



Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permitted without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"ICS/SCADA Security Essentials (Industrial Control Systems 410)"
at <http://www.giac.org/registration/gicsp>

ICS Layered Threat Modeling

GIAC (GICSP) Gold Certification

Author: Mounir Kamal, Mkamal@QCERT.ORG

Advisor: Chris Walker, CISSP

Accepted: 01/01/2019

Abstract

The ultimate goal of building cybersecurity architecture is to protect systems from potential threats that can cause imminent harm to the institution. Often, we hear a common expression in the information security world “security by design,” which is a deeper terminology than it looks, as it requires compiling a list of possible threats against targeted systems. Building a threat model will guide us on how to build a secure architecture and achieve the security by design concept, and this is what precisely the paper aims to explore. This paper is an intensive study to collect accurate and plausible threat models that can help to secure ICS architecture by design.

Table of Contents

1. Introduction.....	4
2. Industrial Control System Modeling	5
2.1. ICS System Modeling Summary	5
2.2. The Proposed Model.....	6
2.2.1. Axis X Where?	8
2.2.2. Axis Y When?.....	8
2.2.3. Axis Z What? (Why, How, Who?)	8
3. Threat Modeling.....	9
3.1. The objective of Threat Modeling.....	9
3.2. Find Threats.....	10
3.3. Proposed Threat Modeling (Blackbox Threat Modeling)	10
Step 1: System Modeling	12
Step 2: Identifying scopes	13
Step 3: Threat Modeling Classes	13
Step 4 Find Threats	14
3.3.1. Step 5 Create the Threat Matrix.....	15
4. Example of the implementation method	16
Step 1 System Modeling.....	16
Step 2-A System Scope (Level-2 SCADA)	17
Step 2-B Lifecycle Scope (Level-2 SCADA)	17
Step 2-C Boundaries Scope (Level-2 SCADA).....	19
External Interactors	19
Step 3 Threat Modeling Class (Level 2 SCADA).....	21
Step 4 Find Threats (Level-2 SCADA)	21
Step 5 Create a Threat Matrix (Level-2 SCADA)	21
<i>Supply Chain Phase Threat Matrix External Interactors</i>	22
<i>Deployment phase Threat Matrix External Interactors</i>	23
<i>Operation Threat Matrix External Interactors</i>	25
5. Conclusion.....	28
6. Bibliography	30
7. Appendices.....	31
7.1. Appendix A: ICS System Modeling.....	31
7.1.1. Purdue Enterprise Reference Architecture Model (PERA)	31

7.1.2.	Smart Grid Architecture Model (SGAM).....	32
7.1.3.	RAMI 4.0 (Reference Architecture Model for Industry 4.0).....	34
7.2.	Appendix B: Threat Modeling Methods.....	35
7.2.1.	STRIDE.....	35
7.2.2.	Attack Tree.....	36
7.2.3.	PASTA.....	38
7.2.4.	CTSA.....	38
7.3	Appendix C: Process Details	39
7.3.1	Appendix C-1 Level 2 SCADA Deployment Process.....	39
7.3.2	Appendix C-2 Level 2 SCADA Operation Process.....	39
7.3.3	Appendix C-3 Level 2 SCADA Maintenance Process	40
7.4	Appendix D: Level-2 SCADA Threat Matrix Internal	40

1. Introduction

Threat modeling is a process of identifying potential threats from various perspectives, including the attacker, risk and software points of view. The purpose of threat modeling is to provide security controls with a systematic analysis of the probable attack vectors to study the targeted assets by the attacker, and thus, being able to identify the attacker's profile. Threat modeling answers questions like "What are the possible threats that may target specific assets?", "Why are the possible threats target specific assets?", "Where are assets most probably vulnerable to attack using a threat?", "How such threat could succeed to target assets?", "When such a threat could succeed to target assets?" and "who may be interesting to target assets?". (Gultom, 2018). By definition, a "Threat" is a possibility of occurrence of an undesirable event. (Dictionary, n.d.) Therefore, threat modeling is to find possible "undesirable events." In our life, there are two types of threats: unintentional threats, intentional threats which design to target specific systems. During the study, the focus will be on the second type of threats, the targeted attacks.

Most security professionals think that threat modeling will not provide the proper return on investment due to setting a gap between the requirements versus valuable results from the business point of view. Also, this is due to lack of understanding all factors that define the threat modeling and defining the exact type that will lead to the required objectives. During the study, a flexible framework will determine the required architecture of a framework to give the required objectives of threat modeling: threat modeling like canvas, the responsibility of the author to adjust it based on the required goals of the organization.

Industrial Control Systems (ICS) or OT (Operation Technology) are terms used in the previous twenty years to differentiate between them and the Information Technology (IT) world.

The two worlds of OT and IT are merging to become one, as automation penetrates everything in our life and the new “Internet of Things” (IOT) world started many years ago to have full integration between both OT and IT. Security threats were identified in each world separately, and now the merge between both of them is creating the complex terminology of threat modeling.

This study attempts to find threats in a systematic method that assures covering key factors as best as possible by searching and analyzing most of the ICS systems modeled by different frameworks, then studying the threat modeling concept with different methods aiming to reach a comprehensive, systematic, and practical method for conducting threat modeling of ICS systems.

2. Industrial Control System Modeling

A comparison study of different system modeling frameworks is helpful to recognize the advantages and disadvantages and decide the best approach to conduct the threat modeling study.

Some of the most known ICS system modeling methods are:

- Perdue Enterprise Reference Architecture Model (PERA)
- Smart Grid Architecture Model (SGAM)
- RAMI 4.0 Model

Details of all listed models are in Appendix (A) ICS System Modeling

2.1. ICS System Modeling Summary

The concept of system modeling is to decompose the systems into subsystems that have a defined process, inputs and outputs. Thus, the study can be conducted on various perspectives such as information flow, architecture, integration, and security.

As we go through the previous system models, there are some conventional methods in modeling most of the systems, such as in PERA model which divides the ICS into different zones from the functionality perspectives. (Refer to Appendix A)

One of the advanced system models that cover more detailed aspects is SGAM, which is a 3D model that studies layers, levels, and domains to encompass the complex nature of the systems. SGAM model is mostly designed for the smart grids where the power is the moving element that transfers from one field to another.

RAMI is also a 3D model that studies layers, lifecycle, and levels. Lifecycle starts from the development phase in the factory till the operation on the customer's site.

Ideally, the ultimate system model would be the one that combines all aspects and elements to address possible operations and security threats.

2.2.The Proposed Model

The proposed model aims to study cybersecurity in a more architectural and systematic method. Modeling the targeted system is essential to analyze it. The decomposing and analyzing system requires to answer the following question elements:

- **What?** Describes the system and its components that may reflect on the threat modeling study
- **When?** Describes the system's lifecycle phase such as design, development, distribution, implementation, operation, or maintenance phase.
- **Where?** Describes the components of the system in the context of locations and zones in which a specific subsystem is located.
- **Why?** Describes the primary business objective of subsystems that work collaboratively to accomplish the primary goal

- **Who?** Describes the human factor which is the most important one, also the human factor in each system's phase
- **How?** Describes how this subsystem will conduct the required function to achieve the primary goal.

These questions were inspired by Zachman framework questions for Enterprise Architecture (ZACHMAN, 2008) . To present the model in more detail, let us assume the following:

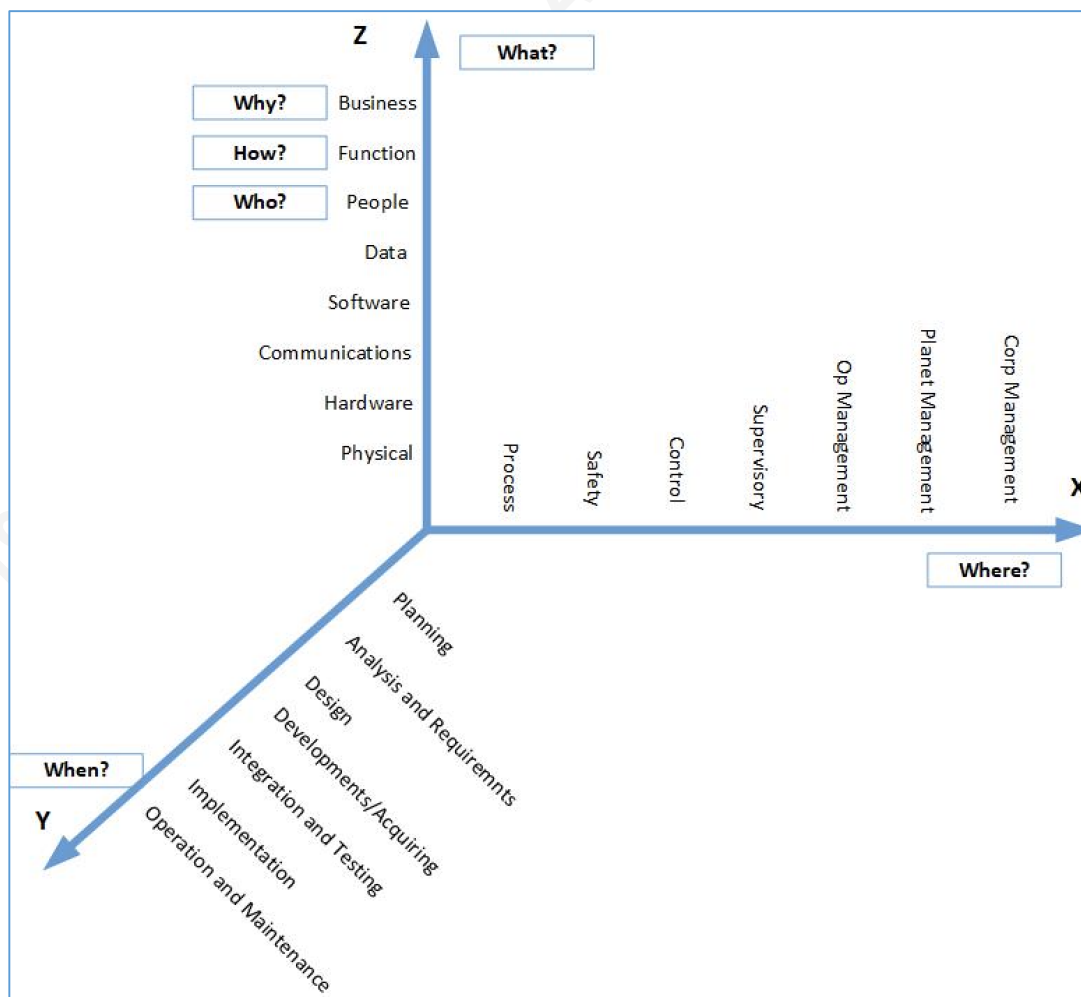


Figure 1 Proposed Model

The 3D axis model answers the presented questions to identify the most common ICS systems. Axis X defines the system architecture and the location of zones and components and helps to address the boundaries and entry points. Axis Y represents the system's life cycle, and axis Z represents the domain.

2.2.1. Axis X Where?

Axis X, represents the "Where" which identifies in which zone subsystems are located relative to the overall system. Using the PERA Model (see Appendix) to define the arrangement of zones within the ICS system although it is very generic, it enables us to oversee multiple systems. The "Where" axis include elements such as processes, safety, control, supervisory, Op management, planet management, and corporates management

2.2.2. Axis Y When?

Axis Y, represents the current phase of the system development lifecycle, starting from the planning and going through design, development and maintenance phases. In the same axis, there is an essential factor that represents the supply chain cycle starting from vendors, system integrator, until the product reaches the site for deployment and operation. It is critical to highlight the importance of axis Y because it focuses on "security by design" concept as it covers the system's development lifecycle and supplies chain issues from the security perspective. System's lifecycle includes phases such as planning, analysis, and requirements, design, development or acquiring, integration and testing, implementation, operation, and maintenance.

2.2.3. Axis Z What? (Why, How, Who?)

Axis Z, represents the subsystems and included components such as information, applications, and networks. At the same time, there are significant issues could assist in answering other questions as well, such as "Why?" which explains the purpose of having any

subsystem from the business perspective, and “How?” which focuses on the function of the subsystem by addressing the received inputs and produced outputs, and finally, the human factor which exists everywhere! We can summarize the levels within axis Z as Business (Why?), Function (How?), People, Data, Software, Communications, Hardware, and Physical (Who). When we answer these questions, we will be able to create the security threat model.

3. Threat Modeling

There are specific properties should be existing within the model to create a genuine threats matrix which are:

- Comprehensive: Inclusive and dealing with almost most possible threats within a specific system.
- Layered: Threat modeling should be multilayered, and each layer could represent a standalone process
- Modular: Consists of flexible modules, so the assessor can run the modules separately if needed to achieve a specific objective (Shostack, 2014)

3.1.The objective of Threat Modeling

Whenever there is a particular system(s), there should be a list of threats that may affect its performance. Studying threats possibilities is the best proactive procedure to ensure the resiliency and strength of the system. Objectives of threat modeling include but not limited to the following:

- Create an attack library listing most possible attaches that could target the system/s and address the most countermeasure of each attack in this library
- Secure the design and development processes of the software.

- Assess vulnerabilities and weak points within the system/s.
- Deliver the concluded information to risk assessment to address possible threats that may harm the system.
- Communicate the situational security awareness level to the developers, security-team, and business owners (Shostack, 2014)

3.2.Find Threats

Through the previous section, we have covered the first phase of the threat modeling process, and now we need to find possible threats based on the proposed system model. There are many methods to find threats within any systems, during the next sections we are going to study the most known methodologies that make addressing threats more systematic by comparing among them to find the strengths and weaknesses of each one within the ICS system. There are threat modeling methods such as STRIDE, Attack Tree, PASTA, and CTSA

Details of Threat Modeling Methods in Appendix B

3.3.Proposed Threat Modeling (Blackbox Threat Modeling)

Threat modeling is a lengthy and complex process that vendors and customers might need but may try to avoid; a valid inquiry may arise from the business stakeholders which is “processes have already been existing since a long time, and everything is going fine, why we need to go through these complex and costly threat modeling processes?” The ultimate answer would be, let us see the cost of being secured against the damage of being vulnerable!

By answering the protocol questions to identify any threat “What, Why, How, Who, Where, and When,” we will be able to create operational threat modeling in a simpler method. The following table shows the proposed threat modeling steps and details.

Please note that there was a presentation having the same name of Blackbox Threat Modeling from the concept of using penetration testing concept terminology as a principle to apply for threat modeling modeling, but the proposed method in this study takes a different perspective of scoping and boundaries identification (Avid, 2017)


Threat Modeling Steps 			
	Objective	Details	Q To A
1- System Modeling	Define the System : People Data Software Communications Hardware Physical	Define the target system to conduct threat modeling	
2-A System Scope	Full System Level Component	Choose the scope if it will be full system, specific level, or one component to define the blackbox created	What?
2-B Lifecycle Scope	Supply Chain Deployment Operation Maintenance	Define the phase of life cycle threat modeling will be conducted	When?
2-C Boundaries Scope	Data Flow	Create the black box around the selected scope and study the data flow	Where?
3- Threat Modeling Class	Class-H Class-M Class-L	Choose the level of threat modeling if it will be Principle, TTP, or In-Depth	Why?
4- Find Threats	ST ----> N ID ----> N-1 RE ----> N+1	Find the threats based on three levels of STRIDE (Spoof-Tamper), (Information disclosure- DOS) , and (Repudiation - EoP)	How?
5- Creating Threat Matrix	Threat Matrix	Based on all the previous findings to create the threat matrix diagram	

Table 1 Threat Modeling Summary

Step 1: System Modeling

In the proposed system modeling, there are three attack categories, which are domains, levels, and lifecycle. Due to the complexity of threat modeling analysis, it makes sense from the business owner's perspective to exclude certain components that are not essential from the study.

To comply with the previous logic of simplifying the process as much as possible, axis Y represents supply chains, implementation, operation, and maintenance, which enables us to address any threat during various phases before the implementation in the premises by looking to one diagram only, as it summarizes the system model and attack surface (refer to figure 2)

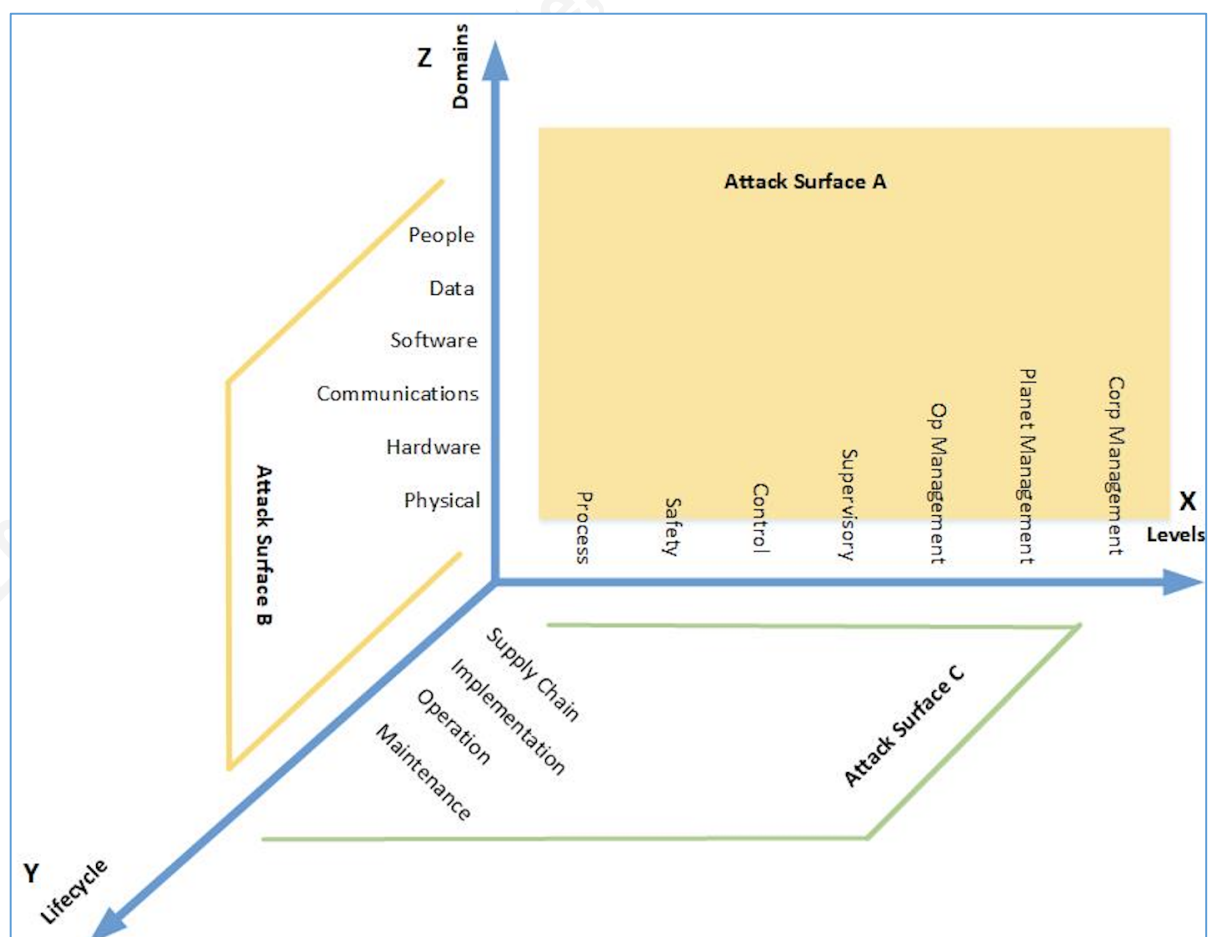


Figure 2 System Modeling & Attack surface

$$\{\text{Attack surface B}\} \cup \{\text{Attack Surface C}\} \cup \{\text{Attack Surface A}\} = \text{Threat Matrix}$$

On simpler expression

Mounir Kamal, mkamal@qcert.org

Using attack domains (Z) \cup During phase (Y) \cup Threat Targeting Level (X) = Threat Matrix

Step 2: Identifying scopes

2. A Business Scope

(Define the business scope we are dealing with, is it corporates management, planet management, operation management, supervisory, control, safety or process) Which is presented in axis X of the System Model- Figure 2.

2. B Lifecycle scope

(Define the scope of the lifecycle we are in, are we in supply chain, implementation, operation or maintenance phase) which is presented in axis Y of the System Model - Figure

2. C Boundaries scope

(Define the scope of interaction, is it “lower level” which indicates are we dealing with physical components or it is “higher-level” which indicates we are dealing with people, data, applications, operating systems, and communication) which is presented in axis Z of the System Model - Figure 2

Step 3: Threat Modeling Classes

Identifying the threat modeling classes:

- Component-oriented threat modeling level that focuses on exploiting specific vulnerabilities concerning certain CVE number, it is very comprehensive on the component level but cannot be applied on higher ones.
- Techniques oriented threat modeling level that can be used for defending against certain attack without mentioning specific CVE vulnerability, such as ATTACK model from MITRE which addresses the TTP using higher level techniques attempting to compromise different systems.
- Concepts oriented threat modeling level, which is the highest level as it operates on the level of the principle of information security threats without going deep into details, and only describes what the high-level threats will be, such as Microsoft STRIDE or MITRE CAPEC (Strom, et al., 2018)

As defined previously, there are three common threat modeling classes, and before the threat modeling process starts, the class of threat modeling should be determined, and the team must indicate which if the following classes to be used:

- Class-H Principle Threat Modeling (i.e., STRIDE, CAPEC MITRE)
- Class-M TTP Threat Modeling (i.e., ATTACK MITRE)
- Class-L In-depth Threat Modeling (i.e., contains all details on CVE level)

Step 4 Find Threats

One of the generic methods to evaluate threats is using STRIDE which is covered in detail in Appendix 6-B, but this time we will take the subject from a different approach. The following figure shows the idea of using STRIDE into three different segments ST, ID, and RE

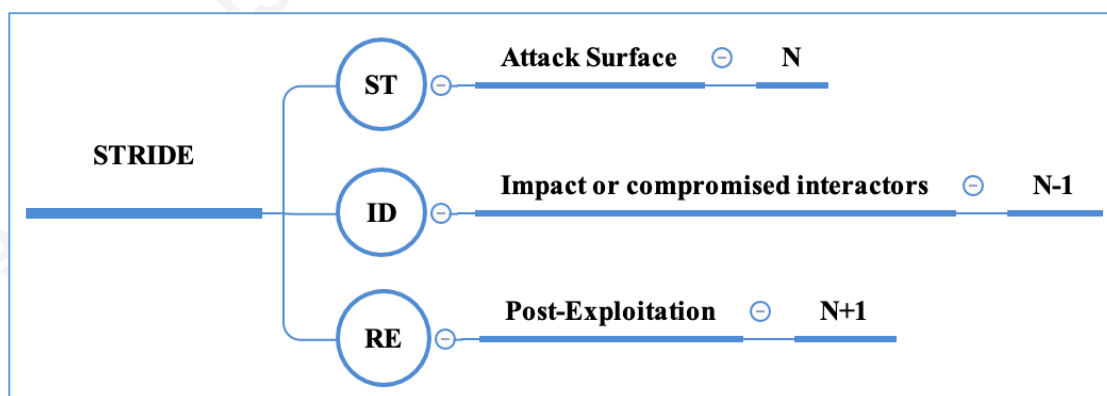


Figure 3 Find Threats Diagram

ST: presents the attack surface or initial access points of the targeted system/zone of the threat modeling which is at level (N) that are initiated by threat actors and not due to misconfiguration or vulnerability of the system. The attack is initiated by either bypassing authentication or tampering data in one of the three phases of processing the

data inside the memory modules, transmission packets over the network, or storing data on storage devices.

ID: presents the impact of the threats on the system, such as information disclosure or denial of service, and it could be any interaction around the targeted system (N-1) such as compromising credentials that enable the attacker to reach components in step (N)

RE: present the post-exploitation activities such as lateral movement, escalation of privileges, and evasion techniques including denying the responsibility of the performing the attack by clearing logging activities or evasion the presence. In such case, the attacker either going deeper to reach more critical zones of the primary target or moving to higher privileges systems/zones (N+1).

3.3.1. Step 5 Create the Threat Matrix

Proposed system modeling will be presented in a two-dimension matrix that contains potential threats for the entire system, level, or even one sub-system.

Using attack domains (Z) \cup during the lifecycle phase(Y) \cup Threat Targeting (System/Level/Components) = Threat Matrix

In the Blackbox approach:

- The black box concept used to divide the system target for threat modeling into three main process input, output, and the black box. And the boundaries of the black box to avoid go deep from more details which makes the threat modeling process impractical. (Bunge, 1963)
- There is one global “people” threat matrix covers the human factor at all phases from the supply chain until the end of the lifecycle. Also, it should exist in both internal and external arrays. The resultant people threat matrix that has been studied previously should merge with all the following threat matrices in the next phases.

- One specific array which is called internal array focuses on hardware, and physical components and requires internal access, any threats within such zone should use those domains.
- Many external arrays include the domains of attacks that can reach the zone remotely either by using data, software or network, any component within the zone is subject for threat modeling, even if it is assumed to be secure.

4. Example of the method implementation

Let us start with the first axis in the model which represents “where” including levels or zones of a typical example of the ICS system. Most of ICS planets will be similar to the following figure 4.

During the threat modeling, there are some assumptions which are:

- All created threat matrixes will be intentional threats and not accidental threats or due to a misconfiguration in the systems
- People threat matrix should be studied separately because it will be covering the entire lifecycle of the system
- Cyber Security Threat is available most of the time and everywhere, a vulnerable system that allows a threat to become a risk for specific assets.
- The threat could change the status of any system from secure to insecure

Step 1 System Modeling

ICS/SCADA system contains hundreds if not thousands of connected hardware, software, and applications that manage and control the physical system which is the core of the business.

As shown in figure 4 is a typical example of ICS/SCADA system contains all levels and domains, as shown in figure 2 system modeling. (NIST, 2015)

Step 2-A System Scope (Level-2 SCADA)

The system scope will focus on Level-2 SCADA system. Level-2 SCADA has been defined as one of the known samples of ICS/SCADA systems which is responsible for acting as an interface between the automation process level and plant management level. The main function of level-2 is monitoring the automation process, updating it, and coordinate the logging information from the automation process with the management, and business database for reporting and statistics information. (NIST, 2015)

Target ICS level-2 contains various components, zooming in the diagram of figure 4 and take only the required level-2 as the target for threat modeling

Step 2-B Lifecycle Scope (Level-2 SCADA)

During the case study, threat modeling will cover the entire life cycle from the supply chain until the maintenance phase.

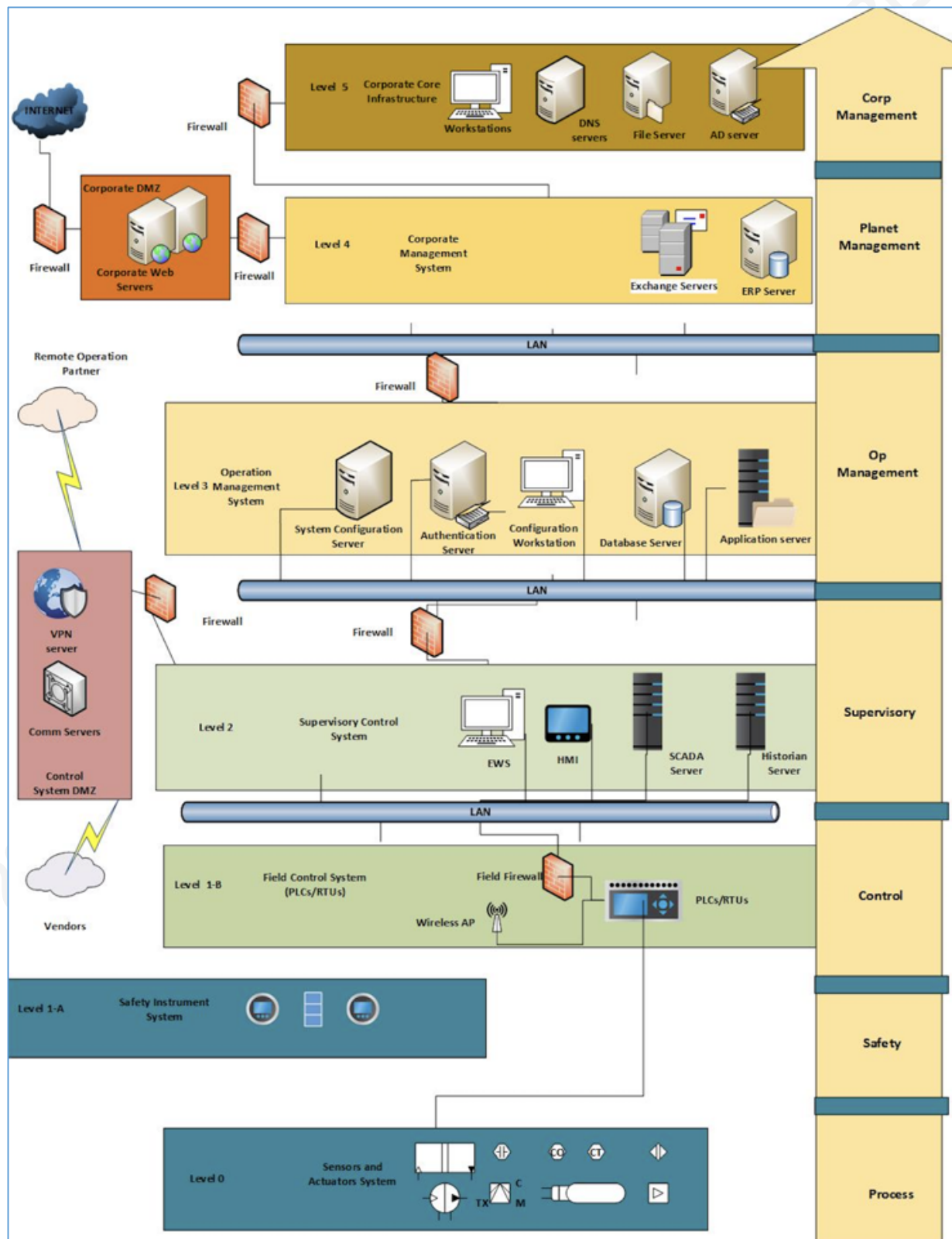


Figure 4 ICS/SCADA example system

Step 2-C Boundaries Scope (Level-2 SCADA)

Boundaries scope should be supported by data flow diagram containing all hosting services that require interface with any external party outside the scope considering all domains presented in axis-X as shown in Figure 5

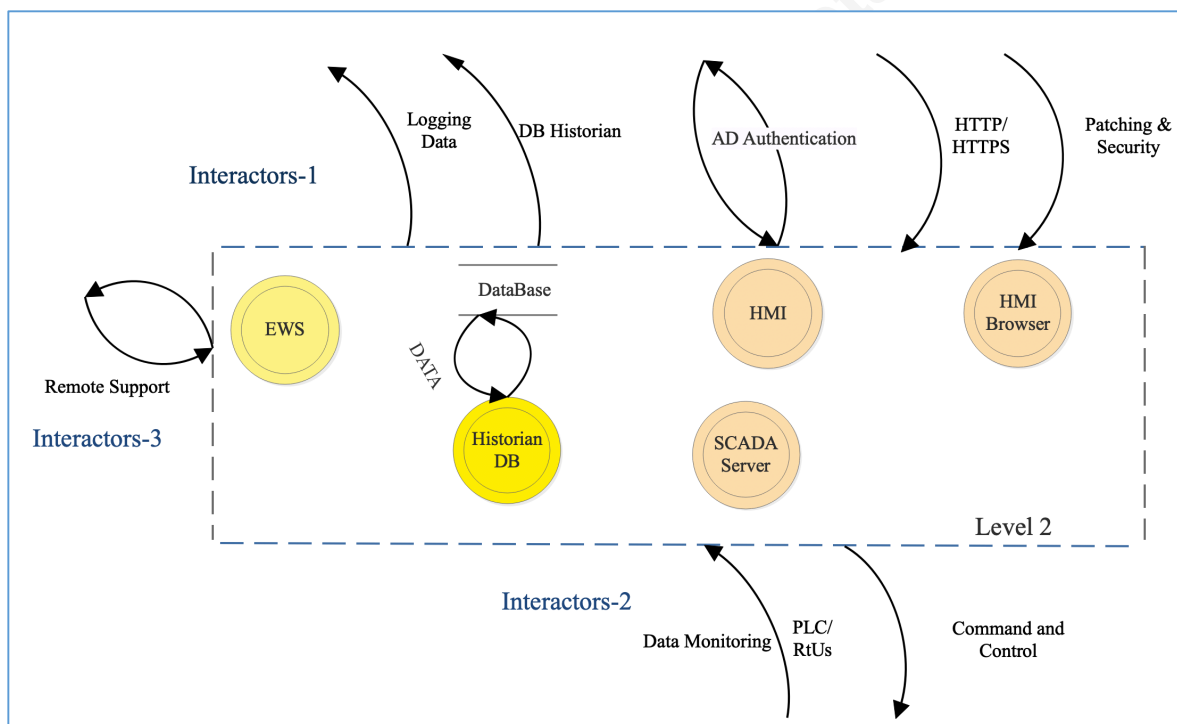


Figure 5 Level 2 Interactors Data flow Details

External Interactors

Interactors-1

There are four main interactors in this interface, the authentication server, business historian, logging services, and configuration server. Interactors contain global communications between hosts in level-2 SCADA and service of authentications, logging, configuration, and finally security servers. There is dedicated communication between Level -2 historian server and the upper business historian server in some cases web-interface of level-2 historian may be open for certain conditions.

In Interactors-1 the data flow between level-2 and level-3. At higher levels, some services require certain data, software, and protocols.

- Microsoft Active Directory authentication based on Kerberos protocols.
- Database service for replication the SCADA data historian servers with business historian
- Web HMI accessible from level-3 via HTTP/HTTPS Protocols.
- Security and patching function via servers in upper levels.
- Logging systems to collect logs of hosts an in level-2 to upper levels.

Interactors-2

Interactors-2 is the bottom interface in the lower-level which is the most critical level to control the operation interface, as it contains the core services network protocols to manage the system. Interactors-2 is the interface with the most critical infrastructure as it controls the automation of the process, most of the traffic comes from PLC/RTU which is responsible for reading the information from the sensors and commanding the actuators and relays which compiles the raw data to travel over specific protocols such as MODBUS, IEC 62443 series, and TCP/IP. Communication media between Level-1 and Level-2 could be LAN, WAN, or Wireless.

Interactors-3

Interactors-3 is the interface with a DMZ that is responsible for handling any remote VPN connection that might be required for third-party access. Most of the critical communication use Interactors-3 for remote technical support or upload data to a third-party entity. A landed server in the DMZ will be the interactor between remote access and target host in Level-2 subject of threat modeling. Interactors-3 is the interface for vendors and system integrators to perform remotely support; it allows access from trusted external third-party to drop into jump server and collect some new logs to perform remote support in case of emergencies.

Step 3 Threat Modeling Class (Level 2 SCADA)

As defined in the method details, there are three classes of threat modeling, and for this case, class-H will be used to simplify the process to study the strategic threats within the target zone.

Step 4 Find Threats (Level-2 SCADA)

Find threat from attack surface will focus only on ST as shown in figure 3, the study of initial access or entry point of spoofing and tampering will be the targeted threats in the case study.

Step 5 Create a Threat Matrix (Level-2 SCADA)

People threat matrix

*Using attack domains (People) \cup during the lifecycle phase (ALL) \cup Threat Targeting (Level-2)
= Threat Matrix*

People matrix will be applied to all phases and associated with all other domains.

Humans do all of the design, development, and many other functions over the full lifecycle of the system, therefore, it needs to study from the beginning until the end, as it is always the starting point behind any attack.

People domain could be used as threats to conduct attack differently in two areas of threats and precisely people initial access could be used either by spoofing to pretend to be someone else or using some social engineering tactics to collect and analyze information.

One of the best resources to study the cyber-attack is MITRE CAPEC that could use the human factor as the domain of attack and represent one of the most critical attacks surface entry point. It is difficult to be prevented due to lack of security awareness, and the impossibility to control human behavior to detect such an attack (MITRE, 2018)

The figure is a high-level model showing the principle of threat matrix using the human domain to initiate an attack entry point on a surface.

People Threat Matrix	
Spoofing	<u>Spoofing Customer Services</u>
	<u>Spoofing Technical Support</u>
	<u>Spoofing Delivery person</u>
	<u>Spoofing Via Phone</u>
	<u>Spear Phishing via Email</u>
	<u>Spear Phishing via social network</u>
Information Disclosure	<div> <div>Using Incentive</div> <div> Financial Social Ideological </div> </div>
	<div> <div>Influence perception</div> <div> Scarcity Authority Consensus or Social Proof Reciprocation Commitment and Consistency Liking </div> </div>
	<div> <div>Physiological Principle</div> <div> Modes of Thinking Eye Cues Micro-Expressions The Human Buffer Overflow Interview and Interrogation Instant Rapport </div> </div>

Figure 6 People Domain Threat Matrix

Supply Chain Phase Threat Matrix External Interactors

Using attack domains (Supply Chain) \cup during lifecycle phase (ALL) \cup Threat Targeting (Level-2) = Threat Matrix

- In supply chain phase there is no live data or configuration yet

Supply Chain Threat External		
Software	AD Authentication	T MS Platform have been tampered in design phase MS Platform have been tampered in development phase MS platform have been tampered during distribution
	Historian	S Spoofing Vendor certificate S Spoofing Vendor Website T Historian Software has been tampered in design T Historian Software has been tampered in development T Historian Software has been tampered during distribution
	Patch & Security	S Spoofing Vendor patch source T Malicious or malfunction patches and fixes
	Engineering Software	T EWS software has been tampered in design T EWS software has been tampered in development T EWS software has been tampered in distribution
	HMI	T HMI software has been tampered in design T HMI software has been tampered in development T HMI software has been tampered in distribution
	SCADA Server	T SCADA software has been tampered in Design T SCADA software has been tampered in Development T SCADA software has been tampered in Distribution
	Wireless	T Tamper firmware in Factory
	WAN	T Tamper firmware in Factory
	LAN	T Tamper firmware in Factory for switch , router, and firewall
	Protocols	T Exploit standard protocols within factory setting

Figure 7 Summary of Supply Chain threat matrix external interactors

Deployment phase Threat Matrix External Interactors

Using attack domains (Software, Communications) \cup During the lifecycle phase (Deployment)
 \cup Threat Targeting (Level-2) = Threat Matrix

Deployment phase includes the installation, configuration, integration, and testing.

Compromising the target system during the installation process requires some predefined access

points to specific interactors and defending breaches by the implementer that requires studying the matrix and defining the attack surface during the deployment phase.

In the deployment phase, most of the configurations are the default ones; the attacker may use this advantage to attack the target system in the proper timing and with the lowest cost. Deployment phase will make the attacker's mission easier.

To conduct a robust threat matrix, we need to determine all detailed processes of level-2 and apply a separate threat modeling study on each one of them. Detailed Deployment process list in Appendix C-1

Deployment Threat External			
Software	Configuration	T	Misconfiguration allow unauthorized access for any of interactors
	Web Apps	S	Default Credential could allow unauthorized access during deployment by interactors
	Apps	S	Spoofed interactors from outside the zone
	Op System	S	Historian
			Spoofed content of updates
			Spoofed source of certificate
			Spoofing Logging server
		T	Malicious source of OP Systems during deployment
			Malicious update source during deployment
			Malicious code deployed in joining to AD doamin controlers
			Malicious code deployed from security and patch server
			Bootkit deployed during installation
			Compromised deployment staff could compromised system during deployment
Communications	Firmware	S	Spoofing vendor Web site during deployment
		T	Malicious Firmware update during deployment
	Protocols	T	Malicious routing during deployment
	LAN	T	Malicious firmware deployed on LAN devices

Figure 8 Threat matrix deployment phase external interactors

Operation Threat Matrix External Interactors

Using attack domains (Software, Communications) \cup During the lifecycle phase (Operation) \cup Threat Targeting (Level-2) = Threat Matrix

The operation is the primary phase and ultimate goal of any system's lifecycle to fulfill the core business of the organization. Operation phase is the day to day processes running through the system to function — detailed operation processes of Level-2 SCADA systems in Appendix C-2. There is one common factor in the operation phase that could be the source of most of the threat, which is the coding. The code of an operating system, application, web platform, and even protocols of a network is an essential factor associated with any exploitation. In the operation phase, these security vulnerabilities can be accepted.

Operation Threat External	
DATA	Processing <ul style="list-style-type: none"> S Spoofed content could exploit software from any of external array T Tamper memory content to execute injected code in any of the hosts
	Transmission <ul style="list-style-type: none"> T Malformed request could exploit software Sending malicious code over network
	Storage <ul style="list-style-type: none"> T Legitimate call for malicious file on storage
Software	Operating System <ul style="list-style-type: none"> S <ul style="list-style-type: none"> Spoofed malicious group policy Compromised high privileges accounts outside level-2 Compromised high privileges computer outside level-2 Spoofed Internal DNS server Spoofing RDP/SSH from jump server from Array-3 T <ul style="list-style-type: none"> DNS poisoning Tamper routing table on any of the hosts Remote Execution/Powershell/WMI on one of the hosts Exploit MS windows unknown vulnerability in RPC/DCOM Exploit RDP/SSH protocols in MS windows from Array-3
	Security and updates <ul style="list-style-type: none"> S Spoofed content of fixes and updates T Tampered security patches / updates
	Logging <ul style="list-style-type: none"> S Spoof distention of logging server to collect logs
	Historian <ul style="list-style-type: none"> S <ul style="list-style-type: none"> Spoofed destination of Business historian in higher level Authentication bypass from business historian level-3 Authentication abuse from business historian level-3 T <ul style="list-style-type: none"> Remote Code Injection and execution from compromised interactors Remote Command Injection and execution from compromised interactors i.e SQL injection
	WEB Historian/HMI <ul style="list-style-type: none"> S <ul style="list-style-type: none"> spoofing client request from higher levels Authentication bypass from higher level-3 Authentication abuse from higher level-3 Server Side Include (SSI) Injection T <ul style="list-style-type: none"> Tamper Web content by Cross site scripting from compromised interactors Remote Code Injection and execution from compromised interactors Remote Command Injection and execution from compromised interactors i.e. SQL injection
	SCADA Server <ul style="list-style-type: none"> S <ul style="list-style-type: none"> Spoofing Controller in level-1 Authentication bypass from lower level-1 T Exploit Software by remote code execution from compromised devices level-1
	EWS <ul style="list-style-type: none"> S Spoofing project file content T Exploit EWS software by code injection
	HMI <ul style="list-style-type: none"> T Exploit HMI software by code injection
Communication	LAN <ul style="list-style-type: none"> S Spoof MAC address to bypass security control T <ul style="list-style-type: none"> Maliciously tamper LAN device configuration Rogue network device injected in the LAN
	Protocols <ul style="list-style-type: none"> S Spoof source IP address to bypass security control T Malicious routing due to compromised interactors

Figure 9 Operation phase threat matrix external interactors

Maintenance Threat Matrix External Interactors

Using attack domains (Software, Communications) \cup During the lifecycle phase (Maintenance)

\cup Threat Targeting (Level-2) = Threat Matrix

During the maintenance phase, there is some operation process that may require a higher level of integrity of the automation process. The maintenance phase may include upgrading or fixing the system, the network, or the software. A detailed sample of the process in Appendix C-3.








Maintenance Threat External 	
Software	S  Spoofed Vendor website content used for upgrade Malicious spoofed content of the new software including Apps, OS, and firmware/
	T  Intentionally malicious changes of configuration Malicious compromised workstation used by vendor or system integrator Tamper software during remote support by compromised vendor or system integrator
Communications	Protocols  T  Malicious routing during maintenance
	LAN  T  Deploy system on LAN for the purpose of sniffing traffic

Figure 10 Maintenance phase threat matrix external interactors

Level-2 SCADA Threat Matrix Internal

Hardware and physical components are the most static elements in the domain, and any changes with them will always be associated with the human behavior who interact directly with the assets whether it is intentionally or unintentionally. The created internal threat matrix for both hardware and physical domains could be applied to all other levels in an ICS system. A complete Threat Matrices of Level-2 SCADA Threat Matrix Internal in Appendix D

5. Conclusion

Imagine looking through a magnifying glass to see the details of a complicated system such as ICS which has been layered in three-dimension axes. By moving the magnifying glass to a 3D mode, right, left, up and down, and then zooming on a specific system, zone, level, or other subsystems. At each movement, you will observe a different angle. The goal is to conduct a threat modeling for the target system or subsystem to define the seven steps of the proposed threat modeling method.

During the study of threat modeling, there is a need to build a system pilot that will help and prepare the system to achieve the target of creating easy and systematic threat modeling. As ICS systems are complex and it is not easy to conduct a comprehensive threat modeling, we find the value of the Blackbox concept as it helps to reduce studying an infinite combination of attacks and only focuses on the attack surface that might be reachable in any targeted zone. Threat modeling is very flexible and helpful in many classes, three classes of threat modeling could be chosen based on the objective of the threat, and determine if it is high medium, or low. Determining the class of a threat is based on the target which could be the full architecture, specific services, or one system (component). Consequently, we will be able to determine the used method, whether we need to change principle, tactics, or technical procedures. Finding threats will focus on attack surface (initial access) based on the Blackbox and STRIDE principle to choose only S (spoofing) and T (Tampering) which will be applicable on targeting or non-targeting attacks. Before going through threat modeling configuration, we will be required to define many factors based on the threat modeling objective. Threat modeling configuration should contain the system model, scope and lifecycle to build Blackbox. By defining the level of threat, we will be able to create the threat matrix finally.

The internal team can manage many threats in the operation environment of ICS systems. However, more complicated procedures require special skills and capabilities.

People threats and especially inside intruders present a real challenge and require a special understanding of concepts, operations, and technologies. Security awareness will support a secure operating environment by protecting the working environment and the flow of information within the team working inside the planet.

Supply chains threats are something challenging to recognize and should be solved with collaborative work, and with a great vetting lab capability that can support an organization to detect such threats. With some dedicated effort, many threats can be managed by the planet's team by screening attack surface breaches and conducting proper consistency of strategy on the tactics, and components levels.

Threat modeling is not a procedural process that consists a defined step anyone can start and end to reach a result, but it has a changeable nature. Using the previous table-1 helps to identify the system modeling by representing the context, which will answer the question of “WHAT” and this identifies the subject of the threat modeling — then limiting the scope to determine if the weakness is in the automation process, systems, sub-systems, or components from the vendor, network protocols, or even single transaction. The next step will be drawing the boundaries as a Blackbox to answer the “WHERE” question which will help to recognize the interactors including the communications, people, hardware, supply chain, and physical components. Sometimes the scope may focus only on one phase of the system lifecycle to define the “WHEN” question in the threat modeling, which will lead us to the “WHO” is conducting each step. Based on the objectives of the threat modeling levels, threat modeling will explain

“WHY” we use a particular level of threat modeling. And define “HOW” security professionals will determine the depth of each level and finally, create the required threat matrix.

6. Bibliography

- Avid. (2017, June). *Black Box Threat Modeling*. Retrieved from <https://www.peerlyst.com/posts/bsidestlv-2017-black-box-threat-modeling-avid>
- Bunge, M. (1963). A general black-box theory. *Philosophy of Science, Vol 30, No4,,* 346-358.
- Chris Salter, O. Sami Saydjari, Bruce Schneier, & Jim Wallner. (1998). *Toward A Secure System Engineering Methodology*.
- Consortium, I. I. (2017). *Architecture Alignment and Interoperability An Industrial Internet Consortium and Plattform Industrie 4.0 Joint Whitepaper*.
- Dazhuang, H., Lobov, A., & Moctezumas, L.E.G. a. (2012). IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society. *An approach to use PERA in Enterprise Modeling for industrial systems*.
- Dictionary, C. (n.d.). *Cambridge Dictionary*. Retrieved from <https://dictionary.cambridge.org/dictionary/english/threat>
- Gervais, A. (2012). *Security Analysis of Industrial Control Systems*.
- Gultom, R. (2018). Introducing the Six-Ware Cyber Security Framework Concept to Enhancing Cyber Security Environment. *International Conference on Cyber Warfare and Security, Academic Conferences International Limited*, (p. 262).
- Jackson Wynn, Joseph Whitmore, Geoff Upton, Lindsay Spriggs, Dan McKinnon, Richard McInnes, . . . Lauren Clausen. (2011). *MITRE Threat Assessment and Remediation Analysis (TARA)*.
- Loren Kohnfelde, & Praerit Garg. (1999). *The threats to our products*.
- Mitre. (2018). *CAPEC Common Attack Pattern Enumeration and Classification*. Retrieved from List of Attack Pattern: <http://capec.mitre.org/index.html>
- MITRE. (2018, July). *CAPEC-403: Social Engineering*. Retrieved from CAPEC: <https://capec.mitre.org/data/definitions/403.html>
- NIST, N. I. (2015, May). *NIST SP 800-82 Rev 2, Guide to Industrial Control Systems (ICS) Security*. Retrieved from <http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-82r2.pdf>.
- Shostack, A. (2014). *Threat Modeling Designing for Security*. John Wiley and Sons, Inc.
- Smart Grid Coordination Group, C.-C.-E. (2012). *Smart Grid Reference Architecture*.
- Strom, B., Applebaum, A., Miller, D., Nickels, K., Pennington, A., & Thomas, C. (2018). *MITRE ATT&CKTM: Design and Philosophy*.
- Tony Ucedavelez, & Marco M. Morana. (2015). *Risk Centric Threat Modeling*. John Wiley & Sons.
- Tyson Macaulay, & Bryan Singer. (2011). *Cybersecurity for Industrial Control Systems: SCADA, DCS, PLC, HMI, and SIS*. Auerbach Publications.
- ZACHMAN, J. A. (2008). *The Concise Definition of The Zachman Framework by: John A. Zachman*. Retrieved from ZachmanInternational: <https://www.zachman.com/about-the-zachman-framework>

7. Appendices

7.1. Appendix A: ICS System Modeling

7.1.1. Purdue Enterprise Reference Architecture Model (PERA)

PERA is an abbreviation for (Purdue Enterprise Reference Architecture) which describes all elements of enterprise engineering and integration and to compile such a model to an Industrial control system (ICS). PERA defines the network architecture that manages assets such as (programmable logic controllers [PLCs], historians, servers...etc.) in separate levels to achieve an adequate response, resolution, reparability. And consequently security. In the PERA system modeling every step toward higher to the business level. Information security goals will change from availability of the system to the system integrity and confidentiality as shown in (Dazhuang, Lobov, A., & Moctezumas, L.E.G. a, 2012) (Tyson Macaulay & Bryan Singer, 2011)

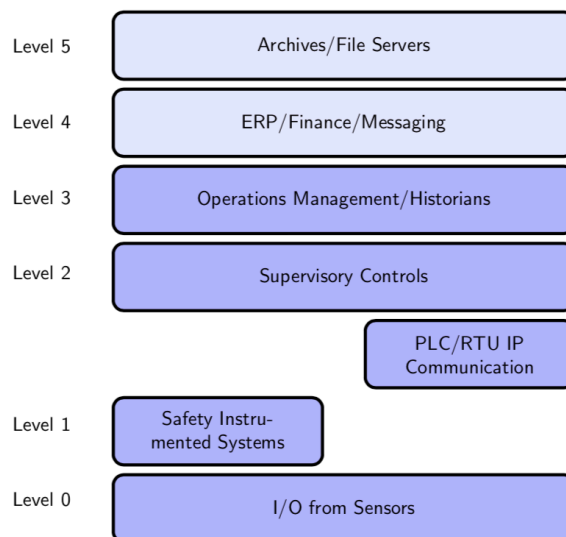


Figure 11 PERA Model

Why PERA Model has six levels need to be studied from the security perspective. PERA model is a purely functional compilation of most primary ICS systems; each level may correspond to a Security Zone that has some

boundaries of System, Network, and Application security controls. Also, PERA clearly defines which type of protocols that travel within and between different levels which makes it easier to normalize the behavior of all mentioned components of System, Network, and Applications. PERA is suitable for small ICS system or could be applied in repeated units of ICS systems.

Limitations of PERA are generic, high-level model, and not detailed. It will be difficult to rely on PERA to conduct a complete process modeling of ICS systems

7.1.2. Smart Grid Architecture Model (SGAM)

Smart Grid Architecture Model (SGAM) framework has been developed for the smart grid that allows the validation of smart grid use cases. The SGAM framework is established by merging the concept of interoperability layers with domain and zone, as shown in the figure adopted from the source (Smart Grid Coordination Group, 2012)

Interoperability comes for clear presentation and simplicity of the architecture model. Therefore, it comes with five primary layers Business, Function, Information, Communication, and Components as follow adopted from (Smart Grid Coordination Group, 2012)

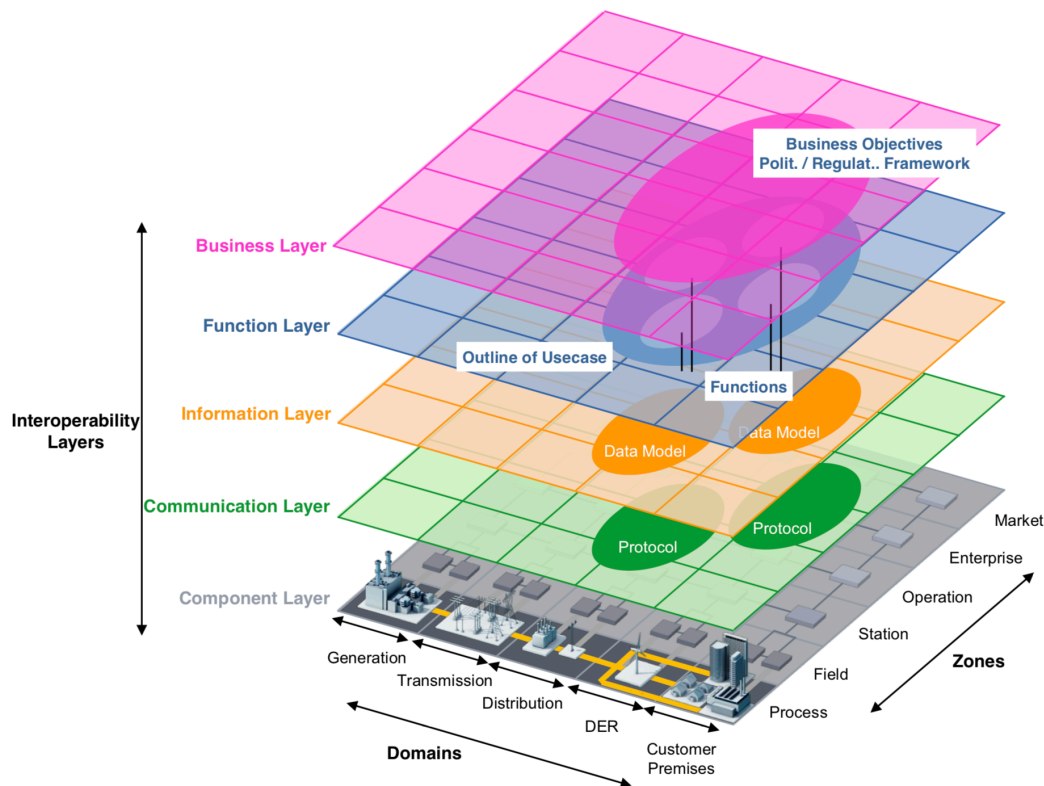


Figure 12 SGAM Model

SGAM Interoperability comes for clear presentation and simplicity of the architecture model. It comes with five primary layers Business, Function, Information, Communication, and Components

SGAM Domains comes from the Electricity life cycle chain starting from Generation, Transmission, Distribution, DER (Distributed Energy Resource), and finally to Customers.

SGAM Zones which is close or drives from PERA (Purdue Enterprise Reference Architecture Model) Model with minor differences

SGAM model is adapted for Smart Grid systems, governing the business layer to integrate the automation process with the business requirement with linked to all marketing and economics part of the production. SGAM is the integration of other frameworks like TOGAF, PERA model, the chain of an end to end electricity generation to use.

Limitations of SGAM framework has been developed and customized to fit with the smart grid enterprise architecture. (Smart Grid Coordination Group, 2012)

Mounir Kamal, mkamal@qcert.org

7.1.3. RAMI 4.0 (Reference Architecture Model for Industry 4.0)

RAMI 4.0 is a three-dimensional coordinate architecture which allows decomposition of complex systems into simple subsystems that could be studied more straightforwardly as shown in the figure adopted from (Consortium, 2017)

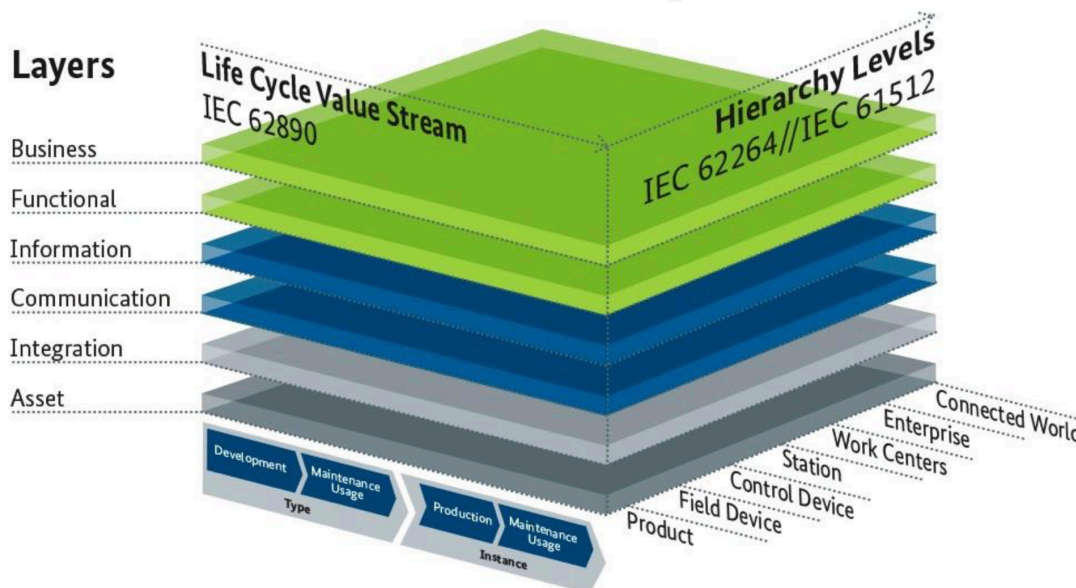


Figure 13 RAMI 4.0

The three-dimensional coordinate architecture consists of three axis's Layers, Hierarchy, and life cycle & value stream.

Layers Axis consists of six layers related to the definition of what is the layer present from business, function, information, communication, integration, and Asset. Layers axis is very close to what have studied before in SGAM, and TOGAF Enterprise Architecture with a small modification of dedication of integration layer in RAMI4.0 and keep the Asset instead of the components in the previous models.

Hierarchy levels are the IEC62264/IEC61512 international standards series from control systems representing the zones of OT (Operation Technology) from field devices, control devices, station, work centers, enterprise, and finally connected world which somehow aligned with both PERA and SGAM except the last zone of the connected world.

Life Cycle Value Stream based on IEC 62890 for lifecycle management in which there are two levels of lifecycle one at the vendors while design, develop a product and the other at Customers in implementation, operation, and maintenance phases.

RAMI 4.0 is very close to SGAM but because SGAM is smart grid focus the timeline factor exist on the domains access focus on the electricity. Which the processing starts from generation, transmission..., till customers while in RAMI 4.0 focus on the product itself used for automation from development in the factory premises till reach on the customer's production which represents some areas were not defined in SGAM. Also, in RAMI 4.0 integration is described in a separate layer of actors which highlight the importance of integration between different vendors.

7.2. Appendix B: Threat Modeling Methods

7.2.1. STRIDE

STRIDE is an abbreviation of most common six threats categories Spoofing, Tampering, Repudiation, Information disclosure, Denial of services, Escalation of privileges. Loren Kohnfelder and Praerit Garg have invented STRIDE threat modeling ((Loren Kohnfelde & Praerit Garg, 1999). This framework and mnemonic were designed to help people developing software to identify the types of attacks that software tends to experience. (Shostack, 2014). The classification of STRIDE makes most of the threat categories let go for everyone and study it in details

STRIDE can be summarized into the following main items:

- 1- Create an architecture overview: during these steps Identify what the application does, create an architecture diagram, Identify the technologies used.
- 2- Creating a system model which help in analyzing and decomposing a system from most of its components.
- 3- Creating DFD (Data Flow Diagram) showing the Trust Boundaries, Data Flow, Entry points, and Privileged code.
- 4- Create Attack library with STRIDE: One of the strongest attack libraries to be used as a source of attack library is CAPEC and ATT&CK frameworks from MITRE.

Tools of implementation: STRIDE was the framework that Microsoft has built Microsoft Threat Modeling

Tools as shown in the following link which automates the full process of threat modeling

[Getting started with the Threat Modeling Tool](#)

STRIDE has the advantage of being Systematic and Conceptual, so it is very clear when it comes to the principle of threats and covering most of the known threats, STRIDE also has the tools that could inspect any system in the graphical interface and easy to use. From another point of view, STRIDE has issues with a sequence of threats and difficulties to categorize each real-life threat into the six categories of STRIDE. With STRIDE you have to categorize Threats Entry point and real use of threats or impact over the system and not in flat. For example, Attacker has obtained User-name and password, if the password is never used, there are no threats, but when the attacker uses it to access mailbox, then it will be combined with information disclosure, and if it uses in-memory injection which is tampering memory content after Escalation of Privileges to extract the administrator credential it will become so complex. This case is an example on one host only, what about huge infrastructure with thousands of applications.

7.2.2. Attack Tree

Attack tree Methodology uses logical diagram showing all possible attacks that could affect a system using a basic logical symbol starts from the attacker objective and ends by leaves that represent all possible threats.

"Attack trees provide a formal, methodical way of describing the security of systems, based on varying attacks. You represent attacks against a system in a tree structure, with the goal as the root node and different ways of achieving that goal as leaf nodes" (Chris Salter, O. Sami Saydjari, Bruce Schneier, & Jim Wallner, 1998). The paper showing the steps required to create an attack tree by

- Creating the root node which defines the attacker goals.
- Using OR symbol to define most of the scenario that may be used by the attacker.
- A node represents by “AND” and “OR” relations. The paper suggests assigning values or labels to the nodes.
- Suggests a leaf node gives values or labels for each node, like probability, Cost, Difficulty, and Legality for doing this task.

Attack Tree Method assumes security analyst thinks like an attacker and the first thing attacker decides is to define the target and goals of attack. Each goal has a separate attack tree. Starting each attack tree by a goal which

Mounir Kamal, mkamal@qcert.org

represents a root node in attack tree. List all possible scenarios to achieve your goal will be defined as sub goals and so a set of sub-node under the root node. Every sub-node evaluated for the possibility of success, Cost of Attack, and particular requirement. Using OR and AND as a conditional symbol OR used to represent alternative ways to achieve your node or sub-node, and AND describes different steps to achieve the same node or sub-node. In AND case, an attacker cannot achieve the goal till all ANDed sub-goal are satisfied. By the end of the tree, for each scenario, there are endpoints which represent tree leaf. For each scenario, a group of leaves has to satisfy the established goal. (Chris Salter, O. Sami Saydjari, Bruce Schneier, & Jim Wallner, 1998). The following diagram shows the process of creation of Attack Tree Adopted from (Gervais, 2012)

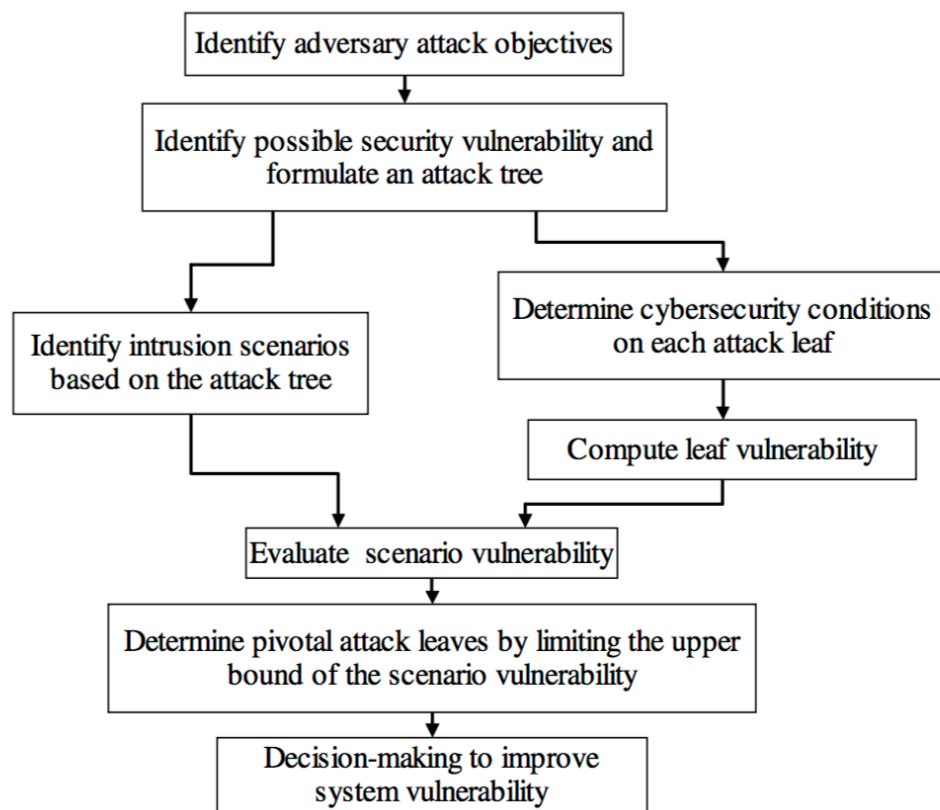


Figure 14 Attack Tree

Thinking like an attacker browse all possible ways to achieve the root goals and leads to some ideas about how to work around security controls that exist in a specific system. An additional cost of attacks that helps in defining the cost of security controls may be required to avoid such threat in risk mitigation strategy.

7.2.3. PASTA

PASTA is a process for Attack Simulation and Threat Analysis. It is a RISK based Threat Modeling Methodology. PASTA starts with defining an object, Tech Scope, App Decomposition, Threat Analysis, Vulnerability Detection, Attack Enumeration, and Risk/Impact Analysis. (Tony Ucedavelez & Marco M. Morana, 2015)

7.2.4. CTSA


CTSA is Cyber Threat Susceptibility Analysis. Part of TARA (Threat Assessment and Remediation Analysis) developed by MITRE. CSTA in combination with CRRA Cyber Risk Remediation Analysis to complete the TARA process.

CTSA objectives are to create a threat matrix of Tactics, Techniques, and Procedures (TTP) that the adversary can apply against systems. CSTA consists of the following steps:


- **Establish assessment scope** define the set of system assets to be evaluated, the type of attack TTP to study, and types of adversaries.
- **Identify candidate TTP** to evaluate the security architecture and its capabilities against TTP. Creating TTP catalog may include different sources such as MITRE CAPEC, ATT&CK, CVE and CWE
- **Eliminate Implausible TTP** by studying the created TTP and eliminating TTP considered implausible or prerequisites to conduct special types of TTP such as exploit MS windows if the operating system of the host is Linux, on other words the inapplicability of TTP against the specific system.
- **Applying scoring model** Ranking of TTP within CTSA methodology have standards TTP scoring spreadsheet that defines 12 factors with a standard range of values from 1 to 5 example of scoring model adopted from the factor range cover Proximity, Locality, Recovery Time, Restoration Costs, Impact on Confidentiality, Integrity, and Availability, Prior Use, Required Skills, Required Resources, Stealth, and Attributions.
- **Constructing the threat matrix** is the final phase of CTSA study by producing a Threat Matrix defines the score, target assets, and adversary type. (Jackson Wynn, et al., 2011)

7.3 Appendix C: Process Details


7.3.1 Appendix C-1 Level 2 SCADA Deployment Process

Deployment steps 				
	Main	Installing	Configuration	Patching and enhancement
Step 1	Prepare the zone from network area	Firewalls Switches	Define open port Define VLAN	Configure Comm Matrix Security setting
Step 2	Deploy Hosts operating systems	Deployment from Media	Configure OS Network configuration Join to AD domain	Patching and updates by configuration Configuring logging procedure Documentation of configuration
Step 3	Deploy Applications	Historian Server HMI EWS software SCADA Server	Applications setting	Deploy latest updates and patch

7.3.2 Appendix C-2 Level 2 SCADA Operation Process

Operation Process 				
	Systems Used	Interactors (mandatory)	Supportive Interactors	Interactors (Optional)
Monitoring	HMI	Level 1 Controllers	AD Patching Logging Server Configuration Server	Higher Level Monitoring Remote Support from array-2
Engineering of Automation	EWS	SCADA servers PLC/RTUs	AD Patching Logging Server Configuration Server	
Logging of Data in all levels	Historian	Higher level Historian SCADA Server HMI	AD Patching Logging Serevr Configuration server	Remote support from Array-2
Communications with Controllers	SCADA Server	PLC/RTUs ALL Level 2 Systems		

7.3.3 Appendix C-3 Level 2 SCADA Maintenance Process

Maintenance Processes 				
	Process Details	Systems Used	Interactors (mandatory)	Supportive Interactors
Upgrading	Upgrading Software Upgrading Network	HMI EWS Historian SCADA Server	Level 1 Controllers	AD Patching Logging Server Configuration Server
Support and fixing	Supporting Software Supporting Network	HMI EWS Historian SCADA Server	Higher level Historian SCADA Server HMI	AD Patching Logging Serevr Configuration server

7.4 Appendix D: Level-2 SCADA Threat Matrix Internal

Supply chain phase Threat Matrix Internal

Using attack domains (Hardware, Physical) \cup during the lifecycle phase (Supply Chain) \cup

Threat Targeting (Level-2) = Threat Matrix

Supply chain Threat Internal	
Hardware	Computers or Others
	<div> <div>S</div> <div>Hardware counterfeiting</div> </div> <div> <div>T</div> <div> <div>Tamper hardware in manufacturing</div> <div>Tamper hardware Design</div> <div>Tamper hardware in distribution</div> </div> </div>
	Media
	<div> <div>S</div> <div>Spoofing vendor media</div> </div> <div> <div>T</div> <div>Malicious or Bad Media USB, Media storage</div> </div>
	Cabels
Physical	Zones
	<div> <div>T</div> <div> <div>Attack physical facility in Distribution</div> <div>Attack physical facility in manufacturing</div> </div> </div>

Figure 15 Threat Matrix Level 2 Supply chain Internal (Mitre, 2018)

Deployment phase Threat Matrix Internal

Using attack domains (Hardware, Physical) \cup During the lifecycle phase (Deployment) \cup

Threat Targeting (Level-2) = Threat Matrix

Deployment Threat Internal	
Hardware	Computers or Others ⊖ T ⊖ Hardware module could allow unauthorized access System Integrator tamper hardware of other systems
	Media ⊖ S ⊖ Spoofing vendor media of installation T ⊖ Malicious Media storage used during installation
	Cabels ⊖ T ⊖ Bypassing Security Controls ⊖ Routers Firewall Switches
Physical	Zones ⊖ S ⊖ System Integrator bypass physical saecurity System Integrator has unauthorized access to different security zone
	Systems ⊖ S ⊖ System Integrator keep access after deployment

Figure 16 Threat matrix internal deployments phase

Operation phase Threat Matrix Internal

Using attack domains (Hardware, Physical) \cup during the lifecycle phase (Operation) \cup Threat

Targeting (Level-2) = Threat Matrix

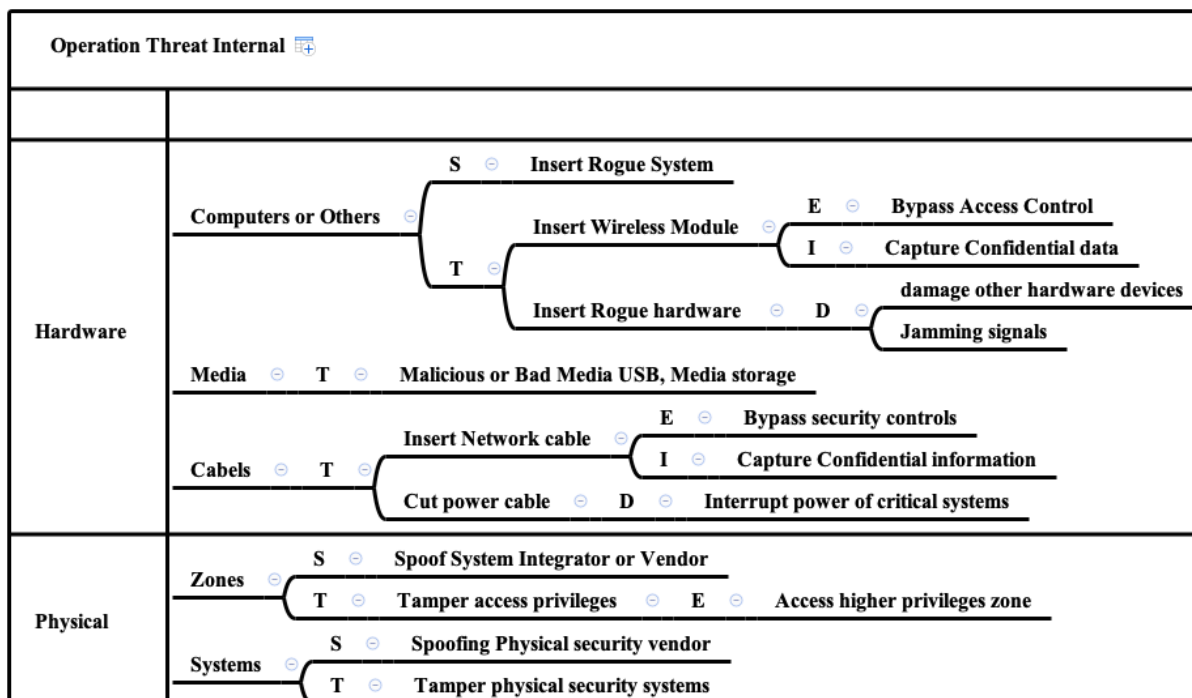


Figure 17 Threat matrix internal Operation phase

Maintenance phase Threat Matrix Internal

Using attack domains (Hardware, Physical) \cup during the lifecycle phase (Maintenance) \cup

Threat Targeting (Level-2) = Threat Matrix

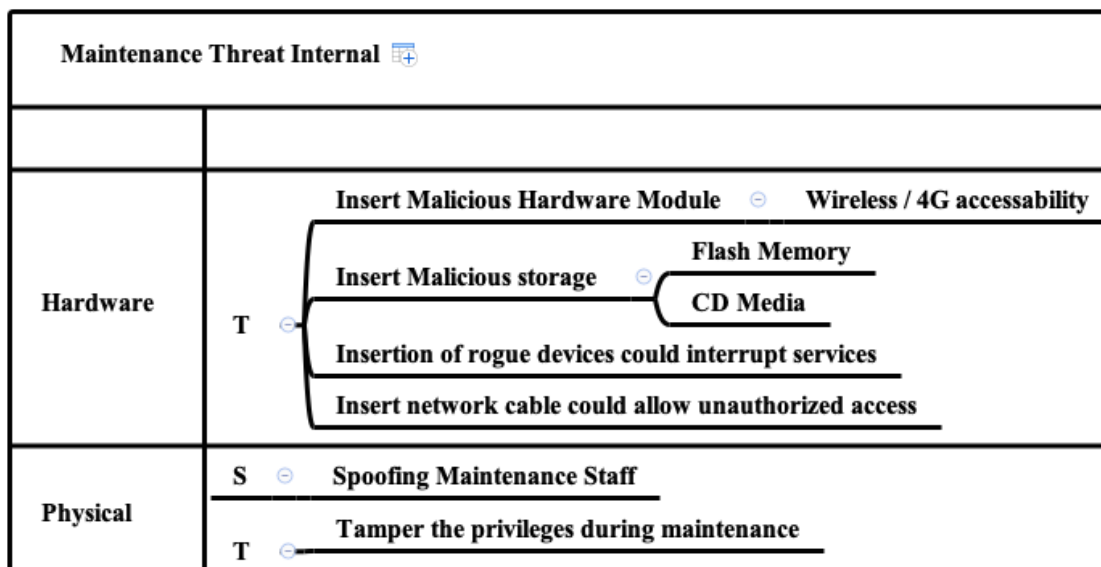


Figure 18 Threat matrix internal Maintenance phase