Threat analysis

Tuomas Aura
CS-C3130 Information security

Aalto University, autumn 2017
Outline

- What is security
- Threat analysis
- Threat modeling example
- Systematic threat modeling
WHAT IS SECURITY
What is security

- When talking about security, we are concerned about **bad events caused with malicious intent**
  - Security vs. reliability?

- Terminology:
  - **Threat** = bad event that might happen
  - **Attack** = someone intentionally causes the bad thing to happen
  - **Vulnerability** = weakness in an information system that enables an attack
  - **Exploit** = implementation of an attack
  - **Risk** = probability of an attack $\times$ damage in euros
Security Goals

- **Confidentiality, Integrity, Availability “CIA”**
  - **Confidentiality** — protection of secrets
  - **Integrity** — only authorized modification of data and system configuration
  - **Availability** — no denial of service, business continuity

- Examples: web server, customer data

- The CIA model is a good starting point but not all:
  - **Access control** — no unauthorized use of resources
  - **Privacy** — control of personal data and space
  - What else?

- Security is a non-functional property of a system
Some goals not covered by CIA

- **Authentication** for access control and accountability
- Accounting, payment
- Content protection
- **Protection of services and infrastructure in a hostile environment** (e.g. Internet)
- Anonymity, freedom of expression
- Control and monitoring
Who is the attacker?

- We partition the world into good and bad entities
  - Honest parties vs. attackers, red vs. blue
  - Good ones follow specification, bad ones do not
  - Must consider different partitions to cover all viewpoints

- Typical attackers:
  - Curious individuals
  - Friends and family
  - Dishonest people — for personal gain, making and saving money
  - Hackers, script kiddies — for challenge and reputation
  - Companies — for business intelligence and marketing
  - Organized criminals — for money
  - Governments and security agencies — NSA, SVR RF, GCHQ, DGSE, etc.
  - Military SIGINT — strategic and tactical intelligence, cyber defense

- Insiders are often the greatest threat
  - Employee, administrator, service provider, customer, family member

- Often, not all types of attackers matter
  - Who would you not want to read your email?
THREAT ANALYSIS
Threats

- **Threat** = something bad that can happen
- **Given an system or product**
  - What **assets** are there to lose and to protect?
  - Who are the **stakeholders**?
  - What are the **threats** against the assets?
  - How serious are the threats i.e. what is the **risk**?
- **Threat analysis requires both security and domain expertise**
Threat modeling approaches

- Different angles to threat modeling:
  - **Assets**: what is valuable in the system and how could it be lost?
  - **Attackers and their motivations**: who would want to do something bad and why?
  - **Engineering**: what parts are there in the system and how could they fail?
    - Draw diagrams of the system architecture, data flows etc.
  - **Countermeasures**: what is or could be done to prevent or mitigate attacks?
  - **Checklists, lessons learned**: what has gone wrong in the past?
THREAT MODELING EXAMPLE:
STUDENT LUNCH DISCOUNT

See a separate document for the details
SYSTEMATIC THREAT MODELING
Basic security goals

- Consider first the well-known security goals:
  - Confidentiality
  - Integrity
  - Availability
  - Authentication
  - Authorization, access control
  - Non-repudiation

- Which goals apply to the system? How could they be violated?
Checklist: some threats to consider

- Typical assets: the product or service, money paid or saved, customer and business information, reputation, and the system components
- Crime motivated by money
  - Theft, fraud
  - Industrial espionage, dishonest ways to competitive advantage
  - Corruption: tax evasion, subsidy fraud, bribery, theft by those in power
- Threats to customer data and personal privacy
- Insider threats
  - Employees, IT administrators, trusted entities
  - Curiosity and pretty theft
  - Misincentivized employees “doing their best”
- Privilege escalation and stepping stones to further attacks
  - Threats to accounts, devices and administration
  - Weaknesses in authentication credentials: issuing and use
  - Bypassing controls, misuse of reputation systems
- Social-engineering threats
- Threats related to error handling and failure recovery
- Threats to business continuity
  - Denial-of-service attacks, crisis management issues, business risks
- Public safety threats
  - Critical infrastructure, vehicles, food safety
- Threats against brands and reputation
- Political threats:
  - Nation-state actors, terrorism, authoritarian governments, dependence on hostile powers
Threat trees

Lecturer’s opinion: Threat trees are pretty useless as an analysis tool, but can be ok as a way to present the results of systematic analysis by experts.

Each leaf is a secondary threat to be evaluated.
STRIDE

- STRIDE model used at Microsoft:
  - Spoofing vs. authentication
  - Tampering vs. integrity
  - Repudiation vs. non-repudiation
  - Information disclosure vs. confidentiality
  - Denial of service vs. availability
  - Elevation of privilege vs. authorization

- Idea: divide the system into components and analyze each component for these threats
  - Note: security of components is necessary but not sufficient for the security of the system

Note: STRIDE is intended for analyzing vulnerabilities in PC or server software, with access to the code. When applying it to distributed computing, humans or cyber-physical systems, you need to think creatively.
STRIKE

- Model the software system as a **data flow diagram (DFD)**
  - **Data flows**: network connections, RPC
  - **Data stores**: files, databases
  - **Processes**: programs, services
  - **Interactors**: users, clients, services etc. connected to the system
- Also mark the **trust boundaries** in the DFD
- Consider the following threats:

<table>
<thead>
<tr>
<th></th>
<th>Spoofing</th>
<th>Tampering</th>
<th>Repudiation</th>
<th>Information disclosure</th>
<th>Denial of service</th>
<th>Elevation of privilege</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data flow</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data store</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Process</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Interactor</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Risk assessment

- Risk assessment is very subjective, many definitions:
  - Risk = probability of attack × damage in euros
  - 0 < Risk < 1
  - Risk ∈ { low, medium, high }

- Numerical risk values tend to be meaningless:
  - What does risk level 0.4 mean in practice?

- Usually difficult to assess absolute risk but easier to prioritize threats

- Risk assessment models, e.g. DREAD
  - Damage: how much does the attack cost to defender?
  - Reproducibility: how reliable is the attack
  - Exploitability: how much work to implement the attack?
  - Affected users: how many people impacted?
  - Discoverability: how likely are attackers to discover the vulnerability?
Pitfalls in risk assessment

- The systematic threat analysis methods help but there is no guarantee of finding all or even the most important threats.
- You need to understand the system: technology, architecture, stakeholders and business model.
- Attackers are clever and invent new threats while threat analysis often enumerates old ones.
- Always start by considering assets and attackers, not technology or security mechanisms.
Security “pixie dust”

- Security mechanism are often applied without particular reason
  - Cryptography, especially encryption does not in itself make your system secure

- If there is no explanation why some security mechanism is used, ask questions:
  - What threats does it protect against?
  - What if we just remove it?
  - Is there something simpler or more suitable?

Must understand threats before applying security mechanisms
SUMMARY
List of key concepts

- Security, threat, attack, vulnerability, exploit, risk, countermeasure
- Confidentiality, integrity, availability
- Asset, attacker, insider
- Threat trees, STRIDE, DREAD
After threat analysis

- After identifying threats, we must assess risks, prioritize the threats and decide countermeasures.
- Threat analysis is an *iterative process*: the analysis must be repeated after designing the system with countermeasures – and over time.
- More detailed threat models can be done for each system component.
- Threat analysis should be done during system design but can also be applied to existing systems.
Reading material

- Dieter Gollmann: Computer Security, 2nd ed., chapter 1.4.3; 3rd ed., chapter 2
- Swiderski and Snyder, Threat modeling, 2004

Online resources:
- MSDN, Uncover Security Design Flaws Using The STRIDE Approach, MSDN Magazine 2016/11 (search for copies)
Exercises

▪ Analyze the threats in the following systems:
  – Oodi student register
  – My Courses
  – Remote read electric meter
  – University card keys
  – Contactless smartcard bus tickets
  – Traffic light priority control for public transportation

▪ What are the assets and potential attackers?
▪ Apply the STRIDE model or threat trees; this will require you to model the system first