PASTA Methodology for Application Security

Stage I: Define Business and Security Objectives

Summary:
These objectives are defined early by asking broad questions about the purpose of the application. For example, how does the app make the business money? Understanding the answer to these questions helps guide the detailed work that will follow.

Recommendations:
A shopping application like this will need to process payments. Based on this description, we know certain technologies are required to keep information private and secure and that everything will need to be compliant with PCI-DSS.

Stage II: Define the Technical Scope

Summary:
The objective here is to understand the attack surface by identifying the technologies being used by the application and understanding their dependencies.

Recommendations:
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. APIs can be more prone to security vulnerabilities because there’s a larger attack surface.

Stage III: Decompose the Application

Summary:
Stage three builds upon the previous stage by investigating how the application's components communicate together. The objective here is to review how the application works and how security controls are currently implemented.

Recommendations:
The sample data flow diagram shows how a typical search request passes through multiple layers. One thing you might review here would be to ensure the MySQL database is using prepared statements when queries are input.

Stage IV: Threat Analysis

Summary:
The main objective of stage four is to consider the types of threats that might affect your application. This relates to the technologies you've already scoped. Another thing to consider is the types of data your application will be processing.

Recommendations:
Injection attacks are common for SQL databases. Session hijacking is possible because the app communicates cookies between multiple layers. It's important to consider your technological attack surface and any relevant threats to your product to effectively implement your information security responsibilities.

Stage V: Vulnerability Analysis

Summary:
Stage five is about associating asset vulnerabilities with potential threats. The objective here is to identify what is wrong with the design of the app or its codebase based on your security testing.

Recommendations:
A lack of prepared statements can make our SQL database vulnerable to injection attacks. Session hijacking is possible if cookies are mishandled between input and output sources.

Stage VI: Attack Modeling

Summary:
In this stage, the objective is to link the threats and vulnerabilities identified in the previous steps using attack trees. The purpose of using attack trees here is to show that the potential threats that you've identified are actually viable. Resources like MITRE ATT&CK and the CVE® list are useful references to find evidence that validates the information that you've modeled in your attack tree.

Recommendations:
This sample attack tree models how user data is vulnerable to the attacks that were identified earlier. Like the sample data flow diagram, an actual attack tree for a mobile application would be much more complex than this.

Stage VII: Risk Analysis and Impact

Summary:
The objective of the final stage of PASTA is to identify ways to mitigate the risks that were identified from stages IV - VI and plan for any remaining risks that can't be remediated.

Recommendations:
SHA-256, incident response procedures, password policy, and principle of least privilege are a few examples of technical, operational, and managerial controls that can be implemented before launch to reduce risk.