

Case Study:

Apply the Process of Attack Simulation and Threat Analysis (PASTA) Threat Model Framework

Scenario:

You're part of the growing security team at a company for sneaker enthusiasts and collectors. The business is preparing to launch a mobile app that makes it easy for their customers to buy and sell shoes.

You are performing a threat model of the application using the PASTA framework. You will go through each of the seven stages of the framework to identify security requirements for the new sneaker company app.

PASTA Report

Stages	Sneaker company
I. Define business and security objectives: Make 2-3 notes of specific business requirements that will be analyzed.	<ul style="list-style-type: none">• Users can create member profiles internally or by connecting external accounts.• The app must process financial transactions.• The app should be in compliance with PCI-DSS.
II. Define the technical scope: List of technologies used by the application:	<ul style="list-style-type: none">• Application programming interface (API)• Public key infrastructure (PKI)• SHA-256• SQL <p>APIs facilitate the exchange of data between customers, partners, and employees, so they should be prioritized. They handle a lot of sensitive data while they connect various users and systems together. However, details such as which APIs are being used should be considered before prioritizing one technology over another. So, they can be more prone to security vulnerabilities because there's a larger attack surface.</p>
III. Decompose application	PASTA 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.	<ul style="list-style-type: none">• Injection• Session hijacking
V. Vulnerability analysis: List 2 vulnerabilities in the PASTA worksheet that could be	<ul style="list-style-type: none">• Lack of prepared statements• Broken API token

exploited.	
VI. Attack modeling	PASTA Attack Tree Diagram
VII. Risk analysis and impact: List 4 security controls that can reduce risk.	SHA-256, incident response procedures, password policy, principle of least privilege
