Cyber Threat Intelligence and Threat Hunting in the Energy Domain

CSP006_S_E

PRESENTATION BY:
DR. STEFAN SCHAUER
DR. ABDELKADER SHAABAN
AIT AUSTRIAN INSTITUTE OF TECHNOLOGY

Acknowledgement

- *Co-Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or HADEA. Neither the European Union nor the granting authority can be held responsible for them.*
- *Project Agreement no. 101083594*

Cyber Threat Intelligence and Threat Hunting in the Energy Domain

Overview

- Topic-1: Introduction to Threat Intelligence and Threat Hunting
- Topic-2: Data Sources and Collection
- Topic-3: Threat Actors and Tactics
- Topic-4: Practical Threat Modelling and Security Investigation

Agenda

01. What is Threat Modeling

02. Threat Modeling Methods

03. Threat Analysis

04. ThreatGet

05. Practical: ThreatGet

Objectives

At the completion of this module, participants should be able to:

1. Overview of different methods of threat modelling.
2. Appreciate that thinking like an attacker can improve your cybersecurity defences.
3. Explain the ‘kill chain’ and how to use it to anticipate attacks and defend a network
4. Describe threats according to tactics, techniques, and procedures that are used to attack
5. Identify how threat characteristics can inform cyber security planning.

What is Threat Modeling

Threat Modeling

- Threat modeling is a building block in security engineering that identifies potential threats in order to define corresponding mitigation in different industrial domains.
- With threat modelling, a representation of security is applied to a representation of a system to identify potential security issues
  - Threat-modeling methods are used to create
    - an abstraction of the system
    - profiles of potential attackers, including their goals and methods
    - a catalog of potential threats that may arise

Threat Modeling Methods

Overview

- STRIDE
- The Process for Attack Simulation and Threat Analysis (PASTA)
- LINDDUN (linkability, identifiability, nonrepudiation, detectability, disclosure of information, unawareness, noncompliance)
- Common Vulnerability Scoring System (CVSS)
- Attack Trees
- Persona Non Grata (PnG)
- NIST Special Publication 800-154
- Security Cards
- Hybrid Threat Modeling Method (hTMM)
- Quantitative Threat Modeling Method (Quantitative TMM)
- Trike
- Visual, Agile, and Simple Threat (VAST) Modeling
- Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE´)

https://insights.sei.cmu.edu/blog/threat-modeling-12-available-methods/

Overview

- Invented in 1999 and adopted by Microsoft in 2002
- most mature threat-modeling method
- threat-specific tables
- variants STRIDE-per-Element and STRIDE-per-Interaction.

	Threat	Property Violated	Threat Definition
S	Spoofing identify	Authentication	Pretending to be something or someone other than yourself
T	Tampering with data	Integrity	Modifying something on disk, network, memory, or elsewhere
R	Repudiation	Non-repudiation	Claiming that you didn't do something or were not responsible; can be honest or false
I	Information disclosure	Confidentiality	Providing information to someone not authorized to access it
D	Denial of service	Availability	Exhausting resources needed to provide service
E	Elevation of privilege	Authorization	Allowing someone to do something they are not authorized to do

https://insights.sei.cmu.edu/blog/threat-modeling-12-available-methods/

STRIDE

- Invented in 1999 and adopted by Microsoft in 2002
- most mature threat-modeling method
- threat-specific tables
  - variants **STRIDE-per-Element** and STRIDE-per-Interaction.

	S	T	R	I	D	E
External Entity	✓		✓			
Process	✓	✓	✓	✓	✓	✓
Data Flow		✓		✓	✓	
Data Store		✓	?	✓	✓	

https://insights.sei.cmu.edu/blog/threat-modeling-12-available-methods/

STRIDE

- Invented in 1999 and adopted by Microsoft in 2002
- most mature threat-modeling method
- threat-specific tables
  - variants STRIDE-per-Element and **STRIDE-per-Interaction**.

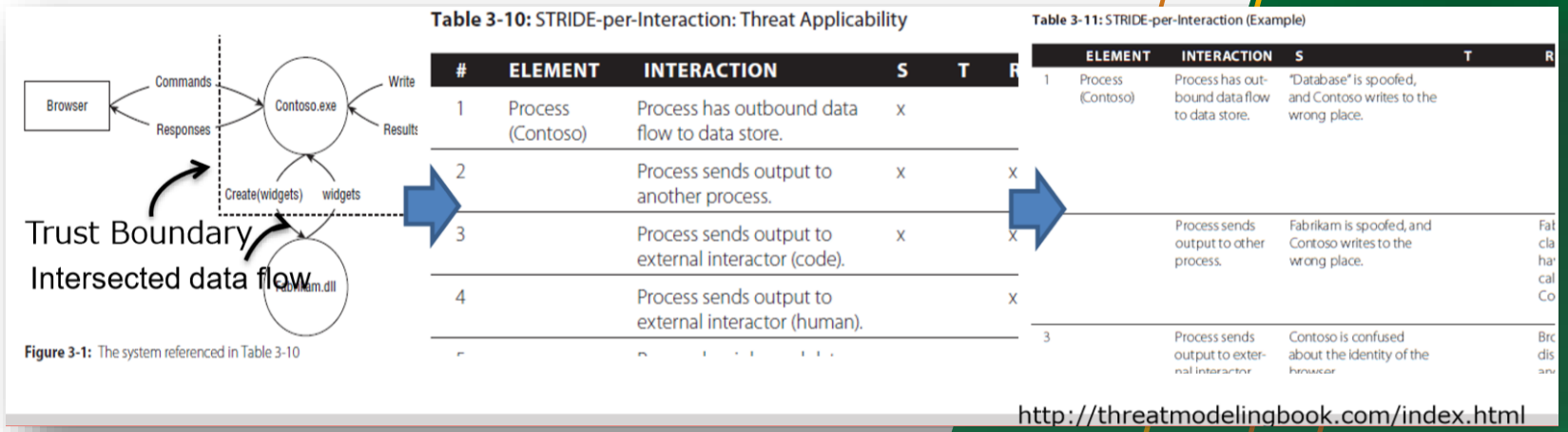


Figure 3-1: The system referenced in Table 3-10

Table 3-10: STRIDE-per-Interaction: Threat Applicability

#	ELEMENT	INTERACTION	S	T	R
1	Process (Contoso)	Process has outbound data flow to data store.	x		
2		Process sends output to another process.	x		x
3		Process sends output to external interactor (code).	x		x
4		Process sends output to external interactor (human).			x

Table 3-11: STRIDE-per-Interaction (Example)

	ELEMENT	INTERACTION	S	T	R
1	Process (Contoso)	Process has out-bound data flow to data store.	"Database" is spoofed, and Contoso writes to the wrong place.		
		Process sends output to other process.	Fabrikam is spoofed, and Contoso writes to the wrong place.		Fat cla ha cal Co
3		Process sends output to exter-	Contoso is confused about the identity of the		Bro dis

http://threatmodelingbook.com/index.html

https://insights.sei.cmu.edu/blog/threat-modeling-12-available-methods/

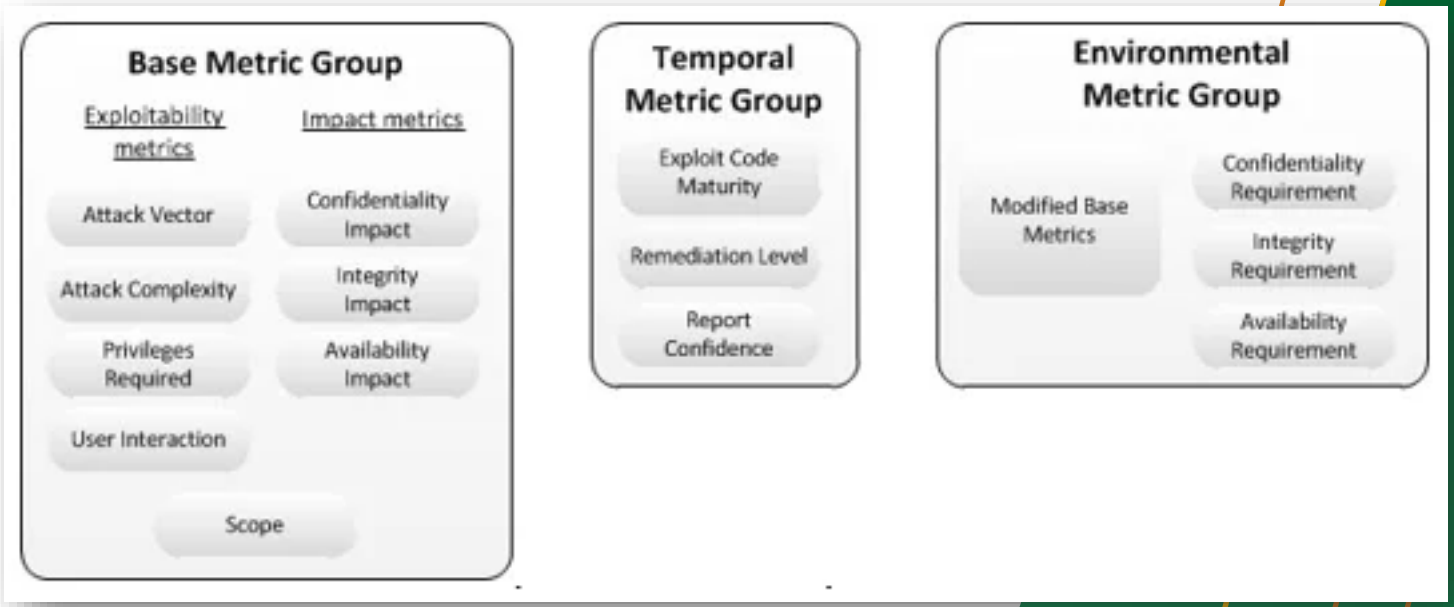
Process for Attack Simulation and Threat Analysis (PASTA)

- Process for Attack Simulation and Threat Analysis (PASTA)
- Risk-centric threat-modeling framework
- Combine business objectives and technical requirements
- Requiring security input from operations, governance, architecture, and development.

https://insights.sei.cmu.edu/blog/threat-modeling-12-available-methods/

The Common Vulnerability Scoring System (CVSS)

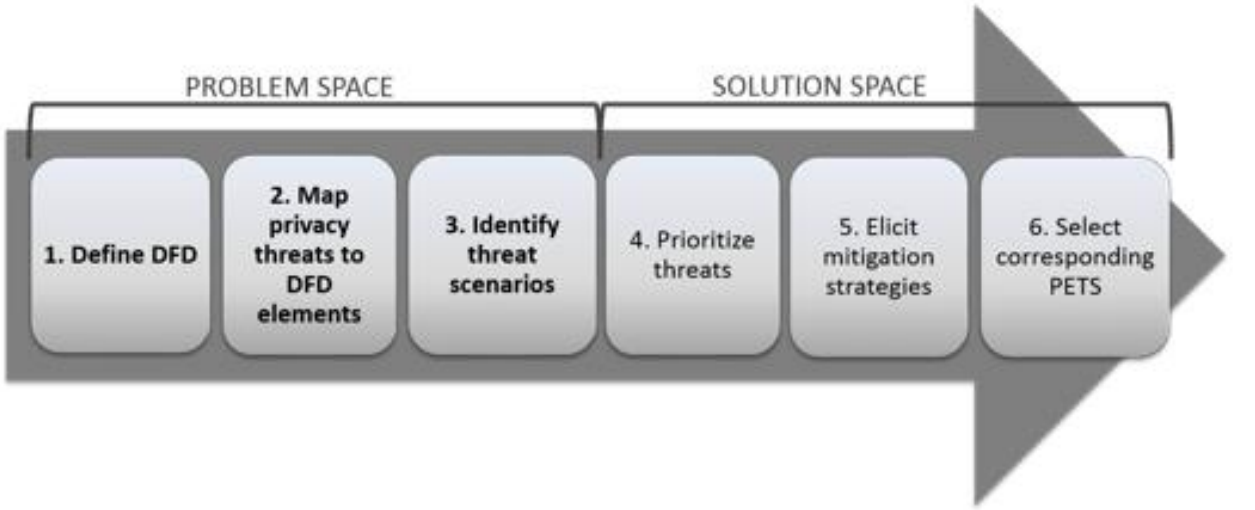
- captures characteristics of a vulnerability
- produces a numerical severity score



https://insights.sei.cmu.edu/blog/threat-modeling-12-available-methods/

LINDDUN

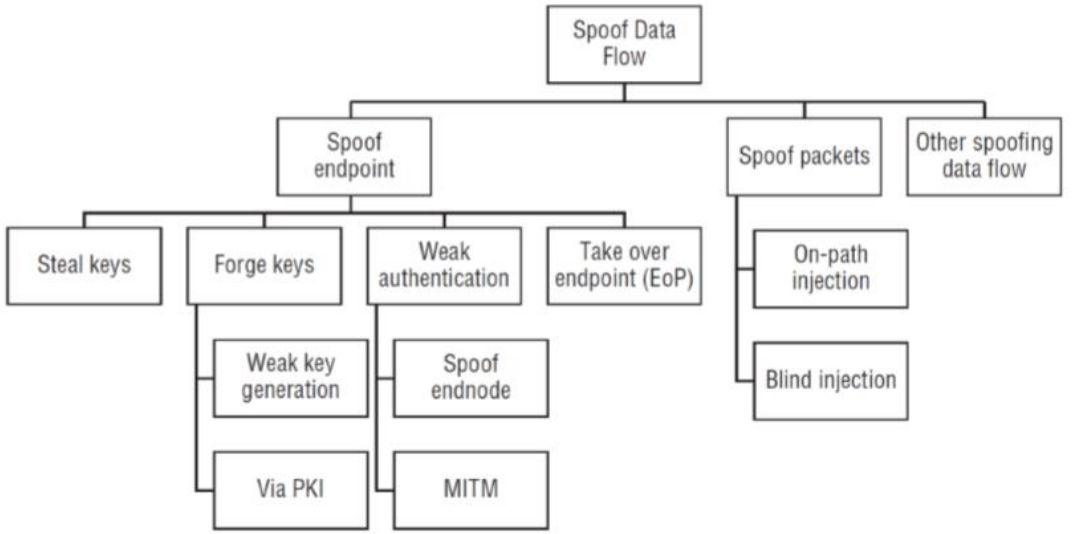
- LINDDUN (linkability, identifiability, nonrepudiation, detectability, disclosure of information, unawareness, non-compliance
- Focuses on privacy
- Can be used for data security



https://insights.sei.cmu.edu/blog/threat-modeling-12-available-methods/

Attack Trees

- Diagrams that depict attacks on a system in tree form
- Tree root is the goal for the attack
- Leaves are ways to achieve that goal
- Can be used in combination with other techniques such as STRIDE, CVSS, or PASTA



//www.theseus.fi/bitstream/handle/10024/220967/Selin_Juuso.pdf?sequence=2&isAllowed=y

https://insights.sei.cmu.edu/blog/threat-modeling-12-available-methods/

Persona non Grata

- View the system from an unintended-use point of view
- can be helpful in the early stages of the development
- examine the attacker's skills, motivations, and goals

Description: Mike worked as a contractor installing SCADA radio-controlled sewage equipment for a municipal authority. After leaving the contractor, Mike applied for a job with the municipality but was rebuffed. Feeling bitter and rejected, Mike decides to get even with the municipality and his former employer.

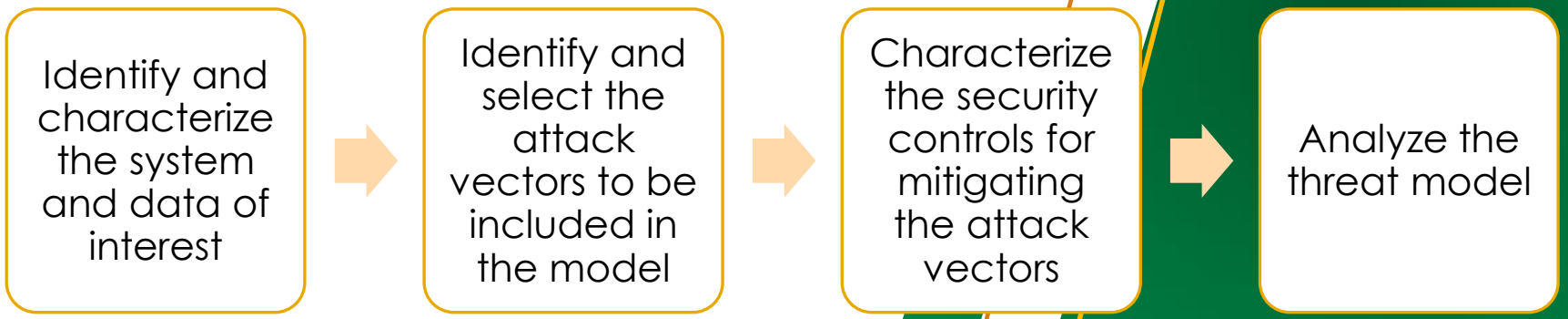
Goals: Cause raw sewage to leak into local parks and rivers and make the events appear as malfunctions. Create a public backlash against the contractor and municipality.

"Mike" is based on the true story of Vitek Boden, who was convicted of causing the release of sewage in Maroochy Shire Council in Queensland, Australia in 2000 after hacking the associated SCADA system. See Abrams & Weiss, 2008.

https://insights.sei.cmu.edu/blog/threat-modeling-12-available-methods/

NIST Special Publication 800-154

- Data-centric system threat modeling
- Focus on protecting particular types of data within systems



https://insights.sei.cmu.edu/blog/threat-modeling-12-available-methods/

Security Cards

brainstorming technique

Card set

- Who might attack?
- Why might the system be attacked?
- What assets are of interest?
- How can these attacks be implemented?

https://insights.sei.cmu.edu/blog/threat-modeling-12-available-methods/

Hybrid Threat Modeling Method (hTMM)

Identify	Identify the system to be threat-modeled.
Apply	Apply Security Cards based on developer suggestions.
Remove	Remove unlikely PnGs (i.e., there are no realistic attack vectors).
Summarize	Summarize the results using tool support.
Continue	Continue with a formal risk-assessment method

https://insights.sei.cmu.edu/blog/threat-modeling-12-available-methods/

Quantitative Threat Modeling Method

Attack trees, STRIDE, and CVSS methods applied in synergy

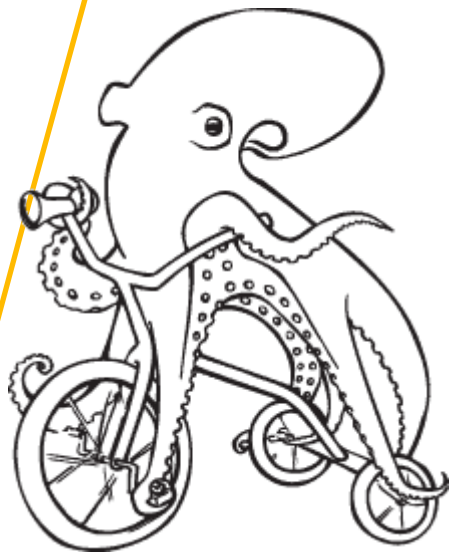
build component attack trees for the five threat categories of STRIDE

dependencies among attack categories and low-level component attributes

CVSS method to calculate scores for the components in the tree

https://insights.sei.cmu.edu/blog/threat-modeling-12-available-methods/

Trike



- security audit framework that uses threat modeling as a technique
- **Version 1**
  - automatic threat generation at the requirements level
  - automatic generation of attack trees.
- **Version**
  - interim bridge between version 1 and version 2.
  - Highlights include improved automatic threat generation at the requirements level, security objectives, the complete absence of threat trees, and HAZOP analysis
- **Version 2**
  - semi-automatic threat generation at the architectural level and attack chaining.
  - Version 2 is under active developmen

https://insights.sei.cmu.edu/blog/threat-modeling-12-available-methods/

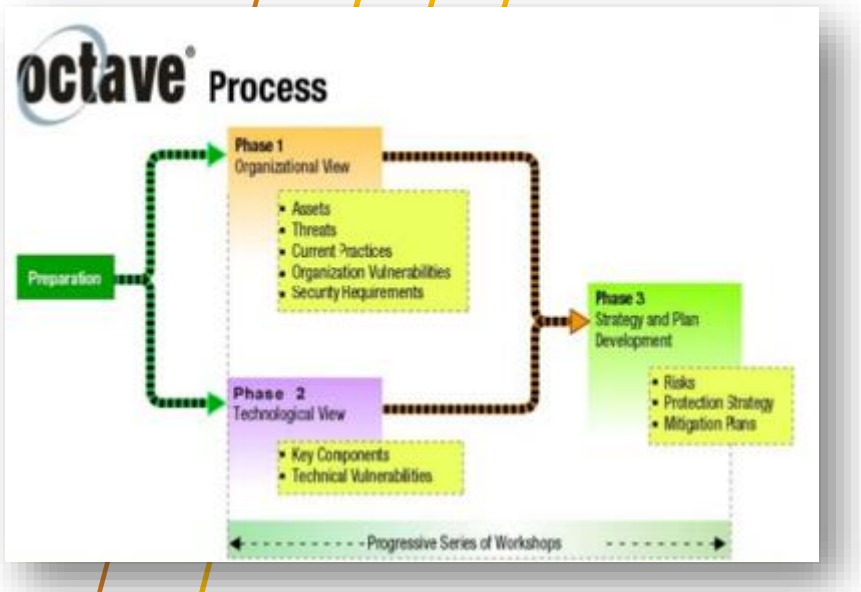
VAST

- Visual, Agile, and Simple Threat (VAST) Modeling
- Based on ThreatModeler, an automated threat-modeling platform
- Two types of models:
  - Application threat models
    - Process-flow diagrams, representing the architectural point of view
  - Operational threat models
    - Attacker point of view based on DFDs

https://insights.sei.cmu.edu/blog/threat-modeling-12-available-methods/

OCTAVE

- Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE)
- Risk-based strategic assessment and planning method for cybersecurity
- Assessing organizational risks and does not address technological risks
  - Build asset-based threat profiles
  - Identify infrastructure vulnerability
  - Develop a security strategy and plans



https://insights.sei.cmu.edu/blog/threat-modeling-12-available-methods/

Threat Analysis

Threat analysis and attack lifecycle

- Threat information can serve as a basis for building and selecting security measures

- Various Tactics, Techniques, and Procedures (TTPs) could be used against energy sector targets during the various phases of a cyber attack (or attack lifecycle)

Threat and Threat Analysis

How do we define threat, threat actors, and how do we analyse them?

Defining the Attack Lifecycle

Can understanding the phases of a cyber attack improve security and reduce risk?

Threat Attributes

What are the threat attributes that are most concerning?

Classifying and Prioritising Threats

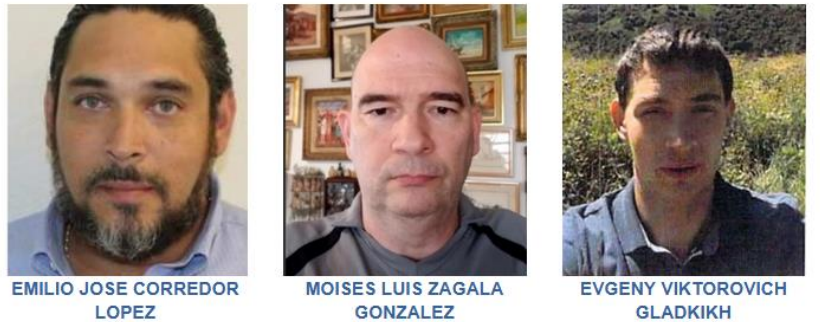
What is the process of classifying and prioritising threat attributes?

What is the difference between threat analysis and attack lifecycle?

Successful cyber-attacks

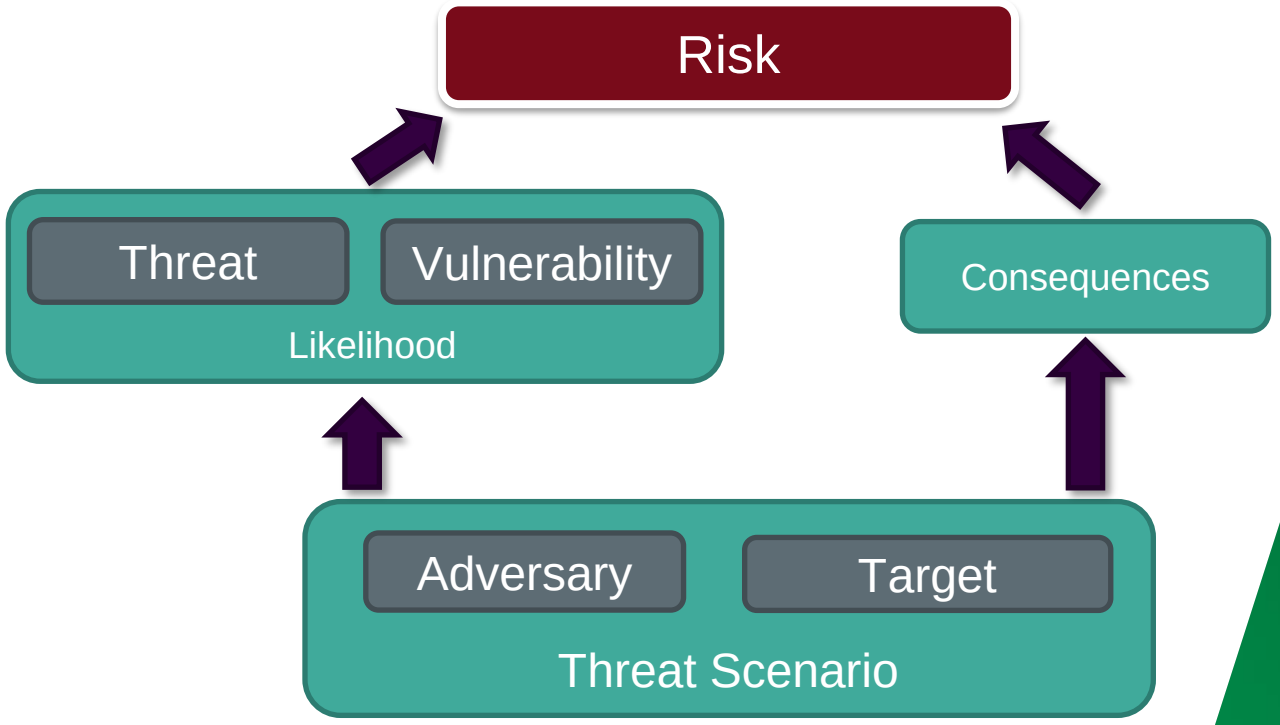
- There are several conditions for a successful cyber-attack:
  - A target must have discoverable vulnerabilities or weaknesses in systems and processes
  - A threat actor must have sufficient resources to use the vulnerabilities and exploit the system
  - The threat actor must believe they will benefit by performing the attack – attractiveness

- We can use threat data to inform security:
  - We can design and defend our systems better if we know how to classify threats (and their methods/ attributes)



Some of the FBI's "Most Wanted" hackers.
Source: fbi.gov/wanted/cyber

Successful cyber-attacks



Threat actor Characterisation

- **Motivation and Intent**
  - Reason and goal the adversary seeks to achieve; financial vs. ideological or theft vs. sabotage

- **Capabilities**
  - Ability and tools of adversary to successfully achieve intent – their Tactics Techniques and Procedures (TTPs)

- **Opportunity**
  - Knowledge of vulnerabilities and the threat actor's ability to leverage and exploit them to breach the system

Types of cyber-attack

Information Attacks

- Capability
  - Many tools available
- Motive
  - Financial Gain
  - Espionage
  - Competitive advantage
  - Prep for OT attack
- Intent
  - Acquire and exploit information

Successful information attacks can tell the attacker what to do for an attack on OT systems

What kinds of capabilities are needed for the different types of attacks?

Operational Attacks

- Capability
  - Requires specialised expertise
- Motive
  - Political
  - Terror
- Intent
  - Physical damage
  - Disruption of critical infrastructure

Threat analysis

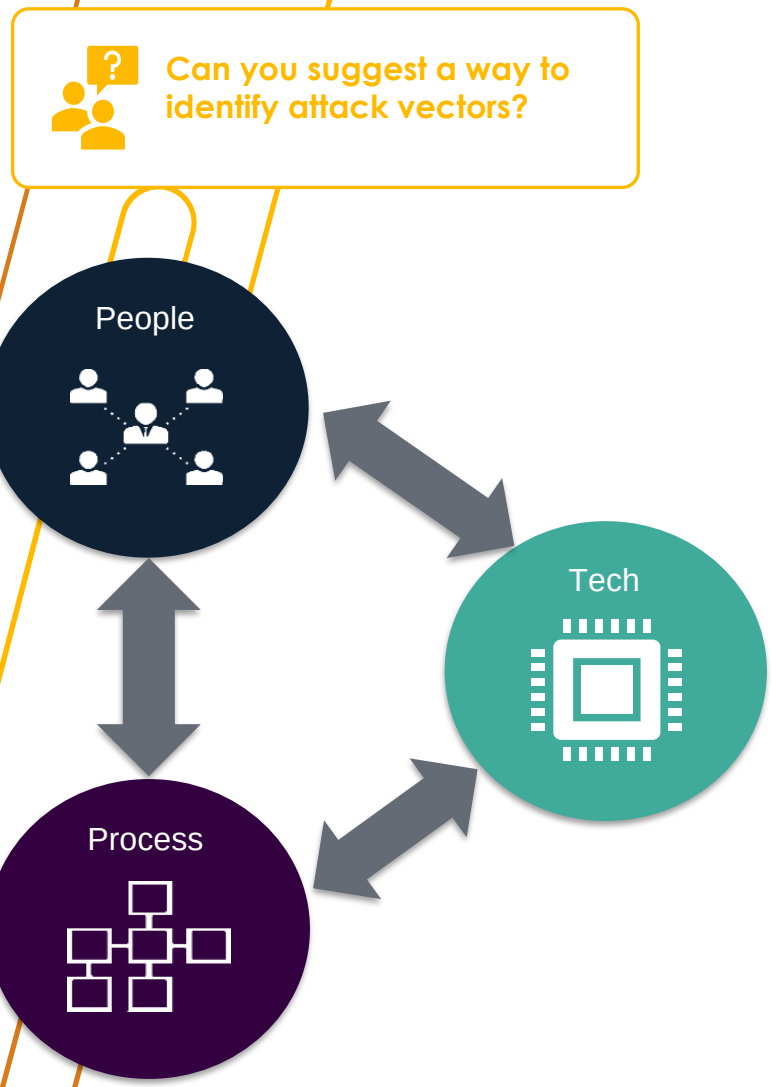
- What is it?
  - An analysis of threat actor characteristics and their activities
  - Gathering and comparing threat data to identify threats and their intent, motivation, and capabilities

- Why do it?
  - Analysing threat activity trends can be used to create effective defences
  - Create situational awareness of threat trends
  - Identify potential detection points if adversaries try the same TTPs

Where can we get data about threats?

Analysing Cyber-attacks

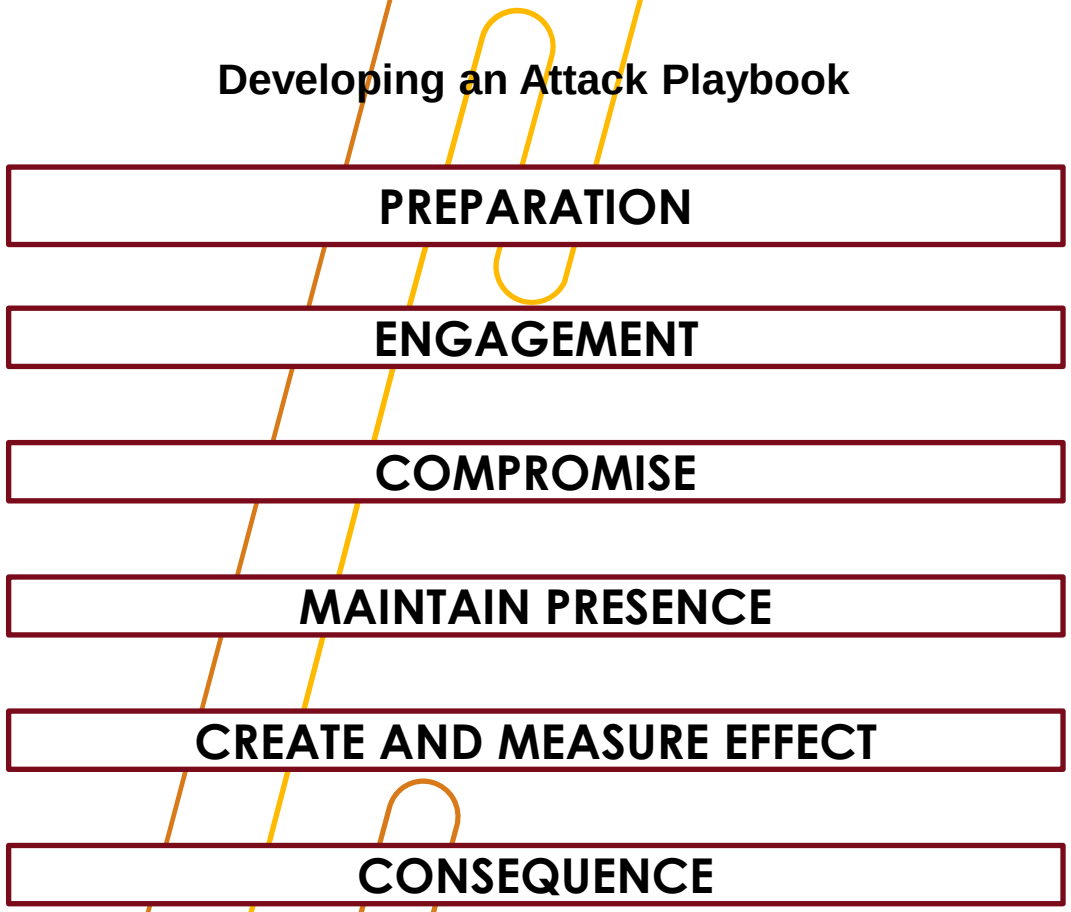
- Attack vectors are pathways a threat uses to propagate or infect systems
  - These vectors exploit people, process, and/or technology
- Blended attacks are coordinated acts that utilise both cyber and physical aspects
- Cyber-attack frameworks depict the phases of an attack and help to identify the capabilities an adversary must have to achieve an objective
- Threat-based defence uses these phases to provide defenders more opportunity to discover and respond to an attack



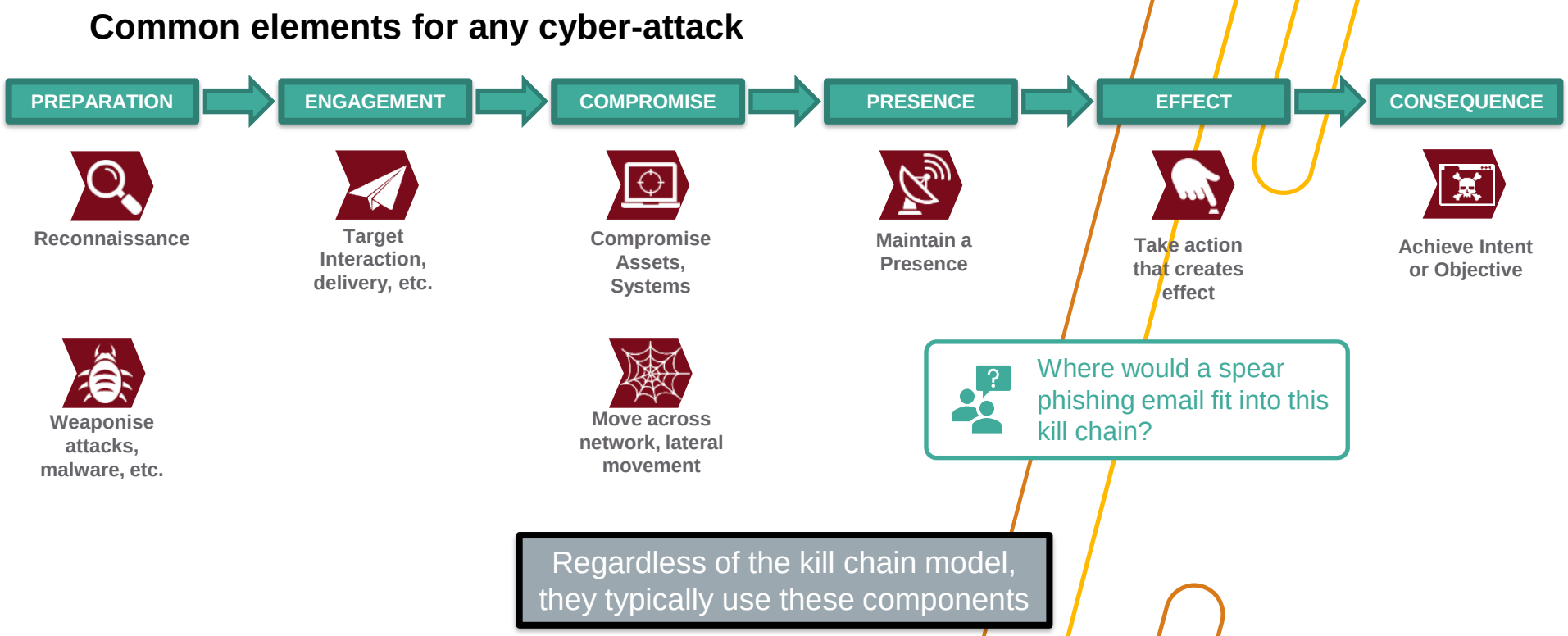
Attack frameworks

Attack actions a threat actor must do…

1. Choose a target and prepare for attack
2. Collect and use intelligence
   - Define vectors
   - Validate the opportunity
   - Assess if capability is adequate
3. Engage target
4. Compromise target and get access
5. Maintain presence and advance
6. Determine if attack is working and adjust as required (feedback)
7. Complete their goal



Kill Chain


Common elements for any cyber-attack
PREPARATION
ENGAGEMENT
COMPROMISE
PRESENCE
EFFECT
CONSEQUENCE
Reconnaissance
Target Interaction, delivery, etc.
Compromise Assets, Systems
Maintain a Presence
Take action that creates effect
Achieve Intent or Objective
Weaponise attacks, malware, etc.
Move across network, lateral movement
Where would a spear phishing email fit into this kill chain?
Regardless of the kill chain model, they typically use these components

An Example Kill Chain Model: SANS ICS Kill Chain

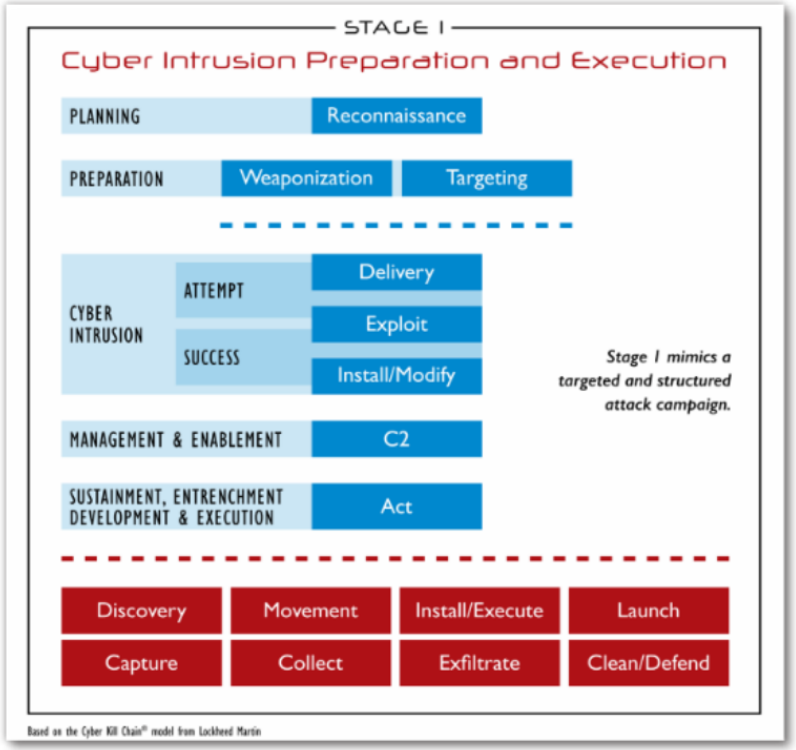


Figure 1. Stage 1: Cyber Intrusion Preparation and Execution

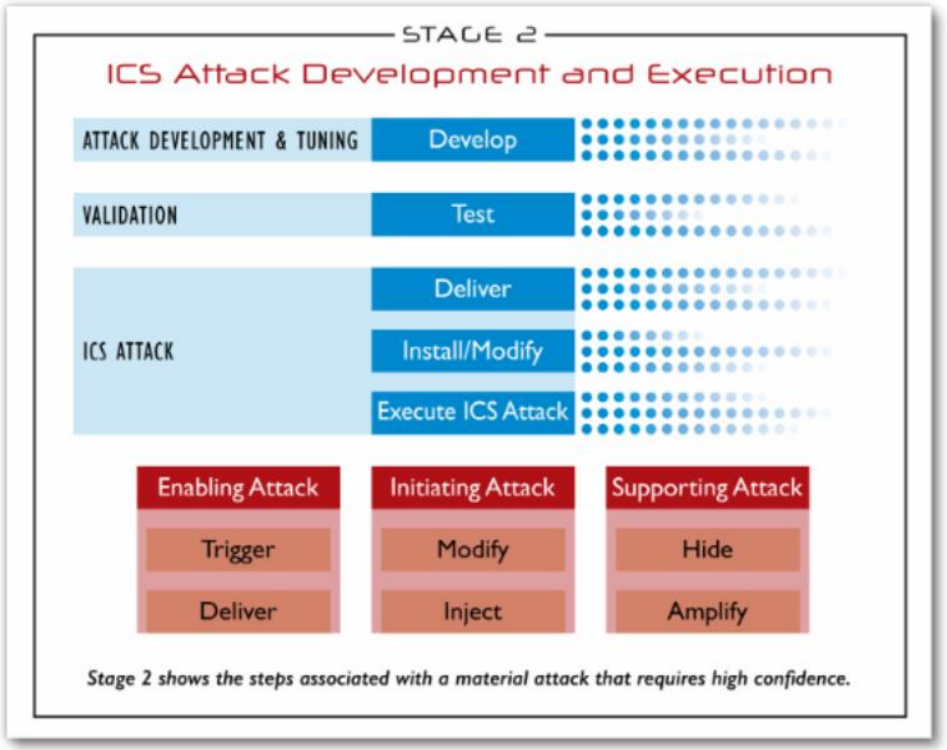


Figure 2. Stage 2: ICS Attack Development and Execution

Source: M.J. Assante, R.M. Lee, The Industrial Control System Cyber Kill Chain, SANS Institute, October, 2015.

Analysing intent and motivation

- Ask yourself, what is an attractive asset to target?
  - This depends on the threat actor's motivation and intention

- Potential targets might include:
  - Personal data
  - Financial systems (e.g. related to payment)
  - Operational systems (sabotage)
  - …

Can you think of other targets and what would motivate a threat actor to exploit them?

Focus on capabilities

Adversary types based on their capabilities

Develops full spectrum & injects vulnerabilities & exploits

Develops sophisticated exploits against vulnerabilities

Discovers system vulnerabilities

Exploits known vulnerabilities with published exploits

Relating capability to kill chain

No.	Description	Threat Class 1	Threat Class 2	Threat Class 3	Threat Class 4
1	Define target	X	X	X	X
2.	Find and organise accomplices			X	X
3.	Build or acquire tools			X	X
4.	Research target infrastructure and employees		X	X	X
5.	Test for detection			X	X
6.	Deployment		X	X	X
7.	Initial intrusion	X	X	X	X
8.	Outbound connection initiated	X	X	X	X
9.	Expand access and obtain credentials		X	X	X
10.	Strengthen foothold			X	X
11.	Exfiltrate data	X	X	X	X
12.	Cover tracks and remain undetected			X	X

What makes a successful attack

Think like an attacker:

- An attack is designed to optimise success and minimise effort
- The ‘human’ element should be exploited
- Data is accessed, stolen, and/or manipulated
  - Collecting as much about the target network environment as possible
  - Include data on people, processes, policies
  - Theft of credentials and/or sensitive information often required for access
- Any “trust" between systems must be exploited
  - Pivot from a single asset/domain into another simply by association
  - Leverage access that already exists
- Access has to be maintained so success can be measured
- Where possible use physical attacks to get access to assets and networks behind security countermeasures

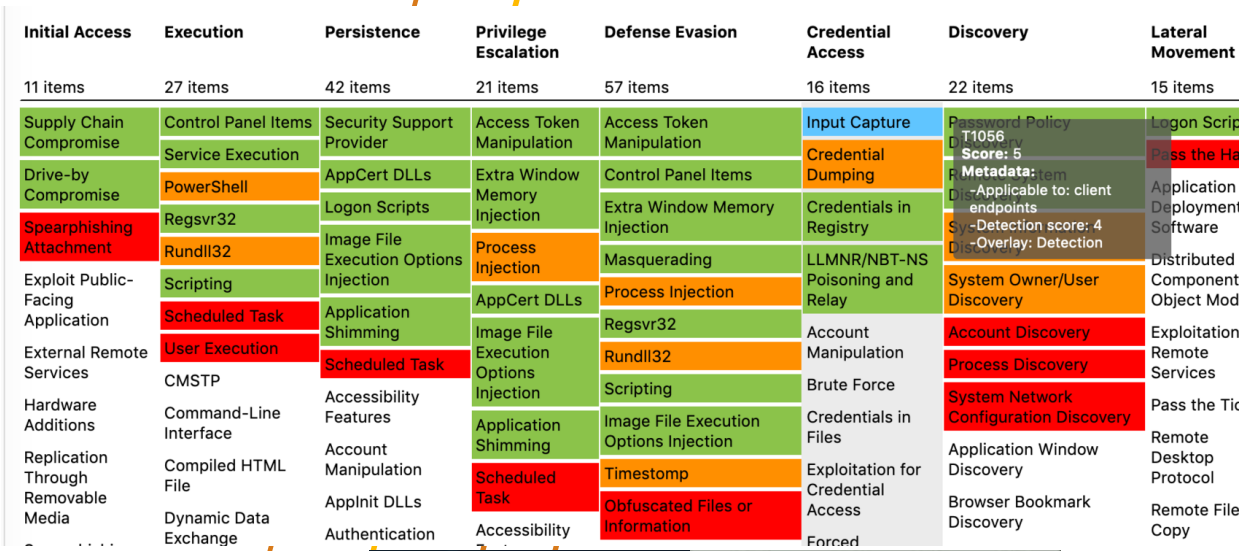
What can we do?

- Understand where our systems provide opportunity to an attacker
- Proactively determine how an attacker could attack
- Assess how seemingly unrelated information can inform the adversary
- Understand and manage our vulnerability
- Determine what capability must be required for an adversary to exploit a vulnerability
- Defend our assets so the work effort of an adversary is increased with the goal of ensuring:
  - Capability is insufficient to attack
  - Opportunity landscape is too small
  - Desired consequence/intent is unachievable
- Mitigate the effects of malicious events before a more severe consequence is realised

What example specific activities can we carry out to build this knowledge and capability?

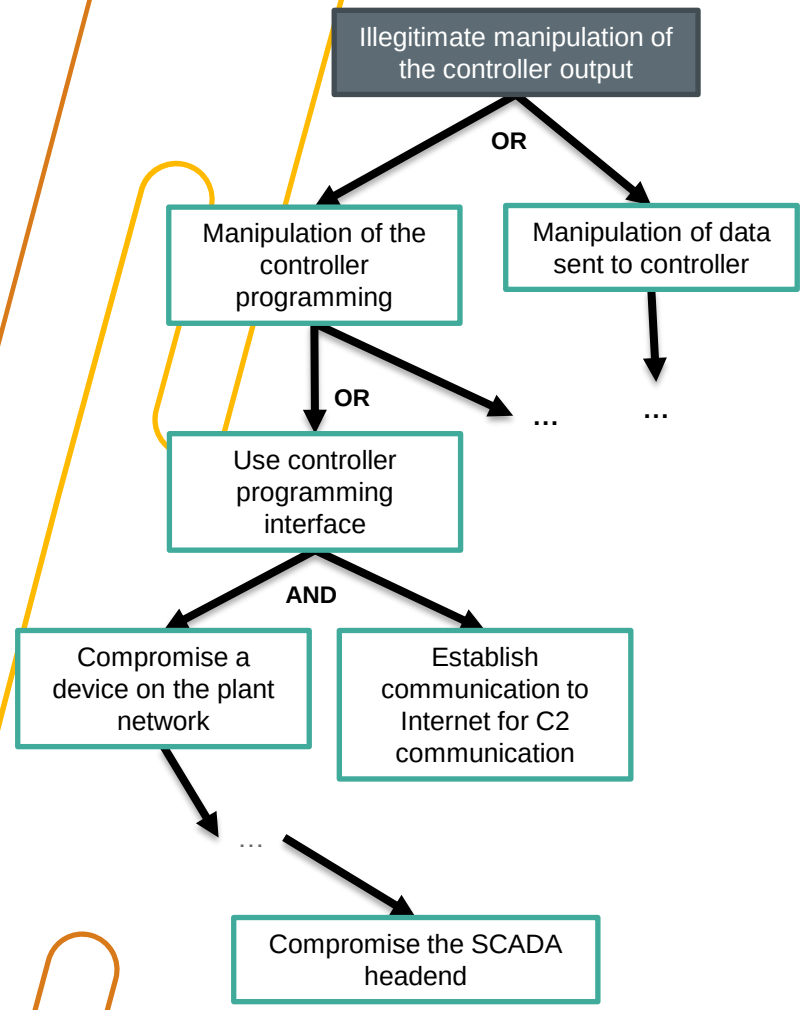
The MITRE ATT&CK Framework

- A structure knowledge-base describing adversary TTPs, arranged into matrices
- Several matrices existing, including for enterprise and industrial control systems
- Knowledge is based on reported incidents from the cybersecurity community
- Can be used to support security automation tasks to evaluate threats and an organization's readiness



Attack trees

- A structured analysis of attack vectors from an attacker's perspectives
  - Identification of the attack goal as the root of the tree
  - Decomposition into sub-goals until sufficient fine granularity is reached (AND, OR)
  - Evaluation of leaf trees with respect to likelihood
  - Propagation of values to the root of the tree
    - AND: minimum of children
    - OR: maximum of children
  - Identification of major attack paths (subgraph)



ThreatGet

Threat Modeling

Threat modeling is a building block in security engineering that:

- Detect potential security weaknesses in a system model
- Define the corresponding security mitigations

https://www.theprepperjournal.com/2018/02/02/hardscaping-projects-backyard/

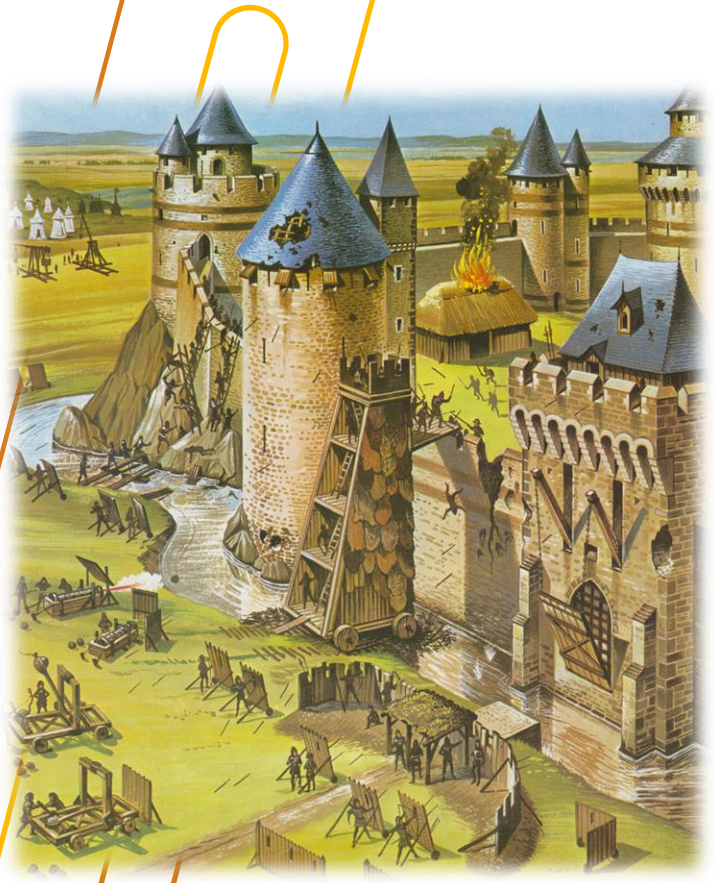
Schmittner C, Tummeltshammer P, Hofbauer D, Shaaban AM, Meidlinger M, Tauber M, Bonitz A, Hametner R, Brandstetter M. Threat Modeling in the Railway Domain. InInternational Conference on Reliability, Safety, and Security of Railway Systems 2019 Jun 4 (pp. 261-271). Springer, Cham.

Threat Modeling

Threat modeling is a building block in security engineering that:

- Detect potential security weaknesses in a system model
- Define the corresponding security mitigations

https://www.theprepperjournal.com/2018/02/02/hardscaping-projects-backyard/



http://history.parkfieldprimary.com/medieval/attacking-a-castle

Schmittner C, Tummeltshammer P, Hofbauer D, Shaaban AM, Meidlinger M, Tauber M, Bonitz A, Hametner R, Brandstetter M. Threat Modeling in the Railway Domain. InInternational Conference on Reliability, Safety, and Security of Railway Systems 2019 Jun 4 (pp. 261-271). Springer, Cham.

Why ThreatGet?

- ThreatGet is a threat analysis tool developed by AIT, that aims to automate the threat analysis approach to identify potential threats in a system model due to the existence of security vulnerabilities.
- ThreatGet helps you innovate this expensive and subjective process by automating the analysis and formalizing threat information.
- Its analysis results are reusable, and all mitigations and design decisions are traceable through the development process.
- ThreatGet helps save cost, and due to the updatable threat catalog, the analysis stays up-to-date automatically.

AUTOMATED SECURITY ASSESSMENT

ThreatGet automatically identifies threats and supports ongoing risk management. The tool extends the well-established Enterprise Architect modeling platform and is designed to support use cases in different domains.

EXTENSIBLE MODEL LIBRARY

ThreatGet contains domainspecific security-relevant elements for system modeling. Company specific model elements and threats can also be added. All model elements contain predefined security parameters to consider existing security concepts.

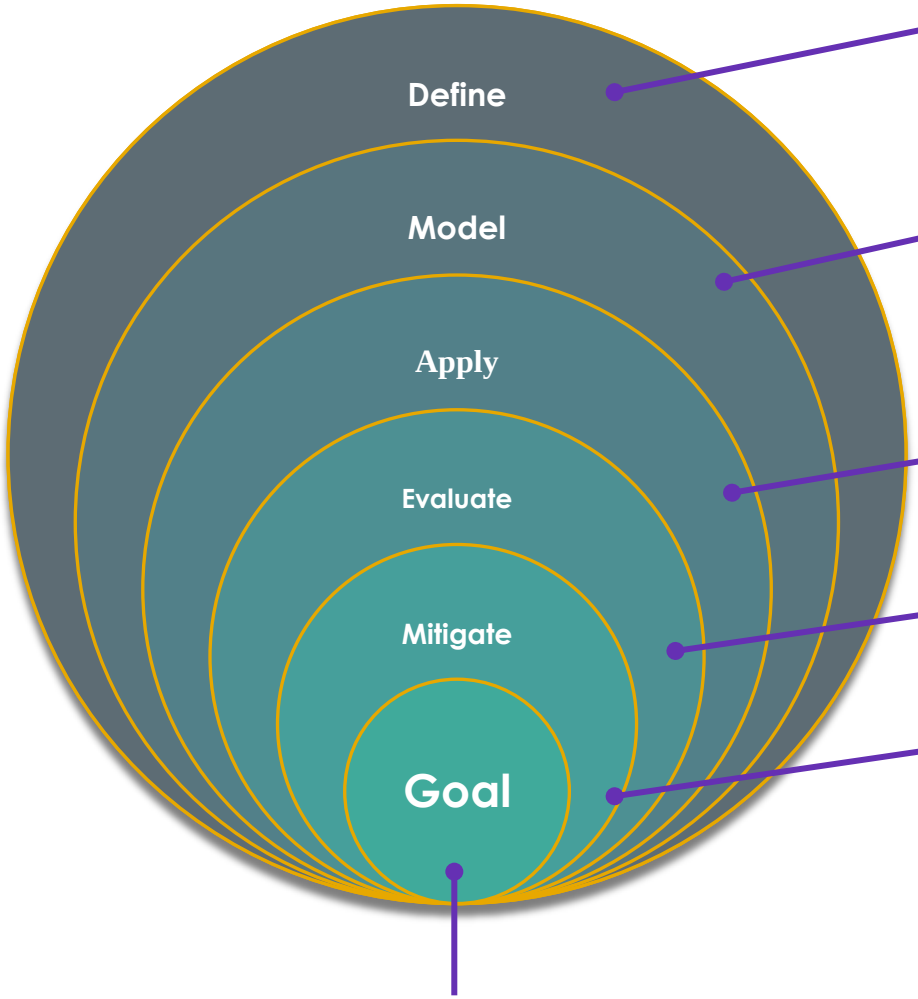
AUTOMATED SUGGESTION OF MITIGATIONS

ThreatGet will automatically assess the system model to find potential security issues in the design and will also suggest mitigations.

AUTOMATED THREAT INTELLIGENCE UPDATES

Stay up to date with ThreatGet and receive the latest cybersecurity threats with a threat-database subscription.

ThreatGet Modeling Steps



Step 1: Define the system components with all security related assumptions and necessary information

Step 2: Creating a system diagram

Step 3: Apply ThreatGet to the system model to identify potential threats

Step 4: Evaluate risk severity for each potential threat

Step 5: Update the system model with new security assumptions

Ensure security-by-design.

ThreatGet Features

Modelling Process: the most common elements and objects in the Automotive domain.

Security Parameters: easily adapt security values of the model elements for some security mitigation.

Rule-Based Analysis: A collection of rules, describing when a specific threat is relevant. This rules are defined, using names and properties of stencils, and stored in a threat database.

Threat Categories: STRIDE threat classification is the guide for threats analysis and detection process.

Results: the detected threats are displayed in separated diagrams in a neat structure.

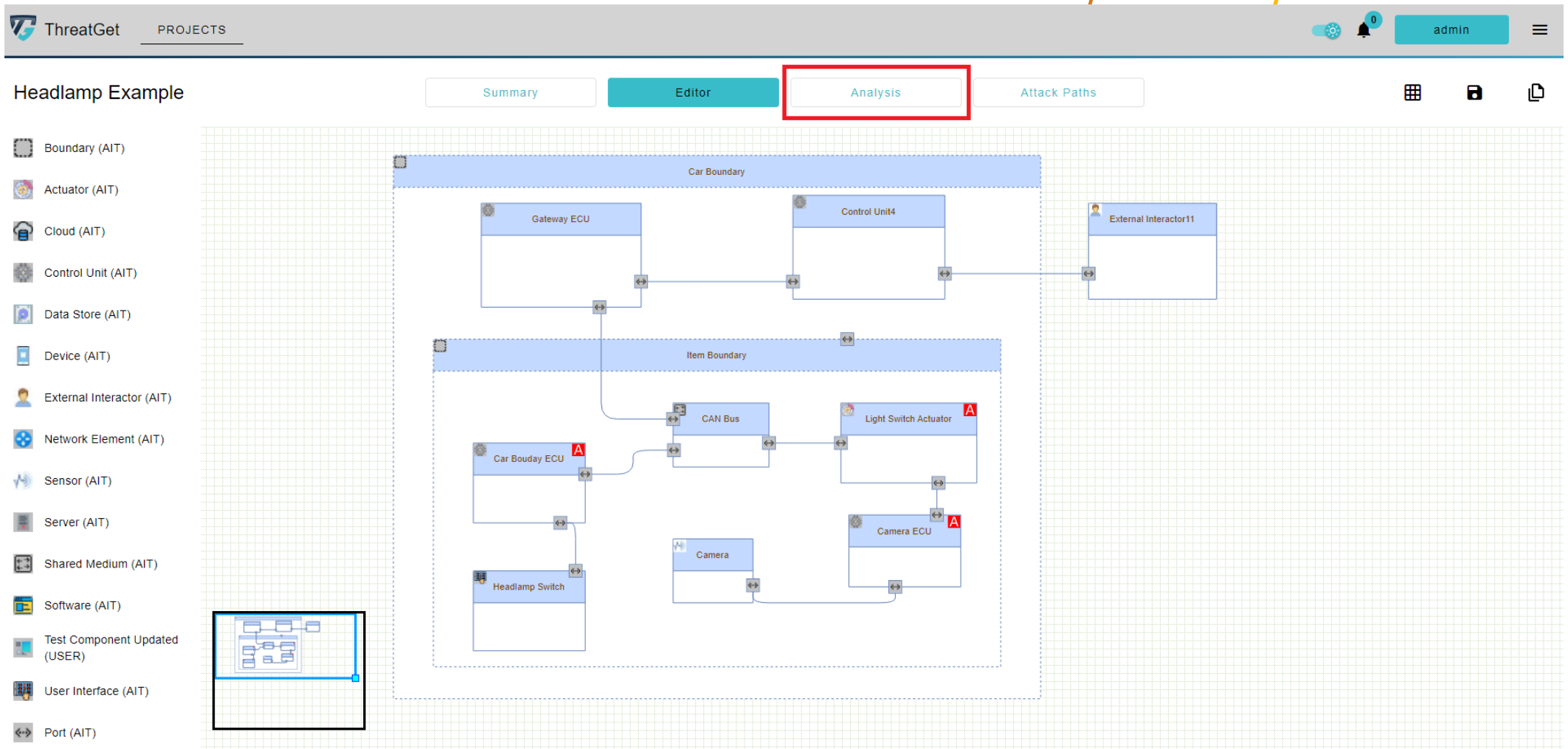
Traceability: ThreatGet aggregates all generated threats into one table and each threat traceable to the respecvtive element (-> Traceability is in the model)

Risk Assessment: ThreatGet is evaluating the degree of risks of the identified potential threats.

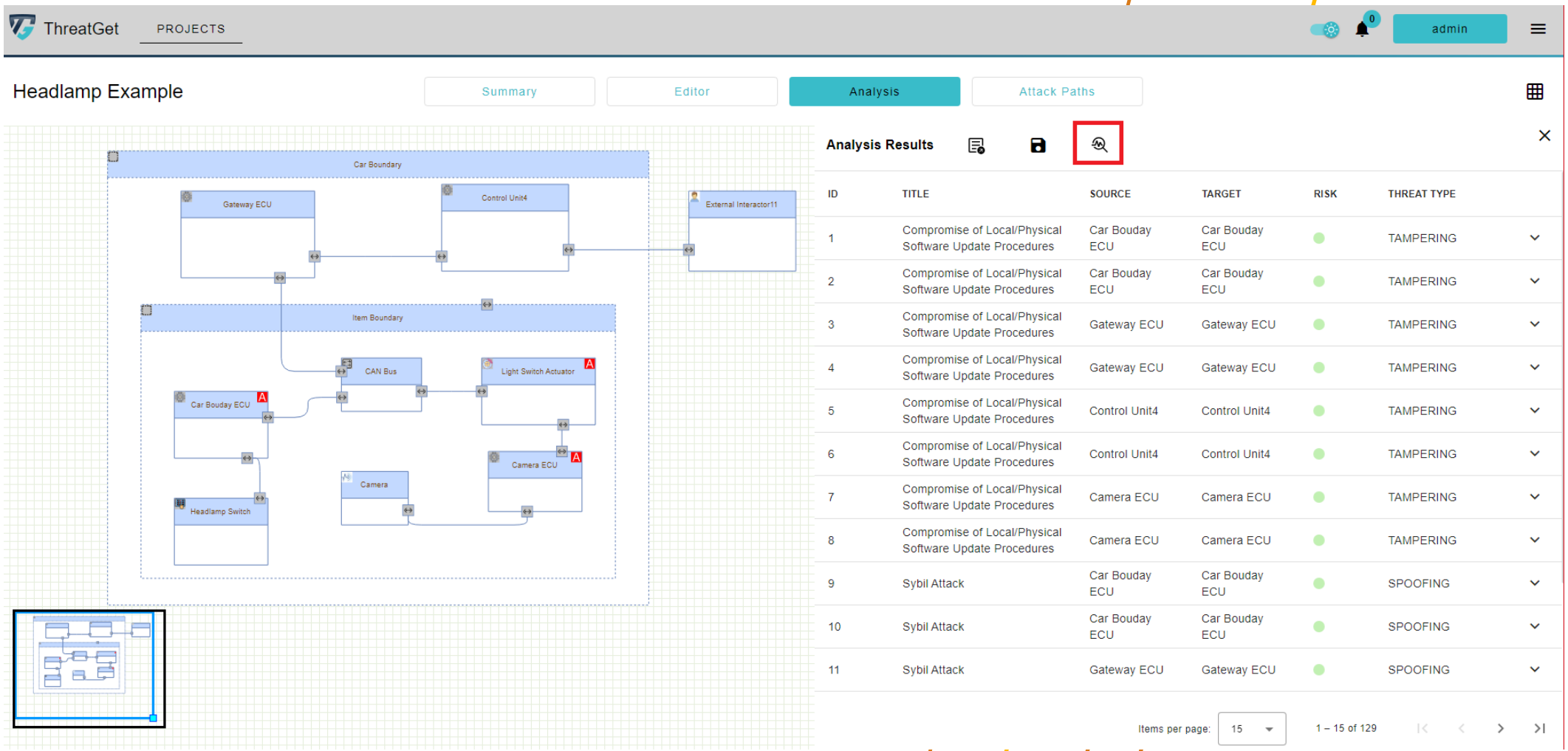
Documentation: ThreatGet generates a complete report has the all identified threats with a picture of the related model element.

Example: ThreatGet

ThreatGet: Modelling Example



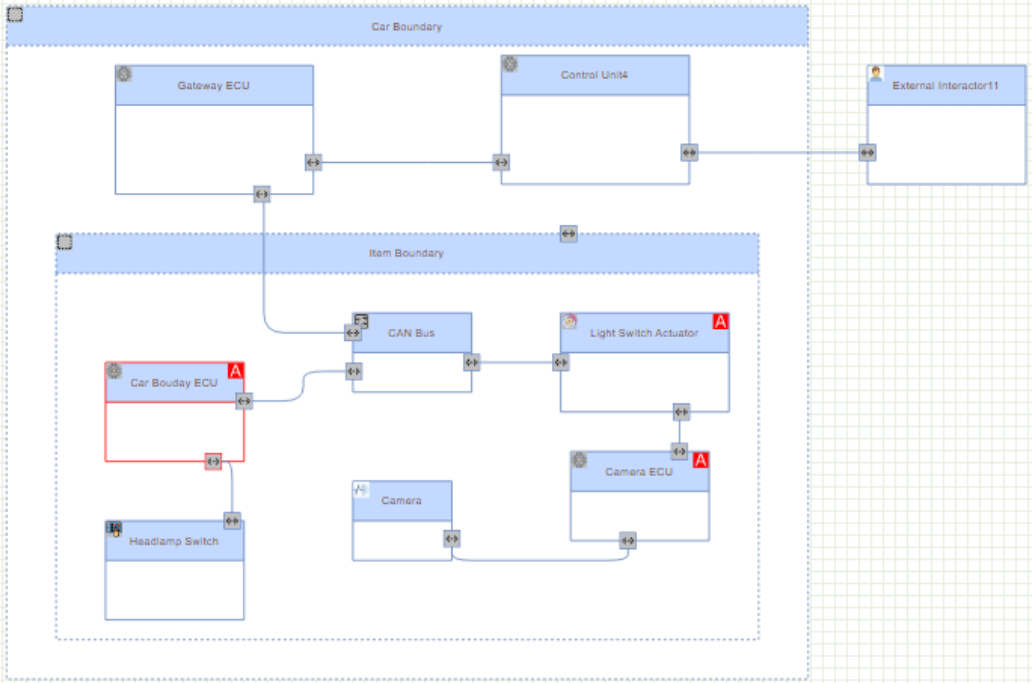
ThreatGet: Analysis Outcomes



ID	TITLE	SOURCE	TARGET	RISK	THREAT TYPE
1	Compromise of Local/Physical Software Update Procedures	Car Bouday ECU	Car Bouday ECU		TAMPERING
2	Compromise of Local/Physical Software Update Procedures	Car Bouday ECU	Car Bouday ECU		TAMPERING
3	Compromise of Local/Physical Software Update Procedures	Gateway ECU	Gateway ECU		TAMPERING
4	Compromise of Local/Physical Software Update Procedures	Gateway ECU	Gateway ECU		TAMPERING
5	Compromise of Local/Physical Software Update Procedures	Control Unit4	Control Unit4		TAMPERING
6	Compromise of Local/Physical Software Update Procedures	Control Unit4	Control Unit4		TAMPERING
7	Compromise of Local/Physical Software Update Procedures	Camera ECU	Camera ECU		TAMPERING
8	Compromise of Local/Physical Software Update Procedures	Camera ECU	Camera ECU		TAMPERING
9	Sybil Attack	Car Bouday ECU	Car Bouday ECU		SPOOFING
10	Sybil Attack	Car Bouday ECU	Car Bouday ECU		SPOOFING
11	Sybil Attack	Gateway ECU	Gateway ECU		SPOOFING

Items per page: 15 1 – 15 of 129

ThreatGet: Analysis Outcomes



1	Compromise of Local/Physical Software Update Procedures	Car Bouday ECU	Car Bouday ECU		TAMPERING

Title:
Compromise of Local/Physical Software Update Procedures

Risk Level: 1

Description:
Compromise of Local/Physical Software Update Procedures involves the physical manipulation of a software update mechanism by external or internal malicious actors. This type of cyberattack is distinct and particularly concerning because it involves direct, physical intervention in the update processes. Unlike remote cyber threats, this attack vector exploits the physical accessibility, allowing attackers with technical knowledge of software systems to alter or fabricate system update programs or firmware. The direct nature of this threat implies that the attacker might be someone with insider access, such as a service technician, or an external party who has managed to gain physical access. The implications of such an attack are severe, as they can lead to the unauthorized introduction of malicious functionalities or access capabilities within the software systems, potentially compromising safety, operational integrity, and user privacy. This threat targets the local or physical aspect of the software update mechanism, which is particularly vulnerable to manipulation when an attacker gains physical access. By modifying, replacing, or introducing harmful elements into the firmware or software updates, attackers can initiate a range of unauthorized actions. These could range from subtle system malfunctions to more overt takeovers of control systems or access to sensitive user data. To effectively counter this risk, it is crucial to implement robust physical security measures that restrict unauthorized access to the software update systems. Secure software delivery protocols are also necessary to ensure the authenticity and integrity of software and firmware updates, even when performed physically. Additionally, specific measures to detect and prevent tampering or spoofing of software updates, especially those involving physical access, are critical components of a comprehensive defense strategy.

Likelihood:
HIGH

Category: TAMPERING

Impact:

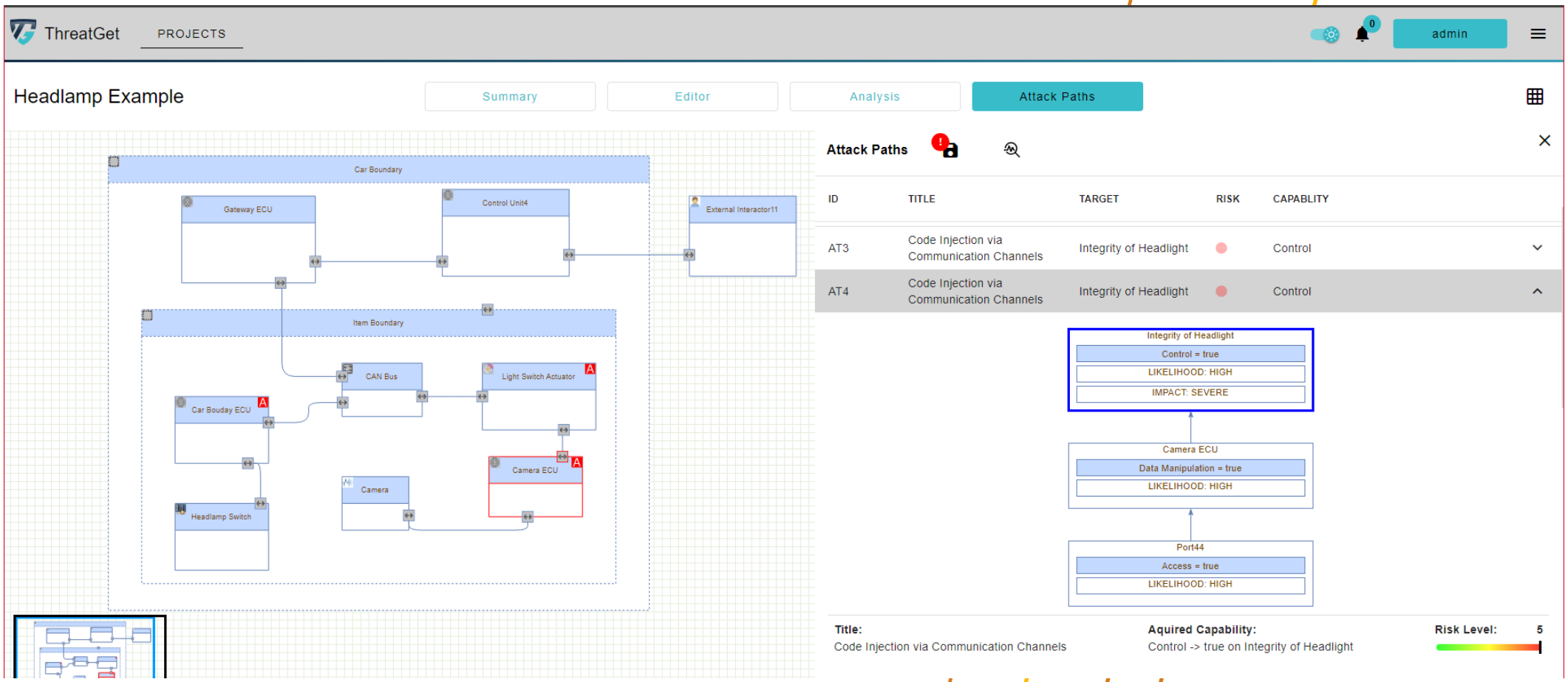
S	F	O	P
NEG.	NEG.	NEG.	NEG.

Attack Feasibility:

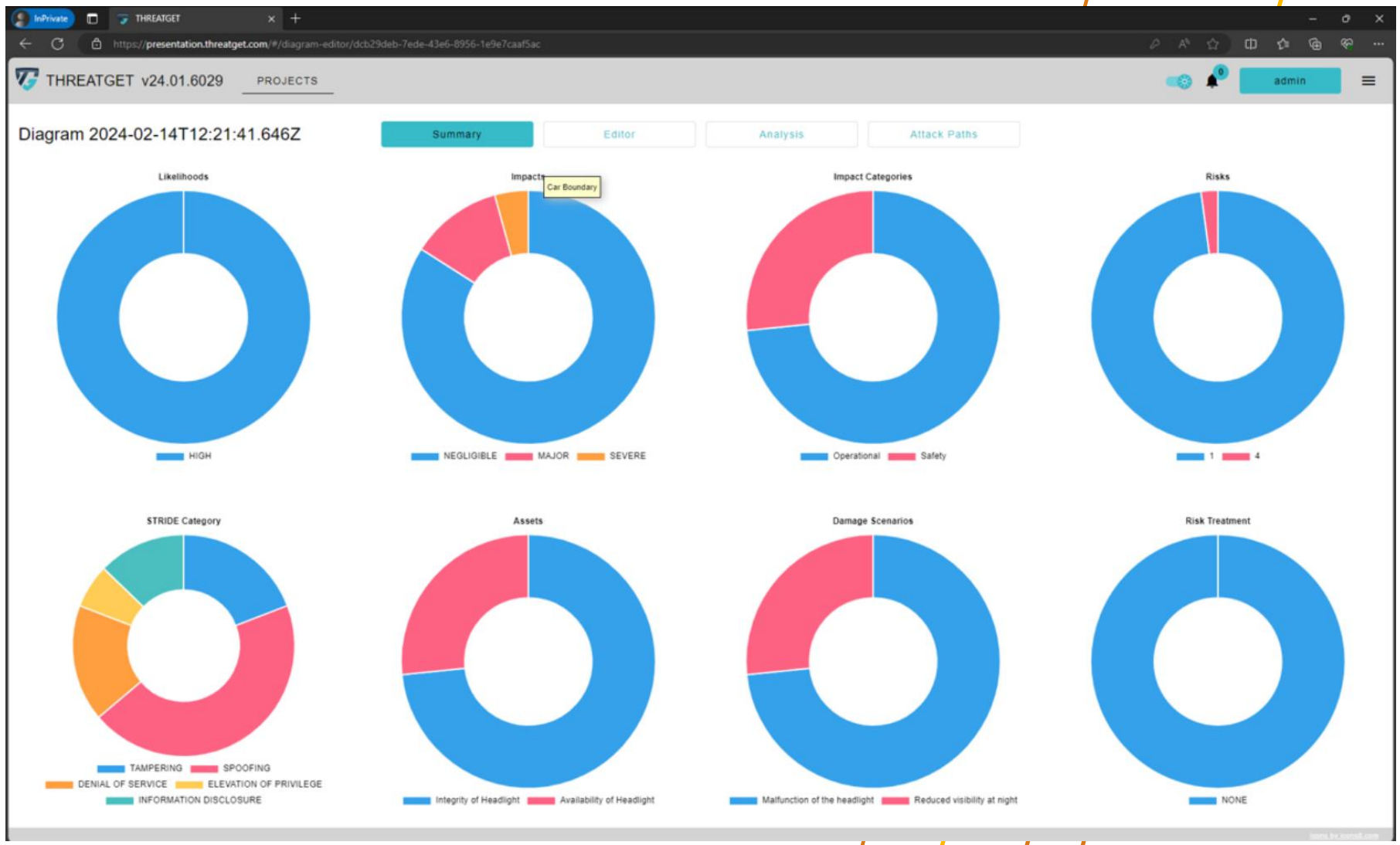
Elapsed Time: <=One Day
Expertise: Layman
Knowledge: Restricted
Window of Opportunity: Moderate
Equipment: Standard

Risk Treatment: NONE

ThreatGet: Attack Path



ThreatGet: Analysis Summary



Practical: ThreatGet

Practical Work

1. **Utilize ThreatGet**: Develop a model of the critical infrastructure for the energy sector using ThreatGet.
2. **Identify Critical Assets**: Define critical assets and outline all related damage scenarios.
3. **Initial Threat Analysis**: Conduct a preliminary threat analysis before implementing any security properties in the system model.
4. **Security Property Specification**: Identify and define the necessary security properties for each system component.
5. **Revised Threat Analysis**: Perform the threat analysis again, this time with the applied security properties.
6. **Reporting and Documentation**: Report the findings from both the initial and revised analyses. Document the differences in outcomes attributable to the application of security properties.

Summary

- Threat information can be used to manage cybersecurity risk and inform security decisions
- The goal is to implement measures that force the adversary to alter tactics, techniques and procedures using threat information
- It is important to know how you can be targeted by threat actors and understand your role in recognising attributes of a cyber-attack and how operations can be impacted

Connect with CyberSecPro:
How to register and other practical information

1. Website:
   www.cybersecpro-project.eu
2. X (Twitter):
   https://twitter.com/CyberSecPro_eu
3. LinkedIn:
   https://www.linkedin.com/company/cybersecpro-euproject/

Co-funded by the European Union

Co-funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or HADEA. Neither the European Union nor the granting authority can be held responsible for them.

Project Agreement no. 101083594

Thank you

Please send all questions to:
Stefan Schauer
Stefan.Schauer@ait.ac.at
Abdelkader Shaaban,
abdelkader.Shaaban@ait.ac.at