Traits of an effective threat model

Threat modeling is the process of identifying assets, their vulnerabilities, and how each is exposed to threats. It is a strategic approach that combines various security activities, such as vulnerability management, threat analysis, and incident response. Security teams commonly perform these exercises to ensure their systems are adequately protected. Another use of threat modeling is to proactively find ways of reducing risks to any system or business process.

Traditionally, threat modeling is associated with the field of application development. In this reading, you will learn about common threat modeling frameworks that are used to design software that can withstand attacks. You'll also learn about the growing need for application security and ways that you can participate.

Why application security matters

Applications have become an essential part of many organizations' success. For example, web-based applications allow customers from anywhere in the world to connect with businesses, their partners, and other customers.

Mobile applications have also changed the way people access the digital world. Smartphones are often the main way that data is exchanged between users and a business. The volume of data being processed by applications makes securing them a key to reducing risk for everyone who's connected.

For example, say an application uses Java-based logging libraries with the Log4Shell vulnerability (CVE-2021-44228). If it's not patched, this vulnerability can allow remote code execution that an attacker can use to gain full access to your system from anywhere in the world. If exploited, a critical vulnerability like this can impact millions of devices.

Defending the application layer

Defending the application layer requires proper testing to uncover weaknesses that can lead to risk. Threat modeling is one of the primary ways to ensure that an application meets security requirements. A DevSecOps team, which stands for development, security, and operations, usually performs these analyses.

A typical threat modeling process is performed in a cycle:

- Define the scope
- Identify threats
- Characterize the environment
- Analyze threats
- Mitigate risks
- Evaluate findings

Ideally, threat modeling should be performed before, during, and after an application is developed. However, conducting a thorough software analysis takes time and resources. Everything from the application's architecture to its business purposes should be evaluated. As a result, a number of threat-modeling frameworks have been developed over the years to make the process smoother.

Note: Threat modeling should be incorporated at every stage of the software development lifecycle, or SDLC.

Common frameworks

When performing threat modeling, there are multiple methods that can be used, such as:

- STRIDE
- PASTA
- Trike
- VAST

Organizations might use any one of these to gather intelligence and make decisions to improve their security posture. Ultimately, the “right” model depends on the situation and the types of risks an application might face.

STRIDE

STRIDE is a threat-modeling framework developed by Microsoft. It’s commonly used to identify vulnerabilities in six specific attack vectors. The acronym represents each of these vectors: spoofing, tampering, repudiation, information disclosure, denial of service, and elevation of privilege.

PASTA

The **Process of Attack Simulation and Threat Analysis** (PASTA) is a risk-centric threat modeling process developed by two OWASP leaders and supported by a cybersecurity firm called VerSprite. Its main focus is to discover evidence of viable threats and represent this information as a model. PASTA's evidence-based design can be applied when threat modeling an application or the environment that supports that application. Its seven stage process consists of various activities that incorporate relevant security artifacts of the environment, like vulnerability assessment reports.

Trike

Trike is an open source methodology and tool that takes a security-centric approach to threat modeling. It's commonly used to focus on security permissions, application use cases, privilege models, and other elements that support a secure environment.

VAST

The Visual, Agile, and Simple Threat (VAST) Modeling framework is part of an automated threat-modeling platform called ThreatModeler®. Many security teams opt to use VAST as a way of automating and streamlining their threat modeling assessments.

Participating in threat modeling

Threat modeling is often performed by experienced security professionals, but it's almost never done alone. This is especially true when it comes to securing applications. Programs are complex systems responsible for handling a lot of data and processing a variety of commands from users and other systems.

One of the keys to threat modeling is asking the right questions:

- What are we working on?
- What kinds of things can go wrong?
- What are we doing about it?
- Have we addressed everything?
- Did we do a good job?

It takes time and practice to learn how to work with things like data flow diagrams and attack trees. However, anyone can learn to be an effective threat modeler. Regardless of your level of experience, participating in one of these exercises always starts with simply asking the right questions.

Key takeaways

Many people rely on software applications in their day to day lives. Securing the applications that people use has never been more important. Threat modeling is one of the main ways to determine whether security controls are in place to protect data privacy. Building the skills required to lead a threat modeling activity is a matter of practice. However, even a security analyst with little experience can be a valuable contributor to the process. It all starts with applying an attacker mindset and thinking critically about how data is handled.