CONTACT

Questions or suggestions?
Please contact us!

Team Q-Sec
q-sec@otto.de

Teams: EC-Security-Champions
PQ 421

THREAT MODELING

6 COUNTERMEASURES

Countermeasures are derived and documented for each threat in the form of an evil user story / abuse story, according to their priority.

Risks can never be completely eliminated and measures should always be proportionate. So you can also accept a risk.

ID/ JIRA ID	Threat	Asset	Prio	measure with Acceptance	Implemented
?	As a Git function user, I want to change the code so that all user data is sent to me.	User data	4,7	1. Finetune the GitWorkflow	?

7 NEXT STEPS

DO THE WORK! Documentation of the stories, writing the tickets and scheduling the work.

SIMPLE THREAT MODELING

The gradually guidance

INTRODUCTION

Threat modeling is part of the design phase in the Software Development Lifecycle (SDLC). In an agile environment, these phases are continuously carried out. Threat modeling helps in the early identification of design flaws that may lead to potential threats and to derive countermeasures against them.

The threat modeling process described below contains simplified elements of well-known methods, such as: Microsoft STRIDE, DREAD etc. Another method is the definition of „evil user stories“ or „abuse stories“ that is described below, which should be used for every feature request.

PREPARATION

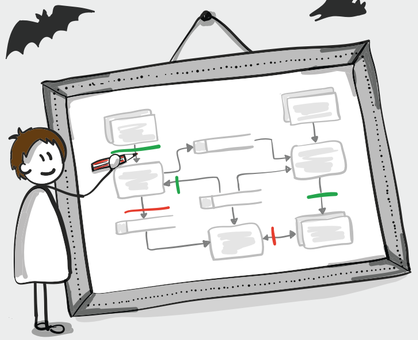
Threat modeling sessions are team events and should be carried out initially by the entire team, since threats present themselves not only at a technical level. In order to promote continuous improvement, threat modeling should be carried out before major software releases or after six months at the latest.

The room for the execution of threat modelling sessions should offer enough space for diagram drawings and post-its wallpapers. It is the same with time: Initially, threat modeling sessions commonly take at least 3 hours. The threat modeling poster can help you to carry out small-scale threat modeling yourself. For moderation support, feel free to contact Q-SEC at q-sec@otto.de

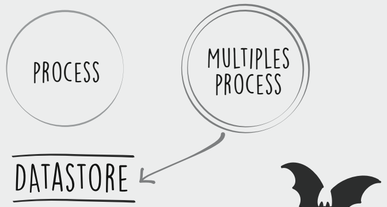
EXECUTION

During a threat modeling session, it is advisable to base discussions according to facts and not to fall into a deep technical discussions at every single step. Additionally, the scope of the session should be defined: Do we want to model the entire system or a specific service? The software build systems and relevant environments, like Jenkins, should each be modelled as individual services.

1 BIG PICTURE



A simplified Data-Flow-Diagram (DFD) with the following symbols helps: Process, Service, Datastore, Flow with type of connection



Develop the DFD with the team from the ground up. Ideally with Post-Its, so that everything can be rearranged at a later time

2 ASSETS AND ACTORS

All assets and actors involved in the information flow are identified and assigned to elements in the previously drawn DTD.

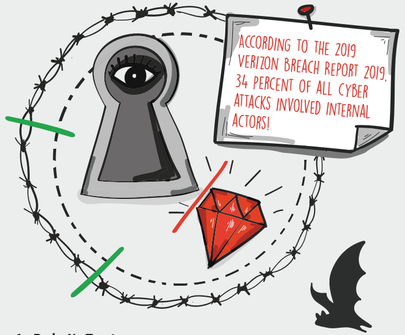
- **Assets = The „Crown Jewels“ of a company**
  - Information and properties that compose the system
  - Abstract: availability, integrity, reputation, credentials, ...
  - Examples: Account balance manipulation, no sexual content wherever children's products are expected, accessibility and response times, API keys?
- **Actors = Personas = Roles**
  - Attackers can only interact with designated roles
  - Examples: Functional users, AWS Admin, Team members, GitHub Users, all employees with company Accounts

3 TRUSTED BOUNDARIES

Trusted boundaries are established in order to help to evaluate the relationships between services (internal and external)

Zero trust environments are of the highest form and many approaches should be applied by default in cloud environments:

- HTTPS with valid certificates,
- Appropriate Authentication and Authorization,
- Language-specific tools for sanitizing, encoding and escaping; For Example: https://github.com/owasp/json-sanitizer,
- Technical review should always be done as detailed as possible - validation,
- Monitoring, to define thresholds and alarming



1. **Red = No Trust**
   (mainly external access and services)
   ❗ Should be in focus
2. **Green = Trust**
   ❗ Should NOT be neglected

4 THREAT IDENTIFICATION

After we have defined the system in detail, we let our „evil creativity“ run wild.

Now it's time to change our perspective to the one of an attacker: Imagine that someone offers you a lot of money for a successful attack! Here are some ideas:

Threat	Desired property	Description
Spoofing	Authenticity	Phishing, falsifying IP packets, ▶ Tchibo instead of OTTO is displayed
Tampering	Integrity	Log File changes, obfuscate Alarms ▶ Data is loaded in order to damage the reputation ▶ Only products from a specific retailer are displayed
Repudiation	Nonrepudiability	No mapping of who did what - functional user, multiple assignment of API keys ▶ To obtain access without being able to track it back to you
Information disclosure (Data leak)	Confidentiality	Leak sensible information ▶ Employee information on a public S3 Bucket ▶ User Credentials are exposed
Denial of Service	Availability	Flood a service with useless requests to the point of failure over capacity limits ▶ The response times are brought down to the floor
Elevation of Privilege	Authorization	Obtain additional rights with existing access rights, For example, to compromise the build system

5 THREAT PRIORITIZATION

The threat scenarios are presented one by one so that everyone understands and can prioritize them. Similar scenarios are summarised before prioritization.

Six risk elements are assessed separately for prioritization. The rating ranges from 1 (very low) to 5 (very high). The average determines the prioritization.

- **1-Complexity, Know- How**
  1. Low complexity = 5
  2. Very complex = 1
- **2-Detection and reaction possibility**
  1. Prompt detection and reaction= 1
  2. Prompt detection but slow reaction= 3
  3. No detection and reacted to late= 5
- **3-Number of Actors (in relation to the system)**
  1. Few Actors = 1
  2. Many Actors = 5
- **4-Extend of Damage**
  - What is the expected damage upon occurence
  - 4% of sales if personal data is lost
- **5- Repeatability**
  - How often can the threat be successfully executed
  - Does the functionality allow mitigation or not
- **6-Number of affected users**
  - The more users are affected, the greater the financial impact
- **7-Motivation of the attacker (Optional)**
  - How high is the motivation of an attacker to take advantage of this threat = what value does it have for him?

The average rating for each threat now specifies the order of prioritization. If necessary, the individual elements can be weighted. Please don't go too scientific and trust your gut. Should there be any very different assessments in the team, you can discuss them, similar to „Planning Poker“.