Secure Product Lifecyle

Risk Analysis and Threat Modeling

Yagoba GmbH, Austria, www.yagoba.com, info@yagoba.com

Today‘s Agenda

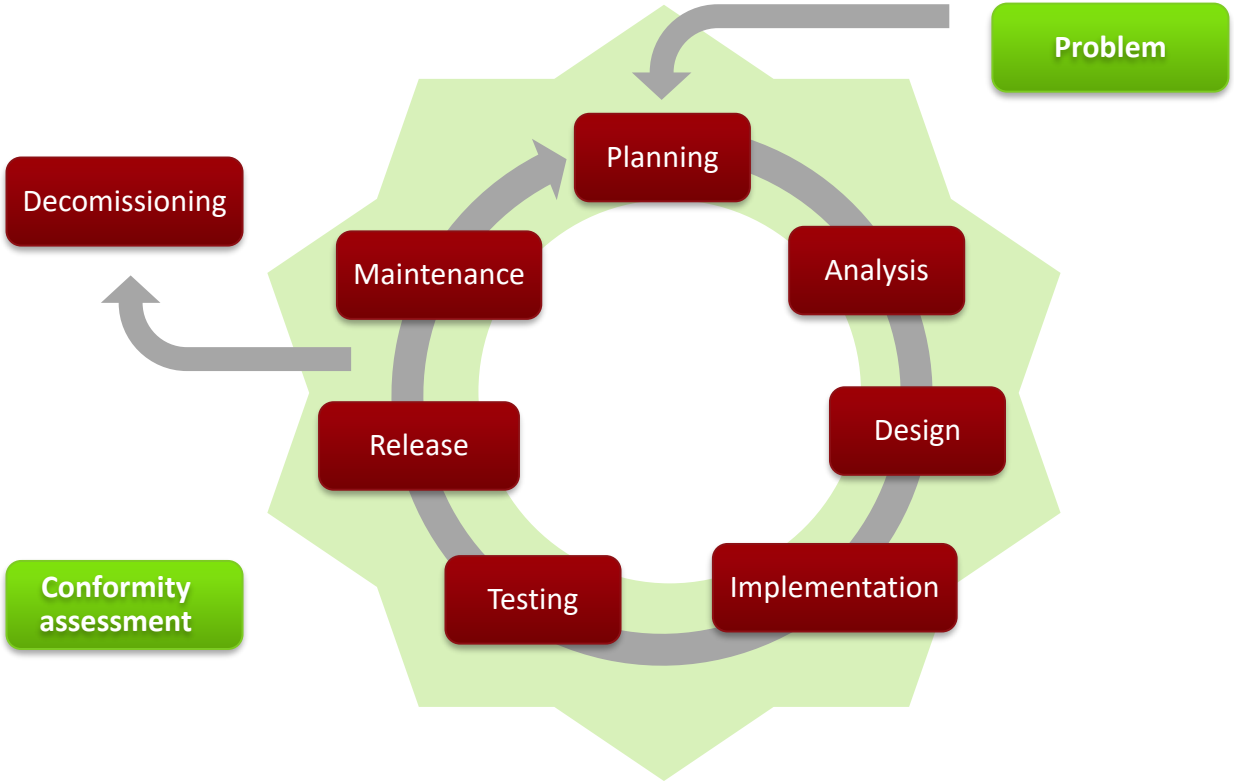
- Context within SDLC - Recap
- Threat modelling
  - Introduction to MS STRIDE
- Risk analysis
  - Thinking about threat consequences
  - Considering impact and likelihood of attacks

Communication

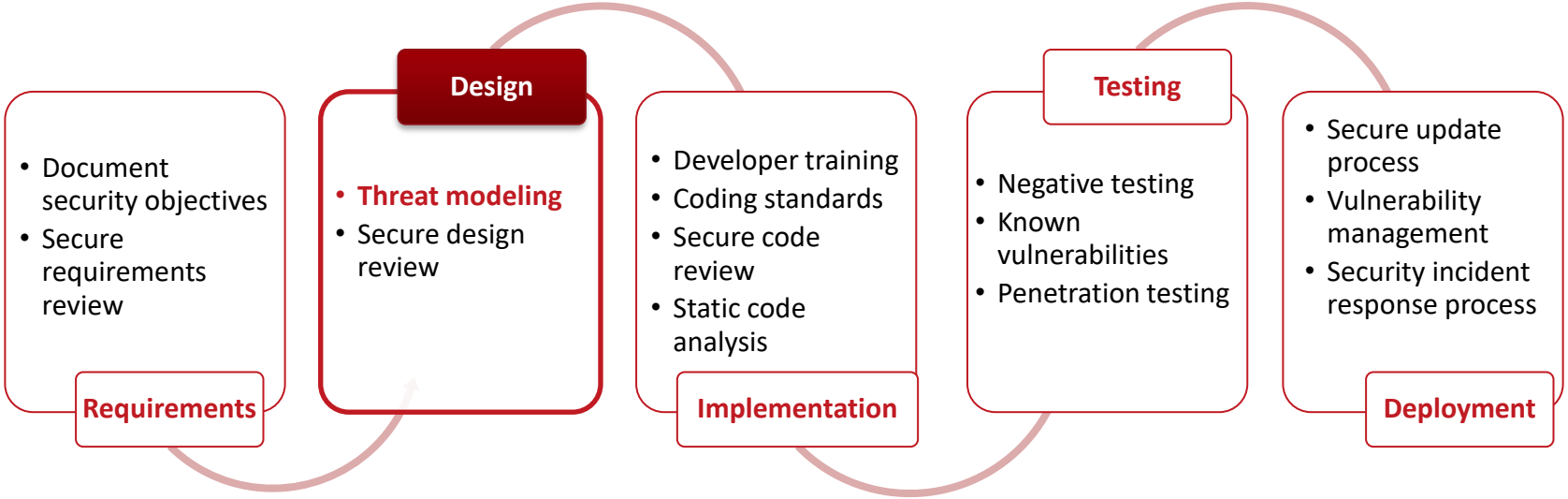
- Christoph Herbst christoph@yagoba.com
- Karin Maier k.maier@yagoba.com

- Discord: #spl

Secure **Product** Lifecycle



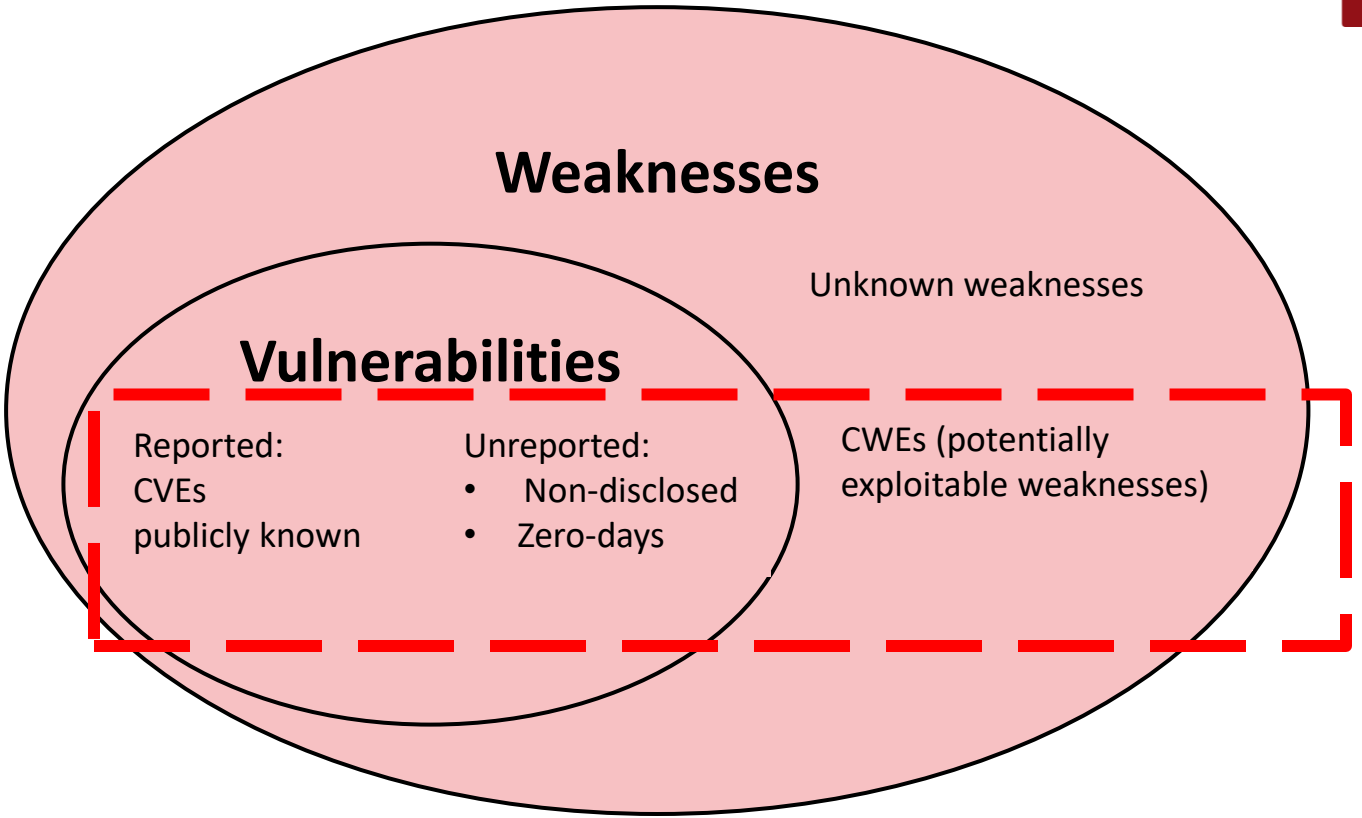
Secure **Development** Lifecycle



Motivation

- Security is not only a matter of technology
- Security needs to be considered over the full product lifecycle
- Relevant security standards like Common Criteria, ISO 62443, etc consider the full lifecycle
- Threat modelling and risk analysis required
  - FDA premarket/postmarket guidance documents
  - UL 2900
  - ISO 21434

Definitions



Definitions

- **Threat**
  - *A potentially successful attack, utilizing specific techniques and resources to take advantage of specific vulnerabilities or lack of risk controls within a product.*
- **Risk**
  - *The potential for harm or damage, measured as the combination of the likelihood of occurrence of that harm or damage and the impact of that harm or damage.*
- **Vulnerability**
  - *A weakness identified in the product for which an exploit does exist, such that it can be directly used by an attacker.*

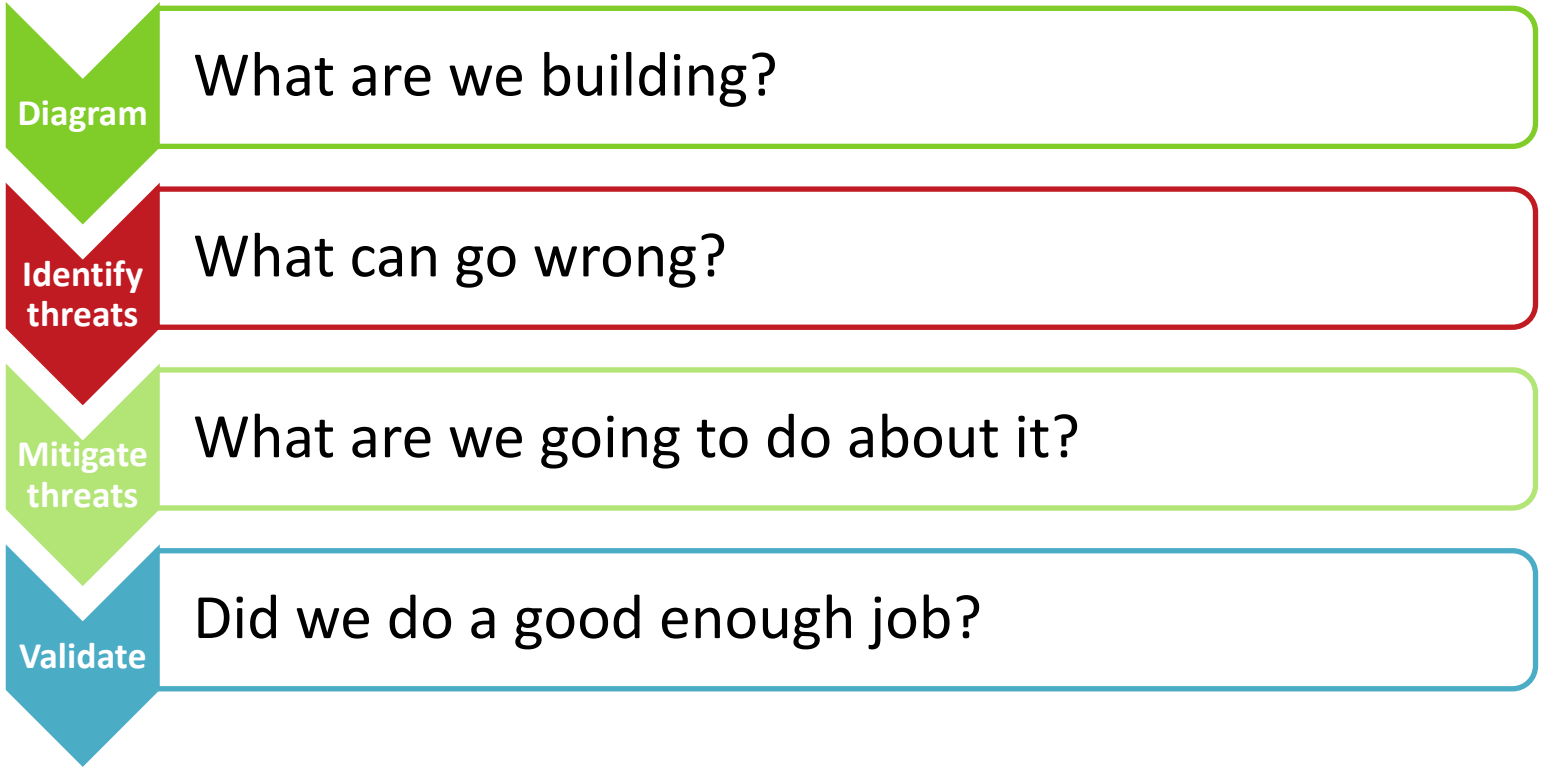
Introduction

- Threat modelling – General aspects
  - Analysis of design to discover threats and to define mitigations
  - Think like an attacker, focus on assets, focus on impact, …
  - When defining what a system must do, also consider what a system must not do
  - When defining use cases, also consider misuse / abuse cases
  - Threat modelling during design phase
    - What are the assets to be protected? (e.g., database)
    - What are the threats? (e.g., malicious user gains credit on account)
    - What are the attacks to realize a threat? (e.g., user sets negative price value)
    - What are conditions to make the attack successful? (e.g., no validation on price value)
- Threat modelling and risk analysis
  - Business risks of successful exploits
  - Costs of liability, redevelopment, and damage to brand image and market share

Introduction

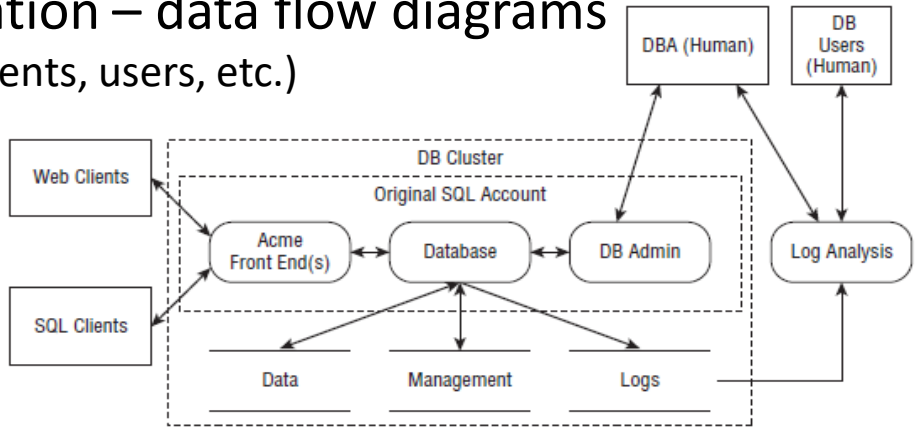
- Why threat modelling?
  - Understand security requirements
  - Identify security issues early in the design
  - → Improve product and software security
  - Serves as input for testing, requirements validation, and identification of risks
  - Align involved team on a shared vision on the security of the product
- Threat modelling answers
  1. What assets need to be protected?
  2. What are the threats to these assets?
  3. How important or likely is each threat?
  4. How can we mitigate the treats?

Process / Framework



Diagram

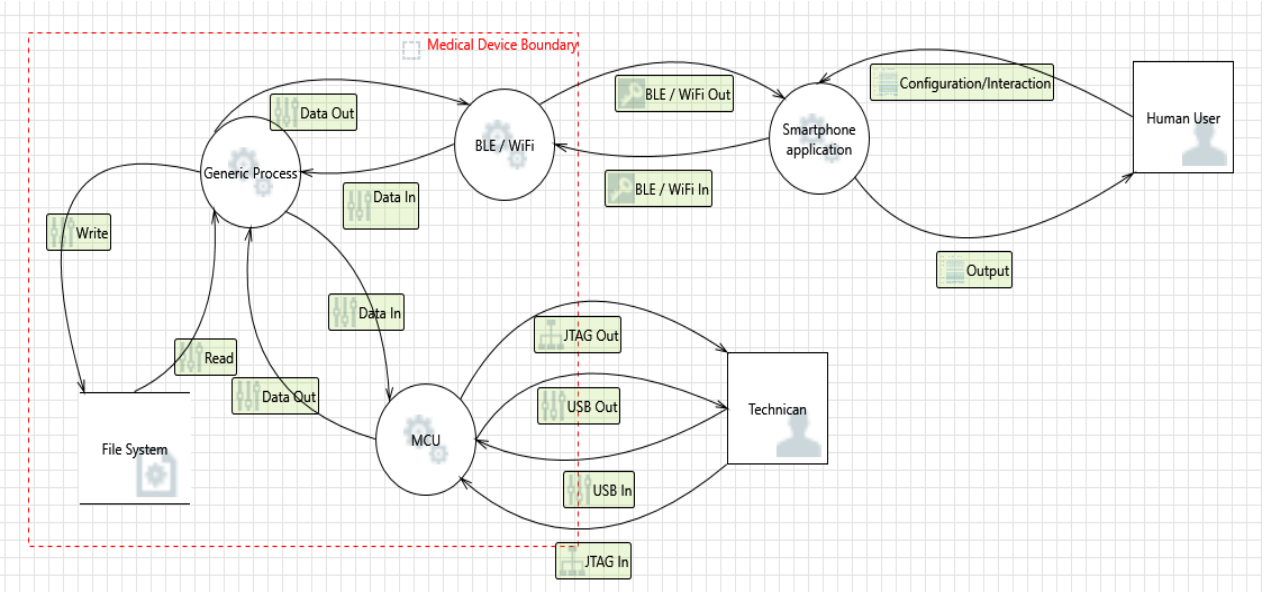
- „What are we building?“ → Data flow diagram to model the system
- Typical information and artifacts
  - Assets (sensitive data, knowledge, etc.)
  - Actors (insiders, outsiders)
- Graphical representation – data flow diagrams
  - External entities (clients, users, etc.)
  - Processes
  - Data stores (DBs)
  - Data flows
  - Trust boundaries



(cf. Shostack, 2014)

Diagram

- DFD from MS Threat Modelling Tool



Diagram

Analyze Potential Threats

- „What can go wrong?“
- Data flow diagram
- Ask questions
  - What security mechanisms are in place to protect assets?
  - Are transitions / interfaces properly secured?
  - Could improper / malicious use of a feature compromise security?
  - Can non-authorized users view / edit confidential data?
- → Use STRIDE for guidance

Popular Threat Models

- Microsoft STRIDE – The most popular threat model
  - Spoofing: Attacker manages to impersonate an entity/process in the system
  - Tampering: Attacker manages to modify data/code
  - Repudiation: Attacker manages to deny having performed an (illegal) action
  - Information disclosure: Attacker manages to get access to information, which should be inaccessible
  - Denial of service: Attacker manages to prevent the service from being available
  - Elevation of privileges: Attacker (or valid user) gains more privileges than anticipated
- Attack trees (either used separately or in combination with STRIDE)
  - Methodical representation of attacks (in a tree structure)
  - Root node: goal
  - Leaf nodes: different ways of achieving that goal
  - After a potential threat has been identified, determine how that threat could manifest itself → threat / attack tree

(Other) Popular Threat Models

- OWASP IoT vulnerabilities projects
  - Information on top IoT vulnerabilities
  - Description of attack surfaces
  - Summary of vulnerabilities
- LINDDUN – Privacy threat analysis methodology
  - Linkability: Being able to determine whether two entities/items are linked
  - Identifiability: Being able to identify a subject
  - Non-repudiation: Not being able to deny a claim
  - Detectability: Being able to determine whether an item of interest exists
  - Disclosure of information
  - Unawareness: Being unaware of consequences of sharing information
  - Non-compliance: Not being compliant with legislation, regulations, or other policies

STRIDE:
Threats and Violated Properties

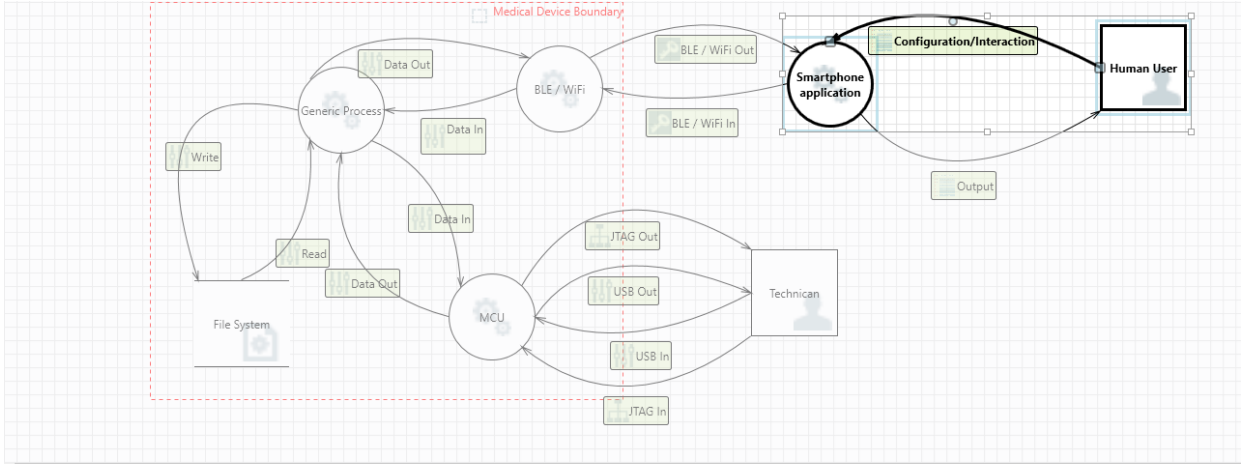
Threat	Violated Property	Description
Spoofing	Authentication	Pretend to be someone else
Tampering	Integrity	Modification of data
Repudiation	Non-repudiation	Deny having performed an action
Information disclosure	Confidentiality	Unauthorized access to data
Denial of Service	Availability	Exhaust resources
Elevation of privileges	Authorization	Performing non-authorized actions

STRIDE per Element

- STRIDE per element: focus on what an attacker might target

	S	T	R	I	D	E
External Entity	X		X			
Process	X	X	X	X	X	X
Data flow		X		X	X	
Data store		X	?	X	X	

MS Threat Modelling Tool



Threat List

ID	Diagram	Changed By	Last Modified	State	Title	Category	Description	Justification	Interaction	Priority
59	Diagram 1		Generated	Not Started	Spoofing of Source Data St	Spoofing	File System may be spoofed by an attacker and this may		Read	High
63	**Diagram 1**		**Generated**	**Not Started**	**Spoofing the Human User**	**Spoofing**	**Human User may be spoofed by an attacker and this n**		**Configuration/I**	**High**
85	Diagram 1		Generated	Not Started	Spoofing the BLE / WiFi Pro	Spoofing	BLE / WiFi may be spoofed by an attacker and this may l		BLE / WiFi Out	High
92	Diagram 1		Generated	Not Started	Spoofing the Smartphone a	Spoofing	Smartphone application may be spoofed by an attacker		BLE / WiFi In	High
9	Diagram 1		Generated	Not Started	External Entity Technican I	Repudiation	Technican claims that it did not receive data from a ny		USB Out	High

Export Csv 61 Threats Displayed, 61 Total

Threat Properties

ID: 63 Diagram: Diagram 1 Status: Not Started

Title: Spoofing the Human User External Entity

Category: Spoofing

Description: Human User may be spoofed by an attacker and this may lead to unauthorized access to Smartphone application. Consider using a standard authentication mechanism to identify the external entity.

Justification:

Interaction: Configuration/Interaction

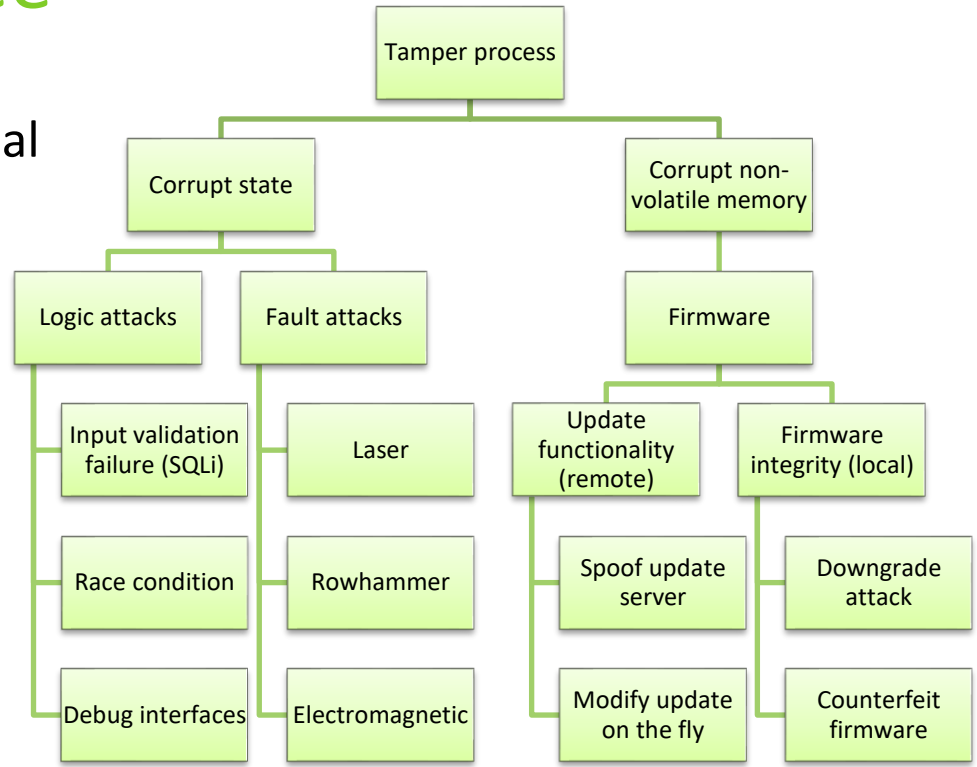
Threat Properties | Notes - no entries

Diagram

Identify threats

Attack Tree

- Root note: goal



Diagram

Identify threats

- Leaf nodes: different ways of achieving that goal (e.g. OWASP Top 10 List)

Addressing Threats

Summary

1. Decompose the product / application using DFDs
2. MS STRIDE (or similar) to identify threats
3. Establish threat trees if appropriate

Overall it does not matter how you identified a threat, but it must be addressed

Respond to threats

- Fix the problem
- Remove the problem – do not include this feature
- Warn the user – allow user to disable feature

STRIDE Examples

Threat/Target	Mitigation Strategy	Mitigation Technique
Spoofing person	Identification/authentication	Username and password, biometrics
Tampering with a NW packet	Cryptography	HTTPS/SSL, IPsec
Repudiation (no logs)	Logging	Log all security-relevant actions
Information disclosure (NW monitoring)	Cryptography	HTTPS/SSL, IPsec
Denial of Service (system resources)	OS	OS quota
Elevation of Privilege (command injection)	Validation	Input validation and sanitization

Addressing Threats: Defensive tactics and technologies

- Spoofing
  - Authenticate a remote service / machine / user
  - Technologies: PKI (SSL, TLS), IPSec, SSH, password, biometric, etc.
- Tampering
  - Protect integrity of files / network traffic
  - Technologies: ACLs, digital signatures, SSL, TLS, IPSec
- Non-repudiation
  - Prevent fraudulent actions, log issues, investigate issues, respond to issues
  - Technologies: Logging, digital signatures
- Confidentiality
  - Protect data at rest and data in transit
  - Technologies: Access Control Lists, Encryption
- Denial of Service
  - Prevent resource exhaustion
  - Technologies: Quotas, load balancing, extra bandwidth
- Elevation of Privilege
  - Prevent unauthorized usage of the system
  - Technologies: Defense in depth, input validation

Validate that Threats are Addressed

- Testing
  - Test implemented mitigations
  - Penetration testing
- QA
  - You are never done with threat modelling, it is a continuous process
  - Does the model match reality?
    - Check the architecture / design against the model
  - Check that identified threats are addressed
    - Traceability matrix (threats ↔ risk control measures)

Threat Assessment & Risk Analysis

- Possible consequences of various threats
  - Damage to brand
  - Financial loss
  - Loss of data
  - Loss of control
  - Compromise of privacy
  - Loss of property
  - Loss of life
  - Environmental damage
  - Service disruption
- Risk analysis / assessment
  - Determine whether identified threat can be neglected or
  - What mitigation technique should be applied

https://pixabay.com/photos/risk-word-letters-boggle-game-1945683/

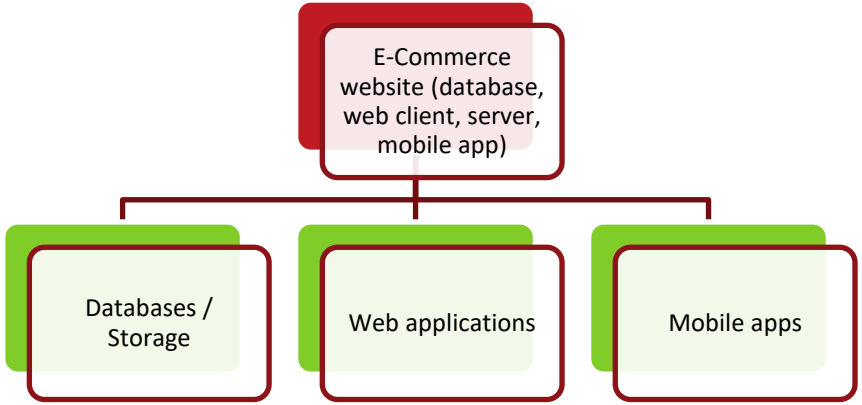
General Scoring Systems

- CWSS (Common Weakness Scoring System)
  - Considers classes of weaknesses (e.g., buffer overflows, OWASP Top 10)
  - Score weaknesses in order to prioritize them
  - E.g., buffer overflows have higher priority than memory leaks
- CVSS (Common Vulnerability Scoring System)
  - Considers already discovered and verified vulnerabilities
  - Score vulnerabilities (reflecting its severity) in order to prioritize them

General Scoring Systems

- CWRAF (Common Weakness Risk Analysis Framework)
  - Rank classes of weaknesses for prioritization
  - Customize CWE rankings based on business domains / vignettes

- Select CWEs related to
  - Databases (SQLi)
  - Web (XSS)
  - Mobile (cleartext storage)
  - Programming language (Java)



General Scoring Systems

DREAD

- Damage: Assess the damage that could result from an attack (e.g., data loss, HW failure, reduced performance, etc.)
- Reproducibility: How easy can the attack be reproduced?
- Exploitability: Effort and expertise required to mount attack
- Affected Users: Number of affected users (1 – 2 vs. 10,000)
- Discoverability: Likelihood that a threat will be exploited

- Score each item (above) with value 1 – 10
- Total score: average of all scores

Risk Assessment

- Bottom up

Risk Assessment

- Identified risks need to be „measured“
  - In general: risk ≈ impact x likelihood
- Impact
  - Consider the assets of the system and what would happen if
    - Confidentiality gets compromised
    - Integrity gets compromised
    - Availability gets compromised
- Likelihood – consider
  - Access to the system (local vs remote)
  - Skills, expertise, and knowledge
  - Motivation (fame, financial gain, IP, anger)
  - Budget

Risk Analysis

- Risk assessment
  - What is considered acceptable (green)?
  - What is considered conditionally acceptable (yellow)?
  - What is considered unacceptable (red)?

		Impact				
		Very low	Low	Moderate	High	Very high
Likelihood	Very low					
	Low					
	Moderate					
	High					
	Very high					

Risk Analysis

Threat / vulnerability	Likelihood	Impact	Risk	Control measure
Leakage of sensitive data / Lack of encrypted communication in transit	High: network traffic can be easily captured	Moderate: disclosure of sensitive information	Unacceptable	Encryption, protect network access, …
…				
…				
…				

Risk Controls

1. By design (prevent access, defense-in-depth)
2. Protective measures (physical proximity, NW restrictions)
3. Documentation

Residual risk evaluation

- Re-evaluate residual risk after applying security risk controls

General Design Principles

Non-exhaustive list

- Secure communication
  - Encryption and integrity protection
  - Input validation
- Data protection
- User authentication
  - No hardcoded credentials
- Device integrity
  - Logging
  - Anti-malware detection
- Software maintenance / update
- Physical access

Practical Considerations

- In many sectors safety & security must be considered
  - Medical
  - Automotive
- Risk-based approach to the design of devices must be followed
- Security risk controls could have a negative impact on safety
  - Risk control for authentication → in-accessible device in emergency situations

Summary

- System model like DFD
- Identify threats (e.g. STRIDE)
- Assess Risk
  - Impact
  - Likelihood
- Address selected risks with mitigation actions
- Derive your requirements (next lecture)

References

- https://github.com/Toreon/threat-model-playbook
- https://docs.microsoft.com/en-us/windows-hardware/drivers/driversecurity/threat-modeling-for-drivers
- Adam Shostack: Threat Modeling – Designing for Security. Wiley & Sons 2014.
- NIST. NIST Special Publication 800-30. Guide for Conducting Risk Assessments.