## PASTA worksheet

|  |  |
| --- | --- |
| **Stages** | **Sneaker company** |
| **I. Define business and security objectives** | * The app wants to connect sellers and buyers of sneakers, so it needs to securely process transactions, adhering to the PCI-DSS standards as to avoid fines. * The app does a good amount of backend operations that handle user data, which the company wants to handle properly and securely, which should comply with ISO/IEC 27002 and GDPR. To compliment, compliance with the NIST CSF should be considered. |
| **II. Define the technical scope** | List oftechnologies used by the application:   * *API* * *PKI* * *AES* * *SHA-256* * *SQL*   The main priority should be preventing SQL injections, since the other technologies are less risky in this scenario. Indeed, SQL injections are very common attacks that could cause tremendous damage, if the SQL database is not properly secured. |
| **III. Decompose application** | [Sample data flow diagram](https://docs.google.com/presentation/d/1ol7y79popTFfNHM-90ES-H-i1Lpd0YNvPShxBlXozjg/template/preview?resourcekey=0-DZAkf7Vzh2PXsP-j3oXV-g) |
| **IV. Threat analysis** | * The most important internal threat is an employee accidentally or maliciously revealing sensitive information like passwords and usernames or entering an SQL injection. * The most important external threat to consider is a hacker that wants to gain access to sensitive database information, like credit card payments, PII or other information in the database, either through a malware or an SQL injection. |
| **V. Vulnerability analysis** | An important vulnerability is the payment system, since if transactions aren’t properly encrypted and compliant to PCI-DSS regulation, they could be exploited by a threat actor.  Another important one is the configuration of the SQL database, which if done improperly could be exploited to gain access to sensitive information or even the app itself. A common solution would be implementing prepared statements to avoid SQL injections and encrypting the database and communications in the application. |
| **VI. Attack modeling** | [Sample attack tree diagram](https://docs.google.com/presentation/d/1FmWLyHgmq9XQoVuMxOym2PHO8IuedCkan4moYnI-EJ0/template/preview?usp=sharing&resourcekey=0-zYPY7AhPJdcClXamlAfOag) |
| **VII. Risk analysis and impact** | A list of security controls that could minimize risks is:   1. Prepared statements 2. Strong password policies 3. Principle of least privilege 4. Encrypt transactions and communications in the application. |