# Modeling Threats to Cyber Physical Systems: A Hypergraph Based Approach

*Thesis submitted in partial fulfillment of the requirements*
*for the degree of*
**Master of Computer Science and Engineering**
of
Computer Science and Engineering Department
of
Jadavpur University

by

**Islaur Rahaman Ruku**
**Regn. No. - 163571 of 2022-2023**
**Exam Roll No. - M4CSE24015**

*under the supervision of*

**Dr. Mridul Sankar Barik**
Assistant Professor

Department of Computer Science and Engineering
Faculty of Engineering and Technology
Jadavpur University
2024

# Certificate from the Supervisor

This is to certify that the work embodied in this thesis entitled **"Modeling Threats to Cyber Physical Systems: A Hypergraph Based Approach"** has been satisfactorily completed by **Islaur Rahaman Ruku** (Registration Number 163571 of $2022-23$; Class Roll No. 002210502020; Examination Roll No. $M4CSE24015$). It is a bona-fide piece of work carried out under my supervision and guidance at Jadavpur University, Kolkata for partial fulfilment of the requirements for the awarding of the **Master of Engineering in Computer Science and Engineering** degree of the Department of Computer Science and Engineering, Faculty of Engineering and Technology, Jadavpur University, during the academic year $2023-24$.

---

**Dr. Mridul Sankar Barik**,
Assistant Professor,
Department of Computer Science and Engineering,
Jadavpur University.
**(Supervisor)**

Forwarded By:

---

**Prof. Nandini Mukhopadhyay**,
Head,
Department of Computer Science and Engineering,
Jadavpur University.

---

**Prof. Dipak Laha**,
DEAN,
Faculty of Engineering & Technology,
Jadavpur University.

**Department of Computer Science and Engineering**
**Faculty of Engineering And Technology**
**Jadavpur University, Kolkata - 700 032**

# Certificate of Approval

This is to certify that the thesis entitled **"Modeling Threats to Cyber Physical Systems: A Hypergraph Based Approach"** is a bona-fide record of work carried out by **Islaur Rahaman Ruku** (Registration Number 163571 of $2022-23$; Class Roll No. 002210502020; Examination Roll No. $M4CSE24015$) in partial fulfilment of the requirements for the award of the degree of **Master of Engineering in Computer Science and Engineering** in the **Department of Computer Science and Engineering, Jadavpur University**, during the period of July 2023 to May 2024. It is understood that by this approval, the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein but approve the thesis only for the purpose of which it has been submitted.

**Examiners:**

(Signature of The Examiner)

(Signature of The Supervisor)

**Department of Computer Science and Engineering**
**Faculty of Engineering And Technology**
**Jadavpur University, Kolkata - 700 032**

# Declaration of Originality and Compliance of Academic Ethics

I hereby declare that the thesis entitled **"Modeling Threats to Cyber Physical Systems: A Hypergraph Based Approach"** contains literature survey and original research work by the undersigned candidate, as a part of his degree of **Master of Engineering in Computer Science and Engineering** in the **Department of Computer Science and Engineering, Jadavpur University**. All information have been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

**Name:** Islaur Rahaman Ruku

**Examination Roll No.:** M4CSE24015

**Registration No.:** 163571 of $2022 - 23$

**Thesis Title:** Modeling Threats to Cyber Physical Systems: A Hypergraph Based Approach

**Signature of the Candidate:**

# Acknowledgement

I am pleased to express my gratitude and regards towards my Project Guide **Dr. Mridul Sankar Barik**, Assistant Professor, Department of Computer Science and Engineering, Jadavpur University, without whose valuable guidance, inspiration and attention towards me, pursuing my project would have been impossible.

Last but not the least, I express my regards towards my friends and family for bearing with me and for being a source of constant motivation during the entire term of the work.

**Islaur Rahaman Ruku**
MCSE Final Year
Exam Roll No. - M4CSE24015
Regn. No. - 163571 of 2022 – 23
Department of Computer Science and Engineering,
Jadavpur University.

# Contents

**Certificate from Supervisor** **I**

**Certificate of Approval** **II**

**Declaration of Originality** **III**

**Acknowledgement** **IV**

**1 Introduction** **1**
1.1 Background . . . . . 1
1.2 Research Statement . . . . . 2
1.3 Contribution of the Thesis . . . . . 2
1.4 Outline of the Thesis . . . . . 2

**2 Hypergraphs and Hypergraph Based Threat Models** **3**
2.1 Introduction . . . . . 3
2.2 Hypergraph Variations . . . . . 4
2.3 Various Hypergraph Definitions . . . . . 5
2.4 Types of Hypergraph . . . . . 5
2.4.1 Undirected Hypergraph . . . . . 5
2.4.2 Directed Hypergraph . . . . . 6
2.5 Few Transformations of Hypergraph . . . . . 7
2.5.1 Line Graph of a Hypergraph . . . . . 7
2.5.2 2-Section Graph of a Hypergraph . . . . . 8
2.5.3 Bipartite Graph of a Hypergraph . . . . . 8
2.6 Applications of Hypergraph . . . . . 9
2.7 Hypergraph Based Threat Models . . . . . 10
2.7.1 Overview . . . . . 10
2.7.2 A Hypergraph-Based Machine Learning Ensemble Network Intrusion Detection System . . . . . 10
2.7.3 A Mathematical Framework to Assess the Security of an Information Infrastructure . . . . . 11
2.7.4 Fusing Hypergraph Spectral Features for Shilling Attack Detection . . . . . 12
2.7.5 High Performance Hypergraph Analytics of Domain Name System Relationships 13
2.7.6 Hypergraph Clustering for Better Network Traffic Inspection . . . . . 14

2.7.7 Hypergraph Clustering Model-Based Association Analysis of DDoS Attacks in Fog Computing Intrusion Detection System . . . 14
2.7.8 Hypergraph-Driven Mitigation of Cyber-Attacks . . . 15
2.7.9 Intrusion Detection with Hypergraph-Based Attack Models . . . 16
2.7.10 Ranks and Partial Cuts in Forward Hypergraph . . . 16
2.7.11 Research and Construction Model of Cyberspace Based on Hypergraph . . . 17

**3 CPS and Security Threats** **19**
3.1 Cyber Physical System . . . 19
3.2 Overview . . . 19
3.3 History of CPS . . . 20
3.4 CPS Components . . . 20
3.4.1 Several Layers of CPS . . . 20
3.4.2 5C Architecture of CPS . . . 22
3.5 CPS Characteristics . . . 24
3.6 CPS Security Issues . . . 26
3.6.1 CPS Vulnerabilites . . . 26
3.6.2 CPS Threats and Attacks . . . 29
3.7 Incidences of Past Attacks on CPS . . . 31
3.7.1 Stuxnet . . . 31
3.7.2 RQ-170 . . . 31
3.7.3 Ukraine Attack . . . 32
3.7.4 Maroochy Attack . . . 32
3.7.5 Jeep Hack . . . 32

**4 Threat Modeling of CPS** **33**
4.1 Threat Modeling . . . 33
4.1.1 Advantages of Threat Modeling . . . 33
4.2 CPS Threat Modeling Methodologies . . . 34
4.2.1 STRIDE . . . 34
4.2.2 PASTA . . . 35
4.2.3 TRIKE . . . 36
4.2.4 DREAD . . . 36
4.2.5 OCATVE . . . 37
4.2.6 LINDDUN . . . 37
4.2.7 CVSS . . . 38
4.2.8 Attack Trees . . . 39
4.2.9 Persona Non Grata (PnG) . . . 39
4.2.10 Security Cards . . . 39
4.2.11 Hybrid Threat Modeling Method (hTMM) . . . 40
4.2.12 Quantitive TMM . . . 40
4.2.13 VAST Modeling . . . 40
4.3 Existing CPS Threat Modeling Solutions . . . 41
4.3.1 A Review of Asset-Centric Threat Modelling Approaches . . . 41
4.3.2 Assessment of Potential Security Risks in Advanced Metering Infrastructure Using the OCTAVE Allegro Approach . . . 42

4.3.3 STRIDE-Based Threat Modeling for Cyber-Physical Systems . . . . . 43
4.3.4 Using Hybrid Attack Graphs to Model Cyber Physical Attacks in the Smart Grid . . . . . 44
4.3.5 A2G2V: Automatic Attack Graph Generation and Visualization and Its Applications to Computer and SCADA Networks . . . . . 45
4.3.6 A Method for Identifying Critical Elements of a Cyber-Physical System Under Data Attack . . . . . 45

**5 Hypergraph Based Threat Models for CPS** **47**
5.1 Attack Graph . . . . . 47
5.2 Attack Hypergraph . . . . . 47
5.3 Attack Hypergraph of CPS: An Example Scenario . . . . . 48

**6 Conclusion and Future Work** **52**
6.1 Conclusion . . . . . 52
6.2 Future Work . . . . . 52

# List of Figures


2.1 An Euler Diagram of an Hypergraph (Left) and It's Incidence Matrix (Right) . . . . 4
2.2 An Example of Undirected Hypergraph (Left) and It's Incidence Matrix (Right) . . 6
2.3 An Example of Directed Hypergraph (Left) and It's Incidence Matrix (Right) . . . . 7
2.4 An Example of Transforming a Hypergraph to a Graph with Line Expansion [18] . . 8
2.5 An Example of Transforming a Hypergraph to It's 2-Section Graph [9] . . . . . 8
2.6 An Example of Transforming a Hypergraph to It's Bipartite Graph [19] . . . . . 9

3.1 Layers of CPS [18] . . . . . 22
3.2 5C Architecture of CPS [29] . . . . . 23

4.1 Phases of OCTAVE Allegro . . . . . 43
4.2 Hybrid Attack Graph Generation [22] . . . . . 45

5.1 SCADA Network Architecture of a Water Treatment System . . . . . 48
5.2 Attack Hyper Graph of SCADA Network Architecture of Figure 5.1 . . . . . 51

# List of Tables

3.1 Platform Vulnerabilities of CPS . . . . . 28
3.2 Network Vulnerabilities of CPS . . . . . 29

5.1 Host Vulnerability Lists . . . . . 49
5.2 Possible Atomic Attacks . . . . . 50

**Abstract**

The key idea of revolutionary Industry 4.0 is none other than the Cyber Physical System (CPS). Its design and component's are incredibly diverse, which makes system's architecture extremely complex. This makes all CPS systems easily vulnerable to attack, which has serious consequences for our day-to-day lives.

This thesis explores the topic of Cyber-Physical Systems (CPS) and emphasizes their importance in the modern world. It draws attention to how CPS's integration of network and digital components is a major factor in technological advancement. The difficulties brought on by the growing size and activity of networks are discussed, especially with regard to securing network systems from cyberattacks.

In complex systems, hypergraphs have appeared as an intuitive and reliable method to illustrate high-order coupling interactions. Although there are lots of challenges to overcome when using high-order information from networks to critical node identification tasks and as well as modeling threats to mitigate the risk on any highly interconnected cyber physical systems.

It is highlighted that simulating cyberattacks with hypergraphs is an extremely useful and very effective tactic for anticipating and blocking possible attacks. We examine the use of hypergraph modeling in threat analysis and demonstrate how well it can capture complex relationships between entities in a network for helping us to reduce the risk optimally.

# Chapter 1

# Introduction

## 1.1 Background

In this modern age, the integration of physical and digital elements leads to several great innovations. Therefor, the enhancements in data transmission and computer processing power have created a need for technical tools and computing systems that extend further conventional functional upgrades. This shift demonstrates how crucial it is to integrate physical devices with information systems, optimize system performance, and efficiently manage resources. Because of this concept of combining computing and communication capabilities with physical processes the term Cyber Physical System (CPS) has caught substantial attraction from all of us [18].

The main objective of CPS is to connect several devices, in contrast to typical embedded technologies, which have been designed to operate as independent units. In today's highly interconnected world, CPS reflects the development of having data and infrastructure easily accessible anywhere [23]. CPS incorporates extensive integration of monitoring, communication, and sensing units into physical systems [14]. This highly interconnected nature of CPS components makes it extremely complex in nature.

As modern industry continues to advance rapidly, there is an increasing need for the integration of Cyber-Physical Systems (CPS) to address deficiencies in networks, technologies, tools, and devices. The complexity and diversity of integrating systems and technologies in CPS aim to create a flexible and inclusive system. However, this very openness and compatibility also create vulnerabilities that malicious actors can exploit, leading to various security challenges [15]. Malicious attacks can therefore have significant effects on national security, business productivity, and human lives. The cybersecurity risks within Cyber-Physical Systems (CPS) have escalated to a critical concern for security experts and professionals [41].

Preventing unauthorized and unauthenticated access to network systems is essential when it comes to cyber security. Organizations use cyber attack modeling, which includes discovering and replicating different kinds of intrusions into their security environments, to accomplish. Selecting a threat modeling method can be a challenging task. The method selected should be align with both the organization's needs and as well as system's requirements. In order to begin with the modeling we need to address two key questions. First, what exactly are threat modeling methods and what different types are available? Second, what criteria should a good threat modeling method satisfy [39]?

With the proliferation of the Internet and the widespread adoption of cloud-based IT infrastructure, there has been a notable surge in cyber security incidents in recent years. Since the enormous expansion in network size and activity, it is extremely challenging to protect our network systems against new and inventive breaches. For these reasons, we need a very effective technique that can predict potential assaults, find a number of security gaps or vulnerabilities in our network system, and even predict issues that haven't yet been identified. Thus, one of the most efficient methods for stopping various cyber attacks is modelling them using hypergraphs. Attack modeling using hypergraphs for any CPS could be able to help us to mitigate the risk optimally as it can demonstrate higher order relationship between the components.

## 1.2 Research Statement

This thesis aims at addressing the research question: *To explore the possibility of using hyper graph based threat models for cyber physical systems.*

## 1.3 Contribution of the Thesis

In this thesis, I have used Hypergraph in order to do effective threat modeling for Cyber Physical System, which can help us to visualize threats more better because it is able to describe higher order interactions between CPS components. The contributions can be outlined as:

1. Discussed about the CPS architecture along with it's security issues like vulnerabilities, threats, and attacks on it.
2. Discussed about existing CPS threat modeling methods and their utilization.
3. I have demonstrated how can we use Hypergraphs to effectively model threat for CPS to mitigate risk on the system maximally.

## 1.4 Outline of the Thesis

The contents of this thesis can be outlined as, it starts with an abstract. After that, in Chapter 1, I have defined Introduction about the thesis which contains Background, Research Statement, Contribution of the Thesis and this Outline of the Thesis. Then Chapter 2 explains about Hypergraphs and Hypergraph Based Threat Models. Chapter 3 covers the details about CPS, it's security related issues and also some previously occurred attacks on CPS. In Chapter 4, I have discussed about the Threat Modeling methods of CPS along with it's existing threat modelings. Chapter 5 contains the main proposal of this thesis, which is Hypergraph Based Threat Models for CPS with an example scenario. At last, Chapter 6 presents the conclusion and the future work about this thesis.

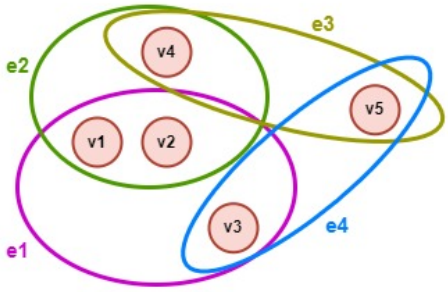
# Chapter 2

# Hypergraphs and Hypergraph Based Threat Models

## 2.1 Introduction

Hypergraphs are important concept in discrete mathematics, which is a generalization of the graph. Therefore, many concepts of hypergraphs can be defined related to the well-known definition of graphs. A hypergraph is defined as a pair of hyper-vertex set and hyperedge set. The hyper-vertex set, also called the vertex set, is a finite set, whereas the hyperedge represents the subset of the vertex set. As the hyperedge can connect any number of vertices, more general types of relationships could be modeled by hypergraphs rather than graphs. The order and the size of the hypergraph can be defined based on the vertex set and hyperedge set, i.e., the order of the hypergraph represents the cardinality of the vertex set, and the size of the hypergraph denotes the cardinality of the hyperedge set [13].

Hypergraphs are a specific type of graph in which an edge can connect to a subset of vertices or nodes instead of just two nodes. Hyperedges, also referred to as generalized edges, are the edges of a hypergraph. In general an undirected Hypergraph $H$ can be defined as pair of $H = (V, E)$, where $V$ is finite, set of non-empty vertices and E is non-empty subsets of $V$, known as hyperedges. Hypergraph can be visually represented by an Euler Diagram and an incidence matrix. Below is an example of a Hypergraph:




| | e1 | e2 | e3 | e4 |
|---|---|---|---|---|
| v1 | 1 | 1 | 0 | 0 |
| v2 | 1 | 1 | 0 | 0 |
| v3 | 1 | 0 | 0 | 1 |
| v4 | 0 | 1 | 1 | 0 |
| v5 | 0 | 0 | 1 | 1 |

Figure 2.1: An Euler Diagram of an Hypergraph (Left) and It's Incidence Matrix (Right)

As shown in above figure, $H(V) = \{v1, v2, v3, v4, v5\}$ and $H(E) = \{e1, e2, e3, e4\} = \{\{v1, v2, v3\}, \{v1, v2, v4\}, \{v4, v5\}, \{v3, v5\}\}$.

## 2.2 Hypergraph Variations

The several kinds of hypergraph variations or properties can be shown as follows:

- **Empty Hypergraph:** The hypergraph with an empty set of hyperedges and vertex points.
- **Trivial Hypergraph:** The hypergraph having an empty hyperedge set and a nonempty vertex set.
- **Simple Hypergraph:** It possesses neither repeated edges nor loops.
- **Non-Simple Hypergraph:** It has repeating edges, which indicates that two or more edges may contain the same set of vertices, or loops, which are hyperedges with a single vertex.
- **Regular Hypergraph:** A regular hypergraph is one in which the degree of each vertex is the same.
- **Uniform Hypergraph:** A uniform hypergraph is one in which the degree of each hyperedge is the same.
- **2-Colorable:** Its vertices can be divided into two classes, $S$ and $T$, so that every hyperedge with at least two cardinality has a minimum of one vertex from each class.
- **K-Partite:** For this type each hyperedge comprises exactly one vertex of each type. The vertices are divided into $k$ parts.
- **Induced Sub-Hypergraph:** A hypergraph whose vertex set is a subset of the given hypergraph and whose hyperedges contain either a single element or at least two intersections with the vertex set is called an induced sub-hypergraph of the provided hypergraph.
- **Sub-Hypergraph:** A sub-hypergraph of the provided hypergraph is one whose set of hyperedges and all of its vertices are subsets of the given hypergraph's subset.

- **Partial Hypergraph:** A hypergraph whose hyperedge set is a subset of the provided hypergraph is said to be partial a partial hypergraph.

## 2.3 Various Hypergraph Definitions

1. **Incidence Matrix:** The incidence matrix of a hypergraph described in [16] is a matrix where the rows correspond to vertices and the columns correspond to hyperedges. The entry $a_{ij}$ is defined as below:

$$a_{ij} = \begin{cases} 1, & v_i \in e_j \\ 0, & v_i \notin e_j \end{cases}$$

   where $v_i$ is the vertex set and $e_j$ is the hyperedge set.

2. **Adjacency Matrix:** For a connected hypergraph $H$, its adjacency matrix defined in [16] $A(H)$ is symmetric, non-negative, and irreducible. The diagonal entries of $A(H)$ are zero, while the off-diagonal entries indicate the number of hyperedges shared by vertices $v_i$ and $v_j$. The entry $a_{ij}$ can be defined as below:

$$a_{ij} = \begin{cases} \sum_{k=1}^{m} a_{ij}^k, & i \neq j \\ 0, & i = j \end{cases}$$

   where

$$a_{ij}^k = \begin{cases} 1, & \text{if } v_i, \quad v_j \in E_k \\ 0, & \text{else.} \end{cases}$$

## 2.4 Types of Hypergraph

Basically, there are two kinds of hypergraphs, which are demonstrated as below:

### 2.4.1 Undirected Hypergraph

The following content is defined in the paper [13]. Let $G$ represent an undirected hypergraph made up of a set of hyperedges $E$ and a set of vertices $V$. Each hyperedge $e \in E$ in a weighted hypergraph has a weight $w(e)$, signifying the significance of the connection relationship over the whole hypergraph. $W$ should be understood as the diagonal matrix of the hyperedge weights, so that $\text{diag}(W) = w(e1), w(e2), \cdots, w(e|E|)$. In general, an incidence matrix $H \in \{0,1\}^{|V|\times|E|}$ represents the structure of a given hypergraph, $G = (V, E, W)$, where each entry $H(v, e)$ indicates whether the vertex $v$ is in the hyperedge $e$:

$$H(v,e) = \begin{cases} 1 & if \quad v \in e \\ 0 & if \quad v \notin e, \end{cases}$$

where $H(v, e)$ indicates the possibility of vertex $v$ assigned to hyperedge $e$ or the importance of vertex $v$ for hyperedge $e$. The degree of hyperedge $e$ and the degree of vertex $v$ are defined as follows:

$$\delta(e) = \sum_{v \in V} H(v, e),$$

and

$$d(v) = \sum_{e \in E} w(e) * H(v, e)$$

The below picture shows example of an Undirected Hypergraph and it's Incidence Matrix denoted by $H$ representation:

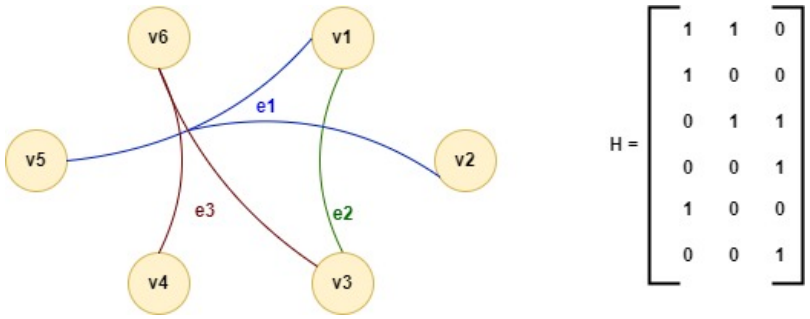



Figure 2.2: An Example of Undirected Hypergraph (Left) and It's Incidence Matrix (Right)

### 2.4.2 Directed Hypergraph

A directed hypergraph defined in [13] $H$ is defined as a pair $(V, E)$ where $V$ is a non-empty set of vertices and $E$ is a set of hyperedges. Each $e \in E$ is an ordered pair $(e^h, e^t)$, where $e^h$ and $e^t$ are non-empty subsets of $V$ and are respectively called the head and tail of the edge $e$.

On directed hypergraph, a trivial definition [13] for the incidence matrix is defined as follows:

$$\hat{H}(v, e) = \begin{cases} -1 & if \quad v \in T(e) \\ 1 & if \quad v \in S(e) \\ 0 & otherwise, \end{cases}$$

where $T(e)$ and $S(e)$ are the target and source vertices for hyperedge $e$, respectively. The below picture shows example of an Directed Hypergraph and it's Incidence Matrix denoted by $H$ representation:

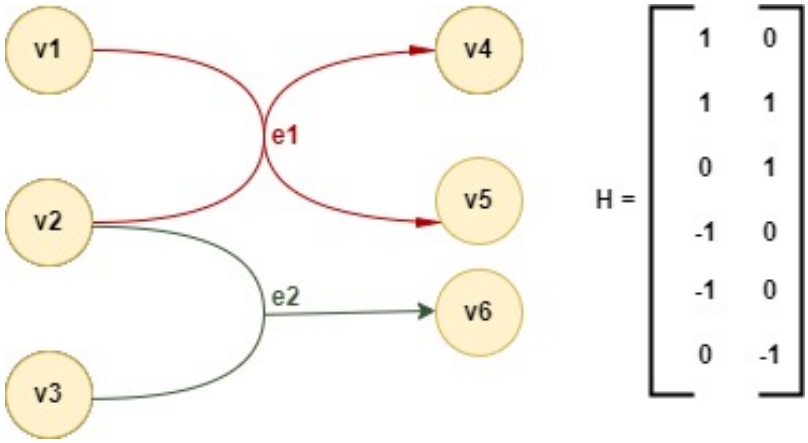



Figure 2.3: An Example of Directed Hypergraph (Left) and It's Incidence Matrix (Right)

Directed hyperedges can represent complex logical dependencies between the sources and targets of the edge [38]. The following is a description of a few significant directed hypergraph concepts:

- **Forward Hyperarc(F-Arc):** A F-arc is a directed hyperedge where the cardinality of the source vertex set is 1.
- **Forward Hypergraph(F-Graph):** A F-graph is a directed hypergraph whose edges are F-arcs.
- **Backward Hyperarc(B-Arc):** A B-arc is a directed hyperedge where the cardinality of the target vertex set is 1.
- **Backward Hypergraph(B-Graph):** A B-graph is a directed hypergraph whose edges are B-arcs.
- **BF-Hypergraph:** A BF-graph is a directed hypergraph whose edges are either B-arcs or F-arcs.

## 2.5 Few Transformations of Hypergraph

We can transform hypergraph into several types of it's equivalent graph. In the following sub-section we are going to see some of the very useful hypergraph transformation.

### 2.5.1 Line Graph of a Hypergraph

Especially in the theory of hypergraphs, the line graph of a hypergraph $H$, designated as $L(H)$, is the graph whose vertex set is the set of the hyperedges of $H$,two vertices are adjacent in $L(H)$ whenever the associated hyperedges have a nonempty intersection in $H$. The below figure shows an example of a hypergraph with it's line graph:

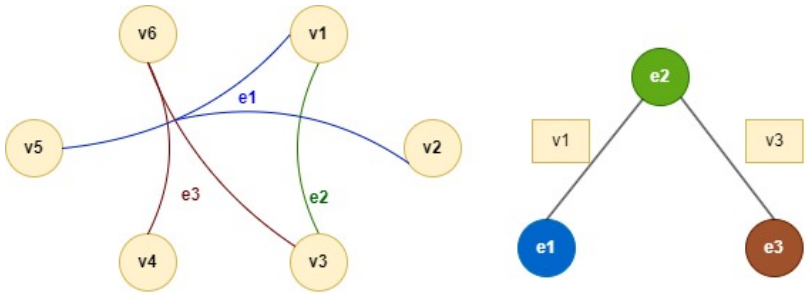



Figure 2.4: An Example of Transforming a Hypergraph to a Graph with Line Expansion [18]

### 2.5.2 2-Section Graph of a Hypergraph

The 2-section graph dictated as $2Sec(H)$, of a hypergraph $H$ is the one if $2Sec(H)$ has the same vertices in $H$. The two vertices in $2Sec(H)$ are adjacent if they are in the same hyperedges of $H$. The below figure shows an example of a hypergraph with it's 2-Section graph:

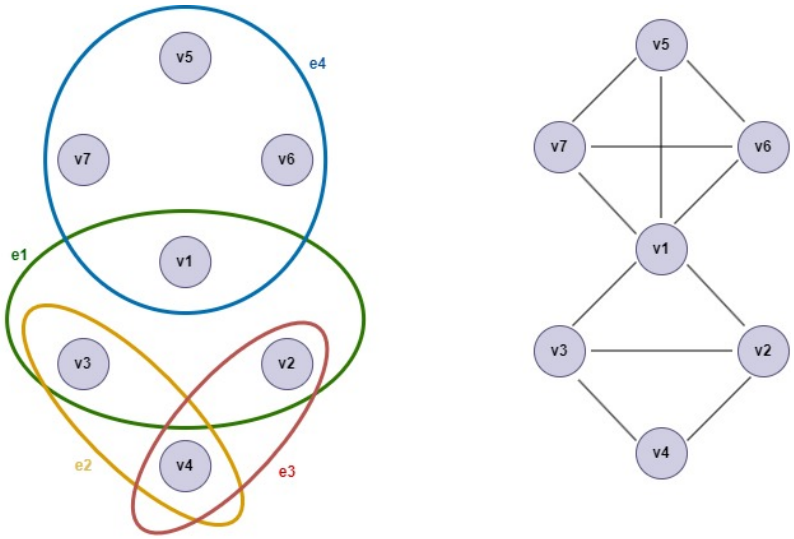



Figure 2.5: An Example of Transforming a Hypergraph to It's 2-Section Graph [9]

### 2.5.3 Bipartite Graph of a Hypergraph

One can represent the bipartite graph as $G = \{U, V, E\}$. The vertices of the bipartite graph can be separated into two distinct and independent sets, $U$ and $V$, in contrast to the simple graph. Only one vertex in set $U$ and one vertex in set $V$ are connected by each edge. Naturally, if the hyperedges are viewed as an additional set of vertices, an undirected hypergraph can be thought of as a bipartite graph.

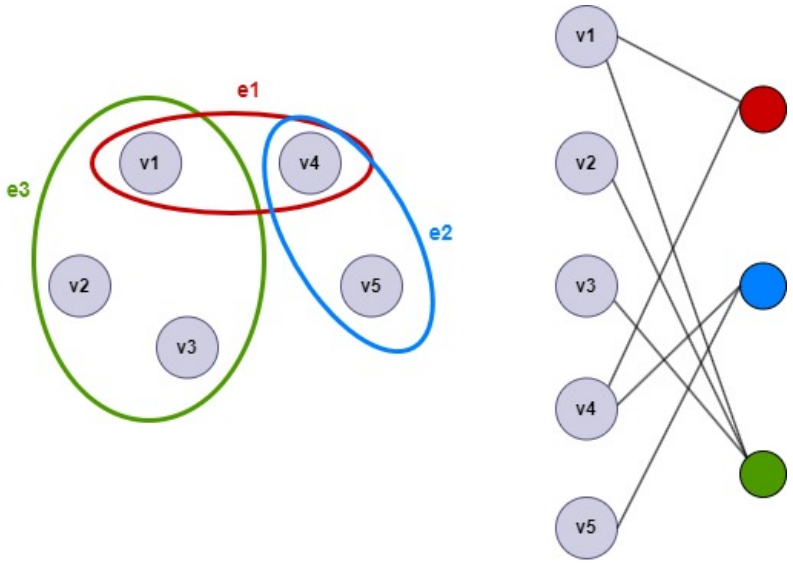



Figure 2.6: An Example of Transforming a Hypergraph to It's Bipartite Graph [19]

## 2.6 Applications of Hypergraph

Hypergraph is one of the best technique at modeling complicated associations, it has been used in many academic fields. This computation techniques are very useful while there exist higher order and complex correlations among data. The subsequent sections will provide insight into several typical uses for hypergraphs [13].

The use of hypergraphs in network security offers a powerful approach to modeling complex relationships and interdependencies within a system, particularly in the context of cyber threats. Hypergraphs provide a more expressive way to represent interconnected attack paths and activities, enabling a deeper understanding of potential vulnerabilities and attack scenarios [21].

The hypergraph is used in recommender systems to establish profile of users, model the user-item network, and further anticipate user preferences. Hypergraph can be utilized to discriminatively establish the high-order connectivities of users and items independently and carry out the collaborative filtering task given the raw user-item network and only the prior interactions between users and items.

Additionally, hypergraph has demonstrated its value in biological and medicinal applications. Massive volumes of biological and medical data have been generated in the last few decades. The data has intricate intra- and inter-data relationships and is multi-modal, diverse, and complicated. The hypergraph can naturally handle multi-modal or diverse information by concatenating hyperedge groups. Furthermore, it can exploit the complimentary information among these data in a discriminative manner by doing this.

Social computing is one example of an application of a hypergraph. Over the past few decades, there has been a significant increase in social media data, which may offer population-level insights. The hypergraph structure can be used to formulate the high-order correlation in social networks, and it is a powerful tool for identifying complicated and hidden connections from the data.

Evaluation of sentiment is a popular but difficult application of social media computing that aims to identify people's actual feelings and attitudes in social media environments. However, the task has become more challenging due to the multi-modality and complicated nature of social media data. For instance, a tweet may contain text, photos, and videos. There are also complex connections between posts in terms of time, geography, and user preferences, among other variables. Thus, it has become critical to figure out how to figure out the complicated connection between tweets and assess user mood. In order to achieve this, a hypergraph can be utilized to create a correlation between each sample and perform reliable and accurate multi-modal sentiment prediction, accounting for the fact that each mood has unique qualities.

## 2.7 Hypergraph Based Threat Models

### 2.7.1 Overview

Hypergraph modeling emerges as one of the most potent methods for addressing several cyber security issues. Hypergraphs offer a versatile framework for representing complex relationships among entities in a network, making them particularly suitable for modeling cyber attacks. Unlike traditional graphs, which connect pairs of nodes with edges, hypergraphs can connect multiple nodes with hyperedges, allowing for more nuanced representations of interactions and dependencies. By leveraging hypergraphs, cyber security professionals can model various attack scenarios, capturing the intricate interplay between attackers, targets, vulnerabilities, and defensive measures. This comprehensive modeling approach enables organizations to anticipate and mitigate a wide range of cyber threats more effectively.

Based on the Hypergraph formalism, extensive research has been done to develop defences against different cyber attacks. We'll describe a few of them in the sub section below.

### 2.7.2 A Hypergraph-Based Machine Learning Ensemble Network Intrusion Detection System

This following is taken from the paper by Lin et al. [31]. In this paper, the authors has suggested to create a Network Intrusion Detection System (NIDS) where port scan assaults can be prevented by combining a hypergraph-based data strategy with an ensemble machine learning NIDS. The vulnerability of network intrusion detection systems to automatically generated port scan attacks has been stated in this paper. In order to track the changing patterns of port scan assaults using a collection of IP addresses and destination ports, they developed hypergraphs. They also noted that ML-generated assaults can succeed against conventional NIDS.

The three primary parts of this model are NIDS Evaluation Framework, NIDS Model Performance Evaluation and Research Data set respectively.

A collection of s-closeness centrality measures that capture the adversary's distinctive pattern of port scan activities on a targeted computer was derived specifically by this work. The essential elements of an ensemble machine learning Network Intrusion Detection System (NIDS) is described as below:

1. **Hypergraph-Based Metrics:** The NIDS employs metrics based on hypergraphs to track the changing patterns of port scan attacks, by examining the connections between IP addresses and target ports. These metrics aid in recognizing distinctive patterns of malicious behavior over time.

2. **Ensemble of Machine Learning Models:** The Network Intrusion Detection System (NIDS) comprises a collection of three classifiers utilizing techniques from Random Forest and Light Gradient Boosting Machine algorithms. These models consist of:
    - **NIDS Model RF:** Trained with raw features using a Random Forest method.
    - **NIDS Model HGI:** Trained with a feature set derived from hypergraph metrics using a Light Gradient Boosting Machine.
    - **NIDS Model HGA:** Trained with another feature set derived from hypergraph metrics using a Light Gradient Boosting Machine.
3. **Real-Time Training and Updating:** The NIDS is structured to consistently train and update its machine learning models in real-time within the network setting. This flexible method enables the NIDS to develop smartly and react dynamically to shifts in network conditions and newly emerging threats.
4. **Behavioral Analytics:** The NIDS integrates behavioral analytics functionalities to aid in real-time detection of port scan activities. Through monitoring network traffic using hypergraph constructs, it can pinpoint and alert potential port scan activities by recognizing signature patterns.
5. **Evaluation Framework:** The assessment technique utilized to gauge the effectiveness of the hypergraph-based NIDS model in identifying port scan attacks consisted of employing a range of hypergraph-based metrics for training an ensemble machine learning NIDS that is both robust and resilient. This approach efficiently observed and identified port scanning activities and unauthorized intrusions while continuously adapting in real-time.

These ideas can be expanded in future study by including more hypergraph measures and by utilising those metrics to examine different kinds of network invasions. A number of alternative methods to generate adversarial examples based on evolutionary computation and deep learning will be further investigated in order to test the adversarial robustness of the ML Ensemble NIDS has been dictated by the authors of this paper.

### 2.7.3 A Mathematical Framework to Assess the Security of an Information Infrastructure

The following is taken from the paper by Baiardi et al. [12]. The authors have introduced a mathematical approach that focuses on analysis of the rights that a threat has obtained through attack sequences in order to assess the security of an information infrastructure.The framework offers various abstraction levels that allow for the evaluation of the most crucial components. How to identify all the rights that a threat obtained via a successful attack is the challenge that must be solved in order to create the framework. These rights rely on how infrastructural elements are connected. For instance, if an attack to take control of a component is successful, any component that trusts the attacked component can also be controlled.

They took into account the three different types of nodes in the hypergraph, which are referred to as users, methods, and components on a high level.

- Users are considered as both authorized users and potential information infrastructure attackers.

- Users who are not attackers have certain privileges that allow them to perform certain functions, which are represented by the methods node in the hypergraph.
- A component can be created from a collection of these methods and some internal state. This infrastructure architecture has been displayed as an ensemble of interconnected components. If a method of one component controls a right of both that component and another component, the hypergraph can be used to show how these components are related.

A labelled hypergraph, in which a hyperarc represents a dependency of the destination node from the source ones, is used to express interactions between components, in order to model components as objects, each of which defines a set of methods invoked by either a user or other components.

In order to define the whole relations among the components they have used following kinds of hyperarcs:

1. **Right Hyperarc:** It shows relation from user to the set of methods.
2. **Source Hyperarc:** The relation among methods that controls the corresponding attributes of the component.
3. **Component Hyperarc:** It defines the dependencies of one components on another components.
4. **Destination Hyperarc:** It defines as the dependency of a method from a set of components.

### 2.7.4 Fusing Hypergraph Spectral Features for Shilling Attack Detection

The following content is taken from the paper by Li et al. [30]. Shilling attacks are one type of cyber attack that are carried out as a result of the recommender system's openness, in which attackers inject fake profiles to alter the recommender system's dependability. We have a number of methods that can assist us in identifying these types of attacks, but they are only practical when identifying well-known attack types and are also quite expensive. Another flaw in this approach is that it primarily emphasizes first-order interactions between users and objects while paying little attention to higher-order interactions amongst users.

In this research paper, they offer the SpDetector shilling attack detection method, which is based on Hypergraph spectral features and incorporates implicit and explicit characteristics to balance effectiveness and generality of the detection model. SpDetector is effective and can overcome the drawbacks of prior approaches. A hypergraph can better capture the high-order relationships between user and items and depict their similarities than a basic binary graph. So, for the purpose of detecting shilling attacks, they introduce hypergraphs.

The main components of this paper can summarized by the following points:

1. They have proposed to develop an model called SpDetector to detect shilling attacks, in which the implicit higher-order information are introduced through hypergraphs and statistic features are extracted to capture the explicit information.
2. They have taken two features called Item Similarity Offset(ISO) and Rating Prediction Error(RPO) and combined these two with User Spectral Features.
3. The three features are entered into the deep neural network, which then produces the likelihood that a user is categorized as a legitimate user. This likelihood is also used as a weight to manage how much of this user's expected rating is contributed.

4. At last, they have evaluated the model against Movie-Lens and Amazon data sets.

The ISO represents the spectral properties of the highest and lowest rated products by each user. Greater similarity between the two products is shown by the larger product, and fake users are more likely to be present.

The additional feature RPE enters the picture because it is clear that legitimate users' preferences are simpler to forecast than attackers'. If a user's predicted rating for a specific item differs too much from the actual rating at the time of prediction, the user is more likely to be an attacker.

### 2.7.5 High Performance Hypergraph Analytics of Domain Name System Relationships

The following is taken from the paper by Joslyn et al. [24]. The Domain Name System (DNS) is nothing more than a distributed database used to translate any host's or website's domain name into a numeric IP address. DNS is utilized because it is exceedingly challenging for humans to remember IP addresses in order for the host to receive any information from it. The authors of this paper present the application of cutting-edge mathematical techniques to hypergraph analytics on a significant amount of DNS data. In network research, hypergraphs generalize graphs to more accurately depict complicated multi way relations among the domain names and the IP addresses. Normally, the relationship should be one-to-one, meaning that one IP address should correspond to a single domain and vice versa. However, in practice, this is not ideal. For the formal clarity and help they offer for the study of complicated cyber data, the authors are committed to taking on the problem of hypergraph complexity.

As a result of this, the authors adopt a dual strategy for creating their approaches such as:

1. For High Performance Computing hypergraph processing, extensive analysis, and data segmentation, they use the Chapel Hypergraph Library (CHGL), a library for hypergraph computation developed in Chapel programming language.

2. They use HyperNetX (HNX), a Python package being developed by PNNL for exploratory data analytics and visualization of hypergraphs, to examine individual hypergraphs or groups of hypergraphs.

In the framework of CHGL and HNX capabilities, they first describe the fundamentals of hypergraph mathematics and hyper network science. They then used the Georgia Institute of Technology's ActiveDNS data set to view the DNS mappings, which produces DNS lookups for well-known zones and lists for domain names. These lookups include the domain name, the date the data was requested, and a list of DNS server IP addresses. Since they created the hypergraph with IPs acting as sets of vertices and domains acting as sets of edges, each domain can be thought of as nothing more than a collection of IPs. These comprise both the exploratory and targeted finding of tiny components as well as fundamental global statistics like degree and edge size distributions. To make an exploratory finding, simple hypergraph metrics are computed and motif mining is used to look for outliers. Targeted discovery, on the other hand, is driven by known illegal activities. They created a blacklist of IP addresses and domains that adhere to a known pattern utilized by a large-scale criminal enterprise.

### 2.7.6 Hypergraph Clustering for Better Network Traffic Inspection

The following is taken from the paper by Pu and Faltings [36]. Networked computing environments are vulnerable to malicious software attacks, unauthorised users, unwanted activities, and setup mistakes. These can be found by keeping an eye on network activity, but network administrators are overloaded by the volume of information that needs to be examined. Filtering is one of the methods used for network traffic inspection to find anomalies, but in this situation we require predetermined filters that can miss the unidentified abnormalities. The development of a clustering approach based on a hypergraph model is suggested in this paper for more effective and efficient network traffic analysis.

They have mostly concentrated on two issues related to this grouping: user affiliation identification and network service identification.

They came up with a grouping technique, in which a collection of ports with similar capabilities are grouped together, for the purpose of identifying network services. In order for us to quickly identify that as malware, they also stipulated that the group of special interests in network service identification would be minimal. They made the assumption that the ports of connections that share the same application-destination pair might be aggregated.

User affiliation identification is also similar, user are grouped together by taking the assumption that users from same department uses same application to access similar destinations because of their similar nature of work and it can be obtained from user's profile. They have shown that a set of users are likely to be in same affiliation if they use same application to access a particular destination.

In this paper they have transformed the entire data set as subsets of ports and each of these subsets are known as set evidence. Every port and user was taken into account as a feature vector, and if these vectors were discovered in the set evidence, it was set to 1. After that, they used a distance-based algorithm to determine how far apart the feature vectors were from one another and then grouped them accordingly. They then used a hypergraph to illustrate the co-occurrence of the feature vectors.

### 2.7.7 Hypergraph Clustering Model-Based Association Analysis of DDoS Attacks in Fog Computing Intrusion Detection System

The following is taken from the paper by An et al. [8]. Similar to edge computing, fog computing places data, processing power, storage, and applications in between cloud services and data sources. In fog computing, distributed denial-of-service (DDoS) refers to the unauthorized use of fog node's computational and bandwidth resources. In DDoS, the attacker takes control of the devices at the user layer before flooding the fog nodes with a massive volume of illicit requests. This paper models and analyzes DDoS attacks in fog clusters based on hypergraph clustering algorithms and can find the relationship between fog clusters and DDoS attacks. The main contributions of this article are the following:

- The detection layer of FC-IDS establishes a security log database for fog nodes and recognizes DDoS attacks. In addition to monitoring and analyzing DDoS events, the cloud server also describes the behavior of intruders and chooses the best intrusion response for analysis and detection.
- During a DDoS attack, a fog node's network bandwidth resources—including resource ID, bandwidth, power, current transmission rate, channel gain, and resource state—are depicted

using a hypergraph. In this approach, the relationships between the fog nodes are described by the hyperedge. In order to determine the association relationship, a two-dimensional scribing conditional bandwidth allocation hypergraph clustering approach based on the Apriori clustering technique is developed.

They describe the goal, layout, and parameter configuration of a simulation experiment to confirm that the hypergraph clustering concept may be an effective DDoS defense. They model the behavior of the system under DDoS attacks in the wireless fog computing environment from the perspective of legal UE access number and radio resource consumption.

### 2.7.8 Hypergraph-Driven Mitigation of Cyber-Attacks

The following is taken from the paper by Granadillo et al. [20]. The objective outlined by the authors is to predict the strategies of an adversary aiming to disrupt a specific system. In order to accomplish this, they support using a hypergraph framework and suggest adding a dynamic financial meter based on Return on Investment (ROI) ideas. By combining a hypergraph model with a stateful quantitative financial measure, it is possible to evaluate, rank, and choose the best defensive tactics while considering the financial and threat consequences. This method comprises a constant assessment of the system's condition, taking into account both the possible effects of adding or eliminating additional security measures as well as the defensive mechanisms that are currently in place.

They have broken down their elements into the following six definitions in this paper: Attack Graph, Attack Action, Event Mapping, Risk Recalculation, Preventive Risk Calculation, and Countermeasure Selection. The proposal aims to improve how attack actions are represented and predicted by introducing a Bayesian methodology, which builds upon prior cyber security studies. It describes an attack graph as a structured framework comprising nodes representing attack actions, interconnected by links, and characterized by conditional probability distributions. This framework facilitates the analysis of potential attack scenarios and helps pinpoint system weaknesses. The proposal underscores the importance of anticipating and mitigating attack actions by leveraging the attack graph model, enabling organizations to preemptively implement defensive measures for heightened security.

Hypergraph-driven reduction is a methodology utilized to simplify and optimize intricate structures, like attack graphs, within the realm of cyber security. This technique employs hypergraphs to depict the relationships and dependencies among various elements, such as attack actions or vulnerabilities, in a more flexible and expressive manner compared to conventional graphs. Hypergraphs allow hyperedges to link multiple nodes concurrently, facilitating a more accurate representation of complex relationships. During the reduction process, the hypergraph structure is meticulously investigated to identify and eliminate redundant or unnecessary elements, such as cycles or overlapping connections, which could complicate the understanding of the attack scenario. Through this streamlining via hypergraph-driven reduction, the system's complexity is diminished, facilitating easier analysis, interpretation, and response to potential cyber threats. In essence, hypergraph-driven reduction bolsters the efficiency of cyber security analysis by furnishing a more succinct and focused representation of the attack scenario. This empowers security professionals to prioritize and address critical vulnerabilities and attack actions more effectively.

### 2.7.9 Intrusion Detection with Hypergraph-Based Attack Models

The following is taken from the paper by Guzzo et al. [21]. The hypergraph-based attack model outlined in the paper presents a fresh perspective on intrusion detection, diverging from conventional methods in several notable ways:

- **Representation of Attack Scenarios:** Some of the traditional ways of modeling an attack which are based on linear sequences or graphs do not allow a clear presentation of the interconnectedness within an attack. The hypergraph-based attack model is based on a hypergraph, which can have many subgraphs on different levels and enable a much more complex representation of the interconnected attack paths. In this way, relationships between more activities and unified activities can be better modeled in a more expressive way.
- **Flexibility and Expressiveness:** The hypergraph-based model offers significant flexibility in representing diverse attack types and structures efficiently. It enables the incorporation of temporal constraints regarding attack execution, supports various security scenarios, and facilitates the depiction of attack scenarios across different abstraction levels. This adaptability allows for a more intricate and comprehensive modeling of attacks.
- **Constraint Specification:** The model permits the inclusion of diverse constraints concerning potential attacks, including temporal constraints, cardinality constraints, hierarchical constraints, and forbidden paths. These constraints serve to describe the actions of attackers and facilitate the identification of anomalies in the system's usual operations.
- **Detection of Complex Attack Scenarios:** Utilizing hypergraphs enables the model to efficiently depict numerous potential attack scenarios, surpassing the capacity of conventional methods. The intricate structure of hypergraphs facilitates the portrayal of intricate attack routes encompassing various activities and entities.
- **Consistency Checking and Attack Detection:** The system integrates features for verifying the consistency of attack models and identifying instances of attacks within sequences of recorded activities. This functionality allows for the detection of potential breaches by matching them against predefined attack models and constraints.

The vertices in these hypergraph-based attack models are instances of groups of log tuples of the same type, where log tuples are nothing more than the attack model's stated sequence of events. The relationships between these vertices create hyperedges, sometimes referred to as segments. This attack model's primary assumption is that it implies every potential attack follows a path from the beginning to a terminal hyperedge. The approach adapts to the unique network setup and incorporating all security concerns.

The hypergraph-based attack model proposed in the paper offers versatile applications beyond computer networks. It can be adapted to various security domains such as physical security, industrial control systems, IoT security, physical access control, and critical infrastructure protection. By customizing this model to specific security needs, organizations can bolster intrusion detection and risk mitigation across diverse environments.

### 2.7.10 Ranks and Partial Cuts in Forward Hypergraph

The following content is taken from PhD thesis of Swailla [38]. An attack graph is a diagram that shows the paths an attacker could take within a network to go from an initial network device to their

target locations, passing via intermediate network devices. Finding the most important network equipment that can aid in preventing attackers from reaching their target is becoming increasingly challenging.

In this paper, they discuss the difficulties associated with analyzing extensive and complicated networks. They dictated two primary graph-related issues concerning the ranking of vertices and the disruption of connectivity within graphs. Asset relationships can be represented as directed graphs. If $a1$ depends on $a2$ and there are two assets $a1$ and $a2$ as vertices, then the dependency is described as a directed edge from $a1$ to $a2$. Furthermore, they have described the significance of this research within the realm of computer network defense. The main contribution to this paper can be outlined as follows:

- **Vertex Ranking:** This refers to the procedure of attributing significance to individual vertices within a graph, particularly in dependency attack graphs pertaining to network security. The primary aim is to assess the relative importance of attack assets (vertices) in achieving defense goals and understanding security risks in a network. The author depicted into the development of the AssetRank algorithm, designed to rank vertices in dependency attack graphs. This algorithm considers various factors like dependencies between vertices, vulnerability attributes, success likelihood metrics, and personalization values to calculate a ranking reflecting the importance of each vertex to a potential attacker. Vertex ranking involves assigning numerical values to vertices based on their relevance in achieving specific network objectives. Through vertex ranking, network defenders can prioritize security measures, concentrate on critical vulnerabilities, and make informed decisions regarding defense against potential attacks. In summary, vertex ranking within the document's context is a vital element of network security analysis, offering valuable insights into the significance of different assets in the network and aiding in decision-making for defending against cyber threats.

- **Partial-Cut Vertex Sets:** Partial-Cut Vertex Sets are a key concept in network security and attack graphs. They have discussed in the context of disrupting connectivity between two sub-communities in a graph. The primary objective of identifying partial-cut vertex sets is to identify a subset of vertices whose elimination maximally disrupts the connectivity between two disjoint communities in a graph while adhering to a specified budget constraint.

- **Experiments:** They have utilized the MulVAL attack-graph tool suite to generate a dependency attack graph based on network descriptions and user queries. For instance, users might inquire about the possibility of attackers executing arbitrary code on servers. The resulting attack graph was exported to a custom Python module, which normalized the data, calculated AssetRank values, and visualized the graph using Graphviz.

### 2.7.11 Research and Construction Model of Cyberspace Based on Hypergraph

The following is taken from the paper by Peng [35]. Cyberspace is the networked environment in which data and information are exchanged. It is a digital world made possible by networked computer systems. Without being physically there, we can communicate, share information, and carry out a variety of tasks in cyberspace.

The hypergraph model proposed in the paper provides a structured framework for characterizing cyberspace by mapping both homogeneous and heterogeneous entities into a hypergraph structure.

This approach enables the representation of complex relationships and interactions among various entities across different layers of cyberspace. By breaking down the hypergraph model, the paper explains the multi-layered nature of cyberspace and the relationships within it, including intra-relations among homogeneous entities like assets and services, as well as inter-relations among heterogeneous entities.

Through examining the connections among sub-hypergraphs, researchers and analysts can acquire valuable insights into the intricate interplays and inter-dependencies present across various layers of cyberspace.The authors of this paper decomposed the hypergraph model into asset, service, and mission sub-hypergraphs offers a structured approach to understanding the multi-layered nature of cyberspace, which can be described as below:

1. **Asset Sub-Hypergraph (ASH):** Cyber events that have similar asset attributes are grouped together into a hyper-edge in the Asset Sub-Hypergraph (ASH). This sub-hypergraph focuses on showing the relationships between assets in cyberspace by highlighting the dependencies and interactions that result from asset attributes.

2. **Service Sub-Hypergraph (SSH):** This illustrates the connections between cyber events through shared service attributes. Through the creation of hyper-edges using these attributes, this sub-hypergraph demonstrates the relationships and inter dependencies among various services within the cyberspace ecosystem.

3. **Mission Sub-Hypergraph (MSH):** The Mission Sub-Hypergraph (MSH) illustrates the connections between cyber events and the missions influenced by these events. Through linking missions to events via hyper-edges, this sub-hypergraph demonstrates the impact of events on various missions within the cyberspace environment.

# Chapter 3

# CPS and Security Threats

## 3.1 Cyber Physical System

"Cyber-Physical Systems or 'smart' systems are co-engineered interacting networks of physical and computational components. These systems will provide the foundation of our critical infrastructure, form the basis of emerging and future smart services, and improve our quality of life in many areas."[NIST, Engineering Laboratory].

Cyber-physical systems (CPS) represent an innovative class of systems combining computational and physical functionalities, facilitating interaction with humans through diverse channels. This integration of computation, communication, and control empowers advancements in various domains such as aerospace engineering, transportation, autonomous driving, and biomedical engineering. These areas offer significant opportunities for technological innovation, albeit accompanied by unique research challenges [11].

A cyber-physical system (CPS) comprises interconnected computers and physical elements. Embedded computing devices oversee and regulate tangible processes, often incorporating feedback mechanisms, wherein physical activities influence computational processes and vice-versa [28].

## 3.2 Overview

A Cyber-Physical System (CPS) represents the fusion of physical and digital domains, harmonizing the interaction between the tangible world and computational capabilities. CPS integrates physical processes with computational algorithms and networking infrastructure to enable real-time monitoring, control, and optimization of intricate systems. Through the collection of data by sensors from the physical environment and subsequent manipulation by actuators based on computational insights, CPS establishes a closed-loop feedback mechanism. This integration facilitates a wide array of applications across industries like manufacturing, transportation, healthcare, and infrastructure. Utilizing networking technologies, CPS connects distributed components to facilitate communication and coordination within the system. Powered by computational intelligence, which encompasses machine learning and optimization methods, CPS can quickly adapt to dynamic environments, enhance performance, and autonomously respond to evolving conditions. CPS also facilitates with opportunities for innovation and societal advancement.

## 3.3 History of CPS

The term "Cyber-Physical Systems" (CPS) was introduced by Helen Gill in 2006 at the National Science Foundation (NSF) CPS Workshop organized by the US NSF. Currently, CPSs have gained recognition and are listed as priority innovations in the US and various European countries [7].

The foundation of CPS can be traced back to the integration of computational elements with physical systems for automation and control. This period witnessed the development of early analog and digital computers applied in industrial and scientific domains. The emergence of embedded systems marked a significant advancement, where computing elements were integrated into various devices for real-time control of physical processes such as manufacturing and automotive systems. Networking technologies and real-time operating systems enabled the development of distributed control systems, allowing remote monitoring and control of physical processes through interconnected computing devices.

The proliferation of sensors, actuators, and connectivity technologies led to the emergence of the Internet of Things, expanding CPS by enabling the interconnection of physical objects with the internet for data collection, analysis, and control in various domains. Industry 4.0 leverages CPS and IoT concepts to create intelligent production systems. Cyber-Physical Production Systems integrate physical processes with digital technologies like AI and cloud computing for smart manufacturing. CPS has been applied to urban environments to create smart cities, utilizing interconnected sensors, actuators, and control systems for optimized resource management and improved public services.

Advancements in edge computing, machine learning, and autonomous systems are expected to further enhance CPS capabilities in areas like autonomous vehicles, robotics, healthcare, and sustainable energy systems.

## 3.4 CPS Components

Architectural frameworks for developing a cyber-physical system consist of diverse components that together establish the system's structure and operational capabilities. These components play a vital role in guaranteeing the smooth fusion of physical and digital aspects within a CPS. In the upcoming subsections we are going to discuss about the several layers and the various architecture of CPS.

### 3.4.1 Several Layers of CPS

The architecture of a cyber-physical system can be analyzed at different levels. The most common structure of a CPS is segmented into seven primary layers, ranging from the physical layer to the application layer [7] as described below:

1. **Physical Layer:** The physical layer forms the foundation of the CPS architecture. This layer includes sensors and actuators interconnected through wired or wireless networks. Devices at this level generally possess limited memory and processing capabilities. Attacks on this layer primarily originate from external sources.

2. **Data Link Layer:** The data connection layer connects network layer requests with physical layer services by managing the production, transport, and reception of data frames. It is separated into two sublayers: Media Access Control (MAC), which controls physical media access, and Logical Link Control (LLC), which offers network layer functions. This layer is

vulnerable to attacks that change MAC addresses, which may result in issues with device identification.

3. **Network Layer:** At this layer, packets are routed by converting MAC addresses into network addresses using IPv4/IPv6 and RPL protocols. Attacks at this layer can disrupt sensors and actuators, altering information and its source, potentially causing mechanical failures.

4. **Transport Layer:** Packets are divided into smaller fragments at this stage. The main transport layer protocols like TCP, UDP, and ICMP are frequently employed. Security breaches at this level can lead to decreased network equipment speed and service disruptions.

5. **Session Layer:** The session layer manages communication sessions to maintain the proper message sequence in network transmission. It uses labels in long messages to prevent restarting in case of faults. Session-level protocols are typically integrated into the top three layers of the model.

6. **Presentation Layer:** The presentation layer manages data presentation between application processes by converting data formats, and handling encryption/decryption tasks. An example protocol at this layer is SSL, ensuring secure message exchange for application-layer protocols in the TCP/IP stack.

7. **Application Layer:** The information gathered from earlier layers is stored, examined, and updated in this layer. It makes control choices that the virtual prototype interface makes visible. The most crucial concern at this level is the preservation of data privacy.

The below figure shows the layers of CPS on a high level:

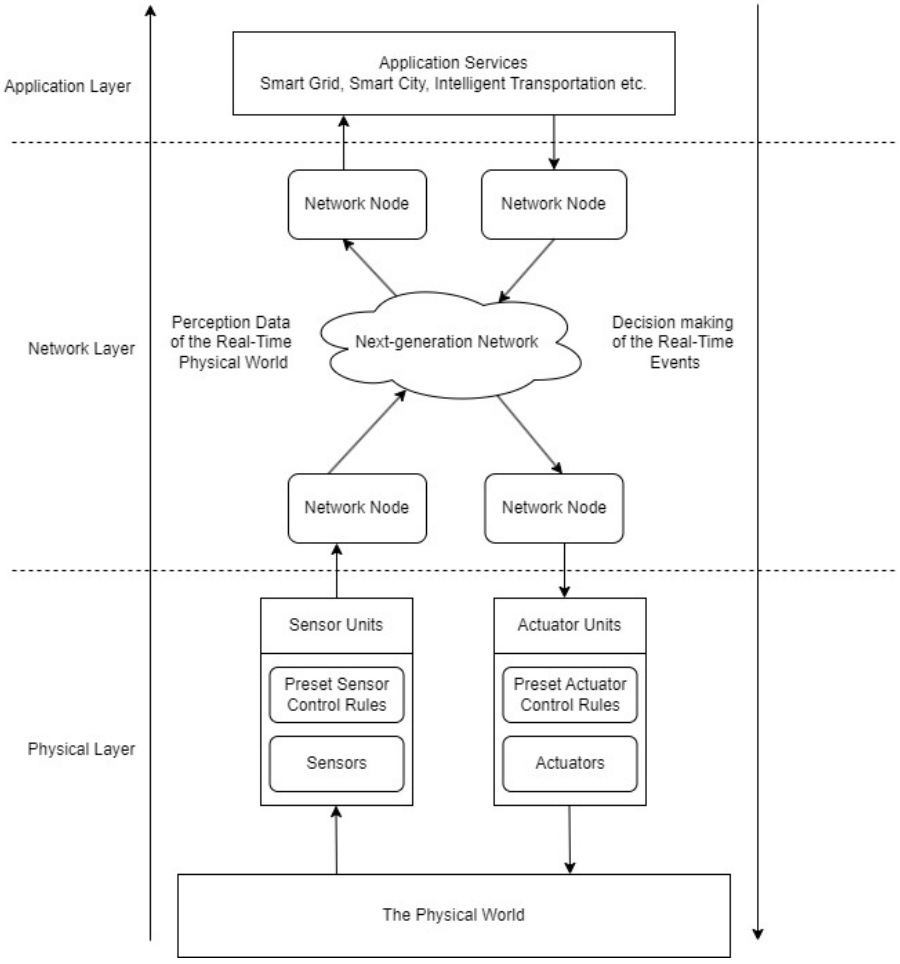



Figure 3.1: Layers of CPS [18]

### 3.4.2 5C Architecture of CPS

The following content is taken from the paper by Lee et al. [29]. Primarily, CPS comprises two primary functional components: (1) advanced connectivity, which guarantees real-time data collection from the physical world and facilitates feedback from cyberspace; and (2) intelligent data management, analytics, and computational capabilities that form the cyberspace. The CPS 5C level architecture offers a structured framework for developing and deploying Cyber-Physical System's by delineating five essential components:

1. **Smart Connection:** The first step in developing a Cyber-Physical System application is to gather accurate and reliable data from machines and their components. This data can be obtained directly via sensors or sourced from controller systems or enterprise manufacturing systems.

2. **Data to Information Conversion:** Valuable insights need to be extracted from the data, and there are numerous tools and methodologies for converting data into meaningful information. There has been a notable focus on the development of these algorithms in recent years.

3. **Cyber:** In this architecture, the cyber level functions as the central information hub. Data is continuously sent to it from all connected machines, creating an integrated network of these machines. With this vast amount of data collected, specialized analytics are necessary to extract further insights, offering a clearer understanding of the status of each individual machine within the network.

4. **Cognition:** Implementing CPS at this stage offers a thorough understanding of the monitored system, facilitating informed decision-making. By accessing comparative data and the statuses of individual machines, tasks can be prioritized to enhance maintenance processes. Utilizing infographics effectively is crucial for conveying the acquired insights to users in a clear and comprehensive manner.

5. **Configuration:** The configuration level acts as a feedback mechanism from the digital realm to the physical domain, serving as a supervisory control that allows machines to self-configure and adjust. This phase functions as a resilience control system (RCS), executing corrective and preventive actions identified at the cognition level on the monitored system.

The following figure depicts the 5C architecture of a Cyber Physical System (CPS):

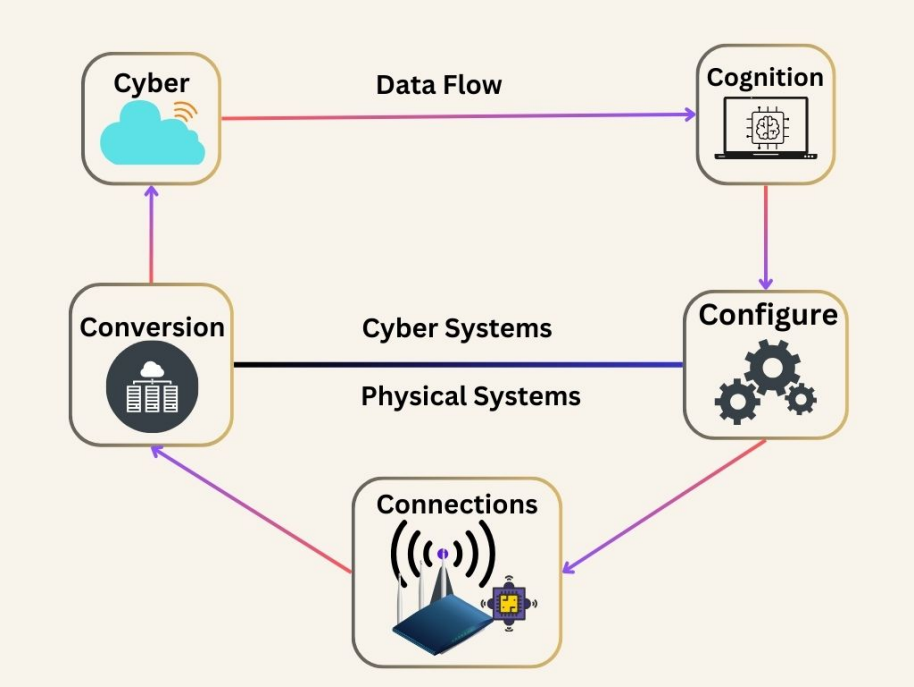



Figure 3.2: 5C Architecture of CPS [29]

## 3.5 CPS Characteristics

Through the use of modern computing, network, and sensor technologies, cyber physical systems comprise built interacting networks that successfully combine both cyber and physical processes and components. Furthermore, cyber-physical systems can be thought of as intricate, multifaceted systems or systems of systems with real-time demands that have the ability to affect both the digital and physical worlds [27].

Now let us discuss about some of the very crucial features or characteristics of CPS, which is taken from the paper by Napoleone et al. [32]. The authors of this paper has described several characteristics of CPS as follows:

1. **Complexity or Heterogeneity Encapsulation:** CPSs frequently have embedded systems (hardware and software) that can produce, exchange, and analyze enormous volumes of data, these heterogeneous nature makes it extremely complex in nature. In summary, managing heterogeneity and complexity is a difficulty associated with integration, and it needs to be resolved by appropriately encapsulating systems and devices. Encapsulation is, in fact, a general design principle that CPSs are born with.

2. **Interoperability:** The capacity of system components to connect, communicate, and function together is known as interoperability. Written differently, interoperability enables CPSs to share mutually understandable data. It can be further described through the following points:

   - **Communication and Connectivity:** Information feedback from the cyberspace and real-time data gathering from the physical world are guaranteed by the communication, or more accurately, connection, feature of CPSs. In fact, CPSs are networks of interconnected entities that span every part of operations, from machine operation and process control to the entire process and distribution network, depending on conditions.
   - **Networking Capability:** With the goal of building intelligence across many domains, CPSs should be made up of interconnected clusters of computational and physical elements across extensive wired and wireless networks via a range of sensors and actuators. The Internet is typically used to connect these fields, this allows for dynamic network participation.

3. **Service Orientation:** The capacity of CPSs to respond quickly in order to address time-sensitive situations comes under service orientation. Within CPSs, digital services contained within service-oriented architectures are also considered cyber elements, such architectures are designed to integrate data and business operations quickly.

4. **Modularity:** The capacity of a CPS to be easily modified, reconfigured, and structured in response to promptly evolving customer demands and product advancements is known as modularity. Therefore, system independence is made possible by modularity, which increases its adaptability. We can also understand this characteristics through the following aspects:

   - **Autonomy and Self Capabilities:** CPS are highly autonomous, dynamic, and complicated in nature. Such autonomy needs to be properly planned to guarantee that components can respond accordingly to interruptions and recover from localized modifications. Autonomy enhances CPS's capacity for self-reliance. In actuality, self-capabilities are examples of autonomy.

- **Decentralization:** CPS components operate independently and make decisions that are in line with the organization's fundamental goals as part of decentralization. Distribution, which is another word for decentralization, describes the CPS control/decision-making system, whose parts are networked computers that communicate and plan actions to achieve shared objectives.

5. **Integration:** In fact, CPSs are the integration of computational and physical processes, which is a crucial and difficult problem. Integrating CPSs into physical systems, software, and platform engineering is a design concern that will enable future design of current process networks connecting production resources to be completed more quickly.

6. **Virtualization:** Virtualization has been defined by numerous writers as the capacity to connect sensor data to virtual factory models and simulation models; put another way, virtualization is the process of building a virtual replica of the real world and maintaining a connection to it over time. It enables CPSs to monitor and remotely evaluate physical processes, emulate actions and, in some situations, enable feedback received from the digital world.

    - **Real Time Capability:** The ability of CPSs to gather and evaluate data on raw materials, equipment, and quality in real time and deliver the resulting insights promptly is known as real-time capability. It is a crucial feature since it enables CPSs to immediately respond to any alteration in the physical processes and guarantee the system's functionality and safety. Therefore, having real-time capacity utilization may also mean taking proactive measures to avoid malfunctions. It is also possible to think of it as virtualization's complement.

7. **Computational Capability:** The definitions given for this feature deals with data management and analytics using approaches and strategies from various fields such as statistics and computer science. As a result, the cyber components of CPSs ought to be capable of handling a sizable portion of the computing and control work that was previously done by humans. These capabilities are now further enhanced by the ability to exchange data and communicate with one another. Furthermore, ubiquitous computing refers to the idea that CPSs should be able to conduct computations from any location. Another feature related to computation can be characterized as following item:

    - **Intelligence:** CPSs are expected to be intelligent, or smart, as they can recognize other components, sense events, communicate with other components, and make decisions on their own. The sophisticated computing, analytics, and data management are capabilities of CPSSs. CPSs provide physical components intelligence by enabling computation and communication. These kinds of intelligence are dispersed among entities. More precisely, a CPS may perform intelligent control over physical systems to adjust to changing situations and environments by using sensors and actuators to gather data about physical operations in real-time.

8. **Co-operation and Collaboration:** Although collaboration is often a stronger idea involving resource sharing and shared aims, co-operation and collaboration are regarded as comparable characteristics in this work. In reality, a form of consistency from CPS co-operation to collaboration is becoming noticeable in the modern conversation. They may regulate their behavior

to adjust to changes and work with other CPSs to complete particular tasks. Co-operation in CPS refers to a distributed system's capacity to dynamically select which of its autonomous subsystems will complete a given task in order to maximize efficiency indicators like quickness of response.

9. **Dynamic Reconfigurability and Adaptability:** In general, dynamic reconfigurability refers to the quality that permits prompt reaction to disruptions and changes in the market. According to this viewpoint, reconfigurable operations is made possible by CPSs. In particular, CPSs are capable of dynamically reconfiguring their structures over time, rearranging their functionalities, altering their behavior, and redefining their bounds. Similar to the dynamic reconfigurability, adaptability refers to CPSs' capacity for dynamic reconfiguration and reorganization in response to rapidly evolving circumstances and new demands. Accordingly, some writers noted that CPSs' capacity for self-organization, learning, and evolution is necessary for adaptation.

## 3.6 CPS Security Issues

Because of the complicated and varied nature of the systems and technology integrated into CPS, the system is compatible and open, which regrettably gives adversaries a platform to take advantage of CPS and leads to a number of security problems. Cyberattacks are a common issue that can lead to severe effects or even an overall decrease in system performance [15].

### 3.6.1 CPS Vulnerabilites

In cybersecurity, vulnerabilities denote weaknesses or deficiencies within a system that malicious actors can exploit to jeopardize the confidentiality, integrity, or availability of the system or its data. These vulnerabilities may manifest in different facets of a system, encompassing software, hardware, network infrastructure, and human procedures. According to Gao et al. [18] CPS vulnerabilities can be classified into three distinct categories:

1. **Management and Policy Vulnerabilities:** If the process for developing and implementing policies and guidelines is inadequate, inappropriate, or absent, it can lead to the introduction of vulnerabilities. Safety documentation and management assistance is the base of security in any strategy. Vulnerabilities can be minimized by implementing appropriate security rules. The following describes the potential weakness in the policy program:
    - **Inadequate or Incomplete Security Policies:** Lack of comprehensive security policies or guidelines can leave systems exposed to potential threats.
    - **Absence of Formal Safety Training Programs:** Failure to provide adequate training to employees on security best practices can result in human errors that compromise security.
    - **Lack of Specific Written Safety Procedures:** Not having detailed procedures for handling security incidents or implementing security measures can lead to confusion and inefficiencies during security incidents.
    - **Missing Disaster Recovery Plans:** Failure to have robust disaster recovery plans in place can result in prolonged system downtime and data loss in the event of a security breach or disaster.

- **Insufficient Change Management Processes:** Inadequate controls over changes to system configurations or software updates can introduce vulnerabilities and weaken the overall security posture of the organization.

2. **Platform Vulnerabilities:** In Cyber-Physical Systems (CPS), vulnerabilities may arise from platform defects, misconfigurations, or inadequate maintenance, encompassing hardware, operating systems, and applications. These vulnerabilities can be mitigated through the implementation of various security measures, including patching operating systems and applications, enforcing physical access controls, and deploying security software such as antivirus programs.

| Classification | Vulnerability |
|---|---|
| Platform Configuration Vulnerabilities | Operating System and application software patches develop is delayed or not maintained |
| | Using the default configuration |
| | Important configuration is not saved or backup |
| | No protection for the data in Portable Devices |
| | Lack of adequate password policy |
| Platform Hardware Vulnerabilities | Lack of testing on security changes |
| | No physical protection for critical systems |
| | Unauthorized personnel can physically access the device |
| | remote access to ICS component |
| | dual network interface card Connected to the network |
| | Unregistered assets |
| | Radio frequency and electromagnetic pulse (EMP) |
| | No standby power |
| | Lack of environmental controls |
| | Important component with no redundant |
| Platform Software Vulnerabilities | Buffer Overflow |
| | Installing security functions can't be start through the default stratup |
| | Improper Handling with the undefined, ill-defined, or "illegal" Condition |
| | unsafe Industry-wide agreement |
| | using plain text |
| | Running unnecessary services |
| | proprietary software discussed in conferences or journals |
| | authentication and access control of Inadequate configuration and programming software |
| | Intrusion detection and prevention software has not been installed |
| | Logs is not maintained |
| | Event is not excavated |

**Table 3.1 Continued from Previous Page**

| Classification | Vulnerability |
|---|---|
| Platform Malware Vulnerabilities | Malware protection software of definition of malware is not up to date |
| | protection software of Malware software is not tested in details |
| | Malware Protection is not installed |

Table 3.1: Platform Vulnerabilities of CPS

3. **Network Vulnerabilities:** Vulnerabilities in Cyber-Physical Systems (CPS) stem from defects, inaccurate configurations, mismanagement of CPS networks, or connectivity with external networks. These vulnerabilities can be mitigated or minimized by implementing a range of security measures, including network design defenses, defense-in-depth strategies, network traffic encryption, limiting network links, and enforcing physical access controls on network components, among others.

| Type | Vulnerability |
|---|---|
| Network Platform Vulnerabilities | Fragile network security architecture |
| | Data flow control is not used |
| | Improperly configured security devices |
| | Network device configuration is not saved or backedup |
| | The password is not encrypted during transport |
| | Password devices can exist on the network indefinitely |
| Network Hardware Vulnerabilities | Inadequate physical protection to the network device |
| | Unsafe Physical port |
| | Lack of environmental controls |
| | Non-essential personnel can access devices and network connections |
| | Lack of critical network redundancy |
| Network Perimeter Vulnerabilities | No defined security perimeter |
| | Firewall does not exist or is not configured properly |
| | Applying control network to non-control link |
| | Control network service is not within the control network |
| Network Monitoring and Logging Vulnerabilities | Lack of firewalls and routers log |
| | ICS network with no security monitoring |

**Table 3.2 Continued from Previous Page**

| Type | Vulnerability |
|---|---|
| Communication Vulnerabilities | Not identifying critical monitoring and control path |
| | Using plain text in Standard, archive intact communication protocols |
| | Authenticated user data or device is defective or does not exist |
| | Integrity checks on Communication is in shortage |
| Network Connection Vulnerabilities | No sufficient certification between the client and the access point |
| | No sufficient data protection between The client and the access point |

Table 3.2: Network Vulnerabilities of CPS

### 3.6.2 CPS Threats and Attacks

Numerous threats exist, which makes it possible for CPS to be the target of multiple attacks. The three categories listed below can be used to categorize the cyberattacks on CPS, which is taken form the paper by Luiz et al. [37]:

1. **Availability:** Because tampering with or absence of sensitive data could have terrible impacts, it is crucial to ensure its security. Data availability-focused cyberattacks occur when harmful data is transmitted, congesting servers or networks. As a result, there is a pause or delay in data communications. We characterize this phenomenon as availability attack. It can be further characterised as below:

   (a) **DoS/DDoS:** Attacks known as denial-of-service (DoS) try to stop system communication and overload the network in order to prevent users from completing their service requests. The attacker can use their own computer to carry out direct attacks, or they can use bots which is the hacker-controlled compromised system to carry out indirect attacks. The different types of DoS/DDoS attacks are as follows:

      i. **ICMP:** When delivering Internet Protocol (IP) packets, the protocol responsible for informing clients of faults is called the Internet Control Message Protocol (ICMP). Through these attacks the attacker overflows the information flow, produces and sends a lot of ICMP requests, and takes advantage of the victim's system's bandwidth.

      ii. **HTTP:** The Hypertext Transfer Protocol (HTTP) serves as the foundation for data exchange on the web and is the protocol responsible for sending hyperlinks. In order to overload, block, disrupt, and mislead the transmission of accurate information and the communication of web-based applications that use the HTTP protocol, the target of these attacks receives an enormous amount of GET and POST messages.

      iii. **TCP SYN:** The TCP SYN attack exhausts system memory and prevents users from using services. Furthermore, it performs the intrusion by taking advantage of a flaw in the TCP protocol.

iv. **UDP:** In the TCP/IP paradigm, a protocol that operates at the transport layer is called the user datagram protocol (UDP). To increase network traffic, the attacker produces and delivers a large number of packets with fictitious addresses, which then overflows the system's bandwidth. The server begins to crash because it is unable to properly verify and process requests, due to which authorized users are unable to access system services.

2. **Integrity:** The quality, consistency, and transparency of the data are essential for the proper operation and management of CPS. Attacks occur when periodic measurements or command signals are changed, compromising the data's integrity. One attack that aims to compromise data integrity is False Data Injection (FDI). Important attack techniques aimed at threatening data integrity are outlined here.

   (a) **Cross-Site Scripting:** One of the most frequent and significant kinds of code insertion attack is cross-site scripting (XSS). Through the execution of incorrect code, this attack takes advantage of security flaws in the system. The harmful code is created by the attacker and spreads via the web browser. Sensitive data on the system is accessible to the attacker when the victim visits the compromised website.

   (b) **Data Diddling:** These attacks include changing database data without permission, which is against regulations. In addition, among other improper modifications that compromise the integrity of the data, the attack has the ability to alter a file's status from permanent to temporary or from private to public.

   (c) **Salami:** Salami attacks include carrying out little intrusions on the network data system in order to obtain a sufficient quantity of private data while escaping detection by the security system. These attacks offer a more powerful assault and, as a result, more harm to the business.

   (d) **Session Hijacking:** An attack known as "session hijacking" occurs when a portion of the network is misused, giving the attacker access to the data on that portion of the system and turning them into a participant. The hacker has the ability to pose as a network administrator and deliver false information packets to other users. The goal of the hijacker is to identify and take advantage of the network's vulnerabilities and unprotected protocols.

   (e) **SQL Injection:** This kind of attack aims to get access to database data by taking advantage of SQL statement vulnerabilities. This attack takes place when an attacker gains access to a website's database by using an improper SQL statement.

   (f) **Replay:** One type of cyberattack that attempts to jeopardize the accuracy of the data pertaining to system components is called a replay attack (RA). The goal of this assault is to track and log a genuine series of sensor readings, then replace the actual readings with these previously recorded values during the invasion. Until the attack ends, these captured data are continuously played again and repeated.

3. **Confidentiality:** Security is compromised when unapproved persons get access to the system planning, operational strategies, control, and user data. To maintain data confidentiality, sensitive system information should only be viewed by those who are allowed. The following are a discussions of such confidentiality-related attacks:

(a) **Eavesdropping:** Eavesdropping is an act of stealthily listening in unauthorized talks across a network. The network user's private and sensitive information is accessible to the eavesdropper. The eavesdropper can then read the system data, add fake data to the network, remove it, or use it for any other unauthorized purpose.

(b) **Keylogger:** Through these types of attacks a malicious software program is installed on the machine without the client's knowledge or consent. Its purpose is to deliberately track and record the user's actions. Information confidentiality is compromised by this kind of attack.

(c) **Password Attacks:** In these types of attacks an attacker can find the user's password in a number of methods, such as from combination of all characters contained in the dictionary, using hacking software that tries numerous possible password combinations, by guessing the password from the victim's personal data etc.

(d) **Snooping:** Online snooping is a tactic whereby a hacker watches a target over the internet and collects passwords, sensitive corporate information, and network data from the victim. This is a digital attack because it may also be carried out via hacking into network switches, routers, and security cameras.

(e) **Social Engineering:** A cyberattack that targets the person rather than the system's network architecture is known as social engineering. Manipulation strategies are used in this attack to fool and control victims until they expose private information that the attacker needs.

(f) **Traffic Analysis:** An examination of the communication traffic between the sender and recipient is done beforehand by the attacker in a traffic analysis attack. It seeks to get private data in order to identify any weaknesses in the network. It then goes on and plans how the heist will be carried out. The security and privacy of the user's information are compromised by this passive attack.

## 3.7 Incidences of Past Attacks on CPS

In the paper by Dibaji et al. [14], they have highlighted several significant instances of cyber-physical attacks that have occurred in real systems. The CPS are described in the following subsection.

### 3.7.1 Stuxnet

Stuxnet was a highly sophisticated cyber-physical attack that targeted an Iranian uranium enrichment plant in late 2009. The attackers exploited vulnerabilities in a Programmable Logic Controller (PLC) to manipulate the centrifuges used in the plant's uranium enrichment process. By inserting malware into the system, the attackers were able to manipulate critical system data, replay measurements to monitoring sites, and inject malicious actuation signals, ultimately causing significant damage to the centrifuges.

### 3.7.2 RQ-170

In 2011, the United States experienced a significant event where they lost control of an RQ-170 unmanned aerial vehicle (UAV), resulting in its landing in Iran. The incident involved Iranian

forces reportedly disrupting GPS communications and manipulating GPS signals to influence the drone's landing location. This occurrence highlighted the potential vulnerabilities associated with GPS spoofing techniques, particularly in the context of vehicle control and navigation systems. The attack on the RQ-170 is categorized as a disruption attack, where control was lost, followed by a deception attack, involving the manipulation of signals to deceive the UAV's navigation system.

### 3.7.3 Ukraine Attack

The Ukraine attack in 2015 deviated from traditional power grid protection practices by targeting Ukrainian power distribution networks with a series of cyberattacks. The initial breach involved phishing emails containing the Black Energy malware, enabling the attacker to steal critical data and manipulate control commands. This breach led to a confidentiality breach, followed by an integrity breach. By tampering with firmware in substations, the attacker remotely disabled breakers, causing an availability breach. In 2016, a further attack using the Crash Override malware targeted a transmission station.

### 3.7.4 Maroochy Attack

The Maroochy attack is a notable case of a cyber-physical attack on a sewage control system in Queensland, Australia. In 2000, a disgruntled former employee used his knowledge of the system to release millions of liters of raw sewage into waterways and public areas by manipulating the control systems. This incident demonstrated the potential consequences of insider threats in cyber-physical systems.

### 3.7.5 Jeep Hack

The Jeep hack refers to a notable cybersecurity incident involving a Jeep vehicle in 2015. Security researchers demonstrated how they could remotely hack into a Jeep's systems through its cellular connection while the vehicle was driving at 70 mph on a highway in St. Louis, USA. The researchers were able to manipulate various Electronic Control Units (ECUs) in the Jeep, including those responsible for functions like wipers, brakes, and engine systems. This hack highlighted the vulnerabilities present in modern automotive systems, showcasing the potential risks associated with remote attacks on vehicles. While the researchers conducted the hack under controlled conditions, the incident raised concerns about the potential life-threatening consequences of remote car hacking in the future.

# Chapter 4

# Threat Modeling of CPS

## 4.1 Threat Modeling

Generally, the practice of discovering and mitigating possible risks to a system is known as threat modeling [25]. In application security, threat modeling is a systematic and ongoing process that aims to enhance the security of a system by creating a security-focused model, identifying potential threats, and developing strategies to address vulnerabilities. By viewing the system from an attacker's perspective, threat modeling helps proactively identify and mitigate security risks to strengthen the system's overall security posture. Early in the Software Development Life Cycle (SDLC), especially during the design phase, threat modeling should ideally be carried out. The threat model must be regularly updated, maintained, and improved as the system changes; this is certainly not a one-time process. Threat modeling should not be considered an optional extra, but rather a typical and important stage in the regular SDLC process.

### 4.1.1 Advantages of Threat Modeling

Before outlining the threat modeling procedures, it is beneficial to address why threat modeling is important. Is it essential to incorporate extra tasks into the development process? What benefits does it offer? The following section provides concise answers to these questions.

- **Early Risk Identification:** Threat modeling looks for possible security problems early in the design process. Because of this, security measures can already be implemented on a system. Compared to trying to find and fix security vulnerabilities after a system is put into production, this is significantly more efficient.
- **Risk Mitigation:** In order to mitigate risk, threat modeling is essential. Organizations can mitigate risks and minimize breaches more economically by anticipating possible threats and taking proactive measures to address them, as opposed to reacting to assaults or breaches after they happen. Furthermore, by being aware of potential attack points, firms may design systems and apps that are more secure from the start.
- **Increased Security Awareness:** Threat modeling is an approach which requires substantial human analysis and decision-making in addition to being strictly technological. Threat

modeling fosters an organizational culture that is security-centric and helps teams become more aware of security issues. Individuals get knowledge about possible threats and countermeasures through this approach.

- **Cost-Effective Security:** Because of threat modeling any organizations can save themselves the expensive consequences of data loss and security breaches by detecting threats early on. Organizations can also make sure they get the best return on their investment by concentrating resources on the biggest risks.
- **Easier Compliance:** Threat modeling is a valuable tool for organizations to meet strict cybersecurity compliance requirements by offering a systematic and documented approach to threat analysis. It enables comprehensive threat analysis, ongoing risk assessment, and process improvement, aligning with the demands of various industries and jurisdictions for robust cybersecurity measures.

## 4.2 CPS Threat Modeling Methodologies

Cyber-physical systems (CPS) face a diverse array of threats that can jeopardize system integrity, impacting cybersecurity vulnerabilities and overall safety. To tackle these challenges, organizations typically utilize one or more threat modeling methods at the outset of the development cycle. Choosing the right threat modeling method is a complex task as it needs to align with both the system's requirements and the organization's needs. Therefore, it is crucial to address two fundamental questions when embarking on this endeavor: the types of available threat modeling method and the essential criteria that an effective threat modeling method should fulfill. The various types of CPS threat modeling techniques are discussed in the following subsection:

### 4.2.1 STRIDE

The STRIDE method is a comprehensive approach to analyzing vulnerabilities within a system by examining each component individually. By assessing potential weaknesses in each system element, such as users, processes, or system elements, STRIDE aims to identify points of exploitation that could be targeted by an attacker. This detailed analysis helps in understanding how an attacker could compromise the entire system by exploiting vulnerabilities in specific components. By addressing these vulnerabilities proactively, organizations can enhance their overall security posture and mitigate the risk of system compromise [26]. The STRIDE method identifies six types of security threats mentioned in below:

1. **Spoofing:** Identity spoofing happens when a hacker impersonates someone else, using their identity and information to commit fraud. A common example of this threat is phishing attack.
2. **Tampering:** Data tampering involves the unauthorized alteration or modification of data or information within a system. This attack occurs when a malicious actor manipulates data to fulfill their objectives, such as gaining unauthorized access, causing system malfunctions, or covering their tracks. Bad actors can carry out data tampering through various methods.
3. **Repudiation:** A repudiation threat arises when a malicious actor perpetrates an attack on a system while denying involvement in the malicious activity. In such instances, the attacker

typically seeks to avoid accountability for their actions, complicating efforts to attribute the attack to a specific individual or entity. Repudiation attacks often exploit vulnerabilities in software, networks, or systems lacking adequate controls for identifying and tracing malicious activities back to the perpetrator.

4. **Information Disclosure:** Information disclosure, also termed as information leakage, occurs when an application or website inadvertently exposes data to unauthorized users. This threat can impact the application's process, data flow, and storage. Examples include inadvertent access to source code files through temporary backups, unnecessary exposure of sensitive data like credit card numbers, and disclosing database information in error messages.

5. **Denial of Service (DoS):** Attacks known as denial of service (DoS) prevent a legitimate user from using resources that they are supposed to have access to. This has an impact on an application's procedure, data flow, and data storage.

6. **Elevation of Privilege:** When an attacker uses flaws to get around security protections and gain unauthorized access or raise their access privileges within a system, this is known as privilege elevation. Attackers can now take advantage of, alter, or steal data for their own benefit.

### 4.2.2 PASTA

PASTA described in [42], stands for Process for Attack Simulation and Threat Analysis, is a risk-centric threat modeling framework developed by Tony UcedaVélez in 2012. This framework provides a structured approach to identifying and analyzing potential threats to an organization's systems and applications. PASTA takes a comprehensive view of security by considering multiple layers of abstraction, ranging from high-level business logic to concrete attack vectors. This model comprises of seven stages [34] as mentioned below:

1. **Define Objectives:** This involves identifying organizational goals, aligning security measures with priorities and also ensures adherence to security standards and regulations.

2. **Define Technical Scope:** This step helps identify all the assets of the system. This step helps in understanding the technological landscape and the components that need to be protected.

3. **Application Decomposition:** Breaking down the system into its components and operations is essential for gaining a comprehensive understanding of how the system functions and where potential vulnerabilities may exist.

4. **Threat Analysis:** This stage involves identifying and analyzing potential threats to the system. By understanding the threats, organizations can better prepare to defend against them.

5. **Vulnerability and Weaknesses Analysis:** This step focuses on identifying vulnerable areas across the system. Mapping these vulnerabilities to an attack tree helps in visualizing how threats could exploit these weaknesses.

6. **Attack Modeling:** This stage involves studying the feasibility of exploiting the identified vulnerabilities. By modeling and simulating potential attacks, organizations can assess the likelihood and impact of security breaches.
7. **Risk and Impact Analysis:** In the final stage, organizations analyze and manage the residual risks that pose major concerns to the system. Mitigation strategies are developed to address these risks and enhance the overall security posture of the system.

### 4.2.3 TRIKE

Trike provides an asset-centric threat modeling approach, generating threat models reliably and consistently. It promotes constructive engagement among pertinent stakeholders by offering a standardized framework for discussing the threats a system may face and how to address them [34]. This model facilitates the following information:

- Engage with system stakeholders to ensure mutual acceptance of risks posed to each asset, fostering a shared understanding and agreement on risk tolerance levels.
- Confirm the effective implementation of risk management measures to address identified risks and vulnerabilities.
- Transparently communicate risk mitigation actions and their impact to stakeholders to foster trust and alignment on security measures.
- Enable stakeholders to proactively reduce risks within their domains by involving them in risk management processes, fostering ownership of security responsibilities and contributing to overall risk reduction.
- Trike adopts a defensive approach to threat modeling, prioritizing understanding the system and its operational environment over solely focusing on attacker's capabilities. Comprehensive system knowledge is crucial for accurately assessing and addressing threats.

### 4.2.4 DREAD

DREAD, developed by Microsoft, assesses threat risk by considering Damage potential, Reproducibility, Exploitability, Affected users, and Discoverability [34] as described below:

- **Damage Potential:** This component evaluates potential of a harm to users and the organization resulting from a successful attack, encompassing financial losses, reputation damage, and other adverse impacts.
- **Reproducibility:** Reproducibility assesses the ease of replicating an attack, gauging the effort needed for successful execution. Attacks with higher reproducibility are deemed riskier.
- **Exploitability:** Exploitability focuses on the likelihood of an attacker being able to take advantage of a vulnerability. It distinguishes between exploits that are easily understood and executable by anyone versus those that require specialized skills.
- **Affected Users:** This component considers the number of users who would be impacted if a threat is realized. Threats that have the potential to affect a larger number of users are typically rated higher in terms of risk.

- **Discoverability:** Discoverability evaluates how easily a vulnerability can be uncovered. It differentiates between vulnerabilities that are difficult to detect versus those that are publicly known. Threats that are easily discoverable are considered higher risk.

### 4.2.5 OCATVE

Operationally Critical Threat, Asset, and Vulnerability Evaluation, or OCTAVE defined in [34], is another asset-centric threat modeling technique that is frequently applied in the information security industry. OCTAVE was created by Carnegie Mellon University's Software Engineering Institute (SEI) to assist enterprises in identifying and controlling risks to their vital assets and information systems. The key aspects of this method can be outlined as below:

- **Operational Risk and Security Practices:** OCTAVE focuses on understanding operational risks within an organization, including threats, vulnerabilities, and potential impacts on critical assets. It also evaluates the organization's security practices and how they align with business objectives.
- **Risk Analysis and Prioritization:** OCTAVE helps organizations identify information-related assets that are crucial to their operations. By focusing on critical assets, OCTAVE enables organizations to prioritize risk analysis activities and develop tailored risk mitigation strategies.
- **Protection Strategy Development:** Through OCTAVE, organizations create a practice-based protection strategy to improve security practices and reduce risks to critical assets. This involves evaluating technologies, processes, and organizational structures to enhance overall security posture.

The OCTAVE methodology utilizes stakeholder's expertise to effectively model security threats, fostering a comprehensive understanding of security posture and informed decision-making to safeguard critical assets. However, OCTAVE's complexity and time-consuming nature may require substantial investment in learning and conducting risk assessments. The extensive documentation associated with OCTAVE may also pose challenges for organizations considering its adoption as a threat modeling approach.

### 4.2.6 LINDDUN

A threat modeling technique with a privacy focus is called LINDDUN. It was created by KU Leuven privacy specialists. Early in the development process, LINDDUN threat modeling is able to recognize privacy threats and countermeasures. Threats to privacy and the mitigation of those threats are the focus of LINDDUN [33]. The threat types of LINDDUN are as follows:

- **Linking:** Deeper understanding of an individual or group can be obtained by combining, associating, or connecting different kinds of data or activities. Combining data like phone numbers, IP addresses, and usernames allows for a more thorough picture of the individual or group. More sophisticated methods, such as profiling, can improve the relationships between data pieces even more. This method of connecting data applies to both people and organizations, providing a more comprehensive image.

- **Identifying:** Identification occurs when someone who is initially unidentifiable becomes identified due to data leakage, inference, or deduction. While identifiable data might be indirectly connected to an identity, identified data can be directly linked to a particular individual.
- **Non-Repudiation:** An individual cannot negate a claim or conduct if it can be linked to them. Regarding the Repudiation threat in STRIDE, non-repudiation is beneficial for security but not for privacy. Logs, signatures, and metadata provide proof that connects acts to certain people.
- **Detecting:** LINDUNN makes it possible to ascertain whether a certain item of interest associated with a data subject is there, even if its contents cannot be interpreted.
- **Data Disclosure:** It involves the over-collection, storage, or processing of personal data, which should only be gathered when absolutely necessary. Excessive data processing includes four types: collecting unnecessary data types, excessive data volume, unnecessary processing, and excessive exposure.
- **Unawareness:** It entails an absence of control, feedback, and transparency. People are ignorant of the gathering and processing of data when there is a lack of transparency. Absence of response indicates incomplete knowledge of privacy concerns. People are unable to access or handle their personal data when they lack control.
- **Non-Compliance:** Improper handling of personal data that violates relevant laws, rules, or policies will be identified as a non-compliance issue.

LINDDUN threat modeling offers three methods for conducting privacy threat assessments, each varying in systematic approach and thoroughness: LINDDUN Go, LINDDUN Pro, and LINDDUN Maestro. In general, it comprises of five steps while conducting privacy assessments:

1. Application of a System Design Framework.
2. Describe the external entities, processes, storage, and data flows of the system.
3. Assess whether the system is threatened or not.
4. Create threat trees by carefully going over each model item one at a time.
5. Evaluate them from the viewpoint of threat categorization.

### 4.2.7 CVSS

The Common Vulnerability Scoring System (CVSS) defined in [5] is an open industry standard used to evaluate the severity of security vulnerabilities. CVSS assigns a score to each vulnerability based on its severity, helping security teams prioritize their remediation efforts. This quantitative measure is designed to be easily understood, even by those without technical expertise. It is simple to translate the numerical score that the CVSS threat model gives into a qualitative representation, such as Low, Medium, High, and Critical. Globally used and updated on a regular basis, the CVSS cybersecurity rating system is managed by the CVSS SIG (Special Interest Group).

There are three metric groups in CVSS:

1. **Base:** The Base metric group reflects the inherent characteristics of a vulnerability that remain constant over time and across different user environments. It consists of two sets of metrics: Exploitability metrics and Impact metrics [1].

2. **Temporal:** Temporal metrics measure the current state of exploit techniques or code availability, the existence of any patches or workarounds, or the confidence in the description of a vulnerability, states FIRST, the group that communicates the CVSS approach. This metric category consists of three metrics: report confidence, remediation level, and exploit code maturity [4].

3. **Environmental:** With the use of these parameters, the consumer analyst can adjust the final score based on how important the impacted IT asset is to the user's company, as determined by the availability, confidentiality, integrity, and usage of alternative or complementary security measures. The metrics are the adjusted Base metrics, and their values are determined by the system's location within the organizational architecture [1].

### 4.2.8 Attack Trees

One of the earliest and most popular methods for modeling threats on cyber-only, cyber-physical, and simply physical systems is the use of attack trees. Attack trees were first used as a standalone technique and have subsequently been integrated with several techniques and frameworks. Diagrams known as "attack trees" show system attacks within a tree structure. The attack's goal is represented by the tree's root, while strategies for achieving it are shown by the leaves. The system threat analysis produces a collection of attack trees since each goal is represented as a separate tree [3].

When dealing with complicated structures, it is possible to construct attack trees for individual components rather than the entire system. In order to make informative security decisions, assess a particular sort of assault, and discover whether the systems are susceptible to an attack, administrators might construct attack trees.

### 4.2.9 Persona Non Grata (PnG)

Persona non Grata (PnG) is an approach that leverages user stereotypes to identify prospective system abusers in order to better understand their motivations and abilities. By encouraging analysts to view the system from the standpoint of accidental use, this technique aids in the early visualization of dangers during the threat modeling phase. PnG assists in detecting vulnerabilities from an attacker's point of view by acquainting a technical expert with the abilities, motives, and objectives of a potential attacker. PnG is a good fit for the Agile methodology, which uses personas to improve security analysis and system design [3].

### 4.2.10 Security Cards

In the context of cybersecurity threat modeling, Security Cards are an inventive and informal way to recognize uncommon and sophisticated threats. Security Cards are not so much a formal method as a brainstorming tool to encourage creative thinking and produce thoughts about possible risks. Using a deck of cards with several situations, attack vectors, or threat descriptions on each card

is how the concept is implemented. To start a conversation about possible security threats, participants in a threat modeling session can choose cards at random or according to predetermined criteria. These cards can depict established attack patterns, newly discovered dangers, or speculative situations that conventional threat analysis techniques might not be able to detect right away [3].

### 4.2.11 Hybrid Threat Modeling Method (hTMM)

The Hybrid Threat Modeling Method (hTMM) [3], created by the SEI in 2018, integrates SQUARE (Security Quality Requirements Engineering Method), Security Cards, and PnG activities. This method aims to ensure no false positives, avoid missed threats, produce consistent results regardless of the analyst, and maintain cost effectiveness.

The method outlined involves the following main steps:

- Identify the system for threat modeling.
- Use Security Cards based on developer input.
- Eliminate unrealistic PnGs (i.e., those without feasible attack vectors).
- Summarize the findings with tool support.
- Proceed with a formal risk assessment method.

### 4.2.12 Quantitive TMM

The Quantitative Threat Modeling Method (Quantitative TMM) is a hybrid approach [3] that combines attack trees, the STRIDE model, and the Common Vulnerability Scoring System (CVSS) to address challenges in threat modeling for cyber-physical systems with intricate interdependencies among their components. The steps are described in below:

1. **Building Component Attack Trees:** The first step in Quantitative TMM involves constructing component attack trees for the five threat categories defined by the STRIDE model. These categories include Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege. By creating attack trees for each category, the method illustrates the relationships and dependencies among different attack vectors and the low-level attributes of system components.
2. **Applying the CVSS Method:** Once the component attack trees are established, the CVSS method is applied to assess the severity of vulnerabilities within the system. CVSS provides a standardized framework for scoring vulnerabilities based on their impact, exploitability, and other factors. By calculating CVSS scores for the components identified in the attack trees, the method quantitatively evaluates the security posture of the system.

### 4.2.13 VAST Modeling

The Visual, Agile, and Simple Threat (VAST) Modeling [3] method is a threat modeling approach based on the ThreatModeler automated platform. VAST is designed to be scalable and user-friendly, making it suitable for adoption in large organizations across their entire infrastructure. It aims to

generate actionable and reliable results that cater to the needs of various stakeholders involved in the security process. VAST requires the creation of two types of models to address the distinct concerns of development and infrastructure teams:

1. **Application Threat Models:** These models utilize process-flow diagrams to depict the system architecture from an architectural perspective. They focus on identifying threats and vulnerabilities within the application design.

2. **Operational Threat Models:** Operational models are developed from an attacker's viewpoint based on Data Flow Diagrams (DFDs). These models emphasize understanding potential threats and attack vectors that could target the operational aspects of the system.

By creating both application and operational threat models, VAST enables the seamless integration of threat modeling into the organization's development and DevOps lifecycles.

## 4.3 Existing CPS Threat Modeling Solutions

The process of listing and ranking the risks associated with malicious agents, their attacks, and the potential effects those attacks might have on a system's assets is known as threat modeling. Implementing threat modeling provides four primary advantages. First, it will help prioritize different kinds of attacks to deal with. Secondly, it will aid in risk mitigation more successfully. Thirdly, it will add fresh potential points of attack to assessments. Lastly, it will pinpoint important vulnerabilities that reveal essential business assets as well as defects in business logic [40]. Let's talk about a few of the threat models for CPS that have already been defined, in the subsections below.

### 4.3.1 A Review of Asset-Centric Threat Modelling Approaches

The following content is taken from the paper by Livinus and Stephen [34]. The selection of a threat modeling approach depends on the specific business goals. Therefore, the initial step in choosing a threat modeling technique for a system is to thoroughly comprehend its intended functionality. There aren't inherently good or bad threat modeling methods; rather, there are appropriate and inappropriate approaches depending on the system in question. Three primary threat modeling approaches exist: asset-centric, attack-centric, and software/system-centric. Asset-centric threat modeling centers on the system's assets, analyzing potential information loss or business impact. It extends to identifying attacker motives and discovering security gaps in the system environment. While it doesn't address flaws or insecure practices directly, it can uncover potential threat scenarios.

The authors of this paper has described about some state of the art asset centric threat modeling techniques: DREAD, Trike, Octave and PASTA. According to the authors, asset-centric threat modelling offers several benefits in enhancing information security measures within an organization and it is also having some disadvantages which is described in below:

- **Advantages:**

    1. Asset-centric threat modelling helps organizations identify and prioritize their critical assets, including data, systems, and intellectual property.
    2. By analyzing threats in relation to specific assets, asset-centric threat modelling allows organizations to prioritize risks based on the potential impact on critical assets.

3. This threat modelling enables organizations to develop tailored mitigation strategies that are specific to the assets being protected and also includes assessing the potential business impact of security threats on critical assets.
4. Asset-centric threat modelling encourages collaboration among stakeholders from different departments within the organization.
5. Asset-centric threat modelling is an iterative process that allows organizations to continuously monitor and update their threat models based on evolving threats and changes in the business environment. This continuous improvement approach enhances the organization's overall security posture.

- **Disadvantages:**

1. One limitation of asset-centric threat modeling is that it may not capture all possible threats and vulnerabilities that could impact the organization. This approach focuses primarily on identifying and mitigating risks to specific assets, which could result in overlooking broader systemic vulnerabilities or interconnected threats that affect multiple assets simultaneously.
2. Furthermore, asset-centric threat modeling may require significant resources and effort to identify, prioritize, and mitigate risks to each individual asset. This intensive process could be challenging for organizations with limited resources or complex infrastructures, leading to potential gaps in threat coverage or delays in implementing mitigation measures.
3. Another limitation is that asset-centric threat modeling may not effectively address emerging or unknown threats that do not align with the predefined assets or risk scenarios.

### 4.3.2 Assessment of Potential Security Risks in Advanced Metering Infrastructure Using the OCTAVE Allegro Approach

The following content is taken from the paper by Ali et al. [10]. An essential element of a smart grid is the advanced metering infrastructure (AMI). It combines information and communication technologies with a traditional electricity grid. Before implementing security measures, it's crucial to conduct an information security risk assessment (ISRA) to identify potential risks associated with the valuable data stored in the AMI system. The paper employs the Octave Allegro (OA) risk assessment approach, which integrates qualitative and quantitative methods using relative risk scores. It also utilizes containers to represent data storage, transmission, and processing locations. The risk assessment task with OA consists of four stages encompassing eight steps across ten worksheets.

The various stages of the OA approach are illustrated in the diagram below.

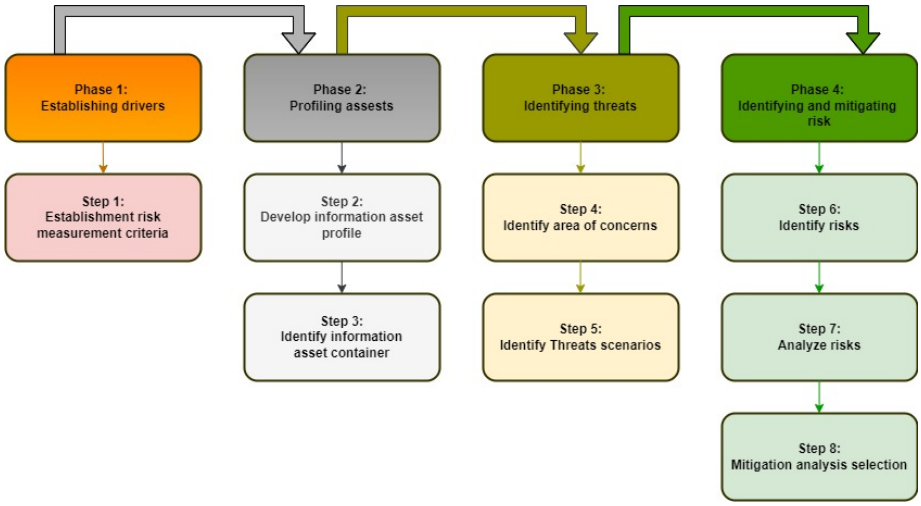



Figure 4.1: Phases of OCTAVE Allegro

AMI systems will undergo assessment using version 1 of the OA worksheets. These worksheets are utilized across different stages of the OA methodology. In the initial stage, Worksheet 1 is employed to identify and prioritize measurement areas according to their significance. From Worksheet 1, three impact areas vulnerable to cyberattacks have been identified.

The second step in organizational analysis involves profiling assets, where each information asset is documented along with its container. This process aims to achieve various objectives, including the identification of critical information assets.

The third phase of the OA approach involves identifying threats, which comprises two steps: pinpointing areas of concern and defining attack scenarios. The initial step involves determining potential incidents that could affect an information asset within an AMI system. This is detailed in the first section of worksheet 10. The threat scenario outlines the threat actor, attack method, motivation, and outcomes of the attack.

In the fourth stage of the OA methodology, the focus shifts to establishing the relative risk score for every risk scenario linked to each information container. This process is outlined in the second part of worksheet 10.

In the last stage of the OA method, mitigation strategies are chosen based on a risk matrix. Risks are categorized into pools based on their risk scores. Mitigation options include acceptance, deferral, mitigation, and transfer. Risks with minimal impact are accepted, while countermeasures are employed to reduce others to acceptable levels. Some risks are deferred to gather more data for further analysis.

### 4.3.3 STRIDE-Based Threat Modeling for Cyber-Physical Systems

The following content is taken from the paper by Khan et al. [26]. Thorough threat modeling during the design and validation phases is crucial for ensuring the reliable operation of complex Cyber-Physical Systems (CPS), including critical infrastructures and industrial control systems. This paper introduces a thorough threat modeling framework for Cyber-Physical Systems (CPS) utilizing the STRIDE methodology, which systematically addresses security concerns at the component level.

Initially, it outlines a practical and efficient approach for implementing STRIDE and subsequently applies it to a genuine synchrophasor-based synchronous islanding testbed in a laboratory setting.

The authors of this paper has mentioned the below five steps for the purpose of threat modeling.

1. **Decompose System into Components:** This involves breaking down the system into its individual components. It's important to emphasize that threat analysis focuses solely on elements vulnerable to cyber attacks and does not include physical components impervious to such threats.

2. **Plot DFD for System Components:** In this step of threat modeling involves creating a Data Flow Diagram (DFD) for each component within the system. This diagram visually represents the flow of data within the system, highlighting how different components interact with each other and with external entities. Each system component is depicted, along with the data flows between them, allowing for a comprehensive understanding of the system's architecture and functionality.

3. **Analyze Threats in the DFD:** To effectively analyze threats, the initial step involves identifying potential attacker motivations or the consequences of those threats. This requires leveraging the expertise about the system to understand what adversaries might aim to achieve or what harm they could inflict.

4. **Identify Vulnerabilities:** Detecting vulnerabilities is crucial for devising robust security strategies. It's imperative to identify potential weaknesses to ensure that security measures are comprehensive and effective.

5. **Plan Mitigation Strategies:** The STRIDE approach provides valuable, easily comprehensible results for system designers, aiding in the development of suitable security solutions. These outputs can inform risk analysis, identifying critical threats and guiding the implementation of effective mitigation strategies.

### 4.3.4 Using Hybrid Attack Graphs to Model Cyber Physical Attacks in the Smart Grid

The following content is taken from the paper by Hawrylak et al. [22]. This paper introduces a cyber-physical attack aimed at a substation within the Smart Grid, a vital component of critical infrastructure. The attack involves inducing overheating in a transformer and is depicted using a hybrid attack graph (HAG), which integrates both cyber and physical aspects of the attack. Through the HAG, the paper offers insights into various potential attack routes. The researchers utilized specialized software for HAG generation, which constructs the HAG by correlating exploit patterns with specific system conditions. The provided figure demonstrates how a HAG is created specifically for an energy-draining attack. The program requires two input files: a CPS description file and an exploit file. The CPS description file outlines details about the system's devices, network layout, software, operating systems, and specific system qualities like temperature alarm thresholds. On the other hand, the exploit file consists of a series of exploits, each with prerequisites that must be met before execution, and descriptions of the alterations made to the system's network state after the exploit is executed.

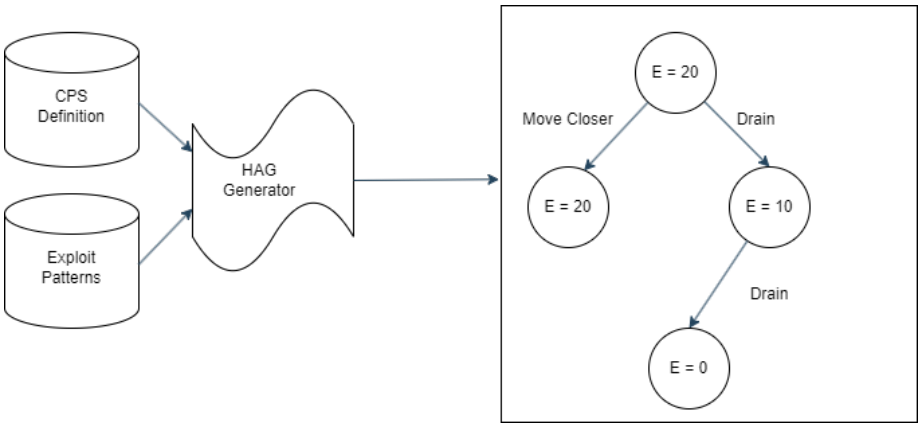



Figure 4.2: Hybrid Attack Graph Generation [22]

### 4.3.5 A2G2V: Automatic Attack Graph Generation and Visualization and Its Applications to Computer and SCADA Networks

The following content is taken from the paper by Ghazo et al. [6]. This paper introduces A2G2V, an automated attack graph generator and visualizer. A2G2V utilizes model-checking tools, an architecture description tool, and custom code to create attack graphs, detailing potential sequences of vulnerabilities exploited to compromise system security. The architecture description tool formalizes the networked system, vulnerabilities, and security properties. A model-checker identifies attack sequences as counterexamples, which are then processed by custom code to reveal all possible attack paths through specification relaxation. Finally, a visualization tool generates graphical representations of the attack graph.

The CPS case study involving a water treatment system demonstrated the efficacy of the A2G2V algorithm in assessing the security of a real-world CPS scenario. In this system, consisting of 12 hosts with designated roles and services, vulnerabilities were identified and encoded using AADL+AGREE. The algorithm successfully produced an attack graph, focusing on scenarios where specific hosts could be compromised. Initially employing a default search depth, the algorithm iteratively adjusted it to uncover all potential attack paths. By amalgamating these paths into an attack graph via GraphViz, it visually depicted vulnerabilities and possible attack sequences within the CPS. This case study underscores the practical application of the A2G2V algorithm in bolstering CPS security, offering automated generation of attack graphs and visualizing potential threats, thus enhancing overall security posture and highlighting exploitable vulnerabilities.

### 4.3.6 A Method for Identifying Critical Elements of a Cyber-Physical System Under Data Attack

The following content is taken from the paper by Fan et al. [16]. The researchers employed hyper-network techniques to pinpoint the vital components of a cyber-physical system (CPS) when subjected to data attacks as per IEC 61850 standards. The Hyper-Network Model of the Substation Auto System (SAS) is developed using hyper-graph theory. This model focuses on capturing the functional attributes of logical nodes within the SAS, allowing for a detailed representation of

the complex interdependencies and interactions among different components. By extending this model to a Cyber-Physical System (CPS) framework, the analysis includes the functional influence after a data attack, providing insights into the system's behavior under security threats. Efficiency indexes are introduced to evaluate the CPS's effectiveness under data attacks, enabling a structured assessment of the system's resilience and performance. The Hyper-Network Model aims to identify critical elements vulnerable to data attacks and enhance vulnerability analysis in cyber-physical systems.

By introducing efficiency indexes and analyzing the network and hyper-network models, the method aims to evaluate the system's resilience and performance under data attacks. The evaluation process provides insights into the damage caused by attacks on different logical nodes within substations, highlighting the importance of considering protection measures such as firewalls and gateways in future studies. The method is designed to guide white box experiments of defense technology against data attacks and emphasizes the need to study more complex composite data attack strategies targeting multiple logical node types in the future.

# Chapter 5

# Hypergraph Based Threat Models for CPS

## 5.1 Attack Graph

Attack graphs are popular modeling tool for multistage attacks. It encodes system security states and attacker actions in form of dyadic relations resulting in a graph. Given the information on system components and their vulnerabilities, network topology and configuration, we can generate attack graphs using open source tools such as MulVAL [2].

Formally, an attack graph is defined as a directed graph $G = (V, E)$. The set of vertices comprises of two types, i.e. $V = E_x \cup S_c$, where $E_x$ is the set of exploits and $S_c$ is the set security conditions. Similarly, the set of edges comprises of two types, i.e. $E = R_r \cup R_i$. Require type directed edges connect security conditions to exploits, i.e. $R_r \subseteq S_c \times E_x$ and imply type directed edges connect exploits to security conditions, i.e. $R_i \subseteq E_x \times S_c$.

## 5.2 Attack Hypergraph

In this section, we present the attack hypergraph formalism. It extends the traditional attack graphs to capture more complex attack scenarios as higher-order relations. To set the context, we first define a directed hypergraph - followed by a suitable example.

In an attack hypergraph, hyperedges are used to represent multi-step attack paths involving multiple vulnerabilities, while nodes represent system states or configurations. In a traditional attack graph, nodes represent individual system states (e.g., configurations, services, or hosts), and edges represent the direct exploitability of a vulnerability to move between states. However, attack graphs often fail to represent more sophisticated attack scenarios that involve sequences of vulnerabilities or multi-step attack chains.

Attack hypergraphs address this limitation by using hyperedges to connect multiple nodes, representing the potential for a series of exploitations or attack steps in a single hyperedge. This allows them to capture the interactions and dependencies among different vulnerabilities and provide a more comprehensive representation of the attack surface and potential attack paths.

The hyperedges allow capturing higher-order relationships and dependencies among nodes, leading to a more expressive representation of complex attack scenarios. [17]

**Definition 1 (Directed Hypergraph)** *A directed hypergraph $\mathcal{H}$ is defined as a pair $(\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is a non-empty set of vertices and $\mathcal{E}$ is a set of hyperedges. Each $e \in \mathcal{E}$ is an ordered pair $(e^h, e^t)$, where $e^h$ and $e^t$ are non-empty subsets of $\mathcal{V}$ and are respectively called the head and tail of the edge $e$.*

## 5.3 Attack Hypergraph of CPS: An Example Scenario

In this section we take an example network of a water treatment system (as given in [6]) and show the corresponding hyper attack graph.

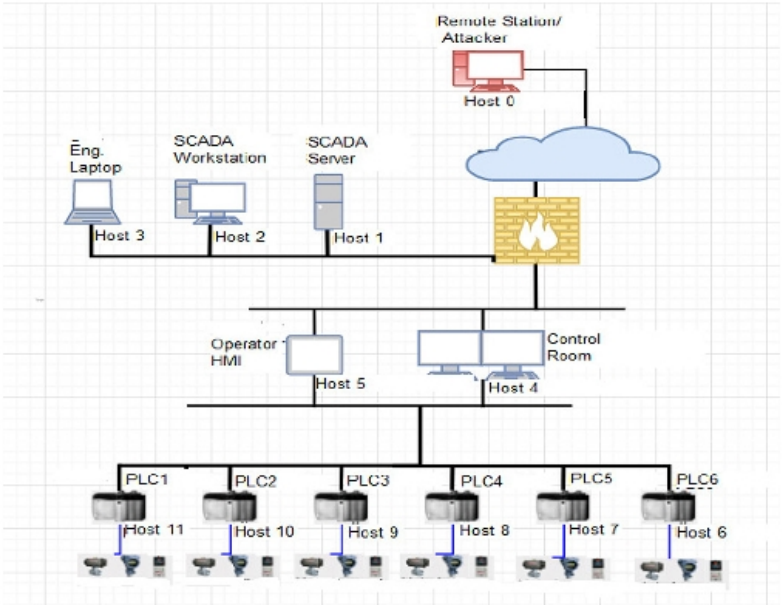



Figure 5.1: SCADA Network Architecture of a Water Treatment System

The system comprises of 12 hosts: Host0 is the remote station and is the attacker's assumed initial location; Host1 is the SCADA server; Host2 is the SCADA workstation; Host3 is the engineer's laptop; Host4 is the operator human machine interface (HMI); Host5 is the control room; Host6 to Host11 are the programmable logic controllers (PLCs) that control the physical process through sensors and actuators. The PLCs (hosts 6–11) are the attacker's main target.

The firewall separates Host0, Host1, Host2, Host3 and the rest of the system. The firewall restricts the access from the remote station to Host1, Host2, and Host3. Also, it allows access from Host1, Host2, and Host3 to the rest of the system. The communication with the PLCs goes through Host5 and Host6. The PLCs communicate with Host5 and Host6 but not with each other.

Table 5.1 provides a list of vulnerabilities of different hosts in the SCADA network. Table 5.2 lists possible attacks that can be launched to exploit those vulnerabilities.

| **Host No.** | **Host Name** | **Vulnerability** |
|---|---|---|
| Host 1 | SCADA Server | "The SCADA server is a SINEMA server, in which:a) iws: the integrated Web servers at Port 4999/TCP and Port 80/TCP may allow unauthenticated remote code execution if an attacker has a network access to the server; b) lmw: local Microsoft Windows operating system users can escalate their privileges if the affected products are not installed under their default path (C:\\ProgramFiles\\∗ or the localized equivalent)." |
| Host 2 | SCADA Workstation | "The SCADA workstation is a SIMATIC Wincc flexible runtime, in which: a) rmm: the remote management module of SIMATIC Wincc flexible panel transmits weakly protected credentials over the network. Attackers capturing network traffic of the remote management module can reconstruct the credentials; b) inws: Siemens Wincc flexible allows remote attackers with user access to inject arbitrary Web script or HTML via unspecified vectors." |
| Host 3 | Engineer's Laptop | "The engineering Laptop runs a Wincc TIA portal, in which: a) bac: when a user logs in, the application sets predictable authentication token/cookie values. This can allow an attacker to bypass authentication checks; b) inws: Wincc TIA portal allows remote attackers to inject arbitrary Web script or HTML via unspecified vectors." |
| Host 4 | Control Room | "The control center runs a Wincc runtime professional, in which: a) rmm: the remote management module of SIMATIC Wincc runtime professional transmits weakly protected credentials over the network. Attackers capturing network traffic of the remote management module can reconstruct the credentials; b) inws: Siemens Wincc flexible allows remote attackers to inject arbitrary Web script or HTML via unspecified vectors." |
| Host 5 | Operator HMI | "The HMI runs a Wincc runtime advanced, in which: a) rmm: the remote management module of Wincc runtime advanced transmits weakly protected credentials over the network. Attackers capturing network traffic of the remote management module can reconstruct the credentials; b) inws: Siemens Wincc runtime advanced allows remote attackers to inject arbitrary Web script or HTML via unspecified vectors." |
| Host 6 | PLC 6 | "The PLCs are Siemens SIMATIC S7-1500 CPU PLC devices, in which: a) rng: the random-number generator on Siemens SIMATIC S7-1500 CPU PLC devices does not have sufficient entropy, which makes it easier for remote attackers to defeat cryptographic protection mechanisms and hijack sessions via unspecified vectors." |
| Host 7 | PLC 5 | -do- |
| Host 8 | PLC 4 | -do- |
| Host 9 | PLC 3 | -do- |
| Host 10 | PLC 2 | -do- |
| Host 11 | PLC 1 | -do- |

Table 5.1: Host Vulnerability Lists

| | |
|---|---|
| 1 | Remote Code Execution (rce): This attack exploits the vulnerability in SINEMA integrated Web servers at Port 4999/TCP and Port 80/TCP if an attacker has network access to the server. This attack gives the attacker user access to the attacked host. |
| 2 | Unquoted Service Paths (usp): This attack exploits the vulnerability in SINEMA server windows operating system which allows a local user to escalate their privileges to gain root access to the attacked host. |
| 3 | User Credentials Construction (ucc): This attack exploits the weakly protected credentials in SIMATIC Wincc flexible, SIMATIC Wincc advanced, and SIMATIC Wincc professional. This attack gives the attacker user access to the attacked host. |
| 4 | Cross-Site Scripting (xss): This attack exploits the Web script and HTML code injection vulnerability in Wincc TIA portal, SIMATIC Wincc flexible, SIMATIC Wincc advanced, and SIMATIC Wincc professional. This attack gives the attacker root access to the attacked host. |
| 5 | Authentication Token/Cookie (atc): This attack exploits the predictable authentication token/cookie values in a Wincc TIA portal. This attack gives the attacker root access to the attacked host. |
| 6 | Cryptographic Protection Mechanisms (cpm): This attack exploits the vulnerability in SIMATIC S7-1500 CPU random-number generator if the attacker has network access to the PLCs. This attack gives the attacker root access to the attacked host. |

Table 5.2: Possible Atomic Attacks

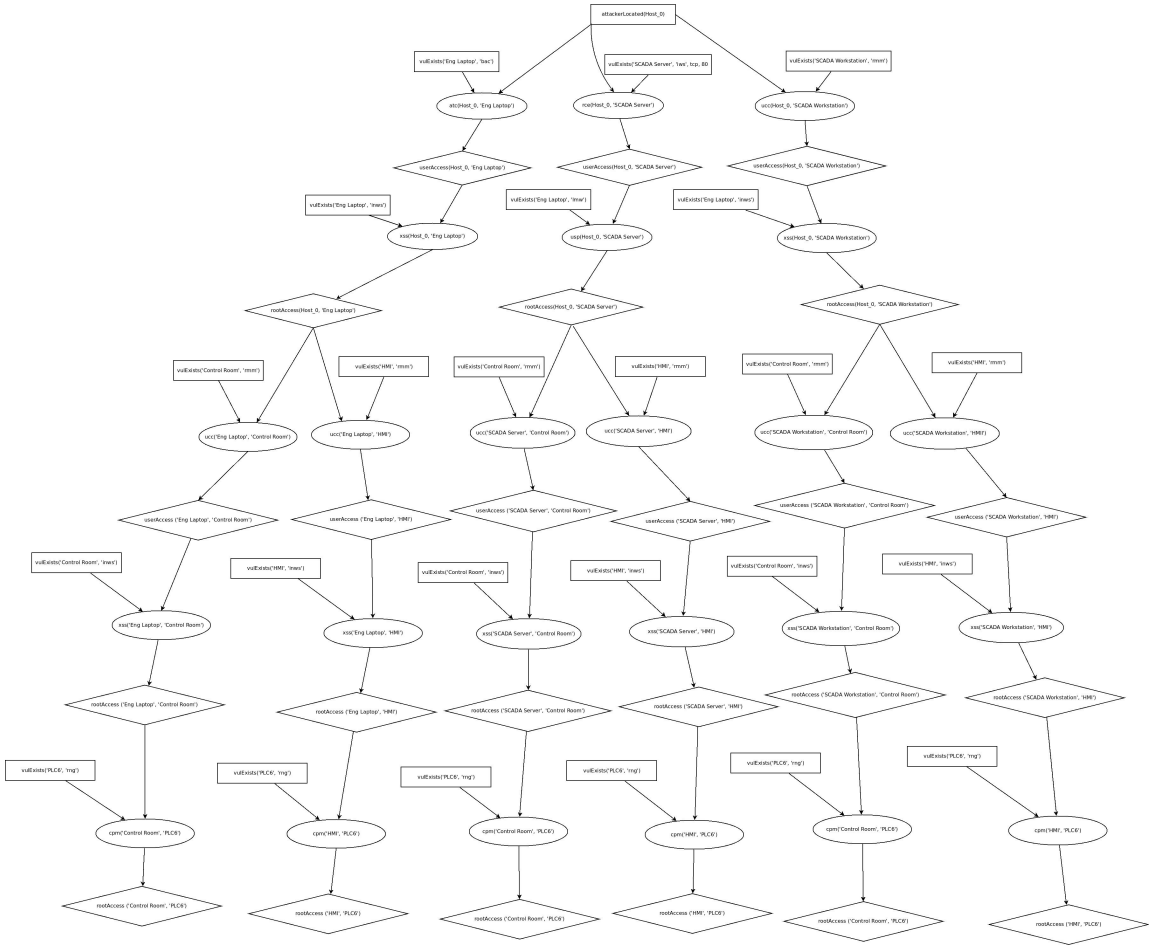



Figure 5.2: Attack Hyper Graph of SCADA Network Architecture of Figure 5.1

# Chapter 6

# Conclusion and Future Work

## 6.1 Conclusion

This thesis presents a comprehensive exploration of utilizing hypergraphs for effective threat modeling in Cyber Physical Systems (CPS). By integrating physical and digital elements, the thesis addresses the critical security issues faced by modern CPS architectures. Through the innovative use of hypergraphs, higher-order interactions between CPS components are visualized, enabling a more accurate understanding of potential threats and vulnerabilities.

The contributions of this thesis lie in its in-depth discussion of CPS architecture, security challenges, existing threat modeling methods, and the application of hypergraphs to enhance threat modeling for CPS. By leveraging hypergraphs, the thesis demonstrates a novel approach to mitigating risks in CPS by capturing complex relationships between system components.

Looking ahead, the research opens ways for further exploration in the realm of CPS security and threat modeling. Future work could focus on refining hypergraph-based models, conducting more extensive experiments, and exploring additional applications of hypergraphs in cybersecurity. By continuing to innovate in this field, researchers can enhance the resilience of CPS against evolving cyber threats and contribute to the advancement of secure digital infrastructures.

Overall, the thesis underscores the importance of adopting advanced modeling techniques like hypergraphs to expand the security status of Cyber Physical Systems, paving the way for more robust and resilient systems in the face of escalating cybersecurity challenges.

## 6.2 Future Work

Future work in the field of modeling threats to Cyber Physical Systems (CPS) using hypergraphs presents exciting opportunities for further research and development. Some potential directions for future work include:

1. **Refinement of Hypergraph Models:** Improving the methods and procedures used to create and evaluate hypergraphs in order to increase the accuracy and efficacy of threat models for CPS that are based on hypergraphs. This might require exploring new approaches for capturing more complex interrelationships amongst CPS elements.

2. **Integration of Machine Learning:** Incorporating machine learning algorithms into hypergraph-based threat modeling to enable automated threat detection, prediction, and response in CPS environments. This fusion of hypergraphs and machine learning could lead to more adaptive and proactive cybersecurity measures.

3. **Real-World Implementation:** Conducting real-world implementations and case studies to validate the effectiveness of hypergraph-based threat models in practical CPS scenarios. This would involve collaborating with industry partners to deploy and evaluate the models in operational environments.

4. **Scalability and Performance:** Addressing scalability and performance challenges associated with applying hypergraphs to large-scale CPS networks. Future work could focus on optimizing hypergraph algorithms to handle the complexity and volume of data generated by extensive CPS infrastructures.

5. **Interdisciplinary Research:** Encouraging interdisciplinary research collaborations between cybersecurity experts, network engineers, data scientists, and domain-specific professionals to leverage diverse perspectives in advancing hypergraph-based threat modeling for CPS. This interdisciplinary approach can lead to more holistic and effective solutions.

6. **Standardization and Best Practices:** Developing standardized frameworks and best practices for hypergraph-based threat modeling in CPS to promote consistency, interoperability, and knowledge sharing within the cybersecurity community. Establishing common guidelines can facilitate the adoption and implementation of hypergraph models across different CPS domains.

# Bibliography


[1] First forum of incident response and security teams, https://www.first.org/cvss/specification-document.

[2] Mulval: Multi-host, multi-stage vulnerability analysis tool. `https://github.com/risksense/mulval`. Accessed: 2023-09-10.

[3] Modeling: 12 available methods2018, https://insights.sei.cmu.edu/blog/threat-modeling-12-available-methods/, Dec. 2018.

[4] Balbix, https://www.balbix.com/insights/temporal-cvss-scores/, Aug. 2020.

[5] Centraleyes, https://www.centraleyes.com/glossary/common-vulnerability-scoring-system/, Mar. 2023.

[6] A. T. Al Ghazo, M. Ibrahim, H. Ren, and R. Kumar. A2g2v: Automatic attack graph generation and visualization and its applications to computer and scada networks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 50(10):3488–3498, 2020.

[7] R. Alguliyev, Y. Imamverdiyev, and L. Sukhostat. Cyber-physical systems and their security issues. *Computers in Industry*, 100:212–223, 2018.

[8] X. An, J. Su, X. Lü, and F. Lin. Hypergraph clustering model-based association analysis of ddos attacks in fog computing intrusion detection system. *EURASIP Journal on Wireless Communications and Networking*, 2018, 10 2018.

[9] A. Antelmi, G. Cordasco, M. Polato, V. Scarano, C. Spagnuolo, and D. Yang. A survey on hypergraph representation learning. *ACM Comput. Surv.*, 56(1), aug 2023.

[10] A. I. Awad, M. Shokry, A. A. Khalaf, and M. K. Abd-Ellah. Assessment of potential security risks in advanced metering infrastructure using the octave allegro approach. *Computers and Electrical Engineering*, 108:108667, 2023.

[11] R. Baheti and H. Gill. Cyber-physical systems. *The impact of control technology*, 12(1):161–166, 2011.

[12] F. Baiardi, S. Suin, and C. Telmon. A mathematical framework to assess the security of an information infrastructure. *First italian workshop on privacy and security, Roma*, 01 2006.

[13] Q. Dai and Y. Gao. *Introduction*, pages 1–17. Springer Nature Singapore, Singapore, 2023.

[14] S. M. Dibaji, M. Pirani, D. B. Flamholz, A. M. Annaswamy, K. H. Johansson, and A. Chakrabortty. A systems and control perspective of cps security. *Annual reviews in control*, 47:394–411, 2019.

[15] W. Duo, M. Zhou, and A. Abusorrah. A survey of cyber attacks on cyber physical systems: Recent advances and challenges. *IEEE/CAA Journal of Automatica Sinica*, 9(5):784–800, 2022.

[16] Y. Fan, J. Li, and D. Zhang. A method for identifying critical elements of a cyber-physical system under data attack. *IEEE Access*, 6:16972–16984, 2018.

[17] G. Gallo, G. Longo, S. Pallottino, and S. Nguyen. Directed hypergraphs and applications. *Discrete Applied Mathematics*, 42(2):177–201, 1993.

[18] Y. Gao, Y. Peng, F. Xie, W. Zhao, D. Wang, X. Han, T. Lu, and Z. Li. Analysis of security threats and vulnerability for cyber-physical systems. In *Proceedings of 2013 3rd International Conference on Computer Science and Network Technology*, pages 50–55. IEEE, 2013.

[19] GeeksforGeeks. Hypergraph its representation discrete mathematics, Nov. 2022.

[20] G. Granadillo, E. Doynikova, I. Kotenko, and J. Garcia-Alfaro. Hypergraph-driven mitigation of cyber-attacks. *Internet Technology Letters*, 1:e38, 03 2018.

[21] A. Guzzo, A. Pugliese, A. Rullo, and D. Saccà. Intrusion detection with hypergraph-based attack models. In M. Croitoru, S. Rudolph, S. Woltran, and C. Gonzales, editors, *Graph Structures for Knowledge Representation and Reasoning*, pages 58–73, Cham, 2014. Springer International Publishing.

[22] P. J. Hawrylak, M. Haney, M. Papa, and J. Hale. Using hybrid attack graphs to model cyber-physical attacks in the smart grid. In *2012 5th International Symposium on Resilient Control Systems*, pages 161–164, 2012.

[23] N. Jazdi. Cyber physical systems in the context of industry 4.0. pages 1–4, 05 2014.

[24] C. A. Joslyn, S. Aksoy, D. Arendt, J. Firoz, L. Jenkins, B. Praggastis, E. Purvine, and M. Zalewski. Hypergraph analytics of domain name system relationships. In B. Kamiński, P. Prałat, and P. Szufel, editors, *Algorithms and Models for the Web Graph*, pages 1–15, Cham, 2020. Springer International Publishing.

[25] S. M. Khalil, H. Bahsi, and T. Korõtko. Threat modeling of industrial control systems: A systematic literature review. *Computers & Security*, page 103543, 2023.

[26] R. Khan, K. McLaughlin, D. Laverty, and S. Sezer. Stride-based threat modeling for cyber-physical systems. In *2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, pages 1–6, 2017.

[27] G. Lampropoulos and K. Siakas. Enhancing and securing cyber-physical systems and industry 4.0 through digital twins: A critical review. *Journal of Software: Evolution and Process*, 35(7):e2494, 2023.

[28] E. A. Lee. The past, present and future of cyber-physical systems: A focus on models. *Sensors*, 15(3):4837–4869, 2015.

[29] J. Lee, B. Bagheri, and H.-A. Kao. A cyber-physical systems architecture for industry 4.0-based manufacturing systems. *Manufacturing letters*, 3:18–23, 2015.

[30] H. Li, M. Gao, F. Zhou, Y. Wang, Q. Fan, and L. Yang. Fusing hypergraph spectral features for shilling attack detection. *Journal of Information Security and Applications*, 63:103051, 2021.

[31] Z.-Z. Lin, T. D. Pike, M. M. Bailey, and N. D. Bastian. A hypergraph-based machine learning ensemble network intrusion detection system, 2023.

[32] A. Napoleone, M. Macchi, and A. Pozzetti. A review on the characteristics of cyber-physical systems for the future smart factories. *Journal of Manufacturing Systems*, 54:305–335, 2020.

[33] Nick. Linddun threat modeling - threat-modeling.com, https://threat-modeling.com/linddun-threat-modeling/, Sept. 2023.

[34] L. O. Nweke and S. D. Wolthusen. A review of asset-centric threat modelling approaches. *International Journal of Advanced Computer Science and Applications*, 11(2), 2020.

[35] C. Peng. Research and construction model of cyberspace based on hypergraph. In *Proceedings of the 2016 5th International Conference on Advanced Materials and Computer Science*, pages 454–459. Atlantis Press, 2016/06.

[36] L. Pu and B. Faltings. Hypergraph clustering for better network traffic inspection. In *International Joint Conference on Artificial Intelligence*, 2011.

[37] L. F. Ribas Monteiro, Y. R. Rodrigues, and A. C. Zambroni de Souza. Cybersecurity in cyberndash;physical power systems. *Energies*, 16(12), 2023.

[38] R. E. Sawilla. *Ranks and Partial Cuts in Forward Hypergraphs*. PhD thesis, Queen's University, April 2011.

[39] N. Shevchenko, B. R. Frye, and C. Woody. Threat modeling for cyber-physical system-of-systems: Methods evaluation. *Software Engineering Institute: Pittsburgh, PA, USA*, 2018.

[40] J. Steven. Threat modeling - perhaps it's time. *IEEE Security  Privacy*, 8(3):83–86, 2010.

[41] S. Walker-Roberts, M. Hammoudeh, O. Aldabbas, M. Aydin, and A. Dehghantanha. Threats on the horizon: Understanding security threats in the era of cyber-physical systems. *The Journal of Supercomputing*, 76:2643–2664, 2020.

[42] A. Wolf, D. Simopoulos, L. D'Avino, and P. Schwaiger. The pasta threat model implementation in the iot development life cycle. 2021.