaker company**
I. Define business and security objectives	Make **2-3 notes** of specific business requirements that will be analyzed. ● *Will the app process transactions?* ● *Does it do a lot of back-end processing?* ● *Are there industry regulations that need to be considered?* Here are **2–3 specific business requirements** that will be analyzed for the sneaker app: ● **The app will process financial transactions**, including handling credit card and digital payment information, which requires secure payment processing and compliance with payment regulations. ● **The app relies heavily on back-end processing**, such as managing user accounts, messaging between buyers and sellers, inventory listings, ratings, and transaction records. ● **The app must comply with industry and data privacy regulations**, such as PCI DSS for payment data and data protection laws (e.g., GDPR or similar), to avoid legal and compliance risks.
II. Define the technical scope	List of technologies used by the application: ● *API* ● *PKI AES and RSA* ● *AES* ● *SHA-256* ● *SQL* Write **2-3 sentences** (40-60 words) that describe why you choose to prioritize that technology over the others. I would prioritize evaluating the **API and SQL technologies** because they directly handle user requests and interact with sensitive backend data. APIs and databases are common attack targets, and weaknesses such as broken authentication or SQL injection could expose user credentials, personal information, and payment data, creating significant security and legal risks.

III. Decompose application	Sample data flow diagram
IV. Threat analysis	List **2 types of threats** in the PASTA worksheet that are risks to the information being handled by the application. ● *What are the internal threats?* ● *What are the external threats?* ● **Credential-based attacks** such as phishing or credential stuffing, which could allow attackers to gain unauthorized access to user accounts and sensitive personal or payment information. ● **Injection and API abuse attacks**, including SQL injection or malicious API requests, that could be used to access, alter, or exfiltrate data stored in the application's backend databases. ● **Internal threats:** Employees or contractors misusing legitimate access, either intentionally or accidentally, such as accessing customer data without authorization or misconfiguring databases or APIs, leading to data exposure. ● **External threats:** Malicious actors outside the organization launching attacks such as phishing, credential stuffing, SQL injection, or API exploitation to steal user data, commit fraud, or disrupt application services.
V. Vulnerability analysis	List **2 vulnerabilities** in the PASTA worksheet that could be exploited. ● *Could there be things wrong with the codebase?* ● *Could there be weaknesses in the database?* ● *Could there be flaws in the network?* ● **Insecure or poorly written code** in the application or API, such as missing input validation or improper authentication checks, which could allow attackers to exploit vulnerabilities like SQL injection or unauthorized access. ● **Database and network weaknesses**, including misconfigured SQL databases, weak access controls, or unencrypted network connections, which could expose sensitive user and payment data to attackers.
VI. Attack modeling	Sample attack tree diagram
VII. Risk analysis and impact	List **4 security controls** that you've learned about that can reduce risk.

	• **Multi-factor authentication (MFA)** to reduce the risk of unauthorized access to user and administrative accounts. • **Encryption of data in transit and at rest** using strong cryptographic standards to protect sensitive information. • **Input validation and secure coding practices**, such as prepared statements, to prevent injection attacks. • **Continuous logging, monitoring, and intrusion detection** to quickly identify and respond to suspicious activity or potential breaches.