VULNERABILITY ASSESSMENT

CS-5156/CS-6056: SECURITY VULNERABILITY ASSESSMENT (SPRING 2025)
LECTURE 3

What is Vulnerability Assessment?

- Process of defining, **identifying**, classifying and prioritizing vulnerabilities in computer systems, applications and network infrastructures

  and

  providing the organization doing the assessment with the necessary knowledge, awareness and risk background to understand the threats to its environment and react appropriately.

  Ref: https://searchsecurity.techtarget.com/definition/vulnerability-assessment-vulnerability-analysis

What is Vulnerability Management?

- Assessment, **classification**, and **mitigation** of vulnerabilities

Why is Vulnerability Assessment Important?

- To be able to respond quickly to mitigate ***potential*** threats – that is, before becoming a victim.

Why Perform Vulnerability Assessment?

Risk analysis could be:

Required by law or legislation

Sarbanes Oxley Audit Standard 5

Federal Information Security Management Act (FISMA)

Recommended to abide by a *standard* – to help minimize business disruption and continue operation in the event of an incident

ISO / IEC standards (**best practice frameworks**)

Useful to protect your organization

From lawsuits

From cybersecurity risks

&& to help you decide where best to invest your security budget.

Why Perform Vulnerability Assessment?

- Vulnerability assessment/management plays a crucial role to an organization in the areas of
  - Financial stability
  - Continued performance
  - Longevity of organization

Terms

- Vulnerability
  - A flaw or weakness in a system's design, implementation, operation or management that could be exploited to compromise the system's security objectives
- Threat
  - Anything that **may** **harm** the assets owned by an application (resources of value, such as the data in a database or in the file system) **by exploiting a vulnerability**
- Risk
  - **Potential for loss** due to some vulnerability being exploited by some threat
  - The combination of the probability of an event and its consequence (ISO Guide 73)
- Test
  - An action to demonstrate that an application meets the security requirements of its stakeholders

Vulnerability/Threat/Risk Example (Seat Belt)

- Vulnerability:
  - Don’t stick to seat well
- Threat:
  - Sudden unexpected stop from accident or failure
- Risk:
  - Bodily harm or death

Vulnerability Examples

- Software and Operating Systems
  - Buffer overflow
    Stack overflow
    Heap overflow
    Return oriented programming
- Languages
  - Pointer management
    String format problems
- Web and Network
  - SQL injection
    Cross Site Scripting
    Broken authentication and session management
    Broken configuration management
    Broken authentication protocols

Threat Examples

- Denial of Service and Distributed Denial of Service
- Social Engineering
  - Phishing – benign appearance, weaponized attachment
    Scareware – you will be arrested unless you call back
    Pretexting – attacker claims to be from Microsoft
- Malware
  - Worm – stand-alone, self replicating, spreads
    Virus – triggered by activation of host (email attachment)
    Rootkit – for remote control by attackers
    Trojan – malware hidden in benign application
    Spyware – steal sensitive data
- Physical
  - Nature – hurricane, tornado, tsunami
    Insider – access to servers

Risk Examples

- Artificial Intelligence
  - Existential – potential for unpredictable malware devel.
    Privacy – monitoring comms for attitudes, opinions
    QoL – e.g. use of robots to replace human services
    Single Point of Failure – e.g. self driving car accidents (death)
  - Weaponization – e.g. drone swarms to harm humans
- Legal Risk
  - Regulatory – changes that affect compliance costs
    Compliance – potential for fines from government
    Contract – damage from failure to meet contract
    Reputation – damage due to loss of reputation
- Other
  - Stolen intellectual property
    Stolen financial assets

What to do with Risk?

- Avoid
- Mitigate
  - Apply some control(s) to reduce impact
- Transfer
  - Contractually engage with a third party who takes on the risk for some money (insurance)
- Accept
  - Do NOTHING and hope it doesn’t happen to you…

The Bottom Line

Assessment is the Practice of

Discovering vulnerabilities posed by an environment
Determining their **negative risk impact**
Documenting these observations for future planning.

This may drive

Modifications to a network or business practice to **eliminate** the vulnerability and reduce its exposure

Implementation of **monitoring** to notify in the event that an identified vulnerability is being exploited in the environment

The Bottom Line

Risk = Likelihood * Impact

What is a risk management framework?

- A ***set of guidelines*** that organizations follow in an effort to **protect themselves from risk** (to help them *manage the risk* to their organization).

Risk Management Framework Components

1. **Identification** of risk

2. **Measurement and assessment** (MUST be measurable!)

3. **Mitigation** and/or **control** (remember: can't *always* eliminate risk)
   - e.g., antivirus, firewall, etc.

4. **Reporting and monitoring** (provides accountability)
   - e.g., constant monitoring of the effectiveness of current security controls

5. **Governance** (ensure that all employees perform duties within the framework)
   - e.g., policy

Security Services To Consider - Triple A

- **Authentication**
  - **Who are you?**
  - **"I am user student and my password validateme proves it."**
- **Authorization**
  - **What can you do? What can you access?**
  - **"User student can access host serverXYZ using Telnet."**
- **Accounting**
  - **What did you do? How long did you do it? How often did you do it?**
  - **"User student accessed host serverXYZ using Telnet for 15 minutes."**

Security Services To Consider - CIA Triad

Example Risks / Controls

CIA	Risks	Controls
Confidentiality	Loss of privacy. Unauthorized access to information. Identity Theft.	Encryption, Authentication Access controls
Integrity	Information is no longer reliable or accurate. Fraud.	Maker/Checker, Quality Assurance, Audit Logs.
Availability	Business disruption. Loss of customer confidence. Loss of revenue.	Business Continuity Plans and tests. Backup storage. Sufficient capacity.

Problem

- Many organizations do **NOT YET USE** security frameworks to help them understand and manage their risks

Solution

- There are MANY frameworks that can be used to simplify analysis
- Some specifically noted as "**risk management frameworks**"
- Others noted as "**security frameworks**" that **include risk management concepts**.
- Frameworks provide a **set of potential recommendations** that may or may not apply to your organization.
- Examples
  - OWASP (Open Web Application Security Project)
  - Microsoft DREAD

OWASP

- Non-profit, charitable organization (good guys) - https://owasp.org
- Aim: improve security of software by making it visible so developers and organizations can make informed decisions about security risks.

Products of OWASP

Application security code of conduct for educational institutions

https://www.owasp.org/images/6/6b/OWASP_Blue_Book-Educational_Institutions.pdf

OWASP Testing Guide

https://owasp.org/www-project-web-security-testing-guide/v42/

Risk Rating Methodology

https://owasp.org/www-community/OWASP_Risk_Rating_Methodology

OWASP Code of Conduct for Educational Institutions

The educational institution must include application security content somewhere in the standard computer science curriculum

The educational institution must offer at least one course dedicated to application security annually

The educational institution must ensure that an OWASP chapter is available to their students and support it

OWASP Testing Guide (Web Application Security Testing)

Information gathering
Configuration and deployment management testing
Identity management testing
Authentication testing
Authorization testing
Session management testing
Input validation testing

Testing for error handling
Testing for weak cryptography
Business logic testing
Client side testing

https://wiki.owasp.org/index.php/Testing_Checklist

https://owasp.org/search/?searchString=testing+checklist

https://owasp.org/www-project-web-security-testing-guide/v42/

OWASP Testing Guide (Tools)

Resource:

https://owasp.org/www-project-web-security-testing-guide/stable/6-Appendix/A-Testing_Tools_Resource

Testing Checklist:

https://owasp.org/search/?searchString=testing+checklist

NOTE: Good as Presentation Topics

OWASP Testing Guide (Tools & Definitions)

Document Object Module (DOM) XSS - https://owasp.org/www-community/attacks/DOM_Based_XSS

AJAX - https://www.w3schools.com/xml/ajax_intro.asp

SQL Injection - https://owasp.org/www-community/attacks/SQL_Injection

Oracle - https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/07-Input_Validation_Testing/05.1-Testing_for_Oracle

SSL - https://owasp.org/www-project-web-security-testing-guide/v41/4-Web_Application_Security_Testing/09-Testing_for_Weak_Cryptography/01-Testing_for_Weak_SSL_TLS_Ciphers_Insufficient_Transport_Layer_Protection

Brute Force Password - https://owasp.org/www-community/attacks/Brute_force_attack

Buffer Overflow - https://owasp.org/www-community/attacks/Buffer_overflow_attack

Fuzzer - https://www.owasp.org/index.php/Fuzzing

Googling - https://owasp.org/www-chapter-ghana/assets/slides/owasp-presentation-googling.pdf

Source code analyzers - https://www.owasp.org/index.php/Source_Code_Analysis_Tools

Acceptance Testing - https://owasp.org/www-project-web-security-testing-guide/v41/6-Appendix/A-Testing_Tools_Resource

Runtime Analysis

OWASP Risk Rating Methodology

Identify a risk in the system

A risk = threat agent + attack + vulnerability + impact

Estimate **likelihood** the risk is realized via attack

Threat agent factors – capability of attackers
Vulnerability factors – chance it will be discovered & used

Estimate **impact** – loss of value to the 'customer'

Light to severe – coarse grained

Determine severity – effort needed to mitigate and fix problem

Low, medium, high

Decide what to fix

Customize the Risk Rating Model – change cutoff points

Identify a risk in the system

Gather **information** about the/all **threat agent(s)**

threat **agent** = capabilities + intentions + past activities

Types of threat agents to consider:

non-target specific: computer viruses, worms, trojans
employees: staff, contractors, maintenance security
organized crime and criminals: bank accounts, IP, CCs
corporations: competitive intelligence

human, unintentional: accidents, carelessness.
human, Intentional: insider (dissatisfied employees, suppliers, contractors, etc.), outsider (competition, hactivists, script kiddie, etc.).
natural: flood, fire, lightning, meteor, earthquakes.

Develop a **template** for each **threat agent**

e.g., https://www.owasp.org/index.php/Logic/time_bomb

Identify a risk in the system

Gather **information** about the **attack**

Develop a **template** for each **attack**

e.g.,
https://owasp.org/www-community/attacks/csrf

Excellent example presentation for this class
https://repository.root-me.org/Exploitation%20-%20Web/EN%20-%20OWASP%20Cross-site%20Request%20Forgery%20CSRF.pdf

Identify a risk in the system

Gather **information** about the **vulnerability**

Develop a **template** for each **vulnerability**

e.g., Buffer Overflow

Description:

consequences: ...
required resources: any
severity: very high
likelihood of exploit: high to very high

Risk Factors:

Language design

Examples:
Related Attacks: format string attack
Related Vulnerabilities: heap buffer overflow
Related Controls: use immune language (haskell?)
Related Technical Impacts: …..

Identify a risk in the system

Gather **information** about the **impact**

Develop a **template** for each **impact**

e.g., Availability

Description:

severity: highest

Risk Factors:

power failure

Denial of Service attack

Examples:

Related Impacts: integrity (*e.g.* service)

Related Controls: uninterruptible power supply

Estimate likelihood the risk is realized (via attack)

Threat agent (TA) factors

Skill level: penetration, network & programming, advanced
Motive: low reward? High reward?
Opportunity: resources needed for TA to exploit vuln
Size: government? Mob? Loner?

Vulnerability factors

What is the likelihood vuln is discovered and exploited?
Ease of discovery: easy, difficult, practically impossible...
Ease of exploit: easy, difficult, theoretical ...
Awareness: how well known is vuln to threat agents?
Detection: how likely is it that attack can be detected?

Estimate impact

Technical Impact Factors

Loss of confidentiality: how sensitive is the data?
Loss of integrity: how much of the data can be damaged?
Loss of Availability: how vital is this service?
Loss of accountability: are threat agents' actions traceable to an individual

Business Impact Factors

Some companies have an asset classification guide and business impact reference to formalize the potential loss

Financial damage: less than cost to fix the vulnerability?

Reputation damage: loss of major accounts?

Non-compliance: how much exposure does

non-compliance introduce? high-profile violation?

Privacy violation: how much personal info can be exposed?

Determine severity of the risk

- Likelihood and impact estimates are put together
- Use a scale from 0-9 for both
- Example for **likelihood**:

Threat Agent Factors			
Skill level	Motive	Opportunity	Size
5	2	7	1
Vulnerability Factors			
Ease of Discovery	Ease of Exploit	Awareness	Detection
3	6	9	2
Overall likelihood: 4.375 (average of all the numbers)			

Determine severity of the risk

- Likelihood and impact estimates are put together
- Use a scale from 0-9 for both
- Example for **impact**:

Technical Impact Factors (loss of)			
Confidentiality	Integrity	Availability	Accountability
9	7	5	8
Overall technical impact: 7.25 (average of all the numbers)			
Business Impact Factors			
Financial Damage	Reputation Damage	Non-compliance	Privacy Violation
1	2	1	5
Overall business impact: 2.25 (average of all the numbers)			

Determine severity of the risk

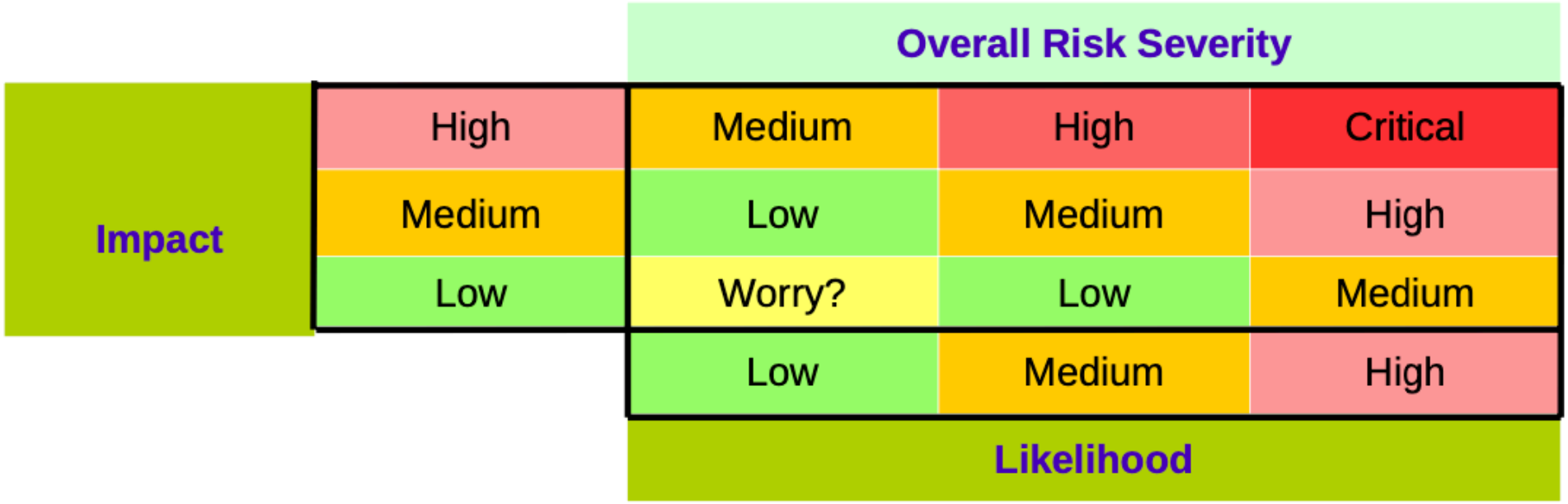
- Likelihood and impact estimates are put together
- Let **0-2** be **'Low'**, **3-5** be **'Medium'**, **6-9** be **'High'**
- Calculation of Risk Severity:

		Overall Risk Severity		
Impact	High	Medium	High	Critical
	Medium	Low	Medium	High
	Low	Worry?	Low	Medium
		Low	Medium	High
		Likelihood		

- Worry? Means maybe it is not worth taking care of the risk
- Compare to https://www.cvedetails.com/

Decide what to fix

- Prioritize risks based on risk analysis
- Assign $ costs to repair and loss for each risk
- Find a point below which there is no net gain to fixing the vulnerabilities – consider fixing those above
- Or find the highest point in the list above which the cost to fix vulnerabilities is greater than the funds available to do so – consider fixing those one above that point



Customize the Risk Rating Model

The model must be tailored to the organization

Add factors:
e.g. military org may add a casualty impact factor
encryption algorithm strength as a likelihood factor

Customize options:
e.g. financial damage may be different for different departments, so one can assign different ratings accordingly

Weight the factors:
In the above, all factors have the same weight

Microsoft Risk Rating Methodology (**DREAD**)

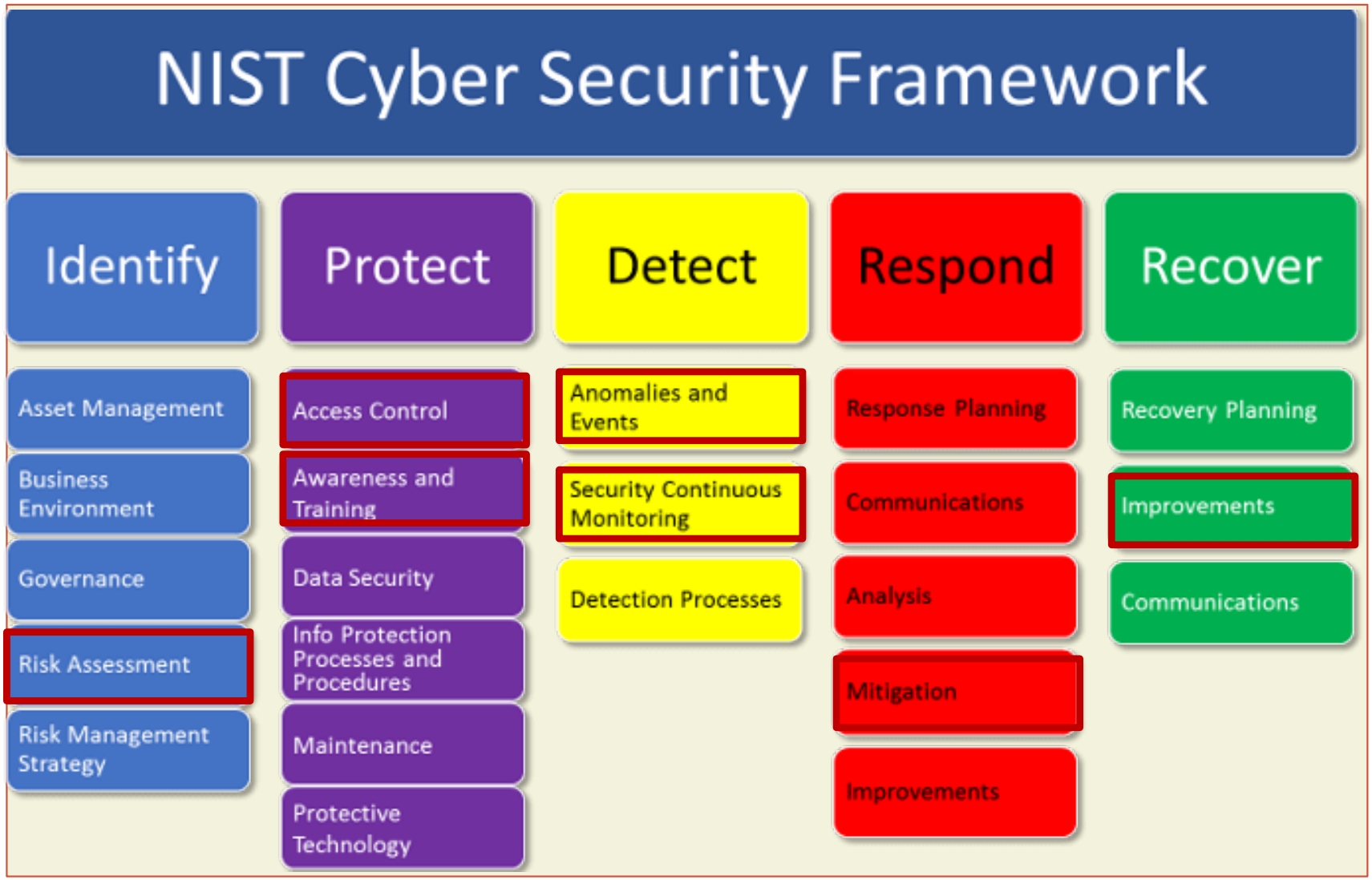
- Classification scheme for quantifying, comparing and prioritizing the amount of risk presented by each evaluated threat
  - Risk = (damage+reproducibility+exploitability+affected-users+ Discoverability)/5
- **D**amage Potential: 0-10 (0=nothing, 10=complete destruction)
- **R**eproduce Threat: 0-10 (0=can't, 10=easily)
- **E**xploitability: 0-10 (0=difficult, 10=easy)
- **A**ffected Users: 0-10 (0=none, 10=everyone)
- **D**iscoverability: 0-10 (0=impossible, 10=right there in browser)

- Ref: https://learn.microsoft.com/en-us/previous-versions/msp-n-p/ff648644(v=pandp.10)?redirectedfrom=MSDN#dread

Other Risk Assessment Related Frameworks

- NIST CSF (Cyber Security Framework)
  - https://csrc.nist.gov/projects/risk-management
  - https://nvlpubs.nist.gov/nistpubs/ir/2020/NIST.IR.8286.pdf
- NIST 800-171 CUI (Controlled Unclassified Information)

NIST CSF


NIST Cyber Security Framework
Identify
Protect
Detect
Respond
Recover
Asset Management
Business Environment
Governance
Risk Assessment
Risk Management Strategy
Access Control
Awareness and Training
Data Security
Info Protection Processes and Procedures
Maintenance
Protective Technology
Anomalies and Events
Security Continuous Monitoring
Detection Processes
Response Planning
Communications
Analysis
Mitigation
Improvements
Recovery Planning
Improvements
Communications

Threat Modeling

- Modeling all kinds of possible threats (actual attacks as well as natural disasters)
- Impossible to defend against ALL threats
- Questions
  - Who are the attackers?
  - How attackers gain their knowledge?
  - How attackers conduct their activities?
  - Why? (possible motivations)
  - What are the common characteristics of an attack?
  - When and how to recognize an attack?

Threat Modeling

- **Threat Model** is a formal framework for grouping threats into discrete categories (classes) in order to plan for security controls
- I.e., a classification scheme
- Examples
  - **Microsoft** STRIDE (older framework)
    - https://learn.microsoft.com/en-us/azure/security/develop/threat-modeling-tool-threats#stride-model
  - OWASP Threat Model
    - https://owasp.org/www-community/Threat_Modeling

Microsoft Threat Modeling Scheme (STRIDE)

Category	Description	Controls
Spoofing	Pretending to be someone else	Authentication
Tampering	Changing data to be inaccurate	Integrity controls
Repudiation	Denying that you did a thing you did	Accountability controls
Information Disclosure	Revealing information that should not be otherwise accessible	Confidentiality controls
Denial of Service	Trying to stop someone else from using a system	Availability controls
Elevation of Privilege	Trying to get a higher level of access than you are currently assigned	Authorization controls

Threat Modeling Goal

- Mapping an Event (e.g., data breach, etc.) to a Formal Threat Model

Example Data Breath (Target)

- Attackers gained access using stolen network credentials from third party HVAC contractor
- Attackers uploaded real-time card stealing malware to POSs
- Users/customers affected: 70 million
- https://krebsonsecurity.com/2014/02/target-hackers-broke-in-via-hvac-company/

Vulnerability Classification

- Many classification schemes exist
- Examples
  - MITRE CWE (**C**ommon **W**eakness **E**numeration)
    - https://cwe.mitre.org/about/sources.html
      - Software Security
        - https://cwe.mitre.org/documents/sources/SevenPerniciousKingdomsTaxonomyGraphic.pdf
      - Web Application Security
        - https://cwe.mitre.org/documents/sources/WASCThreatClassificationTaxonomyGraphic.pdf
  - OWASP Top 10
    - Top 10 Privacy Risks
      - https://owasp.org/www-project-top-10-privacy-risks/
    - Top 10 Web Application Security Risks
      - https://owasp.org/www-project-top-ten/
      - List of Mapped CWEs
        - E.g., Injection: https://owasp.org/Top10/A03_2021-Injection/#list-of-mapped-cwes
  - SANS Top 20
    - Top 20 Software
      - https://www.softwaretestinghelp.com/sans-top-20-security-vulnerabilities/