**What is threat modelling?**

Threat modeling works to identify (This is a threat), communicate (inform responsible people), and understand threats and mitigations within the context of protecting something with value

It is a structured representation of all the information that affects the security of an application. It is the view of an application and its environment through the lens of security.

**Threat model typically includes:**

1. Include the description of the model to be protected. (this is a mobile app that gives offers to people and is scalable….)
2. Assumptions that can be checked or challenged in the future as the threat landscape changes.
3. Potential threats to the system.
4. Actions that can be taken to mitigate each threat.
5. A way of validating the model and threats, and verification of success of actions taken.

**Threat Modeling Process:**

Threat modeling process can be decomposed to three steps:

1. **Decompose the Application:** Understand the application and how it works.
   1. Creating use cases to understand how the application is used.
   2. Take notes of the threat model information. (Application name, version, description, document owner, participants and reviewers).
   3. Identify External Dependencies. (e.g., an Apache server and how it will be hardened).
   4. Identify Entry points which is the interfaces in which the attacker can interact with the application. (E.g., a search engine).
   5. Identify External points which is where the data exists the system. It could be useful for account harvesting, XSS, and information disclosure vulnerabilities.
   6. Identify Assets That could be a target for the attacker.
   7. Identify Trust Levels.

This information is used to produce DFDs, which provides visual representation of how the application processes the data, and helps identify affected components.

1. **Determine and Rank Threats:**

DFDs from step one can be used to identify the potential threat targets from the attacker’s perspective. These threats can be categorized using threat models like STRIDE which categorizes the threats from the attacker’s perspective in a more general perspective. And ASF which categorizes the threats from a defense perspective. The determination for the risk for each threat can be made using a value-based metric risk model such as DREAD (numeric) or a Qualitative Risk Model. (EX, DM, SCOPE(high, low, medium)).

1. **Determine Countermeasures and Mitigations:**

Determine some kind of protective measure. When threats are identified within the category of threat models like STRIDE and ASF… it’s possible to find appropriate counter measurements.

* **STRIDE:**

STRIDE is an acronym for the type of threats it covers. (Spoofing, Tampering, Repudiation, Information Disclosure, DoS, and Elevation of Privilege).

STRIDE is used along with a model of the target system which makes it very valuable for identifying individual systems. It can help identify threats during system design at an early stage of the system software. And while modifying an existing system to evaluate whether any change will introduce new vulnerabilities. It also can be used post-breach it can be used to understand the nature of the attack and to comprehend what went wrong.

STRIDE is a threat modeling method that classifies potential threats into six categories.

Used: during system design, simpler systems (people with no experience with Threat modeling), and in combination with DFDs.

* **PASTA:**

Seven-step risk-centric methodology, the steps include defining the scope, identify and enumerate the threats. Identify vulnerabilities, analyze attacks, analyze weaknesses, correlating information, and producing a report.

It offers an in-depth analysis of potential threats, from identification to attack simulation and weakness analysis.

It is business-centric: more business-focused security approach.

It helps with risk prioritization

Used: risk analysis that has attack simulations and weakness simulations. Business Centric (consider business objectives and related risks) if you want to align your security considerations with your business goals. Complex Systems: good when there is many potential threats.

* **CVSS (Common Vulnerability Scoring System):**

Produce a reliable risk ranking to ensure the exploits are taken seriously. It does not reduce the attack surface. It’s just a scoring system.

Its more complex than STRIDE/PASTA

4 CVSS does not find or reduce the attack surface area (i.e. design flaws), or help enumerate risks within any arbitrary piece of code, as it is just a scoring system, not a modeling methodology. 4 CVSS is more complex than STRIDE/DREAD, as it aims to calculate the risk of announced vulnerabilities as applied to deployed software and environmental factors. 4 The CVSS risk ranking is complex – a spreadsheet is required to calculate the risk components as the assumption behind CVSS is that a specific vulnerability has been identified and announced, or a worm or Trojan has been released targeting a small number of attack vectors. 4 The overhead of calculating the CVSS risk ranking is quite high if applied to a thorough code review, which may have 250 or more threats to rank. IT WONT COVER ASPECT OF BUSINESS