Measuring and Improving the Security Postures of Smart Grid Substations Against Cyber Attacks

Onur Duman

A Thesis

in

The Concordia Institute

for

Information Systems Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy (Information Systems Engineering) at

Concordia University

Montréal, Québec, Canada

December 2023

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Mr. Onur Duman**

Entitled: **Measuring and Improving the Security Postures of Smart Grid Substations Against Cyber Attacks**

and submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy (Information Systems Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

_____ Chair
Dr. Mustafa K. Mehmet Ali

_____External Examiner
Dr. Rongxing Lu

_____External to Program
Dr. Otmane Ait Mohamed

_____Examiner
Dr. Mohsen Ghafouri

_____Examiner
Dr. Amr Youssef

_____Thesis Supervisor
Dr. Lingyu Wang

_____Thesis Supervisor
Dr. Mourad Debbabi

Approved by _____
Dr. Jun Yan, Graduate Program Director (Ph.D.)

_____ 2023

_____
Prof. Mourad Debbabi, Dean
Gina Cody School of Engineering and Computer Science

Abstract

Measuring and Improving the Security Postures of Smart Grid Substations Against Cyber Attacks

Onur Duman, Ph.D.

Concordia University, 2024

Substations, which are utilized to bring voltage levels from high-voltage to low-voltage while ensuring monitoring, protection, and control, are critical components of power systems. Digital substations, in which the operation is managed between intelligent electronic devices (IED), are utilized in the smart grid. Compromising those substations can lead to major consequences, such as transmission line failures or blackouts. Since cyber-attacks against substations can lead to serious physical consequences, securing substations requires quantitative security metrics that consider the cyber and physical aspects of attacks. The goal of designing security metrics is to measure how defense-ready critical systems are so that attacks can be prevented from succeeding in the first place, rather than being detected in later stages. This Ph.D. thesis aims to develop security metrics for substations, provide a framework for improving their security posture, which is measured with the developed security metrics, and provide an online security monitoring framework. More concretely, the first chapter provides a literature review on threat models and security metrics in the smart grid. Consequently, two security metrics are defined to measure how well redundancy is designed from a security perspective. The effectiveness of those metrics is assessed via simulations conducted using realistic attack graph models. After that, three security metrics are provided to measure the security postures of substations concerning supply chain attacks, and their effectiveness is assessed via simulations. Substations may contain devices from different vendors which may be more or less trustworthy. Based on that, measuring the security postures of substations against supply chain attacks is important. Then, a hardening framework is developed to improve

the security postures of substations with respect to supply chain attacks. This framework considers supply chain-related and non-supply chain-related hardening options for finding optimal ways to improve the security postures of substations. The effectiveness of the designed framework is experimented with using realistic scenarios. Lastly, an online security monitoring framework which first generates threat models and then enhances them according to ongoing instances of attacks is provided. This framework first generates threat models based on the static configuration. After that, the generated models are updated based on information about ongoing instances of attacks obtained from system logs. Experiment results highlight that the designed framework scales well and has a reasonable execution time. Finally, even though securing substations is the main focus of this thesis, quantitative security metrics and tools defined in this thesis can potentially be used for other critical systems such as microgrids.

Acknowledgments

First, I would like to thank my supervisors, Prof. Mourad Debabbi and Prof. Lingyu Wang, for their great guidance during this research. I would like to thank Prof. Mourad Debbabi for his continuous support, his inspiring ideas about my research, for providing a nice work environment, for challenging me to perform better research, and for providing us with a great department as the dean. I would like to thank Prof. Lingyu Wang for his continuous encouragement, amazing ideas, constant guidance, motivation even in difficult times during this research, his patience, and his time spent reviewing my works in detail. I am delighted to have two such great supervisors from whom I learned a lot. I hope such a great collaboration continues after my graduation.

Second, I would like to thank my collaborators and lab mates. Special thank goes to Dr. Mengyuan Zhang for her collaboration in creating the first and last contributions of this thesis. Another special thanks goes to Dr. Mohsen Ghafouri for his collaboration in creating the second contribution. In addition, another special thanks goes to Dr. Azadeh Tabiban for her collaboration in creating the third contribution. In addition to my collaborators, I would like to thank my labmates, Saghar Vahidi, Afshin Ebtia, Abolfazl Rahiminejad, Dhiaa Elhak Rebbah, Mark Karanfil, Olivier Cabana, Abdullah Qasem, Pratyusha Bhattacharya, Juanwei Chen, Hang Du, Anis Lounis, Sofiane Benahmed, Dr. Mouatez Karbab, Dr. Andrei Soeanu, and others for their support, friendship, and creating a great work environment.

Third, I would like to thank my former Master's supervisor Prof. Amr Youssef for being my lifetime mentor, for his continuous encouragement, and for introducing me to Concordia University.

Fourth, I would like to thank my committee members, Prof. Amr Youssef, Dr. Mohsen Ghafouri, and Prof. Otmane Ait Mohamed, for their guidance and feedback during this degree.

Fifth, I would like to thank the Professors at Concordia University I took lessons from or worked with either for research or for any teaching duty. I have learned a lot about different topics of cyber security in such a fantastic learning environment.

Sixth, I would like to thank my parents and my extended family for their constant support, love, encouragement, believing in my capabilities, and for always being there for me.

Last, I would like to thank the great people I met during my Ph.D. Special thank goes to Can Telkenaroğlu, Rimliya Telkenaroğlu, Korhan Akçura, Başak Tozlu, Gülveren Özdemir, Nezih Özdemir, and Ekin Özdemir for being like a family to me. In addition, I would like to thank Dr. Muhammad Azam, a former Ph.D. student at Concordia University, for his friendship and constant support. I am grateful to have such great people around me.

Whether you think you can, or you think you can't – you're right.

– Henry Ford

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter, we first discuss the overview of the problem tackled in this thesis along with the motivation. After that, we present research questions and finally, we provide contributions.

1.1 Overview

Smart grids enhance the efficiency and reliability of power systems by integrating information and communication technologies (ICT) into them [7]. Such an integration grants smart grids an essential role in addressing the global challenge of satisfying the energy demand, which grows faster than its supply [8]. On the other hand, the smart grid is a complex system that involves three subsystems: generation, transmission, and distribution. In the generation subsystem, the energy is produced using renewable resources such as wind or non-renewable resources such as coal. Transmission refers to the transfer of the generated energy close to end users. Distribution refers to delivering the energy to end users.

Even though the smart grid enhances the traditional power system by adding ICTs, there are differences between the smart grid and the traditional power system [9]. Firstly, the traditional power system is a closed-loop system in which the power is carried from a few generators to customers. However, the smart grid is an open-loop system in the sense that it uses a two-way flow of electricity and information. Secondly, the power generation in the traditional power grid is centralized but it is distributed in the smart grid. Thirdly, the traditional power grid contains fewer sensors compared

to the smart grid. The transition from the traditional power grid to the smart grid is a double-edged sword in the sense that two-way communication, decentralized generation, and added intelligence make the smart grid better at handling failures and making decisions for efficient usage of the generated power but these also make the smart grid vulnerable to more cyber-attacks compared to the traditional power systems.

Digital Substations are included in the transmission and distribution subsystems of the smart grid. According to a study conducted by the Federal Energy Regulatory Commission (FERC), a coordinated attack on just nine substations (out of $55,000$) can bring down the entire United States (US) power grid [10]. In addition, seven substations were targeted in the real-world attack on the Ukrainian power grid, which caused a blackout resulting in $225,000$ homes without power and lasted for several hours [1]. The Ukraine attack has demonstrated that launching cyber attacks against substations is feasible [1]. Given the fact that such attacks are feasible, it is crucial to protect substations against cyber attacks with major physical consequences. In this thesis, we identify and address important problems in measuring and enhancing the security postures of smart grid substations against cyber attacks with physical consequences.

1.2 Motivation

Firstly, redundancy, usually done by backing up important components in a critical network, is a widely adopted solution to improve the resilience of substations against failures. As indicated in the International Electrotechnical Commission (IEC) 61850-90-4 standard [11], most smart grid substations contain redundant subsystems with identical functionality, designed to handle the failure of one subsystem (e.g., lightning damaging a circuit breaker) by activating another subsystem. However, the effectiveness of this approach in relation to failures due to cyber attacks did not receive enough attention. Even if a substation is designed to have two identical subsystems, attackers may compromise the actively running subsystem and reuse the information gathered during the attack to compromise the other similarly configured subsystems. In this context, designing a substation network with two redundant subsystems may not always be effective against failures resulting from

[1] Analysis of the Cyber Attack on the Ukrainian Power Grid https://www.sans.org/blog/confirmation-of-a-coordinated-attack-on-the-ukrainian-power-grid/ [Online; Accessed 3-October-2023]

cyber attacks. By taking such potential attacks into account, we need quantitative risk assessment methods aimed at measuring the security effectiveness of substations designed as redundant subsystems with equivalent functionality.

Secondly, smart grid substations can become attractive targets of supply chain (hijacking) attacks due to their higher levels of automation and the inclusion of devices from different vendors. These attacks exploit the vulnerabilities injected into devices by a malicious vendor or an attacker who has compromised a trustworthy vendor's distribution channel. In the specific context of substations, the proprietary nature of substation devices and the limited number of vendors supplying such equipment means that a power utility needs to trust the available vendors, even if such a trust is not completely verified. In addition, even a trustworthy vendor may become the target of those attacks. For instance, in the case of SolarWinds[2], the vendor's update channel has been hijacked through a supply chain attack. Moreover, supply chain vulnerabilities have unique characteristics such as being continuously updated by the vendor (which can also be a hijacked vendor) to avoid detection, taking the form of a hidden backdoor, and being specifically designed to cause long-term damages [12]. By taking those attacks into account, in addition to patching known vulnerabilities and placing traditional defense mechanisms (e.g., firewalls and intrusion detection systems (IDS)), there is a need for evaluating the security postures of substations against potential supply chain attacks in order to protect substations against such attacks.

Thirdly, in order to defend substations against supply chain capable attackers, the obvious solution of replacing less trusted vendors with trusted ones may not always be feasible due to operational constraints, such as the limited availability of vendors and prohibitive costs. In addition to quantitatively measuring the security postures of substations against supply chain attacks, the measured security postures need to be aimed at optimal improvement. For this purpose, hardening [13] refers to improving the security postures of critical networks concerning quantitative security metrics. However, the hardening solution for improving the security postures of substations against supply chain attacks should make it possible to improve the posture by utilizing non-supply chain-related security mechanisms (e.g., adding firewalls, placing IDS, and disabling services). Utilizing security

[2] SolarWinds Supply-Chain Attack https://www.sans.org/blog/what-you-need-to-know-about-the-solarwinds-supply-chain-attack/ [Online; Accessed 9-January-2021]

mechanisms not related to the supply chain is important because the obvious solution of replacing vendors may not always be applicable. By taking the fact that it is challenging to improve the security postures of substations against supply chain attacks, there is a need for a system that can measure and aim to optimally improve the security postures of substations against supply chain attacks while considering costs and operational constraints.

Lastly, identifying possible attacker strategies is important for protecting substations against cyber-physical attacks. In this regard, threat modeling refers to obtaining an overview of possible cyber attacks and security threats [14]. Attack graphs [15, 16, 21, 20] are widely used in threat modeling. Attack graphs show possible ways the system can be attacked through potential vulnerability exploits [15]. Once exploits of known vulnerabilities are captured in an attack graph, the attack paths can be enumerated to provide complete coverage of potential ways for reaching the attack goals [16]. In [21], the authors show that there exists a trade-off between the completeness and usefulness of attack graphs and in order to overcome this they provide a framework in which the user can control the amount of information contained in the attack graph. In [20], the authors demonstrate that alert correlation approaches do not take unsuccessful attack paths and true negatives into account, and attack graphs can be used along with alert correlation approaches to overcome this. After building attack graphs, attack graph-based security metrics can be developed in order to quantify the overall resilience of networks against attacks [18]. Given that attack graphs can represent different attack paths, are flexible in the sense that the user can add as many different attack paths as the user needs, and can be used to quantify security levels of critical networks, we utilize attack graph-based security metrics in order to tackle the problem of measuring and improving security postures of smart grid substations with respect to different types of attacks.

Although attack graphs [15, 16] are widely used in threat modeling, their generation and usage for substations may be limited for several reasons. First, most attack graph generation methodologies in the literature require expert knowledge. Second, most attack graph generation tools (e.g., MulVAL [17]) generate attack graphs based on only the static network configuration. Third, attack graphs fall short in modeling the physical aspects of cyber-physical attacks. In addition to the difficulty of generating attack graphs and their limited usage for substations, another problem power

system operators face is differentiating between real alerts and false positives [3] among millions of log alerts received by substations daily. By taking those into account, there is a need for a tool that generates threat models for substations based on both the substation configuration and log alerts while considering the physical aspects of cyber-physical attacks.

To summarize, quantifying the security effectiveness of redundant subsystems, developing security metrics considering supply chain attacks, hardening substations with respect to quantitative security metrics, and enhancing threat models with ongoing instances of attacks are important problems for IEC 61850 substations.

1.3 Research Questions

This thesis aims to provide attack graph-based security metrics, a hardening framework, and threat modeling for substations considering different types of cyber-physical attacks. To this end, this thesis addresses the following main research question:

"How can we measure and improve security postures of IEC 61850 substations with respect to different types of attacks?"

More specifically, the main problem this thesis aims to solve can be described with the following research questions:

- *"How can we quantify the security effectiveness of those redundant designs when IEC 61850 substations are designed as having two or more subsystems with identical functionality?"*
- *"How can we measure the security postures of IEC 61850 substations containing devices from different vendors with respect to supply chain attacks?"*
- *"How can we improve security postures of IEC 61850 substations concerning supply chain attacks considering operational and budget constraints, such as the operator being bound to certain vendors?"*

[3] Survey: 27 Percent of IT professionals receive more than 1 million security alerts daily https://www.imperva.com/blog/27-percent-of-it-professionals-receive-more-than-1-million-security-alerts-daily [Online; Accessed 3-July-2023]

- *"How can we enhance threat models for IEC 61850 substations with real-time log events, such as alerts?"*

In summary, four research questions tackled in this thesis focus on measuring and improving the security postures of substations with respect to different types of attacks. Those research questions complement each other in the sense that we first need to quantify the security postures of substations utilizing security metrics in order to measure and improve the security levels of substations to prevent as many attacks as possible. In order to prevent even more potential attacks, it is important to enhance threat models, which are generated according to static system configurations, with respect to newly discovered attacks.

1.4 Research Contributions

This thesis aims to provide security metrics, threat modeling, and hardening methodology for substations considering different aspects of securing them against cyber-physical attacks. Throughout this thesis, we make the following contributions.

1.4.1 Factor of Security (FoS): Quantifying the Security Effectiveness of Redundant Smart Grid Subsystems

The main contribution of Chapter 3 is as follows. To the best of our knowledge, this is the first effort to quantify the security effectiveness of redundant subsystems in the context of substations. We are also among the first to adapt the factor of safety concept from traditional engineering domains to security. In addition, we show that security practitioners can apply the proposed security metrics to answer practical questions to identify and mitigate the security threats of known and zero-day vulnerabilities.

1.4.2 Measuring the Security Postures of Substations Against Supply Chain Attacks

The main contribution of Chapter 4 is as follows. First, we design a supply chain risk metric, namely SCR, to measure the relative risk of vendors and devices for containing supply chain vulnerabilities through quantifying and integrating various supply chain-related risk factors following a

methodology inspired by the Common Vulnerability Scoring System (CVSS) [4]. Second, we design the $kSupply$ security metric that measures the security postures of substations concerning supply chain vulnerabilities. Third, we design the $kSupplier$ security metric to consider the case where all the devices from a malicious (or trustworthy but hijacked) vendor may contain supply chain vulnerabilities and, therefore, to evaluate how many malicious vendors (instead of vulnerabilities) the substation can survive while keeping the security at an acceptable level. Fourth, we demonstrate how attack graphs can be combined with the Markov Decision Process (MDP) to identify critical attack paths in terms of potential physical consequences in addition to the shortest attack path found by attack graphs. Lastly, we validate the effectiveness of those metrics and models through simulations.

1.4.3 Optimal Security Hardening of Substations Against Supply Chain Attacks

The main contribution of Chapter 5 is as follows. To the best of our knowledge, this is the first interactive system specifically designed to improve the security postures of substations against supply chain attacks under given cost and operational constraints using multi-objective optimization. A unique feature of the designed system is that it can still mitigate supply chain attacks even if supply chain-related hardening options (e.g., replacing vendors or upgrading devices) are unavailable. In that system, we first design a module that allows the operator to automatically generate attack graphs based on the static substation configuration. Then, our metric calculation module takes multiple security metrics into account for measuring the security postures of IEC 61850 substations against supply chain attacks. After that, we developed an MDP generation module to analyze the potential physical damages of cyber attacks. Lastly, we validate the effectiveness of the designed system based on several scenarios representing different real-world substations.

[4] CVSS Version 3.1 Release https://www.first.org/cvss/specification-document. [Online; Accessed 23-January-2021]

1.4.4 SecMonS: A Security Monitoring Framework for Substations Based on Configuration Files and Logs

The main contribution of Chapter 6 is as follows. We first provide an automated tool to generate attack graphs based on the static configurations of substations. Then, we generate automaton models from log files that visualize the behaviors of Intelligent Electronic Devices (IED) in the substation. Then, we merge those two models to provide security monitoring for substations based on substation configurations and log files. After that, we generate MDP models starting from the attack stage the attacker is already in for tackling the state space explosion problem those models face. Finally, we demonstrate the effectiveness of our tool through simulations and show that power operators can utilize it for real-time security monitoring since the tool has a reasonable runtime.

1.5 Summary

In summary, the main contributions of this thesis are as follows:

- We define two security metrics for measuring the security effectiveness of redundant subsystems and verify the effectiveness of those metrics through simulations.
- We define security metrics for measuring the security postures of IEC 61850 substations with respect to supply chain vulnerabilities. Given the fact that IEC 61850 substations may contain devices from different vendors, it is important to quantify the security levels of substations with respect to supply chain attacks.
- We develop a hardening system that considers both supply chain-related and non-supply chain-related hardening options under the same framework. This allows hardening IEC 61850 substations against supply chain attacks when power operators are bound to certain vendors.
- We develop a security monitoring framework that first generates attack graphs according to the static configurations of substations obtained from Substation Configuration Language (SCL) files, and after that it enhances those attack graphs with ongoing instances of attacks obtained from log events.

Chapter 2

Literature Review

In this chapter, we first provide the current state-of-the-art literature on general threat modeling in the smart grid (Section 2.1). After that, since attack trees, attack graphs, and security metrics are commonly leveraged to model threats in the literature, we focus on threat modeling utilizing attack trees (Section 2.1.1), attack graphs (Section 2.1.2), and security metrics (Section 2.1.3).

After providing a literature review on threat modeling in the smart grid, we provide state-of-the-art literature on redundancy and security in critical control systems (Section 2.2). Subsequently, we provide a current literature review on supply chain attacks in the smart grid (Section 2.3). Then, we provide a literature review on network hardening, which refers to applying security measures (such as placing firewalls, disabling services, etc.) in an optimal manner (Section 2.4). Lastly, we provide state-of-the-art literature on generating attack graphs based on different information sources such as logs, events, and alerts (Section 2.5).

2.1 Threat Modeling

The smart power grid is a complex system involving cyber and physical components and communication between those. Increased communication between those components, compared to the traditional power grid, makes the smart grid vulnerable to cyber attacks with major physical consequences. To illustrate, in the case of the Ukraine attack[1], the attackers compromised substations,

[1] Analysis of the Cyber Attack on the Ukrainian Power Grid https://www.sans.org/blog/confirmation-of-a-coordinated-attack-on-the-ukrainian-power-grid/ [Online; Accessed 3-October-2023]

maliciously tripped circuit breakers, and as a result caused a blackout for several hours. In another important security incident, NotPetya ransomware, which initially impacted computer systems in Ukraine, spread quickly and impacted hospitals in the United States and a chocolate factory in Australia [22]. In another security incident, CRASH OVERRIDE [2] malware, which impacted a transmission level substation, showed its capability of manipulating industrial control systems and causing the destruction of data via wiper [2]. Given that the smart grid is vulnerable to sophisticated attacks with physical consequences, it is important to perform threat modeling [23] to gain a better understanding of such attacks and identify mitigation strategies. In [23], the authors define threat modeling as "a process that can be used to analyze potential attacks or threats".

2.1.1 Attack Trees

Attack trees are first introduced in [24]. A sample attack tree is shown in Figure 2.1. The root node of the attack tree represents the attacker's end goal. Leaf nodes represent basic steps that the attacker can take. Attacker steps can be combined using AND and OR operators.

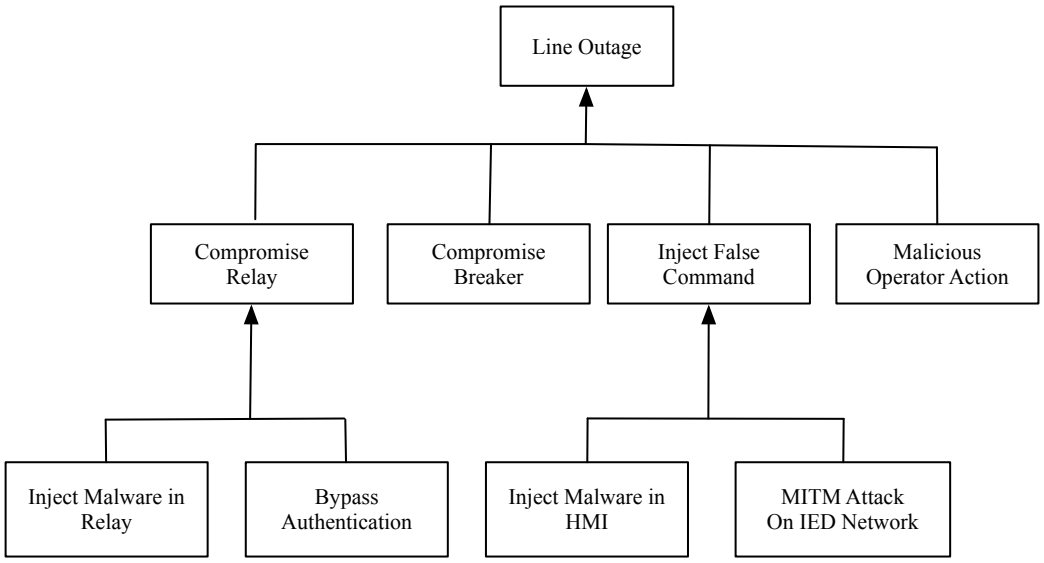


Figure 2.1: Attack Tree Model Used in [1]

There is a rich literature on attack tree modeling in the smart grid for different smart grid components. In [25], the authors develop an attack tree visualizing potential ways whereby power system

[2] CRASH OVERRIDE Analysis of the Threat to Electric Grid Operations https://www.dragos.com/wp-content/uploads/CrashOverride-01.pdf [Online; Accessed 18-June-2023]

control can be attacked by targeting control centers or web-based Supervisory Control and Data Acquisition (SCADA) systems. The attack tree model in [25] is utilized to calculate vulnerability indices of leaf nodes for identifying nodes to focus on for security enhancement. Vulnerability indices are calculated based on evidence of intrusion attempts, existing countermeasures, and password policy enforcement. They demonstrate that by enhancing security measures and preventing the usage of factory default passwords, vulnerability indices of leaf nodes can be improved [25]. In addition, they provide both an attack tree and an attack-tree-based security evaluation [25].

In [26], the authors provide a comprehensive literature review on the security of critical infrastructures. A SCADA system is responsible for controlling and monitoring substations and field devices [27]. They extend the work in [25] by applying attack tree modeling as part of a SCADA security framework [26]. The security framework in [26] consists of four major components: real-time monitoring, anomaly detection, attack impact analysis, and mitigation strategies. Real-time monitoring involves information gathering regarding both the cyber and the physical aspects of the system. Anomaly detection refers to correlating security and event logs for detecting attacks. Attack impact analysis refers to evaluating possible attacker actions. Mitigation strategies refer to decision-making for attack prevention or recovery. They use the attack tree and numerical vulnerability indices obtained from the attack tree as part of the SCADA security framework [26].

In [28], the authors provide an attack tree model showing how an attacker can gain administrative rights in a hydroelectric station monitoring server. After building the attack tree, they claim that there are many possible attack paths for many possible attacker goals, so important security goals should be the focus of securing the monitoring server [28]. In the attack tree in [28], each node represents a network state, and each edge represents a possible attacker action. Details about the attack, such as which vulnerabilities are exploited, are also provided in the attack tree employed in [28]. The authors provide the attack tree but do not provide metric calculations [28].

In [29], the authors categorize attack trees as archetypal and concrete. Archetypal attack trees provide strategies for reaching a given adversarial goal in any system. Concrete attack trees refine the archetypal ones based on specific vendors [29]. They claim that archetypal attack trees can be used for high-level modeling to conduct penetration testing [29]. After generating archetypal attack trees, concrete attack trees can be generated for each vendor by using archetypal attack trees as

templates to deal with device diversity in penetration testing. The authors also provide attack trees for actual attacks, but they do not provide any quantitative risk assessment methodologies using those attack trees [29].

In [30], the authors use attack trees to develop a quantitative risk assessment method for SCADA systems. They first develop an attack tree showing the attacker's goals [30]. After that, the probabilities of the leaf nodes of the attack tree are calculated. Consequently, attack paths which represent a series of events from leaf nodes to the attacker goal (root node), are analyzed by calculating their probabilities. The authors develop a risk assessment framework that considers both cyber and physical aspects of attacks [30].

In [31], the authors develop a minimum cost attack prevention framework, which first develops an attack tree and finds the minimum number of leaf nodes the security operator needs to focus on. The framework [31] makes it possible to improve the security posture in a cost-efficient manner. They demonstrate their method by utilizing an attack tree showing ways the attacker can perform energy theft [31]. Even though the method described in [31] aims to find the minimum cost attack prevention, it does not prioritize between leaf nodes of the attack tree and assumes that each leaf node has identical priority.

In [32], the authors present different ways Wide Area Monitoring Systems (WAMS) can be attacked by providing three attack trees. The first attack tree in [32] shows how field devices can be compromised. The second attack tree in [32] shows possible ways whereby a blackout can be caused. The third attack tree in [32] shows possible ways whereby control commands can be manipulated. They provide attack trees without providing any countermeasures [32].

In [33], the authors propose an automated attack tree generation methodology utilizing artificial intelligence. To verify their methodology, they first manually develop an attack tree for a video surveillance system [33]. After that, they utilize the designed framework for generating an attack tree for the same system [33]. The authors demonstrate that the automated attack tree takes less time to generate, can be generated without domain expert knowledge, and has more depth than the manual attack tree [33]. They generate attack trees in an automated manner, but their model does not provide quantitative risk assessment or countermeasures [33].

In [34], the authors develop an attack tree as part of the risk assessment of cyber attacks against

electric cyber-physical systems. After developing the attack tree in [34], the probabilities of leaf nodes are calculated. Those calculated probabilities are then used to calculate probabilities of physical attacks. The risk assessment methodology in [34] is demonstrated using IEEE 14 and 39-bus systems. The authors develop a risk assessment methodology that takes both cyber and physical aspects of attacks into account. However, cyber attacks described in [34] are high-level.

In [35], the authors extend the attack tree model and develop a preference attack tree model. The preference attack tree model [35] is based on statistics about the number of times each attack has occurred. Each leaf node of the preference attack tree contains the number of times the attacker has utilized each leaf node. They simulate their model using a simple cyber configuration [35]. The authors enhance the attack tree generation methodology [35], but they do not take into account new attack strategies that attackers can utilize in the future since their method employs historical data of attacks that have occurred.

In [36], the authors define Bayesian attack trees to model attacks against substations. In Bayesian attack trees [36], the probability of each node depends on the conditional probability of parent nodes. After calculating probabilities of attack paths utilizing probabilities of each node, they utilize the mean time-to-compromise (MTTC) model (as detailed in [37]) to measure the system's security posture [36]. The authors also utilize attack trees for quantitative risk assessment, but they do not consider countermeasures to improve attack resilience[36].

In [38], the authors claim that compromising the smart grid can be performed by launching multiple attacks exploiting a diverse set of vulnerabilities. They also claim that attack trees are not able to capture the time that an attacker takes between each attack step due to their static nature [38]. To overcome such a challenge, the authors developed a framework for analyzing attack trees based on Continuous Time Markov Chain (CTMC) [38]. They demonstrate their model using case studies and show that countermeasures identified by their framework can reduce the probability of the attacker reaching the goal node and increase the time required for the attack [38]. Also, the authors enhance the attack tree by considering the time taken between nodes and provide a framework with which operators can analyze the impacts of countermeasures. However, they do not provide a framework that identifies optimal countermeasures.

In [39], the authors develop an attack vector for a time delay attack targeting load frequency

control in the smart grid system. The attack vector defined in [39] targets protocol vulnerabilities and puts the grid into an unstable state. The developed attack vector [39] is visualized using an attack tree model. In [39], the authors execute the attack on a testbed and model the attack utilizing the attack tree modeling but they do not provide mitigation strategies.

In [40], the authors generate an attack tree model for WAMS communication. Then, the vulnerabilities of each leaf node are calculated. As a result, vulnerabilities of attack scenarios are calculated. In the end, the vulnerability of the system is calculated. The risk assessment method presented in [40] is used to identify critical ports, and it can be used to analyze the impacts of countermeasures, but it does not provide optimal ways to apply countermeasures.

In [41], the authors develop an attack tree model for the SCADA system in order to develop a vulnerability assessment method where each leaf node of the attack tree is analyzed based on attack cost and attack difficulty. Such an assessment provides the weak links in the system. The vulnerability assessment method developed in [41] considers only cyber aspects of attacks against the SCADA system.

In [42], the authors develop an attack tree visualizing cyber attacks against safety and stability control systems. The attack tree developed in [42] is utilized to calculate the success probabilities of cyber attacks by assigning a score to each attack tree node based on cost, difficulty, and impact. The risk assessment model in [42] also considers the physical consequences of attacks. The developed risk assessment methodology takes both cyber and physical aspects of attacks into account but it does not consider countermeasures.

In [43], the authors provide a vulnerability assessment framework utilizing attack trees for digital secondary systems in substations. The attacks considered in [43] are unauthorized access, tampering, and denial of service. Security metrics are calculated in [43] for leaf nodes of digital secondary systems. Based on the risk assessment, they conclude that process bus point-to-point architecture is the most secure one for digital secondary systems. The authors develop a risk assessment framework for digital secondary systems aimed at finding the most secure architecture [43], but they do not consider cases where the architecture cannot be changed due to operational constraints.

In [44], the authors extend the attack tree model and develop the attack countermeasure tree model which has two attributes that represent the vulnerabilities that can be exploited and possible

countermeasures. The model designed in [44] is utilized for risk assessment in SCADA systems. The attack countermeasure tree model [44] is verified via experimentation of high-risk attack scenarios in the SCADA system. Physical impacts of attacks are not considered in [44].

Lastly, in [45], the authors develop a risk assessment framework for evaluating the resilience of Wide Area Damping Control systems. In the designed framework [45], an attack tree model is developed in order to demonstrate the feasibility of attacks against communication links. The framework is verified through simulations using the Kundur 2-area test system in different loading conditions. The authors consider both cyber and physical aspects of attacks [45], but they do not consider countermeasures.

According to the literature on attack tree usage in the smart grid, some works (such as [28, 29, 32, 33]) only develop attack tree models for demonstrating potential attacks but they do not utilize the developed model as part of risk assessment. Conversely, some works (such as [25, 30, 34, 35, 36, 38, 40]) utilize the attack tree as part of the risk assessment. However, usage of the attack tree for optimal hardening did not receive enough attention in the literature except [31, 44]. In this thesis, we mostly utilize attack graphs for risk assessment since attack graphs contain more information, such as specific vulnerabilities exploited, locations of firewalls, etc., in contrast to attack trees. Papers related to attack graph usage in the smart grid are summarized in the next section.

2.1.2 Attack Graphs

Attack graphs visualize inter-dependent actions taken by attackers to compromise critical assets in the network [15, 16]. Fig. 2.2 shows an example attack graph depicting the steps that an attacker could take to compromise host 7 (details such as risk score, probability, vendor, and device can be ignored for now and will be explained later in Chapter 5). An attack graph consists of two types of nodes: *exploits* and *conditions*.

Exploits (shown by ovals) represent vulnerable hosts. Only exploitable vulnerabilities in each host are considered for the attack graph model. For example, in Fig. 2.2, the exploit $\langle V\_Known\_1, Attacker, 6\rangle$ represents the exploit of a known vulnerability that allows an attacker to compromise $host6$. An attacker can exploit a vulnerability when all of its preconditions are satisfied. For instance, in Fig. 2.2, an attacker can exploit the vulnerability $\langle V\_SCADA, 1, 2\rangle$ when the $SCADA$

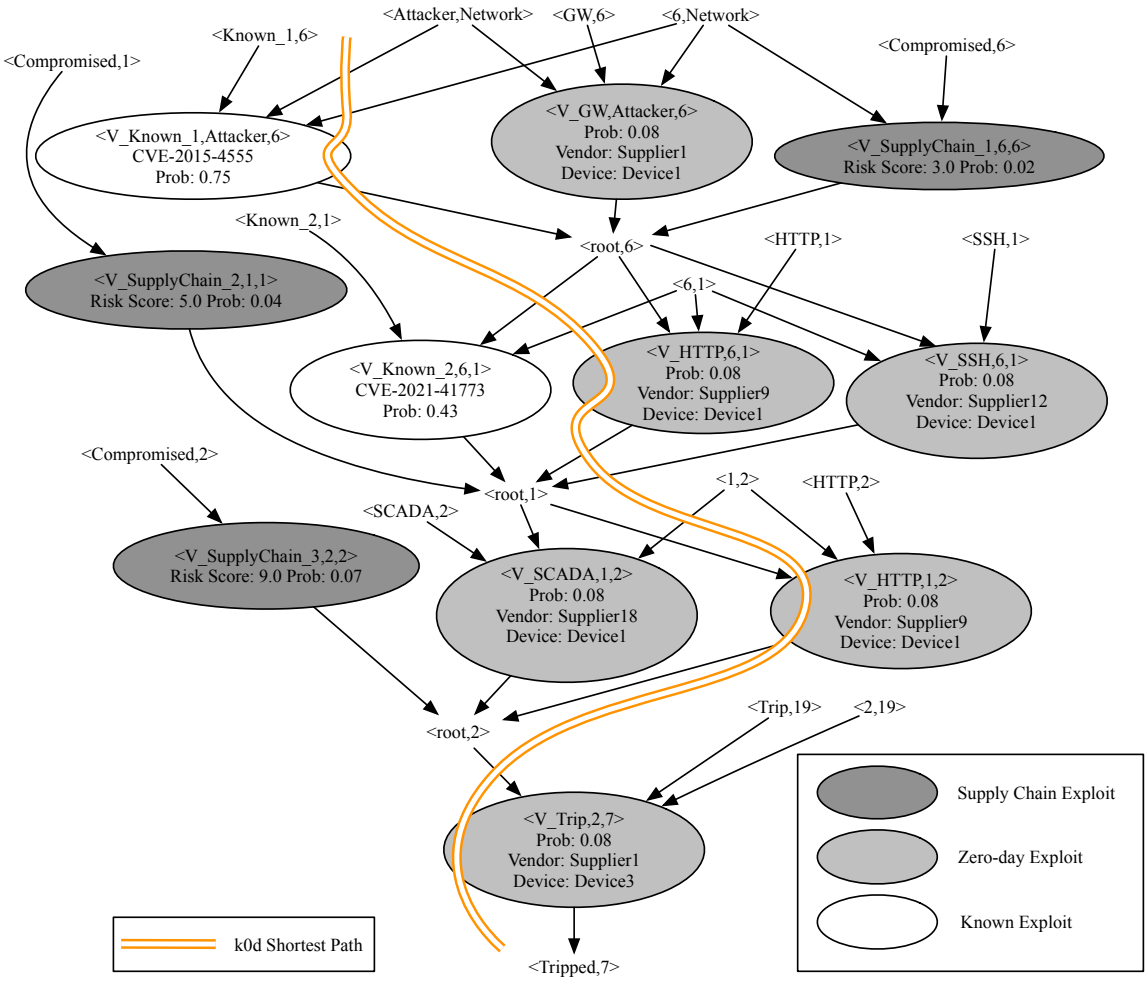


Figure 2.2: Attack Graph for Maliciously Tripping Circuit Breakers

server is running on $host2$, the attacker has compromised $host1$, and $host1$ is connected to $host2$.

Conditions (shown by texts without border) represent what needs to be satisfied for exploiting vulnerabilities and are typically categorized into three main groups: connectivity, privilege, and service existence conditions [46]. A condition is satisfied when either of its parent exploits is satisfied. To illustrate, in Fig. 2.2, the privilege condition $\langle root, 2\rangle$ is satisfied when the attacker exploits any of $\langle V\_SCADA, 1, 2\rangle$, $\langle V\_HTTP, 1, 2\rangle$ or $\langle V\_SupplyChain\_3, 2, 2\rangle$. Any condition without a parent exploit is called an initial condition (e.g., $\langle HTTP, 1\rangle$ in Fig. 2.2). The goal condition (e.g., $\langle Tripped, 7\rangle$ in Fig. 2.2) represents the attacker's end goal.

Various works in the literature discuss the usage of attack graphs in smart grids. In [47], the authors present a hybrid attack model, which models the cyber attacker causing overheating of

transformers. The attack graph in [47] is hybrid in that it models the attack's cyber aspects and physical consequences. Even though the authors [47] develop an attack graph model capturing both cyber and physical aspects, they do not provide security metrics or other quantitative measures based on their model.

In [48], the authors develop attack graphs showing how attackers can trip circuit breakers in the SCADA system. Attack graphs generated in [48] are utilized to calculate the probabilities of compromise. After calculating these probabilities, the authors utilize them in calculating loss of load probabilities [49] for reliability. Moreover, the authors verify through simulations that attackers with higher skill levels can launch more successful attacks with fewer attempts than attackers with lower skill levels [48]. They conduct risk assessment [48] but do not provide avenues whereby the SCADA system can be hardened.

In [50], the authors extend the work in [48] with a more detailed model, including mean time to compromise [37] to estimate the frequency of cyber attacks. Attack graphs are developed for four different attack scenarios: attacking the control center network, power corporation network, substation networks, and man-in-the-middle attack against communication links. After modeling those attacks and calculating reliability metrics, they demonstrate through simulations that reliability decreases as the skill levels of cyber attackers increase [50].

In [51], the authors develop attack graph-based security metrics that consider both cyber and physical aspects of attacks. Moreover, coordinated attacks are considered in addition to attacks conducted by a single attacker. They claim that the developed security metrics can aid in identifying critical assets. In addition, the authors define security metrics and test their metrics on IEEE 9 and 39-buses [51], but they do not conduct simulations on larger bus systems.

In [52], the authors apply the reliability analysis in [50] to wind farm SCADA systems. Attack graphs are developed demonstrating cyber-attacks leading to tripping circuit breakers in wind farm SCADA systems. The difference between wind farms and substations is that a cyber attack resulting in tripping breakers against wind farms may not cause any damage to the system if there is no wind at the time of the attack. The reliability model presented in [52] considers this. They demonstrate through simulations that reliability decreases as the skill levels of attackers increase [46].

In [53], the authors develop attack graph models for two cyber attacks targeting remote terminal

units. The attack graph in [53] calculates the probabilities of exploiting known and zero-day vulnerabilities. In [53], the authors leverage the attack graph model as part of risk assessment for active distribution systems.

In [54], the authors define a novel approach for determining the power system's resilience. They first identify attack sequences, and then the attack graph is generated by combining those sequences. After that, the system's resilience level is calculated as a metric. Lastly, the metric is verified using two different power system communication networks.

In [55], the authors update attack graphs based on real-time intrusion alerts. Moreover, an attack graph is first generated based on existing threat models. After that, the authors [55] experiment with four different score-based learning techniques and identify that K2 learning (as detailed in [56]) outperforms other learning algorithms for the problem of enhancing attack graphs based on alerts.

In [57], attack graphs are generated based on system connectivity and topology. After that, critical components of the power grid are ranked based on cyber-physical betweenness centrality [57] metric. The authors demonstrate that their risk assessment methodology is effective by applying it to an 8-substation power system (whose configuration is provided in [58]).

In [59], the authors generate attack graphs based on substation configuration language (SCL) files. According to them, generated attack graphs can be used to evaluate different security designs. The generated attack graphs in [59] visualize a set of devices compromised by an attacker, but security metrics are not defined based on those generated graphs.

In [60], the authors propose a model for critical component ranking using the cyber-physical betweenness index. The model in [60] aims at providing mitigation strategies to operators based on the impacts affecting the compromised components. The authors demonstrate their model using an 8-substation power system.

In [61], the authors present a framework to automatically generate attack graphs using information about the power grid and cyber attacks against it. This framework allows the discovery of attack surfaces without requiring expert knowledge. They integrate information about vulnerabilities obtained from Common Vulnerabilities and Exposures (CVE)[3] into their framework [61].

In [62], the authors develop a risk assessment methodology by analyzing possible attacks against

[3] CVE Details https://www.cvedetails.com/ [Online; Accessed 3-October-2023]

substations. Attack graphs are generated and employed to calculate attack probabilities. The risk assessment methodology mentioned in [62] is demonstrated using IEEE 30-bus as a case study by calculating the attack risks of each node.

In [63], the authors perform a risk assessment on remote terminal units (RTU) by first conducting penetration testing on RTUs and then analyzing their vulnerabilities. Results of the risk assessment [63] are shown as an attack graph. As a result of this work, some vulnerabilities are found in RTUs due to the lack of reliability.

In [64], the authors develop an online situational awareness framework for attack localization. In this framework, attack graphs are first created using network topology, normal network traffic, and attack traffic to visualize anomalous nodes. The framework described in [64] employs deep learning to the network traffic for the attack graph generation.

In [65], the authors tackle the problem of predicting adversary behavior and develop a prediction framework. The prediction framework uses Bayesian attack graphs where each node contains a single attacker action. The prediction performance described in [65] is assessed using five realistic power systems.

Attack graphs have been used in the smart grid for risk assessment. Those models are utilized to either calculate probabilities (e.g. [48, 50, 65]) or develop security metrics (e.g. [51, 54, 60]). An important usage of attack graphs is finding optimal network hardening [13] options. Based on the limited number of publications tackling the usage of attack graphs for hardening, utilizing them for improving the security postures of substations needs more attention in the literature. The next section describes the usage of security metrics in the smart grid.

2.1.3 Security Metrics

Security metrics are utilized to measure the security postures of critical networks for different kinds of attacks. The literature is rich in attack graph-based security metrics for cyber networks. For example, $k0d$ metric [66] represents the minimum number of zero-day vulnerabilities that an attacker needs to exploit in order to compromise a given network's critical asset. Higher values of $k0d$ for a substation network imply higher resilience against zero-day attacks, as exploiting many distinct zero-day vulnerabilities in a given substation is less likely for an attacker. To illustrate, in

Fig. 2.2, the $k0d$ shortest path $\langle V\_GW, Attacker, 6\rangle \rightarrow \langle V\_HTTP, 6, 1\rangle \rightarrow \langle V\_HTTP, 1, 2\rangle \rightarrow \langle V\_Trip, 2, 7\rangle$ (shown by the gray double line) depicts how an attacker can compromise $host7$ through exploiting the minimum number of distinct zero-day vulnerabilities. This attack path consists of four exploits: two zero-day $HTTP$ exploits in $host1$ and $host2$, a zero-day $Trip$ exploit in $host7$, and another zero-day exploit in $host6$. In calculating $k0d$, the two $HTTP$ exploits of this attack path (corresponding to the same service instance running on $Device1$ of $Supplier9$) are counted as one zero-day exploit, as an attacker capable of exploiting one of those instances would also be able to exploit the other one. Based on those, $k0d$ is 4 for the attack graph shown in Fig. 2.2. Other papers related to security metrics in cyber networks include [67, 68, 69, 46]. In [67], the authors define a metric called Network Compromise Percentage (NCP) which measures the percentage of the hosts where the attacker has obtained user or administrator privileges. In [68], the authors define a security metric that measures the minimal set of initial conditions that the weakest attacker requires to compromise the network. In [69], the authors define an attack graph-based probabilistic security metric that measures the success probability of attacks utilizing multiple vulnerabilities. In [46], the authors define three security metrics for the effectiveness of diversification in critical networks and demonstrate that their metrics are effective through simulations.

Security metrics have also been studied in the smart grid. In [70], the authors define security metrics for IEDs, which aim to quantify the threat to IEDs based on attacker goals, the vulnerability of IED according to security features, and compare IEDs based on their security levels. The IED security metric designed in [70] is verified through simulations. The authors also claim that their metrics can be utilized for auditing by calculating security scores for IEDs [70].

In [71], the authors define a security quantification framework, namely ADversary VIew Security Evaluation (ADVISE). In order to utilize this model [71], the attack steps are initially identified. After that, adversarial attack preferences are determined. Lastly, the authors demonstrate the usefulness of the ADVISE framework as a case study of a SCADA system.

In [72], the authors define a security metric named Elimet. In Elimet, a Competitive Markov Decision Process (CMDP) [73] is first generated based on power system configuration and cyber network configuration. States correspond to the set of compromised hosts, and actions correspond to attackers compromising hosts or operators making security decisions (such as which host to restore

when more than one host is compromised). Elimet [72] is an online security metrics framework in the sense that security metrics for each possible attacker state are updated dynamically based on operator actions. The authors demonstrate that Elimet accurately estimates security metrics for each possible attacker state [72].

In [74], the authors extend the power system contingency analysis (i.e. analyzing possible failure scenarios for the power system, such as what happens if a particular circuit breaker fails) to include contingencies due to cyber attacks in addition to contingencies due to natural faults. Most power systems operate with N-1 contingency criteria [75], meaning the power system should be functional even if one device fails. In [74], the authors develop an MDP based on cyber topology and the power network configuration. States correspond to the set of compromised hosts. Actions correspond to vulnerability exploits where each vulnerability is classified as LOW, MEDIUM, or HIGH based on its CVSS[3] severity. After building the MDP model, the authors [74] identify critical attacker states to identify states to focus on for applying countermeasures.

In [76], the authors develop a risk management framework named CPINDEX which is an automated cyber-physical analysis tool that considers cyber topology, power network topology, and IDS alerts. Interdependencies between cyber and physical components are considered in CPINDEX. The goal of the security analysis in CPINDEX [76] is to predict the most probable upcoming cyber attack incident.

In [1], the authors develop an online security measurement framework considering both cyber and physical networks in power systems. The authors utilize the Partially Observable Markov Decision Process (POMDP) in which there is uncertainty about the exact state the attacker is currently visiting. The authors implement the framework using PowerWorld[4] and Zabbix[5]. PowerWorld is a power system analysis software. Zabbix is a monitoring software. In the developed framework [1], Zabbix is used to monitor security states, and it communicates with PowerWorld for calculating power system dynamics (such as loads on transmission lines after one transmission line is tripped). The online framework [1] is inspired by [74] and extends [74] by considering logs and alerts.

[4] PowerWorld Software https://www.powerworld.com. [Online; Accessed 3-October-2023]

[5] Zabbix: The Enterprise-Class Monitoring Solution for Everyone http://www.zabbix.com/. [Online; Accessed 3-October-2023]

In [77], the authors develop a security assessment framework that first constructs the cyber-physical dependence model. After building the model, a security metric, namely the attack consequence probability index, is calculated for each cyber-physical system node. The authors [77] consider security assessment for distributed cyber-physical systems. The effectiveness of the security assessment in [77] is demonstrated through simulations.

In [78], the authors extend the power system contingency analysis by including concurrent contingencies. Concurrent contingencies are simultaneous failures of two or more components. The authors first find sets of hosts for which simultaneous compromise causes more damage than the sum of individual compromises. After that, MDP is built with security states, each of which refers to a set of compromised hosts. The MDP model is used to calculate security states. The authors of [78] demonstrate that their security assessment framework is practical through simulations using IEEE 14 and 39-bus systems.

In [79], the authors develop a security assessment metric named cyber-physical security assessment metric (CP-SAM), which considers factors affecting power system resiliency. CP-SAM [79], which utilizes attack graph metrics and vulnerabilities in different components as information sources, is designed to evaluate the resiliency of microgrids. The authors of [79] demonstrate the usefulness of CP-SAM through case studies, including the following scenarios: loss of load and loss of link.

In [80], the authors present the Cyber-Physical Energy System Quantitative Security Metric (CPES-QSM) which is a security metric that considers power system dynamics, graph theory, and cyber domain. They verify the metric through simulations using the IEEE RTS-24 bus system.

Lastly, in [81], the authors present Cyber-Physical Resilient Energy Systems (CYPRES) Energy Management System (EMS) for early identification and prevention of threats against the smart grid. Several security metrics, such as those discussed in [60], are utilized for online monitoring as part of CYPRES EMS. The usefulness of CYPRES EMS is demonstrated through a testbed.

As can be seen, the literature on security metrics in the smart grid is rich. However, cyber vulnerabilities are considered in isolation in most of those works (such as [74, 78]). Integrating security metrics designed for cyber networks and metrics calculated according to power system dynamics needs more attention in the literature. Moreover, redundancy has not been considered in most of

the works on security metrics for the smart grid. In addition, known or zero-day vulnerabilities are considered in most of those works, but supply chain vulnerabilities are not considered.

2.2 Redundancy and Security in Critical Control Systems

The research on redundancy and security in critical control systems has received significant attention. In [82], the authors define control areas that are vulnerable to attacks utilizing dependency graphs. As a result of the work in [82], the authors identify five requirements for ensuring advantageous tradeoffs between performance and security in control systems. In [83], the authors propose a situational awareness model for monitoring potential faults and attacks. The model proposed in [83] is utilized for finding the minimum protection in unexpected situations. In [84], the authors propose an attack model that involves attacks through graph vertex removal (where the control system is represented as a directed graph). The methodology proposed in [84] relies on the power dominating set for identifying different attacks against control systems. In [85], the authors propose recovery strategies for controllability in a critical control system following failures in nodes and links. The strategies proposed in [85] provide efficient restoration. In [86], the authors extend the work in [84] by considering combinations of different attack types rather than just vertex removal. Impacts of those different types of attacks are also studied in [86]. In [87], the authors propose a policy enforcement system for the heterogeneous smart grid network. The usefulness of the designed system [87] is demonstrated through a case study. In [88], the authors consider the restoration of the network using redundant edges. They analyze four restoration strategies, and the best option is found in terms of complexity, optimization, and robustness [88]. In [89], the authors utilize backup links to ensure that the system is observable when it is attacked by advanced persistent threats. Utilizing those backup links in [89] ensures reachability between nodes and prevents the attacker from discovering the network topology by identifying paths followed by messages. In [90], the authors provide a redundancy-based restoration approach by taking the standard IEC 62351 into account. The approach provided in [90] is assessed using two case studies and two different adversarial models. In [91], the authors define a checkpoint-based model to produce sufficient data redundancy. The model defined in [91] is evaluated from a practical perspective using realistic case studies. Lastly, in

[92], the authors utilize blockchain technology to develop an architecture to manage secure connections between entities in control systems. They also demonstrate that the provided architecture is suitable for the smart grid. Chapter 3 differs from those aforementioned works in that we focus on a different aspect: measuring the security posture of the system designed as redundant subsystems rather than the recovery of the control structure of the system after being attacked.

2.3 Supply Chain Attacks

Supply chain attacks refer to the exploit of compromised hardware or software where the vulnerabilities are already present before the hardware or software is deployed and configured [93]. In [94], the authors define supply chain attacks as attacks whereby the attacker inserts malicious code or a backdoor into a device before its shipment to the end user. There have been incidents where the attackers utilized the supply chain as their attack vector. For example, in the SolarWinds supply chain attack[6], an attacker inserted malware into the Orion Network Management system (NMS) update channel. As another example, attackers compromised the update channel of accounting software and then inserted NotPetya ransomware[7]. In addition, according to a recent report published by ReversingLabs [95], supply chain attacks also target open source repositories, and attacks on npm and PyPI repositories have increased 271% and 414% respectively. The growing number of attacks and the fact that even trusted vendors can become victims of supply chain attacks (such as in SolarWinds[6] and NotPetya[7] security incidents) show that securing critical infrastructures against supply chain attacks is essential. In order to secure critical infrastructures, such as the smart grid, against supply chain attacks, the first step is to model those attacks in order to identify possible attack vectors. In this section, we provide an in-depth literature review on threat modeling for supply chain attacks in the smart grid.

Supply chain security in the smart grid has been studied in the literature. In [96], the authors discuss key management protocols for substations. A supply chain attack scenario where an attacker compromises an IED inside a substation is among the possible threats discussed in [96]. In [97],

[6] SolarWinds Supply-Chain Attack https://www.sans.org/blog/what-you-need-to-know-about-the-solarwinds-supply-chain-attack/. [Online; Accessed 9-January-2021].

[7] Supply chain attacks https://www.enisa.europa.eu/publications/info-notes/supply-chain-attacks. [Online; Accessed 11-June-2023]

the authors define a multi-scale approach to facilitate the resilience of the supply chain in energy systems. The approach defined in [97] is utilized to monitor the supply chain, and it includes a variety of actors in the process, such as stakeholders, organizational levels, and geographic scales. In [98], the authors perform statistical analysis to detect counterfeit devices based on their behavioral characteristics using system calls from genuine and counterfeit devices. The detection rate of the methodology described in [98] is measured, and it was found that it is high. In [99], the authors analyze how blockchain technologies can help in securing the supply chain. As a result of such an analysis, the authors [99] claim that innovative solutions are needed for securing the supply chain. In [100], the authors develop a Bayesian Belief Network for detecting supply chain attacks. They claim that the provided Bayesian model can be utilized to determine uncertainties, such as attack vectors utilized by an attacker, in cyber attacks. In [101], the authors perform threat modeling among different supply chain stakeholders. The provided threat model is used for identifying countermeasures. Lastly, in [102], the authors propose a blockchain-based identity management system for mitigating supply chain attacks. They verify that their system is effective using case studies. However, to the best of our knowledge, in Chapter 4, we are the first one to quantify the security of IEC 61850 substations considering supply chain vulnerabilities.

2.4 Network Hardening

Network hardening refers to improving the security postures of critical networks. There is a rich literature on network hardening for cyber networks. In [103], the authors tackle the problem of finding minimum-cost hardening options. The hardening is performed in [103] by finding the minimum number of critical exploits (exploits whose removal invalidates attack paths). In [104], the authors solve the hardening problem utilizing the Reduced Ordered Binary Decision Diagram (ROBDD) method. They demonstrate that their approach can be applied to attack graphs. In [105], the authors study heuristic methods for solving the hardening problem. They demonstrate that their methodology finds good enough solutions to the hardening problem in less time for cases where finding the optimal solution is time-consuming. In [106], the authors consider the defense budget for the network administrator in the hardening problem. The hardening problem was solved in [106]

using dynamic programming. In [107], the authors consider dependencies across hardening options and formalize the hardening problem regarding allowable hardening options and hardening cost. They validate that their hardening solution scales linearly with the attack graph size. In [108], the authors present a minimum cost network hardening method using compact attack graphs, named Budget Controlled Network Hardening Algorithm (BCNHA). They show through a case study that their hardening method scales in linear time with respect to attack graph size. In [109], the authors apply genetic algorithm optimization to improve the security posture of critical networks with respect to network diversity [46] metrics. The authors of [109] validated the scalability and accuracy of their approach through simulations. In [110], the authors extend [109] by considering multiple hardening options, such as activating firewalls and relocating services from one host to another. The authors of [110] validate the accuracy and efficiency of their model through simulations. In [13], the authors formulate the optimization problem of diversifying network services under cost constraints using different diversity metrics, such as the probabilistic diversity metric described in [46]. The approach in [13] is validated through simulations based on realistic scenarios. In [111], the authors utilize greedy algorithms for finding cost-effective security solutions for the Internet of Things (IoT). The authors of [111] demonstrate through simulations that the complexity of their hardening approach increases linearly with the attack graph size. In [112], the authors claim that gradient descent search outperforms genetic algorithms for network hardening without specifying the hardening options that should be utilized. The authors of [112] claim that their hardening algorithm requires less computations than NSGA-II. Lastly, in [113], the authors adopt a multi-objective optimization algorithm for improving the security of dynamic networks based on heterogeneous hardening options. The hardening options considered in [113] include patching, isolation of vulnerable hosts, and disabling vulnerable services. In contrast to those general-purpose hardening approaches, in Chapter 5, we focus on combining supply chain-related and non-supply chain-related hardening options for improving the security posture of IEC 61850 substations against supply chain attacks.

2.5 Log-based Attack Graph Generation

Alert correlation refers to the process of finding relationships between many alerts [114]. Alerts correspond to events that require attention (e.g., attacker actions, software faults, failing hardware, etc.). There is limited literature on alert correlation. In [115], the authors convert IDS alerts to a sequence of attack activities and create a probabilistic model. In [116], the authors build attack graphs using intrusion alerts directly without relying on expert knowledge for forensic analysis of prior attacks. They claim that alert-driven attack graphs can potentially become key players in cyber threat intelligence using artificial intelligence. In [117], the authors develop a tool that generates attack graphs directly from alerts which can transform several thousands of alerts into a few attack graphs. In [118], the authors extend the work in [117] and develop a tool that compresses $330,000$ alerts into 93 attack graphs in less than a minute. According to the designed tool, the authors of [118] claim that attackers have more tendency to follow shorter paths even after discovering longer ones in most cases. In [119], the authors present a tool, namely SteinerLog, to correlate alerts and generate attack graphs by finding compromised entities and attacker steps. They show that SteinerLog is effective in detecting compromised hosts and users. In [120], the authors utilize system logs to detect threats, predict attacker actions, and then generate attack graphs based on predictions. Moreover, they demonstrate that their tool can reduce the analysis burden and detect attacker strategies. Lastly, in [64], the authors propose a cyber attack situational awareness tool for substations using deep learning. The tool provided in [64] is capable of identifying attacker locations within substations.

In addition to log-based attack graph generation, there is literature on attack graph generation for substations based on their configurations. In [121], the authors develop a language for creating threat models and use such models to generate attack graphs. The attack graphs in [121] contain assets targeted within the substation, but they are not enhanced according to log events. In contrast, in Chapter 6, we generate attack graphs showing possible ways the attacker can attack the Generic Object Oriented Substation Event (GOOSE) protocol by tampering with GOOSE parameters.

Chapter 3

Factor of Security (FoS): Quantifying the Security Effectiveness of Redundant Smart Grid Subsystems

3.1 Introduction

Among the standardization efforts for smart grid substations, IEC 61850 provides high-level guidelines to establish communication among devices from various manufacturers without elaborating on the security design. IEC 62351 is designed to provide security protection on top of IEC 61850, although it lacks a quantitative approach and is criticized for other weaknesses [122]. For instance, in [122], the authors show two weaknesses in IEC 62351, one that allows the attacker to replay GOOSE and Sampled Values (SV) messages and another in the time synchronization protocol that can place the substation into a vulnerable state.

Although IEC 62443 [123] provides a set of requirements for security risk assessment, it still lacks a quantitative and credible validation [124]. Security metrics in the context of smart grids have been developed previously (a detailed review of the related work is provided in Section 2.1.3). However, most existing works do not employ an explicit security metric like the one proposed in this chapter. To the best of our knowledge, we are among the first to focus on quantifying redundant

subsystems' security effectiveness against known and zero-day vulnerabilities in smart grids.

In this chapter, we introduce a novel security metric, namely, FoS (Factor of Security), to quantify the security effectiveness of redundant subsystems in smart grids[1]. Our key idea is as follows. Although not seen in the context of smart grids or cyber-physical security, the factor of safety is a widely used concept in traditional engineering domains such as mechanical design, which roughly measures how much stronger a system is than it needs to be. For example, the factor of safety would be two if the designed load-carrying capacity (called the strength) is twice as much as the actual load (called the stress). We adopt this concept in the context of cyber-physical security in redundant smart grid subsystems. Specifically, we first design a representative smart grid substation configuration based on IEC 61850-90-4 [11] and existing industrial practices, which provides concrete details about the hardware and software components to facilitate the threat modeling process. Second, based on the well-known attack graph model, we define the strength and stress of a smart grid system as the length of the shortest attack path for the smart grid and its subsystems, respectively; the ratio between the two then yields the FoS, which intuitively gives the equivalent number of subsystems in terms of resilience to attacks.

The following section provides a detailed substation design based on IEC 61850-90-4 [11] and product documentation from key vendors providing substation devices.

3.2 IEC 61850 Substations

A key challenge facing the development of security metrics and threat models for smart grids is the lack of public access to concrete designs of real-world smart grids, including detailed information about the hardware and software components and their vulnerabilities. This fact is understandable since smart grid operators are typically reluctant to disclose details about their infrastructures, especially the vulnerabilities. Although existing standards like IEC 61850-90-4 provide sample substation configurations [11]; those configurations are typically very simplistic and only contain high-level concepts, which is insufficient for our purposes. Therefore, we first, present a concrete

[1]Although we focus on smart grids, the FoS metrics may potentially be applied to other cyber or cyber-physical systems designed with redundancy.

design of a smart grid substation with redundant subsystems, in which one subsystem is the actively running subsystem and others are backups, to facilitate further discussions. In order to ensure that the substation design is sufficiently representative, it is based on IEC 61850-90-4 and product documentation from key vendors. Those key vendors include ABB[2], SEL[3], and Symmetricom[4].

Specifically, Fig. 3.1 shows a substation with two redundant subsystems, A and B. The numbers inside the circles ranging from $A1$ to $A31$ are components of subsystem A, and the numbers from $B1$ to $B31$ are components of subsystem B. In subsystem A, component $A2$ is protected by the firewall F_A since it is a critical component. Similarly, in subsystem B, component $B2$ is protected by the firewall F_B. Components $A7 - A13$ are monitored for anomalies by an IDS, namely, IDS_A_1 [125]. Similarly, components $A19 - A25$ are monitored by IDS_A_2. The same applies to subsystem B components, which are monitored by IDS_B_1 and IDS_B_2. For components for which a replica exists, the component and its replica are numbered correspondingly, e.g., $A1$ is the replica of $B1$ (for those which are not replicated, both numbers refer to the same component, e.g., $A4$ and $B4$). We assume that substations communicate with the control center (*node* 300) through Wide Area Network (WAN), which is also used by remote operators (*node* 200) to manage the substation remotely. Also, the attacker (*node* 100), representing an outsider with unauthorized access to the WAN, is assumed to be connected to the WAN.

According to IEC 61850-90-4 [11], a substation is divided into three levels as detailed in the following:

- **Substation Level:** Human Machine Interface (HMI) is used for monitoring the status of the substation (*nodes* $A1$, $B1$). Workstations (*nodes* $A2$, $B2$) are used for automation in the substation. According to IEC 61850-90-4, workstations at the station level can be used as SCADA servers. Since SCADA servers are critical components [50], incoming and outgoing connections to them are protected by firewalls F_A and F_B. Global Positioning System (GPS) clocks (*nodes* $A3$, $B3$) are used for time synchronization in the substation. All connections to the substation go through substation gateways (*nodes* $A6$, $B6$).

[2] ABB: Substation Automation Solutions SAS 600 Series http://new.abb.com [Online; Accessed 21-March-2017]

[3] SEL https://selinc.com/ [Online; Accessed 21-March-2017]

[4] IEC61850 Smart Substation https://www.techonline.com/directory/symmetricom/. [Online; Accessed 21-March-2017]

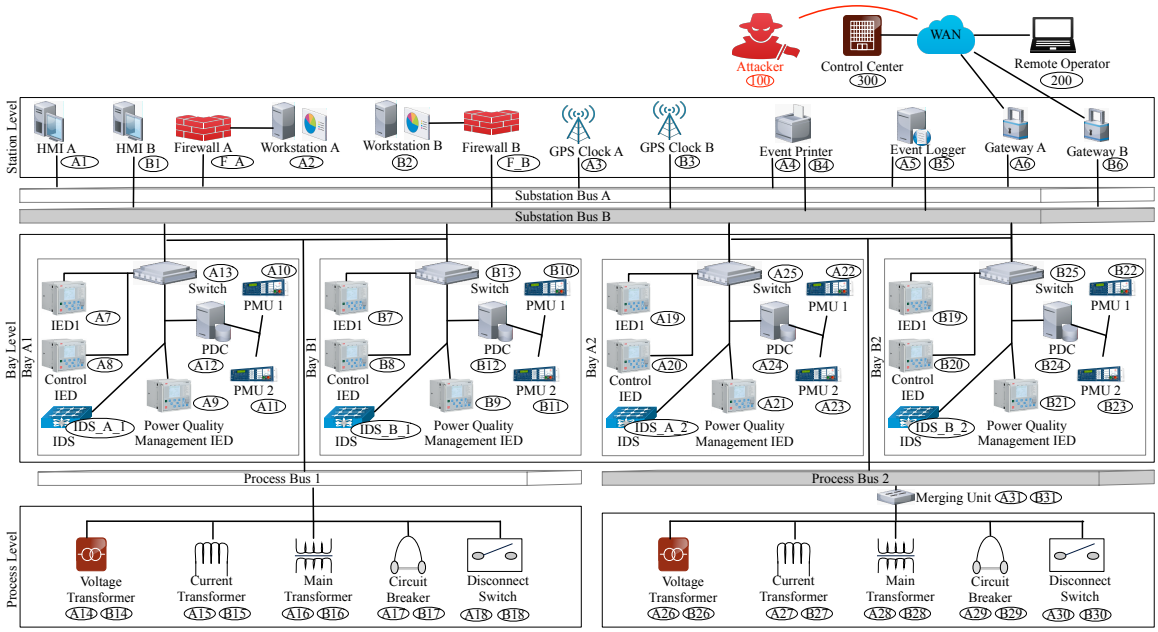


Figure 3.1: Substation Based on IEC 61850-90-4

- **Bay Level:** According to IEC 61850-90-4, the bay level mainly contains IEDs. IEDs improve the automation inside a substation [126]. IEDs in bay A_1 and bay B_1 control field devices which are connected to Process Bus 1, while IEDs in bays A_2 and B_2 control field devices which are connected to Process Bus 2. The IEDs in those bays act as interfaces between cyber components (components at the substation level) and physical components (components at the process level). Each bay contains protection IEDs (*nodes* $A7$, $B7$, $A19$, $B19$), which are used for fault isolation. Control IEDs (*nodes* $A8$, $B8$, $A20$, $B20$) are used for managing the power system for efficient usage. In IEC 61850-90-4 [11], there are also IEDs for power quality measurement, which are included in our design as power quality measurement IEDs (*nodes* $A9$, $B9$, $A21$, $B21$). Even though not seen in IEC 61850-90-4, Phasor Measurement Units (PMU) connected to Phasor Data Concentrators (PDC) are also included in our design since it can be seen in the sample substation design provided by SEL [3]. The PMUs (*nodes* $A10$, $B10$, $A11$, $B11$, $A22$, $B22$, $A23$, $B23$) and PDCs (*nodes* $A12$, $B12$, $A24$, $B24$) inside the bays are used for obtaining synchronized measurements of voltages and phase angles. Lastly, each bay is monitored using an IDS (IDS_A_1, IDS_A_2, IDS_B_1, IDS_B_2).

- **Process Level:** The process level contains field devices, e.g., voltage transformers (*nodes*

Table 3.1: Details of Components in an IEC 61850 Substation

Level	Component	Functionality	References
Substation Level	Gateway (A6, B6)	Sends data to the control center, transfers commands received from the control center [93], and allows remote operators to connect to the substation for monitoring and controlling[a]	IEC 61850-7-1 Clause 8.2.3, [7],[50], [127], [125]
	HMI (A1, B1)	Presents information to the operator about the state of the substation[b].	IEC 61850-7-1 Clause 7, [50], [127], [125], [128], [129]
	Workstation (A2, B2)	Performs automation in a substation, and is also used as a SCADA server [50].	IEC 61850-7-1 Clause 5.2, IEC 62264-1:2013, [7], [130], [122],[124], [50], [51], [127], [125], [126], [131], [132], [133], [129], [134]
	GPS Clock (A3, B3)	Time source in the substation for time synchronization between devices [128].	IEEE 1588-2019, [127], [125], [126], [128], [135], [136]
	Event Printer(A4, B4)	Devices in a substation produce time-stamped events (such as tripping), which are sent to the event printer for monitoring [137].	IEC 61850-90-4 Clause 7.1, [127], [129]
	Event Logger (A5, B5)	Event logger for logging status changes in substation IEDs [138].	IEC 61850-90-4 Clause 7.1, , [127], [126]
Bay Level	Protection IED (A7, B7, A17, B17)	Detects electrical faults and performs fault isolation[c]	IEC 61850-7-1 Clause 5.4, [7], [50], [74], [1], [127], [126], [128]
	Control IED (A8, B8, A18, B18)	Manages the power system for efficient usage and regulates system parameters[c]	IEC 61850-90-4 Clause 7.3, [7], [50], [127], [125], [126], [128], [134]
	Power Quality Measurement IED (A9, B9, A19, B19)	Monitors the power quality [11].	IEC 61000-4-30, [50], [127], [125]
	PMU (A10, A11, B10, B11, A20, A21, B20, B21)	Obtains synchronized measurements of voltage magnitudes and phase angles [d]	IEC 61850-90-5 Clause 6.3, C37.118.2-2011, [7], [127], [126], [131], [136], [132]
	PDC (A12, B12, A22, B22)	Receives, combines and pre-processes synchronized measurements [139].	IEC 61850-90-5 Clause 6.4, C37.118.2-2011, [7], [127]
Process Level	Voltage Transformer (A14, B14, A24, B24)	Measures voltage[e]	IEC 60044-7, [7], [50], [127], [132], [140]
	Current Transformer (A15, B15, A25, B25)	Steps down current levels and measures them[f]	IEC 61869 Part 1, [7], [8], [50], [127], [132], [140]
	Circuit Breaker (A16, B16, A26, B26)	Connects or disconnects transmission lines[g]	IEC 61850-8-1 Clause 6.4, IEC 61850-7-1, Clauses 11.2,12.1, [7], [141], [50], [51], [74], [1], [127], [2], [132], [140]
	Merging Unit (A27, B27)	Converts analog signals into IEC 61850 Sampled Value (SV)[h]	IEC 61850-7-1 Annex B, [127], [125], [132]

[a] DS Agile Substation Gateway https://www.gegridsolutions.com/multilin/energy/catalog/dsagile_substation_gateway.htm [Online; Accessed 23-January-2019]
[b] HMI http://www.subnet.com/resources/dictionary/HMI.aspx [Online; Accessed 9-February-2020]
[c] Protection & Control Relay (IED) http://www02.abb.com/global/seitp/seitp202.nsf/0/137e4d13c980a910c1257e6700337741/$file/Protection+Control+IED-+Ustika.pdf [Online; Accessed 9-February-2020]
[d] SYNCHROPHASORS https://selinc.com/solutions/synchrophasors/ [Online; Accessed 9-February-2020]
[e] Voltage Transformer https://www.globalspec.com/learnmore/electrical_electronic_components/transformers/voltage_transformers [Online; Accessed 23-January-2019]
[f] Current Transformer https://www.electronics-tutorials.ws/transformer/current-transformer.html [Online; Accessed 23-January-2019]
[g] Circuit Breaker http://www.subnet.com/resources/dictionary/Circuit-Breaker.aspx. [Online; Accessed 23-January-2019]
[h] IEC 61850 Stand Alone Merging Units https://www.arteche.com/en/iec-61850-stand-alone-merging-unit-proccess-bus [Online; Accessed 9-February-2020]

$A14$, $B14$, $A26$, $B26$), current transformers (*nodes* $A15$, $B15$, $A27$, $B27$), and circuit breakers (*nodes* $A17$, $B17$, $A29$, $B29$). All the devices receive commands from IEDs and send measurements to IEDs. Some of those devices (*nodes* $A26 - A30$, $B26 - B30$) must be connected to a merging unit (*nodes* $A31$, $B31$) before being connected to the process bus.

Table 3.1 summarizes the role of each component at each level with relevant references, including standards and research papers that address such components. To make the design more representative, Fig. 3.1 also reflects many concepts from industrial practices, e.g., SEL[3], Symmetricom[4], ABB[2], etc., as detailed below.

- The design includes two PMUs (*such as nodes* $A10$ *and* $A11$) connected to one PDC (*such as node* $A12$) in each bay, which is based on a similar configuration by SEL[3].

- The design includes three field devices: a voltage transformer, a current transformer, a circuit breaker, and their intelligent counterparts. Intelligent devices (*nodes* $A14$, $B14$, $A15$, $B15$, $A16$, $B16$, $A17$, $B17$, $A18$, $B18$) can be directly connected to the process bus. Other devices (*nodes* $A26$, $B26$, $A27$, $B27$, $A28$, $B28$, $A29$, $B29$, $A30$, $B30$) need to first go through a merging unit (*nodes* $A31$, $B31$) before being connected to the process bus. This design is based on a similar configuration given by Symmetricom[4].

- The design includes two subsystems with equivalent functionality, with everything replicated except the event printer (*nodes* $A4$, $B4$) and the event logger (*nodes* $A5$, $B5$). This design is based on a similar configuration given by ABB[2] where the HMI (*nodes* $A1$, $B1$) and the GPS server (*nodes* $A3$, $B3$) are replicated.

- The design includes two firewalls (one for each subsystem) at the substation level and IDS in each bay. This design is based on a similar configuration given by [125].

In addition to the detailed configuration's hardware components, we assume the following services are running on top of those components. The GATEWAY service runs on substation gateways (*nodes* $A6$, $B6$). The HMIs and Workstations run the SSH (The Secure Shell) service for remote maintenance. They also run HTTP (Hypertext Transfer Protocol) service to provide substation operators with a user-friendly interface. Workstations can also be used as SCADA servers. GPS clocks run GPS service for time synchronization inside the substation. Services running on IEDs have the same names as the names of the IEDs (such as protection service on protection IEDs). The remote operator's machine is running HTTP and SSH. Even though we made an effort to design a substation based on standard documents and documentation from vendors, substations can have different configurations in a way that IEDs from different vendors can be deployed, and different types of IEDs can be utilized. In the next section, we first discuss a motivating example to build intuitions, and then we define the FoS (Factor of Security) and PFoS (Probabilistic Factor of Security) metrics.

3.3 Factor of Security Metrics

3.3.1 Motivating Example

For the substation depicted in Fig. 3.1, it may seem obvious that, since the substation has two separate subsystems, any faults causing one subsystem to fail can be easily tolerated by switching to the other unaffected subsystem without causing a power outage. However, such reasoning only works for random faults that happen unintentionally. The situation would be quite different when it comes to malicious attacks. For example, consider a remote attacker who wishes to cause a blackout in an area, and he/she has identified this substation as his/her main target. This attacker is an outsider who performs his/her attacks through vulnerability exploitations. Subsystem A is the actively running system, and the attacker targets the HMI in subsystem A (node $A1$) by getting administrator access (such as root in Linux machines) in node $A1$. In order to achieve this, the attacker needs to first compromise the substation gateway (node $A6$) by exploiting a known or zero-day vulnerability in the substation gateway. After that, the attacker needs to compromise the HMI (node $A1$) through another vulnerability exploitation. Consequently, the attacker needs to exploit two vulnerabilities to gain administrator access to HMI. Suppose the operator notices that subsystem A is under attack and switches to subsystem B before the attacker can start to remotely trip circuit breakers [50]. Now the attacker might notice that, although the actively running subsystem has been changed, subsystem B still contains the identically configured components, namely, the gateway (node $B6$) and HMI (node $B1$). Exploiting the same vulnerabilities, the attacker will succeed in compromising subsystem B and subsequently bring the substation down and cause a major power outage. Clearly, even though the substation has been designed to include two subsystems, similar vulnerabilities shared by those subsystems mean that these do not really count as two subsystems in terms of their resilience to malicious attacks since compromising the substation requires almost the same effort and skill as compromising one subsystem.

One option for the operator would be to make the actively running subsystem (e.g., subsystem A) more attack-resilient. Even if the operator makes the actively running subsystem more resilient, the actively running subsystem can still be attacked by more skilled attackers. This implies that

the operators should be prepared for scenarios in which the actively running subsystem has to be switched to the backup system due to attackers compromising the actively running subsystem. In this work, we consider attacks which are purely cyber but which have physical consequences such as maliciously tripping circuit breakers [142, 143].

To better capture such intuitions, we apply the concept of the *Factor of Safety*. The factor of safety is a widely used concept in traditional engineering domains, such as mechanical design [144]. However, it is not seen in the context of the cyber-physical security of smart grids. Roughly speaking, the factor of safety of a system measures how much stronger the system is designed to be (e.g., an airplane is designed with two engines) than it needs to be (e.g., one engine is actually sufficient) in order to ensure the desired level of reliability (e.g., the failure of one engine can be safely tolerated). More formally, the factor of safety is defined as the ratio between the designed load-carrying capacity (i.e., the maximum amount of load that the system can carry, namely, the *Strength*) and the actual load (i.e., the expected amount of load to be carried by the system, namely, the *Stress*), denoted as $Factor\ of\ Safety = \frac{Strength}{Stress}$ [144]. Intuitively, by applying this concept to our previous example, we know that our substation would have a factor of safety of less than 2 from the security point of view (i.e., the substation is not twice as secure as it needs to be).

3.3.2 Threat Model

The following describes different aspects of our threat model.

- **Type of Attackers:** We focus on remote attackers who rely on network connections to launch attacks and escalate privileges by exploiting known or zero-day vulnerabilities in the smart grid components. Therefore, attacks that do not rely on vulnerabilities but employ social engineering, insider misbehavior, or manipulating smart grid measurements [131] will not be considered in our threat model.

- **Attacker's Capacity and Access Level:** In the substation level, we consider attacks performed by exploiting vulnerabilities. The attacker can exploit known or zero-day vulnerabilities to escalate their privileges. The attacker initially does not have access to any devices in the substation except the gateway. Whereas, at the smart grid level, we focus on attacks

performed by disconnecting transmission lines through tripping circuit breakers or disrupting communication lines.

- **Zero-Day Vulnerabilities:** Zero-day vulnerabilities [145] are unknown vulnerabilities whose existence and the way to exploit them are not known to the public. For example, Stuxnet utilized four zero-day vulnerabilities to spread itself [66]. In the threat model, attackers are assumed to be able to exploit such vulnerabilities.

- **Known Vulnerabilities:** Known vulnerabilities can be exploited as long as all preconditions are satisfied. For example, a known vulnerability in the Apache HTTP server, with the identifier CVE-2010-1151[5], allows the attacker to create a race condition and bypass the authentication. If an Apache HTTP server has that vulnerability and if the attacker can connect to that web server with user privilege, then the attacker can exploit that vulnerability to bypass authentication and escalate his/her privilege. In our analysis, tools, and technologies used by the attacker are not relevant since they are not part of attack graphs.

- **Assumptions:** As an assumption of the attack graph model, the attackers are assumed to have all the required capabilities and attacking tools to exploit vulnerabilities as long as all preconditions are satisfied. We assume a series of redundant subsystems will be launched linearly (in any order) after each failure, and the attacker must compromise all the subsystems before causing major damages (e.g., a blackout). The attacker is assumed to reuse any knowledge or skills he/she may have gained while exploiting a vulnerability to attack other similar components in one or more subsystems. We focus on attacks that aim at taking control of critical assets (e.g., circuit breakers) in smart grids, and thus we ignore any reconnaissance steps but consider that attackers already have sufficient knowledge about the smart grid.

3.3.3 Redundant Attack Graph Model

To realize this idea, we first need to model the vulnerabilities (both known and unknown) inside a smart grid or substation, the causal relationships between such vulnerabilities (e.g., exploiting the

[5] CVE-2010-1151 Details https://nvd.nist.gov/vuln/detail/CVE-2010-1151. [Online; Accessed 9-February-2020].

HMI is only possible after exploiting the gateway in Fig. 3.1), and possible ways for compromising the critical assets (e.g., the circuit breakers). To this end, we employ the widely used attack graph model [15, 16]), which is a directed graph with two types of nodes (i.e., exploits and conditions) and edges pointing from preconditions to exploits and from exploits to postconditions. In this chapter, we extend the idea of the attack graph and define the redundant attack graph. A redundant attack graph contains two or more attack graphs merged using an intermediate node. Definition 3.3.1 gives the formal definition of the redundant attack graph.

Definition 3.3.1 *(Redundant Attack Graph) Given a network with a set of hosts H, a set of services S, with the service mapping ser(.): $H \rightarrow 2^R$, set of exploits $E = \{ \langle r,\ h_s,\ h_d \rangle \mid h_s \in H,\ h_d \in H, r \in ser(h_d)\}$, and their pre-and post-conditions C, a redundant attack graph is a directed acyclic graph $G (E \bigcup C, R_r \bigcup R_i)$ where $R_r \subseteq C \times E$ and $R_i \subseteq E \times C$ are pre- and post-condition relations, respectively.*

Our redundant attack graphs can be generated in a similar way as regular attack graphs [15, 16]. Attack graph generation is conducted in three stages [19], which are reachability analysis, attack graph modeling, and core attack graph building. Reachability analysis determines direct reachability between network hosts. Attack graph modeling involves creating attack templates that involve conditions (capabilities required by the attacker to succeed) for attacking network assets and relationships between conditions and network elements. The attack graph building phase refers to algorithms used to build attack graphs. As an example, MulVal [17] is an open-source tool for attack graph generation, and it models exploits as logic propositions and applies logic deduction to reach goal facts from initial facts.

Fig. 3.2 gives an example of our *redundant attack graph model*. In Fig. 3.2, each triple inside an oval indicates an exploit $\langle vulnerability, source\ host, destination\ host \rangle$, such as $\langle v\_SSH, 100, 200 \rangle$, and each pair in the clear text indicates a condition (e.g., a connectivity relationship such as $\langle 100, 300 \rangle$, and a service running on a host such as $\langle SSH, 300 \rangle$). The nodes at the top are examples of initial conditions, such as $\langle Attacker, 100 \rangle$, whereas the node at the bottom is the goal condition, which is $\langle TimeDelay, \{A3, B3\} \rangle$. In addition to the standard notations used in attack

graphs [46, 66], we introduce some new notations. First, the double oval nodes represent network services with no known vulnerabilities. Such network services are considered to contain potential zero-day vulnerabilities. Second, we attach a version number to those nodes, shown in square brackets, to distinguish between different variations of the same service (e.g., Apache[6] for IIS (Internet Information Services) [7] for the HTTP service). Third, oval nodes with CVE [9] numbers (such as $\langle V\_TRIP, ATTACKER, T2\_1\rangle$ in Fig. 3.5) represent known vulnerabilities. Fourth, firewalls ($\langle v\_Firewall, A6, FA\rangle$) are shown in red and are represented in the same way as exploits since from the attackers' perspective bypassing the firewall also requires exploiting a vulnerability in the firewall. The postcondition of the firewall is the connectivity behind that firewall (such as connectivity condition $\langle A6, A2\rangle$ that is behind the firewall F_A). The firewall has only one precondition, which is the name of the firewall (such as $\langle Firewall, F\_A\rangle$). Lastly, the attack graph model in Fig. 3.3 contains intrusion detection nodes ($\langle v\_IDS, A7, IDS\_A\_1\rangle$ and $\langle v\_IDS, B7, IDS\_B\_1\rangle$). The postcondition of those nodes is the failure condition (which is $\langle Fail, Attacker\rangle$). The preconditions of an intrusion detection node include the detection condition (such $\langle IDS\_A\_1, Detect\rangle$), which means that the IDS has succeeded in detecting the attack. The stage where the attack is detected can be any of the privileges acquired by the attacker. If the attack has been detected in the final stage, another condition with the name "Attacked" is added. The preconditions of an IDS also include the existence of the IDS (such as $\langle IDS\_A\_1, A7\rangle$), meaning that node $A7$ is being monitored by an IDS with the name IDS_A_1. If the fact that the attacker has acquired a privilege is not detected by IDS, then the movement of the attacker to acquire the detection condition is stealthy.

Another challenge in applying the attack graph model in our context is to model the existence of multiple subsystems and their relationships. In Fig. 3.2, both sub-graphs for the subsystems share the same topology since subsystems are usually designed as replicas of each other. One subsystem is active each time, and all others are used as backups. Since the operator will switch an active subsystem under attack to another backup subsystem, ideally, the attacker would have to compromise all subsystems before he/she can bring down the entire smart grid or substation. In this sense, the relationship between those subsystems is a conjunction, which can be modeled using an

[6] Apache HTTP Server Project https://httpd.apache.org/. [Online; Accessed 17-August-2019].
[7] Microsoft IIShttps://www.iis.net/ [Online; Accessed 17-August-2019]

intermediate exploit (e.g., node I in Fig. 3.2), which takes the goal condition of each subsystem as its precondition and the final goal condition as its postcondition. The probabilities, the common parent nodes, and the similarity node in Fig. 3.6 can be ignored for now and will be discussed later.

3.3.4 Factor of Security (FoS)

Following our discussions of the motivating example, we now consider how to more precisely capture those intuitions. Our key idea is to define the aforementioned "load" concept (which is used to define both the Strength and Stress of a system and hence its factor of safety) as the system's level of attack resilience. To that end, we apply an existing network security metric, the K-zero day safety ($k0d$) [66], to quantify the level of attack resilience. Considering each remotely accessible service to potentially contain an unknown vulnerability, the $k0d$ metric basically counts the minimum number of such vulnerabilities required to compromise a given network asset. A larger $k0d$ value indicates a more secure network because it is less likely for a large number of unknown vulnerabilities to co-exist and be exploitable by the same attacker. By leveraging this concept, we can define the "strength" as the $k0d$ value of the entire smart grid or substation system with the existence of all redundant subsystems taken into consideration and the "stress" as the $k0d$ value of a subsystem (without considering redundancy). The ratio of the two thus indicates how much more security is provided by the design of redundant subsystems than what is provided by each subsystem. We provide more details in the following.

The Factor of Security Metrics

To define the factor of security metrics based on the redundant attack graph model, we consider three cases.

Case 1 We start with the simplest case, i.e., there is no known vulnerability (only zero-day vulnerabilities), and we are only worried about the attackers who can compromise the critical assets with minimum effort. We first define the *Strength* of a smart grid or substation with respect to a given critical asset as the $k0d$ value for the whole system calculated based on the redundant attack graph model with the final goal condition. This definition indicates the level of security resilience in terms

of the least number of distinct zero-day vulnerabilities required for compromising the whole system, i.e., all the subsystems. Second, we define the *Stress* of the system as the maximum $k0d$ metric value of a subsystem with respect to the same critical asset. We choose the maximum value in order to ensure a proper range of values for the factor of security, which will never exceed the number of subsystems as designed. Finally, we define the *FoS (Factor of Security)* as the ratio between the Strength and the Stress, which intuitively indicates how many redundant subsystems a substation effectively has from the security perspective. In an ideal case, the factor of security should be equal to the number of subsystems physically present in the substation. A lower factor of security value indicates a less-than-ideal design where the redundancy does not deliver as much security resilience as it is assumed to at the design stage. More formally, the strength of a system S, with multiple subsystems S_i, for a given asset μ (μ_i in S_i) is defined as $Strength(S, \mu) = k0d(S, \mu)$, where $\mu = \bigcup_{i=1}^{N} \mu_i$. The stress is defined as $Stress(S, \mu) = \max_{\forall S_i} k0d(S_i, \mu_i)$. Considering Stress and Strength, the factor of security is defined as:

$$Factor\ of\ Security\ (S, \mu) = \frac{Strength(S, \mu)}{Stress(S, \mu)} \tag{1}$$

The factor of security, which is defined as $FoS(S, \mu) = \frac{k0d(S,\mu)}{max(k0d(S_i,\mu_i))}$, where $i = 1 \cdots n$, and n is the number of subsystems, is a semi-metric function since it satisfies the following properties of a semi-metric function [146]:

- **Self-identity:** $FoS(S, \mu) = 0$ iff $k0d(S, u) = 0$, and since $k0d$ satisfies self-identity, as proven in [66], FoS also satisfies self-identity.
- **Positivity:** $FoS(S, \mu)$ satisfies positivity since both $k0d(S, \mu)$ and $max(k0d(S_i, \mu_i))$ are positive (as $k0d$ satisfies positivity [66]).
- **Symmetry:** $FoS(S, \mu)$ satisfies symmetry since both $k0d(S, \mu)$ and $max(k0d(S_i, \mu_i))$ satisfy symmetry [66].

Case 2 In the second case, we consider the co-existence of known vulnerabilities [147] and zero-day vulnerabilities [66]. Compared to zero-day vulnerabilities, which are typically less common

and known to fewer attackers, known vulnerabilities are much easier to exploit; equivalently, they impact the attack resilience much less from the defender's perspective and should count less than a zero-day vulnerability (which counts as 1 in calculating the $k0d$ metric). The probability of successfully exploiting a known vulnerability should reflect the relative difficulty of exploiting that vulnerability [147]. Instead of assigning arbitrary values, we employ the standard CVSS scores of known vulnerabilities [148], which are readily available in the public vulnerability database (which is updated frequently and can be accessed through a web browser [8] or accessed through an API (Application Programming Interface) from [9]). CVSS measures the impact of each individual vulnerability, and it also contains environmental metrics [149]. We normalize the CVSS score of a known vulnerability (which ranges between 0.0 and 10.0) in order to convert the vulnerability score to a probability of exploiting the vulnerability as $\frac{CVSS}{10.0}$, which is usually called the attack likelihood (or probability) of the vulnerability. Normalizing known vulnerabilities in order to estimate their exploit probabilities is also used in [147, 50]. This approach works because the CVSS score represents the intrinsic difficulty of exploiting a vulnerability [147, 50]. On the contrary, CVSS has three different types of scores which are base, impact, and exploitability. Based on that, the CVSS score covers more than exploitability. However, we choose the base score for calculations in this work (same approach as [147, 50]) mainly for two reasons:

- As the exploitability or impact subscores increase, the CVSS base score also increases. This implies that choosing the exploitability score instead of the base score may change average CVSS scores but if two known vulnerabilities are compared, as the exploitability score increases, the base score also increases so their relative difference does not get impacted significantly.

- Base score is available easily. There is a CVSS database that can be accessed through API calls [150], and there is another website from which CVE entries can be downloaded [151]. In both of those databases, the base score is directly accessible. However, impact and exploitability subscores need to be calculated using the CVSS vector.

As an example, the probability of a known vulnerability CVE-2012-0022 (used later in our

[8] CVE Details https://www.cvedetails.com/ [Online; Accessed 14-July-2019]
[9] Circl CVE Search https://cve.circl.lu/. [Online; Accessed 17-August-2019]

discussions) is 0.5 (the CVSS score is 5.0). For zero-day vulnerabilities, we follow the existing approach [147] of modeling them as a special kind of vulnerabilities with the following CVSS base metrics: remediation level "unavailable", report confidence "unknown", and exploitability "unproven that exploit exists". With those values as inputs to the CVSS equation [148], the CVSS score is calculated as 0.8, and therefore a nominal probability 0.08 is assigned to zero-day vulnerabilities. Finally, with probabilities obtained for both known and zero-day vulnerabilities, we can then convert each exploit of a known vulnerability into an equivalent number of exploits of zero-day vulnerabilities and use the result inside the same calculation of the FoS metric as in Case 1 above, using Equation 1. The attack path length for a vulnerability (either known or zero-day) is defined as:

$$log_{0.08}(Probability\ of\ the\ Vulnerability) \tag{2}$$

Case 3 The previous two cases are based on the worst-case scenario (from the defender's point of view), which provides a useful lower bound for the level of security resilience against the best attackers who can always follow the shortest paths. In the last case, we consider other attackers who may not (be able to) follow the shortest path, and we propose an average case-based metric called the *PFoS (Probabilistic Factor of Security)*. For this purpose, we adopt the Bayesian network-based attack graph model introduced in [46]. We first construct a Bayesian network using the attack graph as the DAG (Directed Acyclic Graph). The probabilities assigned to vulnerabilities (CVSS divided by 10 for known vulnerabilities and 0.08 for zero-day vulnerabilities) are interpreted as the conditional probabilities that the exploits of corresponding vulnerabilities can be executed given all the preconditions are already satisfied. Conditional probabilities also need to be defined to encode the logical AND and OR relationships between pre- and postconditions. In addition, we add common parent nodes for exploits which share the exact same vulnerabilities (e.g., same CVE entries for known vulnerabilities, same instances of the same service for zero-day vulnerabilities) to represent the fact that if the attacker exploits one of those vulnerabilities, the rest would have a much higher conditional probability of being exploited. To reflect this, for exploit nodes connected to a

common parent, the conditional probability has a higher value (e.g., 0.9) when all of their preconditions (including the common parent node) are satisfied. To illustrate, in Fig. 3.2, the attacker can compromise the substation gateway (node $A6$) by exploiting either $\langle V\_Gateway, 300, A6\rangle$ or $\langle V\_Gateway, 200, A6\rangle$. Since both of those exploit nodes represent zero-day exploits and both have the same instances of the Gateway (instance 1 shown in square brackets), they are identical. This implies that if the attacker exploits one of those, the probability that the attacker exploits the other one is high. This fact is represented by creating the common parent node(e.g., $\langle Gateway\rangle[1]$ in Fig. 3.2) and connecting this common parent node to both of those exploits. Finally, in order to model two components that are similar but not identical (e.g., two versions of the software), we have introduced the similarity nodes (an example is shown in Fig. 3.6).

Next, we perform Bayesian inferences based on the constructed Bayesian network as follows. We assign all initial conditions with probability 1.0 and calculate the probabilities of the goal condition inside each subsystem and that of the goal condition of the entire system. We then use Equation 2 above to convert these probabilities into the equivalent number of zero-day exploits corresponding to the entire system, denoted as $k0d_S$, and that of the i^{th} subsystem, denoted as $k0d_i$. Finally, the strength of a system S with multiple subsystems S_i for a given asset μ (μ_i in S_i) is defined as $Strength(S, \mu) = k0d_s$, and the stress is defined as $Stress(S, \mu) = \max_{\forall S_i} k0d_i$. With those definitions, the factor of security can be calculated as usual using Equation 1.

3.4 Applying FoS to Smart Grids and Substations

In this section, we illustrate how our FoS metrics can be applied to both the smart grid substations and the smart grid distribution domain to evaluate the security effectiveness of redundant subsystems. Specifically, we apply the metrics to two substation-specific attack scenarios, namely, the Precision Time Protocol (PTP) time delay attack [128] and the tripping circuit breakers attack [50], and two attack scenarios in smart grids, namely, the cascading link failure attack [2] and the coordinated attack against substations and transmission lines [3].

3.4.1 Attack Scenarios in IEC 61850 Substations

PTP Time Delay Attack

PTP is a protocol used to synchronize clocks in a network. The PTP time delay attack aims to delay PTP messages which are used for time synchronization in IEC 61850 substations [128]. Time synchronization among IEDs in substations is important since the lack of time synchronization can lead to the control center making wrong decisions, such as scheduling the power demand inefficiently [135]. In fact, PTP requires a master clock, which acts as the main time source, and slave clocks, which get timing information from the master clock. In our substation model (Fig. 3.1), the GPS clocks (nodes $A3$ and $B3$) are PTP masters for each subsystem. In order to attack this protocol, the attacker needs to delay or modify messages used in PTP for time synchronization [128]. The attacker could be located physically close to the substation such that he/she can utilize electronic spoofing equipment to spoof GPS messages [136]. The attacker could also remotely access the substation to launch the attack, e.g., by compromising substation gateways (nodes $A6$ or $B6$) or by causing malware to be installed on the remote operator's machine (node 200). In this scenario, we do not assume the physical proximity but consider the latter case.

Fig. 3.2 shows our attack graph model for the PTP time delay attack. The attacker can either compromise the remote operator or the control center in order to gain access to the substation gateway. After compromising the substation gateway, the attacker needs to compromise the workstation and then the GPS server in order to cause a time delay. Therefore, to compromise each subsystem, the attacker needs to exploit in total five vulnerabilities (one in SSH version 1 or SSH version 2 or HTTP version 1, followed by one in the gateway, one in the workstation, one in the GPS server, and also the one in the firewall) to compromise each subsystem, and the $k0d$ value of each subsystem is 5 (since all vulnerabilities are assumed to be zero-day). On the other hand, in order to compromise the whole system, the attacker needs nine distinct vulnerabilities in total (one in SSH version 1, one in Gateway version 1, one in Gateway version 2, one in SCADA version 1, one in SCADA version 2, one in GPS version 1, one in GPS version 2, one in Firewall version 1, and one in Firewall version 2), so the FoS (Factor of Security) can be calculated as the following.

$$FoS = \frac{9}{max(\{5,5\})} = 1.8 \tag{3}$$

Observation: This simple application of the FoS metric can lead to the following observations. First, as it can be seen, the FoS value is less than two because part of the attack graph (exploits of the remote operator's machine or network services of the control center) is shared by both subsystems. The insight is that while it is common for redundant subsystems to share certain external components, such shared components may lead to common attack surfaces across the subsystems and hence reduce the overall security effectiveness. Second, we can see that diversity in the gateway (nodes $A6$ and $B6$) and GPS servers (nodes $A3$ and $B3$) between the two subsystems (i.e., the two subsystems are using two different versions of those services) is the key to improving the FoS value, even though diversifying such components may or may not be feasible in practice.

For the calculation of the PFoS (Probabilistic Factor of Security), we perform Bayesian inference on the goal condition of each subsystem, which is $\langle TIME\_DELAY, A3\rangle$, and $\langle TIME\_DELAY, B3\rangle$, respectively, and on the goal condition of the whole system, which is $\langle TIME\_DELAY, \{A3, B3\}\rangle$. Probabilities for those nodes can be calculated as follows.

- Pr($\langle TIME\_DELAY, A3\rangle$) = 0.0000082037
- Pr($\langle TIME\_DELAY, B3\rangle$) = 0.0000082037
- Pr($\langle TIME\_DELAY, \{A3, B3\}\rangle$) = 0.0000000003

By taking those probabilities into account, PFoS can be calculated as follows.

$$L0 \leftarrow log_{0.08}(0.0000082037) = 4.63665$$

$$L1 \leftarrow log_{0.08}(0.0000082037) = 4.63665$$

$$L2 \leftarrow log_{0.08}(0.0000000003) = 8.681549$$

$$PFoS = \frac{L2}{max(L0, L1)} = 1.87237$$

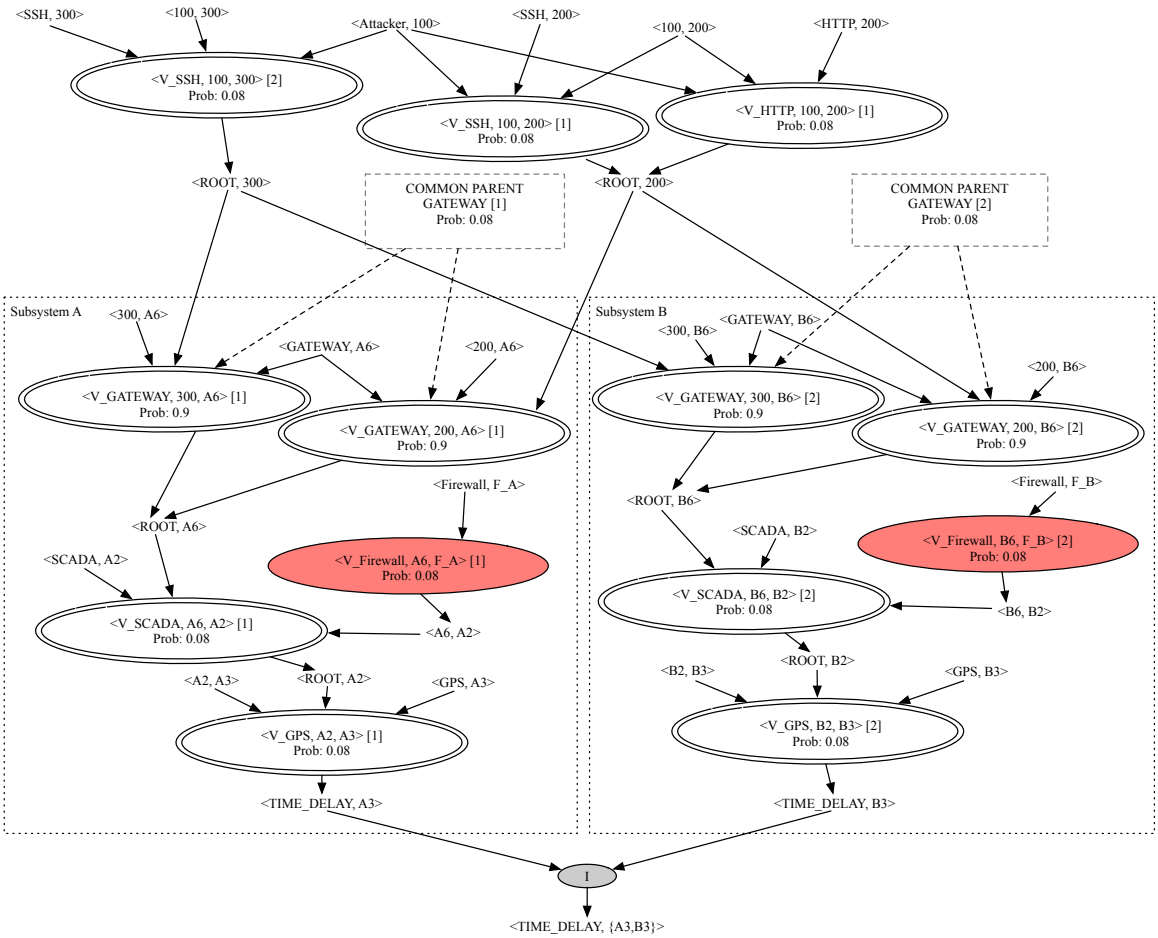


Figure 3.2: Attack Graph for PTP Time Delay Attack

Observation: We can observe that, in contrast to the calculation of FoS, the Strength and Stress under PFoS both become slightly lower. This is mainly due to the existence of multiple attack paths, which are taken into consideration under PFoS, whereas only one (the shortest) path is considered under FoS. While the difference is not significant in this case (since there are only a few attack paths), in general, the two metrics may have different trends as the number of attack paths changes. (hence it is important to consider both), especially when many attack paths exist. We will investigate this further through simulations in Section 3.5.

Tripping Circuit Breakers Attack

Circuit breakers protect transmission lines from damage due to excess current by disabling transmission lines. As an example, transmission line failures were seen in the 2003 Northeastern

blackout [152]; even though this blackout was not a result of a cyber attack, this could as well be the case since an attacker who has access to the substation HMI will be able to send trip commands from the HMI to Protection IEDs remotely [50]. In the Ukraine attack [141], attackers tripped circuit breakers by sending remote commands, and they also changed the firmware contained in IEDs to delay repair attempts.

Fig. 3.3 shows the attack graph representation of the tripping circuit breakers attack. The attacker must compromise protection IEDs in order to trip circuit breakers. In order to do so, the attacker needs to first compromise the gateway and the HMI to gain access to protection IEDs. For each subsystem, the attacker may be detected at the last step of the attack using IDS_A_1 and IDS_B_1, respectively. In Fig. 3.3, the attacker needs to exploit four different vulnerabilities for each subsystem. On the other hand, to compromise the whole system, the attacker needs six different vulnerabilities (note that the same version number of both protection IEDs, A7, and B7, indicate these are identically configured), so the FoS (Factor of Security) in this scenario can be calculated as follows.

$$FoS = \frac{6}{max(\{4,4\})} = 1.5 \tag{4}$$

For the calculation of the PFoS (Probabilistic Factor of Security), we perform Bayesian inference on the goal conditions of subsystems, $\langle TRIPPED, A7\rangle$ and $\langle TRIPPED, B7\rangle$, and on the goal condition of the whole system, $\langle TRIPPED, \{A7, B7\}\rangle$. Probabilities for those nodes can be calculated as follows.

- Pr($\langle TRIPPED, A7\rangle$) = 0.00003544017
- Pr($\langle TRIPPED, B7\rangle$) = 0.00003544017
- Pr($\langle TRIPPED, \{A7, B7\}\rangle$) = 0.000000007

By taking those probabilities into account, PFoS can be calculated as follows:

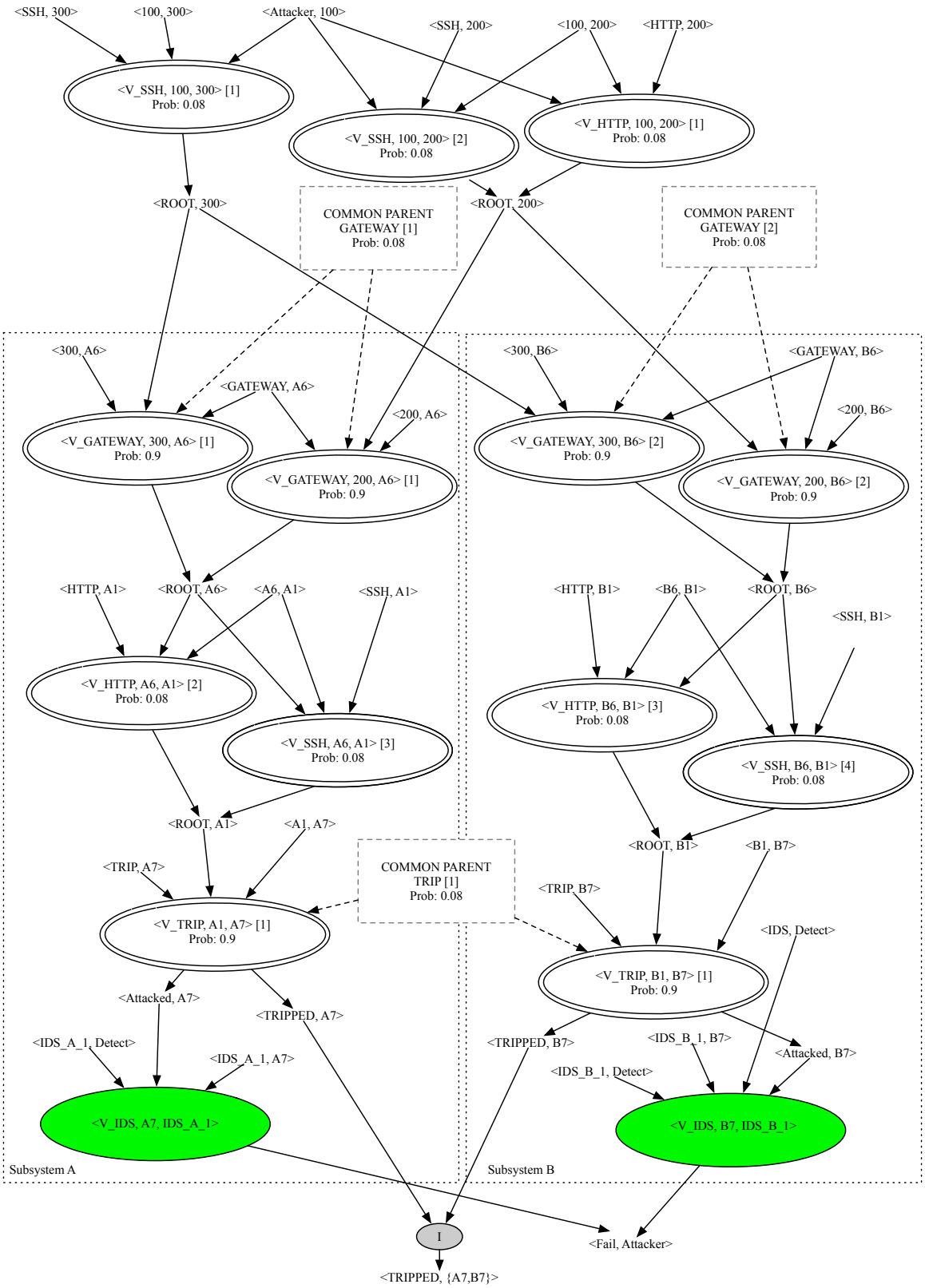


Figure 3.3: Attack Graph for Tripping Circuit Breakers

$$L0 \leftarrow log_{0.08}(0.000000007) = 7.434431$$

$$L1 \leftarrow log_{0.08}(0.00003544017) = 4.057310$$

$$L2 \leftarrow log_{0.08}(0.00003544017) = 4.057310$$

$$PFoS = \frac{L0}{max(L1, L2)} = 1.832354$$

Observation: Comparing those two attack scenarios, we can observe that the FoS metrics yield different values for different attack scenarios, even though they are both evaluated with respect to the same substation. In practice, the power grid operator will need to assign weights to those results to reflect the relative importance of those different threats and aggregate the weighted results to evaluate the overall security effectiveness of the design. Also note that the existence of IDS does not affect FoS since FoS only depends on the shortest attack path, which would not include an IDS since it will cause the attack to fail. On the other hand, PFoS may be affected by IDS since PFoS considers all attack paths and an IDS may reduce the probability of the attack by causing some attack paths to diverge to the failure nodes.

3.4.2 Attack Scenarios in the Smart Grid Distribution Domain

Cascading Link Failure Analysis

The Cascading Link Failure Analysis is introduced in [2]. In this attack scenario, the smart grid network is modeled as a directed graph where the nodes include producer nodes (power generators), intermediate nodes (substations), and consumer nodes (end users), and the edges represent power and communication lines. The goal of attackers is to cause as much power loss as possible by attacking the minimum number of transmission and communication lines. If attackers disconnect all incoming power or communication lines to a consumer node, this will cause a blackout at the consumer node. Fig. 3.4 shows a smart grid network that consists of a transmission network and a communication network. In this system, $N1$ and $N2$ are consumer nodes, $S1$ and $S2$ are transmission substations, $S3$ is a distribution substation, $T1$, $T2$, $\cdots$, $T6$ are transmission lines.

$C1$, $C2$, $C3$, and $C4$ are communication lines. Assuming that $S1$ and $S2$ are secure and cannot be attacked, in order to cause a blackout in the area $N1$, attackers could choose to compromise any of the following combinations, $T2$ and $T5$, $C1$ and $C3$, or both.

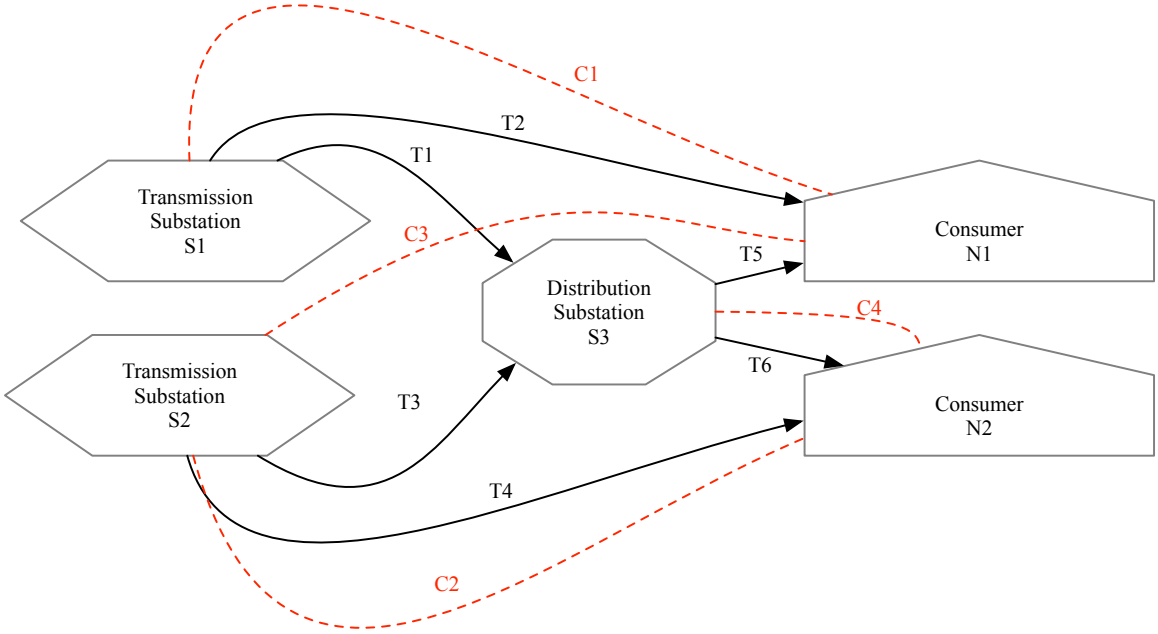


Figure 3.4: Smart Grid Transmission and Communication Networks [2]

In Fig. 3.5, the attacker needs to follow one of the three paths to compromise each subsystem. Among those paths, the shortest path is the path containing two known vulnerabilities (CVE-2012-1442, CVE-2012-0022). Therefore, the attacker can trip $T2\_1$ and $T5\_1$ to cause a blackout in area $N1\_1$. The $k0d$ value for the shortest path can be calculated as follows.

$$k0d\ (T2\_1\ \rightarrow\ T5\_1, \langle Blackout, N1\_1\rangle) = \log_{0.08}(0.43) + \log_{0.08}(0.5)$$
$$= 0.3341 + 0.2744 = 0.6095$$

Similarly, the same two known vulnerabilities can be used to compromise the second subsystem, and the $k0d$ value can be calculated as $k0d\ (T2\_2\ \rightarrow\ T5\_2, \langle Blackout, N1\_2\rangle) = 0.6095$. For the whole system, the attacker can use the same two known vulnerabilities to compromise both subsystems and reach the goal condition "$\langle Blackout, \{N1\_1, N1\_2\}\rangle$". Therefore, the $k0d$ value for the whole system can be calculated as:

$$AL\ (T2\_1\ \rightarrow\ T5\_1\ \rightarrow\ T2\_2\ \rightarrow\ T5\_2, \langle Blackout, 0\rangle) = 0.6095$$

By taking those into account, the FoS value can be calculated as follows.

$$FoS\ =\ \frac{0.6095}{max(0.6095, 0.6095)}\ =\ 1.0 \tag{5}$$

For the PFoS metric, we perform Bayesian inference on subsystem goal conditions, which are "$\langle Blackout, N1\_1\rangle$" and "$\langle Blackout, N1\_2\rangle$", and the goal condition of the whole system. Probabilities can be calculated as follows.

- Pr($\langle Blackout, N1\_1\rangle$) = 0.227665
- Pr($\langle Blackout, N1\_2\rangle$) = 0.13325522
- Pr($\langle Blackout, \{N1\_1, N1\_2\}\rangle$) = 0.107674496

By taking all of those probabilities into account, the PFoS can be calculated as follows.

$$L0 \leftarrow log_{0.08}(0.107674496) = 0.882376$$
$$L1 \leftarrow log_{0.08}(0.13325522) = 0.7979832$$
$$L2 \leftarrow log_{0.08}(0.227665) = 0.585922$$
$$PFoS\ =\ \frac{L0}{max(L1, L2)} \approx 1.105757$$

Observation: It can be seen that, in this special case, the FoS value of 1 implies that the design of two subsystems does not provide any advantage in terms of security resilience against attackers who follow the shortest attack paths. On the other hand, the PFoS value indicates a slightly more optimistic scenario where attackers may not necessarily (be able to) follow the shortest paths, and hence the redundancy still provides a little more security resilience against such attackers. This is

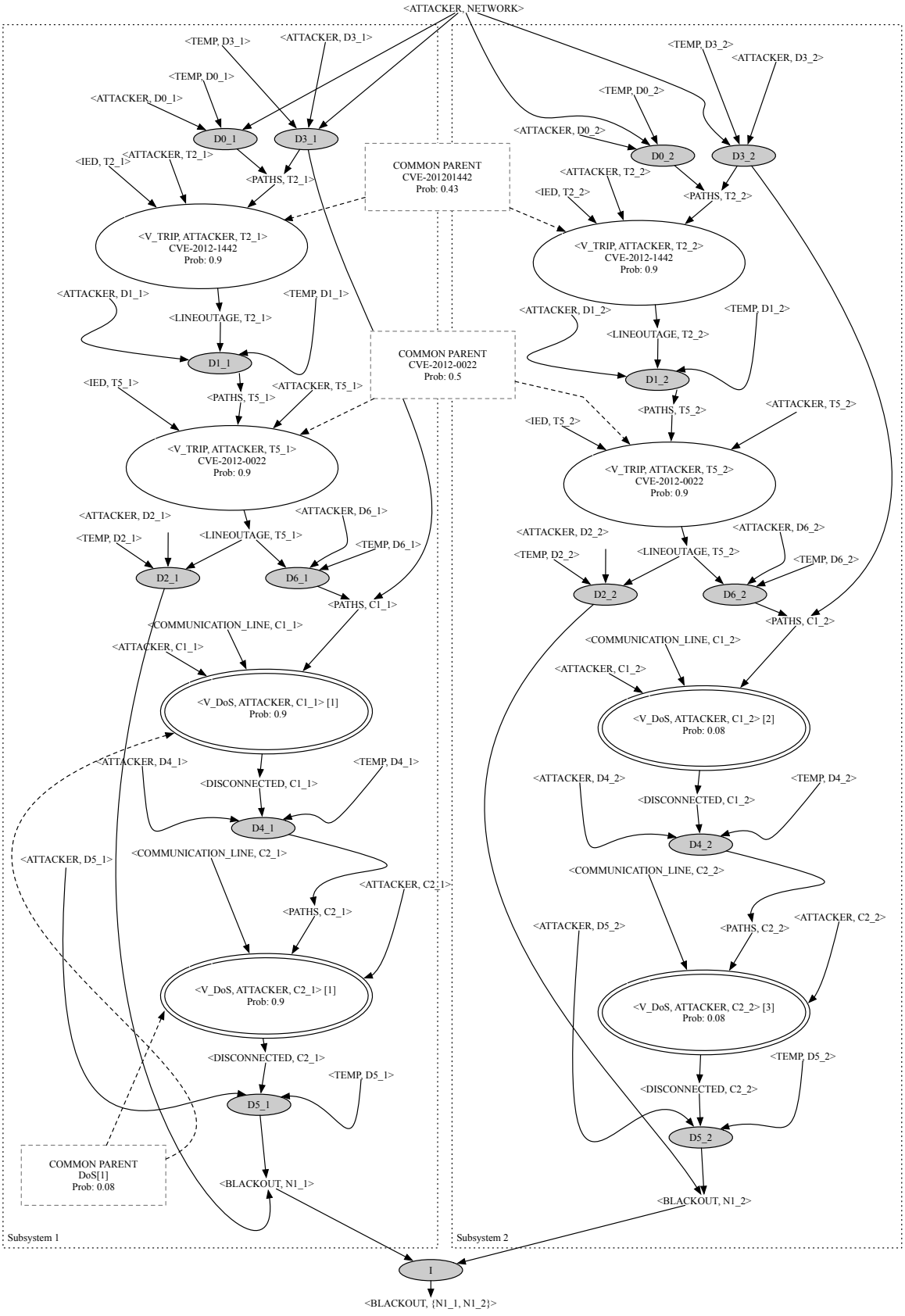


Figure 3.5: Attack Graph for Cascading Link Failure Analysis

also usually (not always) true in general, i.e., the PFoS is slightly more optimistic than FoS since the former is designed to reflect the average case scenario (i.e., attackers may or may not have the skills and resources to take advantage of the shortest paths).

Coordinated Attack Against Substations and Transmission Lines

Originally, coordinated attacks against substations and transmission lines were presented in [3]. In these attacks, attackers target combinations of substations and transmission lines concurrently. Assuming that substations $S1$ and $S2$ are secure and cannot be attacked, the attacker has to target $S3$ and transmission line $T2$ and optionally target transmission lines $T1$ and $T5$ in order to cause a blackout in the area $N1$.

Fig. 3.6 demonstrates the attack graph of the coordinated attack against substation S3 and transmission line T2. Attackers can attack substations by compromising substation gateways and attack transmission lines by tripping circuit breakers. In Fig. 3.6, the attacker can first trip the circuit breaker, $T2\_1$, and then compromise the gateway of the substation $S3\_1$, which gives the shortest attack path with the least $k0d$ value. Since $T2\_1$ contains a zero-day vulnerability, exploiting it counts as one toward the $k0d$ value calculation. $S3\_1$ contains a known vulnerability with probability 0.75, so the equivalent $k0d$ value for $S3\_1$ is $log_{0.08}0.75 = 0.1139$. Therefore, the shortest path has a $k0d$ value of 1.1139. A similar calculation applies to the other subsystem, and the shortest path also yields a $k0d$ value of 1.1139. For the whole system, the attacker can reuse known vulnerability CVE-2012-0461 for exploiting both $S3\_1$ and $S3\_2$, so he/she needs two zero-day vulnerabilities (those in $T5\_1$ and $T5\_2$) and one known vulnerability (CVE-2012-0461) to compromise the whole system, and the $k0d$ value for the whole system is $2 + 0.1139 = 2.1139$. By taking these into account, the FoS can be calculated as follows.

$$FoS = \frac{2.1139}{max(1.1139, 1.1139)} = 1.897747 \tag{6}$$

For the calculation of PFoS, we also consider the similarity node in Fig. 3.6 since $T2\_1$ and $T2\_2$ are considered as different but similar IEDs (e.g., different versions). After adding that similarity

node, Bayesian inference is performed on goal conditions of subsystems, $\langle BLACKOUT, N1\_1\rangle$ and $\langle BLACKOUT, N1\_2\rangle$, and for the goal condition of the whole system, $\langle BLACKOUT, \{N1\_1, N1\_2\}\rangle$. The probabilities for those nodes are calculated as follows.

- Pr($\langle BLACKOUT, N1\_1\rangle$) = 0.0603624
- Pr($\langle BLACKOUT, N1\_2\rangle$) = 0.048270
- Pr($\langle BLACKOUT, \{N1\_1, N1\_2\}\rangle$) = 0.018565262

By taking those probabilities into account, the PFoS can be calculated as follows.

$$L0 \leftarrow log_{0.08}(0.018565262) = 1.57834$$
$$L1 \leftarrow log_{0.08}(0.048270) = 1.200027$$
$$L2 \leftarrow log_{0.08}(0.0603624) = 1.1115164$$
$$PFoS = \frac{L0}{max(L1, L2)} = 1.31525374$$

Observation: It can be seen that, in this special case, the PFoS value is actually smaller than the FoS value, which is opposite to most cases seen above and to the general pattern that PFoS would be slightly more optimistic. Therefore, the behavior of those metrics is not always straightforward, and we will investigate this further in the coming section through simulations.

3.5 Simulations

3.5.1 Experimental Design

Our simulations are performed on both the substation level and the smart grid level. We are using IEEE 14-bus [153] for the smart grid level. For substations, we use the substation configuration given in Fig. 3.1 as the seed graph and generate different substation configurations by randomly adding more hosts and connections between hosts. The IEEE 14-Bus does not originally include communication lines, so we make additional efforts to add communication lines to IEEE 14-Bus

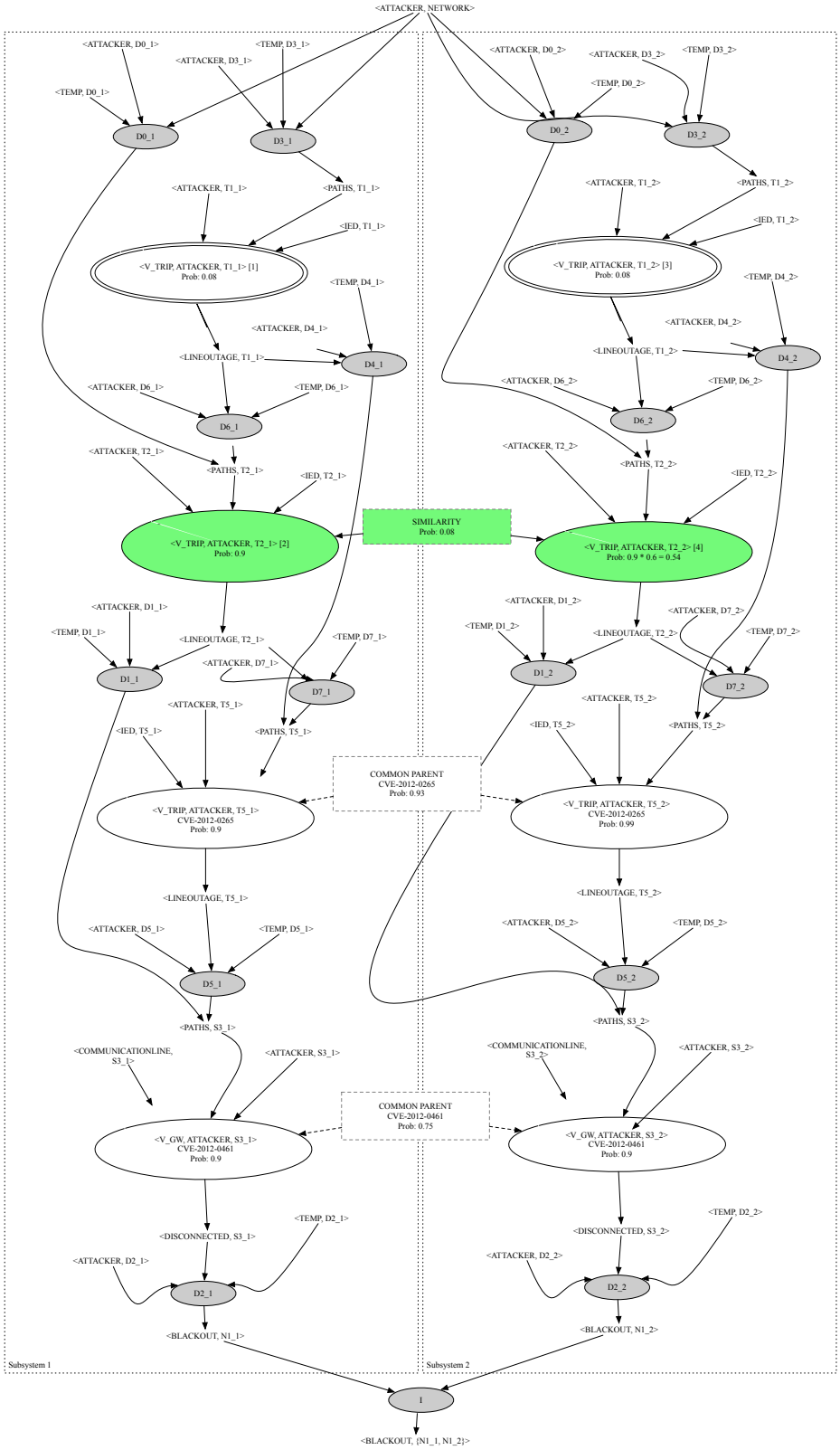


Figure 3.6: Attack Graph for the Coordinated Attack [3]

following the literature [2, 132]. In [132], the authors provide communication infrastructure topology for the IEEE 14-Bus system, and the same communication topology is used in [2]. After adding communication infrastructure to IEEE 14-Bus, we develop attack graphs that include ways whereby Bus 5 in IEEE 14-Bus can be attacked in order to cause a blackout, similar to the ones in Figs. 3.5 and 3.6. The energy components and control elements are not considered as parts of the attack graph modeling. The attack graph for the IEEE 14-Bus and other attack graphs shown in the previous section are taken as seed graphs for generating attack graphs. For a given seed graph, hosts are added with a randomly chosen set of services, and random connections are added between the newly added host and existing hosts. Known vulnerabilities are chosen from the CVE database [8]. Zero-day vulnerabilities are assumed to exist in most services. After adding hosts, firewalls and intrusion detection systems are added to attack graphs. Firewalls are added by choosing a random connection condition in the attack graph and placing that connection condition behind a firewall. IDSes are added by choosing a random privilege condition (which is not included in initial conditions) and adding a connection from the chosen random privilege condition to the IDS exploit and from the IDS exploit to the failure condition.

All simulations are performed on MacOS Mojave 10.14.1 with a 2.9 GHz Core i7 processor and 16 GB RAM. The attack graph generation and FoS calculation are implemented in Python 3.6, and the Bayesian inference is based on the automated reasoning tool SamIam[10]. Our experimental setup is summarized in Table 3.2. Section 3.5.2 compares the FoS and PFoS in different substation setups. Section 3.5.3 studies both metrics with security aspects, and Section 3.5.4 provides an extensive study with extra security mechanisms. Section 3.5.5 studies the behavior of metrics with respect to different types of attacks and their combinations.

3.5.2 Comparison Between FoS and PFoS

In this section, we evaluate the behavior of the metrics based on substation setups. Our first simulation aims to determine how our metrics would behave when applied to systems of different sizes, e.g., when the utility expands its smart grid network or substations. This simulation is based

[10] SamIam: Sensitivity Analysis, Modeling, Inference, and More http://reasoning.cs.ucla.edu/samiam/ [Online; Accessed 23-March-2018]

Table 3.2: Experimental Setup

	Smart Grid	Substation
Type of Attacker	Remote attacker who has access to IED controlling transmission lines, communication lines and substation gateways.	Remote attacker who resides in the same network as the substation gateway.
Attack Scenarios	Cascading link failure analysis and coordinated attack against transmission lines and substations.	PTP time delay attack and tripping circuit breakers attack.
Attack Method	Disconnecting transmission lines, disrupting communication lines or compromising substation gateways.	Exploiting vulnerabilities, delaying PTP messages, tripping circuit breakers.
Number of Nodes in Attack Graphs	10-600	10-500
Characterization of The Power Network	IEEE 14-Bus which includes 14 buses, 5 generators, 11 loads, 17 transmission lines, with communication lines added similar to [2]	The power network is not considered.
Deployment of Firewalls and IDS.	Connectivity conditions are chosen randomly before which firewalls are added. IDSes are added as special types of exploits in which the precondition is one of the postconditions gained by the attacker and if the detection succeeds, the attack fails.	Connectivity conditions are chosen randomly before which firewalls are added. IDSes are added as special types of exploits in which the precondition is one of the postconditions gained by the attacker and if the detection succeeds, the attack fails.

on over 5,000 attack graphs. First, we group attack graphs with similar sizes together and calculate the average FoS and PFoS values for each group. In addition, we generate a set of capabilities (i.e., which vulnerabilities may be exploited) for 500 fictitious attackers where the number of zero-day vulnerabilities exploitable by each attacker follows a normal distribution, and the number of known vulnerabilities the attacker can exploit depends on the CVSS score of the known vulnerability (e.g., for a vulnerability with a CVSS score of 9.0, the attacker would be able to exploit it with probability 0.9). We compare the predicted attack resilience results of the FoS metrics to the actual success rate in terms of the number of attackers who can successfully reach the goal condition with their assigned capabilities.

Results and Implications: As can be seen in Fig. 3.7, as the attack graph size increases, the metrics generally decrease since a larger attack graph would mean more identical components across the subsystems, and hence the redundancy becomes less effective in terms of providing security resilience. We can also observe that, in contrast to FoS, PFoS decreases more slowly since it is based on all attack paths. Finally, as metrics decrease, the number of successful attackers increases following a reversed trend, which means the metrics can reflect the expected security effectiveness of the design.

Our second simulation aims to analyze how metrics behave with more diversity available, e.g., when the utility invests in having different types of communication equipment, IEDs, and other devices deployed in the same substation. Those different choices of resources are modeled using the version numbers in our attack graphs, as discussed in the previous section, and we will call the collection of such choices the *resource pool*. We perform simulations by varying the size of the resource pools. This simulation is based on 900 attack graphs, and the resource pool sizes are

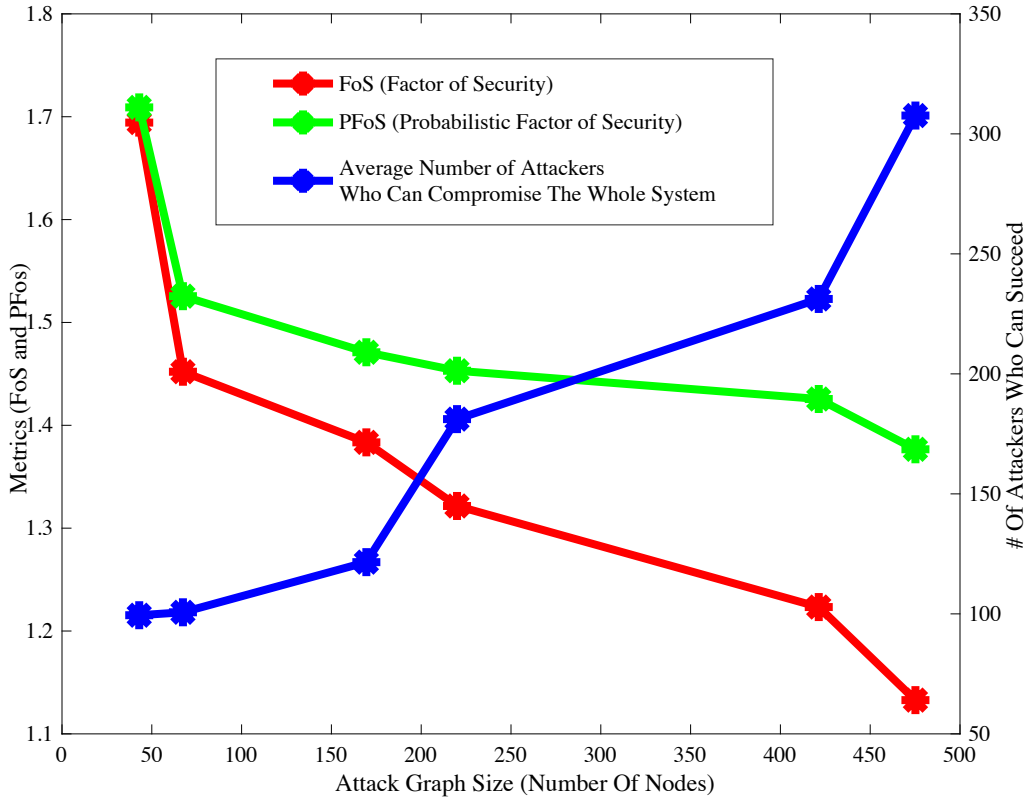


Figure 3.7: FoS and $PFoS$ in Sizes of Attack Graphs

4, 8, 12, 16, 20, and 24.

Results and Implications: As it can be seen in Fig. 3.8, as the resource pool size increases, both metrics generally increase. However, the improvement gradually flattens out as the resource pool size further increases since other factors, such as the shared components across subsystems, become more dominant. Therefore, investing in diversity may improve security effectiveness to some extent, but ultimately the system design (the topology) becomes the determining factor. Nonetheless, as metric values increase, values of the number of successful attackers decrease.

Our third simulation aims to determine how metrics behave at the smart grid level. We developed attack graph models for the IEEE 14-Bus system with communication lines [2], and such attack graphs are taken as seed graphs to generate more attack graphs. We perform simulations to see how the number of successful attackers would change as both metrics increase.

Results and Implications: As it can be seen in Fig. 3.9, in IEEE 14-Bus systems, the number of

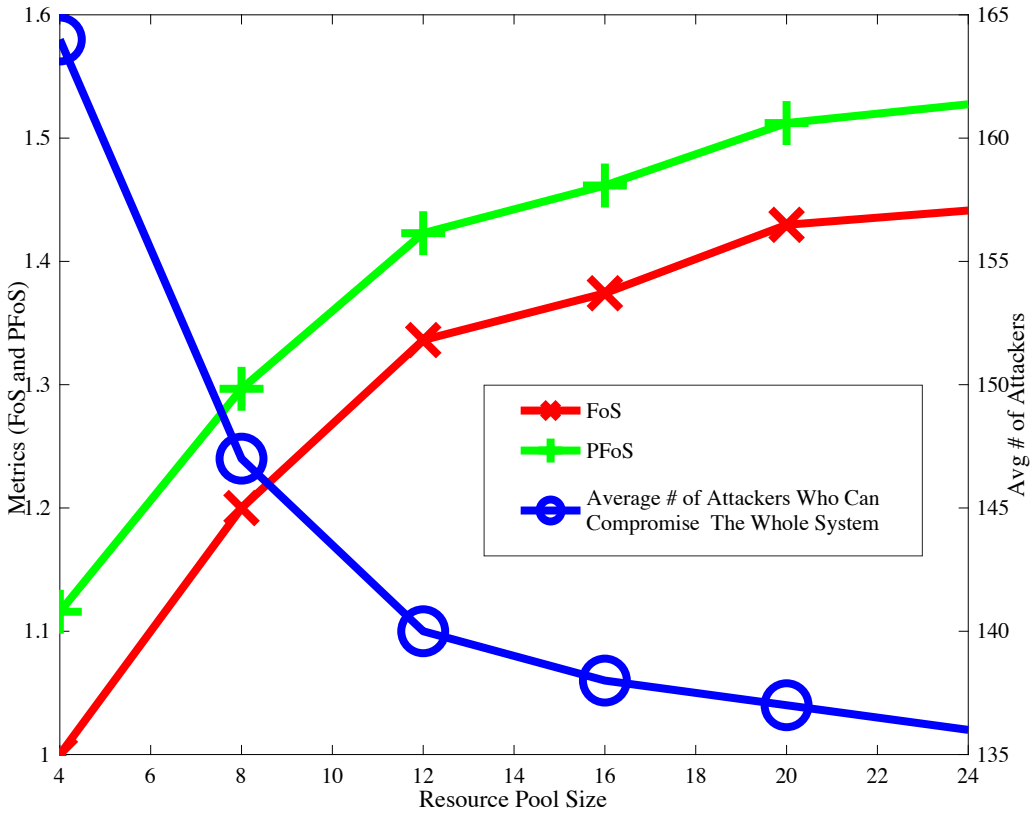


Figure 3.8: FoS and $PFoS$ in Sizes of Resource Pools

successful attackers almost decreases linearly in FoS. The FoS also covers the full range of values between 1 and 2. The PFoS only starts from 1.2 and ends around 1.9, and there is a sharper decrease in the number of attackers before the PFoS value of 1.4. PFoS values have a limited range due to the average nature of PFoS (e.g., no systems could yield a value less than 1.2 when taking into consideration all the attack paths). This also explains the initial sharp decrease in the number of attackers with respect to PFoS.

Our fourth simulation aims to determine the relationship between FoS and PFoS. We group attack graphs with similar ranges of FoS values and analyze the relationship between their PFoS values and the number of attack paths (since the key difference between FoS and PFoS is the number of paths that is taken into consideration).

Results and Implications: As can be seen in Fig. 3.10, for smaller values of FoS (lower curves), PFoS clearly increases in the number of paths. As FoS gets larger, the trend becomes less obvious

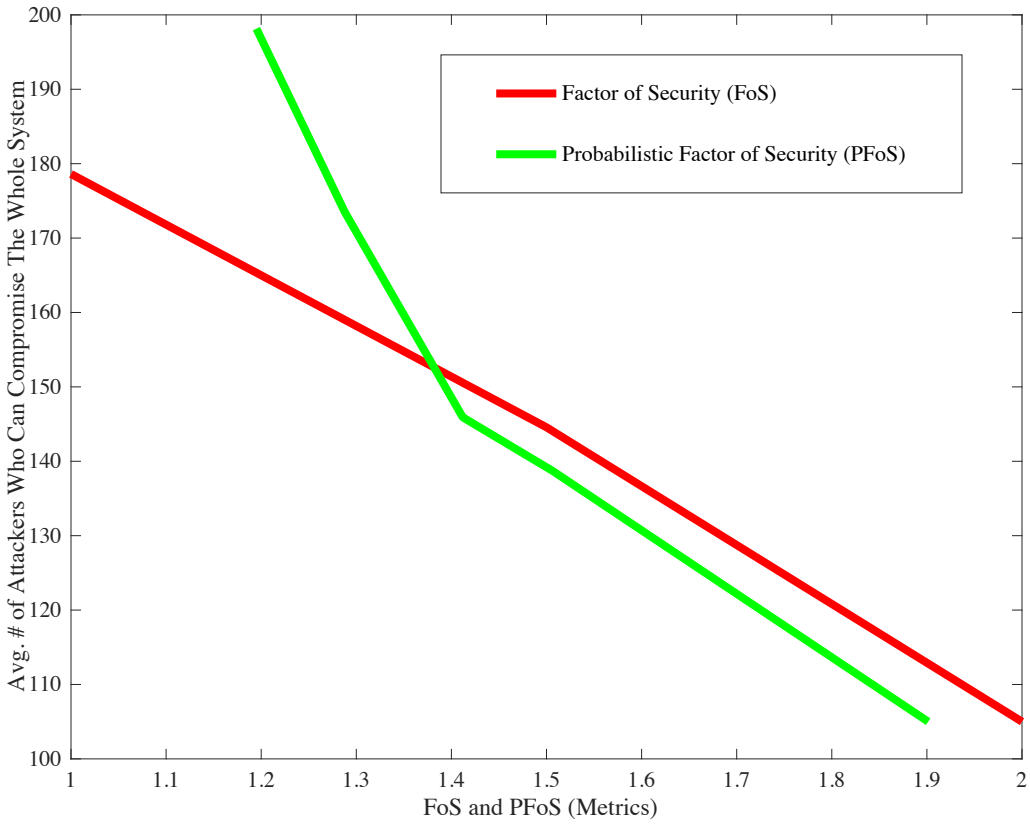


Figure 3.9: FoS and $PFoS$ Applied to the IEEE 14-Bus

(i.e., there is no significant difference between FoS and PFoS). This indicates that, for systems with a smaller value of the FoS metric but a large number of attack paths (this implies that the system is not well designed in terms of both security and redundancy), the difference between those two metrics (i.e., PFoS is more optimistic than FoS) becomes more significant, and hence it becomes more important to consider both metrics at the same time. Conversely, for better-designed systems with higher FoS values or fewer paths, the two metrics behave similarly, and thus using one of those metrics might be enough.

3.5.3 Metrics versus Security Aspects

In this section, we simulate various security aspects to observe the behavior of FoS and PFoS. One unique aspect of cyber-physical systems like smart grids is that some components may only be available from a small number of manufacturers, and diversifying such components in practice may

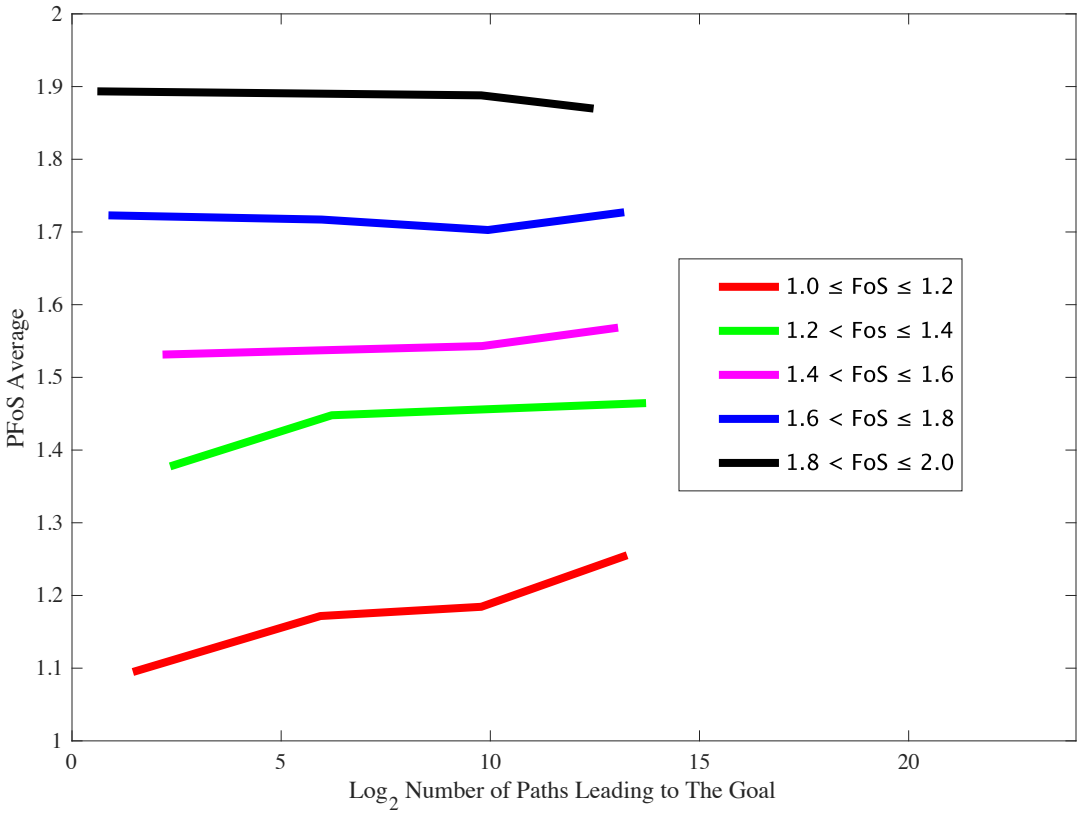


Figure 3.10: FoS and $PFoS$ in Number of Paths

not be feasible. Therefore, our next two simulations analyze the effect of such components on the security effectiveness of redundant systems. The fifth simulation analyzes how our metrics behave when IEDs cannot be diversified.

Results and Implications: In Fig. 3.11, we can see that both metrics yield lower values when IEDs cannot be diversified. While FoS almost linearly decreases in graph sizes when IEDs can be diversified, the decrease flattens at a certain point in the other three cases since the role of diversity diminishes and other factors, such as the topology, become more dominant.

Our sixth simulation analyzes how metrics behave when communication equipment cannot be diversified. Since communication in smart grid networks or IEC 61850 substations must follow specific protocols, this is a realistic scenario.

Results and Implications: In Fig. 3.12, as it can be seen, the gap between metric values (with

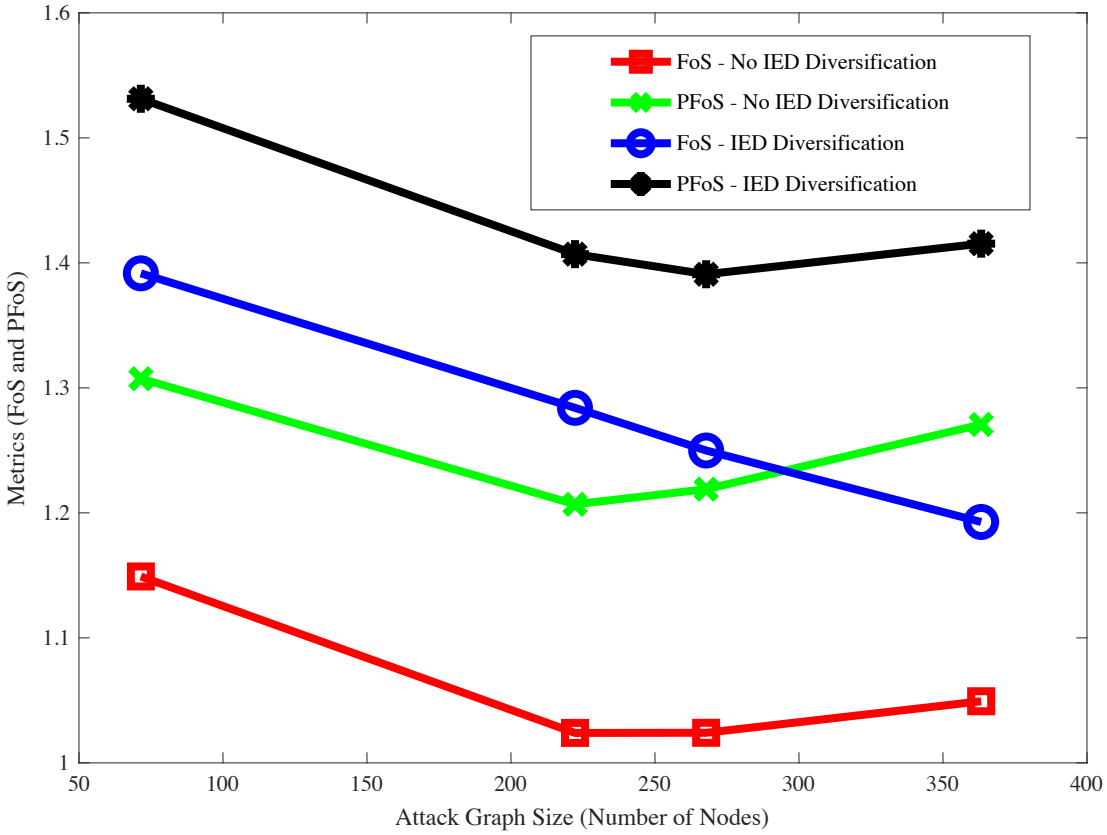


Figure 3.11: FoS and $PFoS$ when IEDs Can(not) be Diversified

and without diversity) becomes smaller compared to the previous case (diversification of IEDs). This implies that the diversity of different components in a system may be of different significance w.r.t. the effect on the overall security effectiveness of the redundant design, e.g., diversifying the IEDs may yield more benefit than diversifying the communication equipment.

Our seventh and eighth simulations analyze how the percentage of zero-day vulnerabilities to known vulnerabilities affects the metrics. First, we conduct experiments by making all the vulnerabilities in the substation zero-day. After that, we experiment by making half of the vulnerabilities zero-day and making the other half known.

Results and Implications: Fig. 3.13a shows how metrics behave when all vulnerabilities are assumed to be zero-day. As can be seen, metrics behave similarly when all vulnerabilities are zero-day since all vulnerabilities will be treated the same. Fig. 3.13b shows how metrics behave when

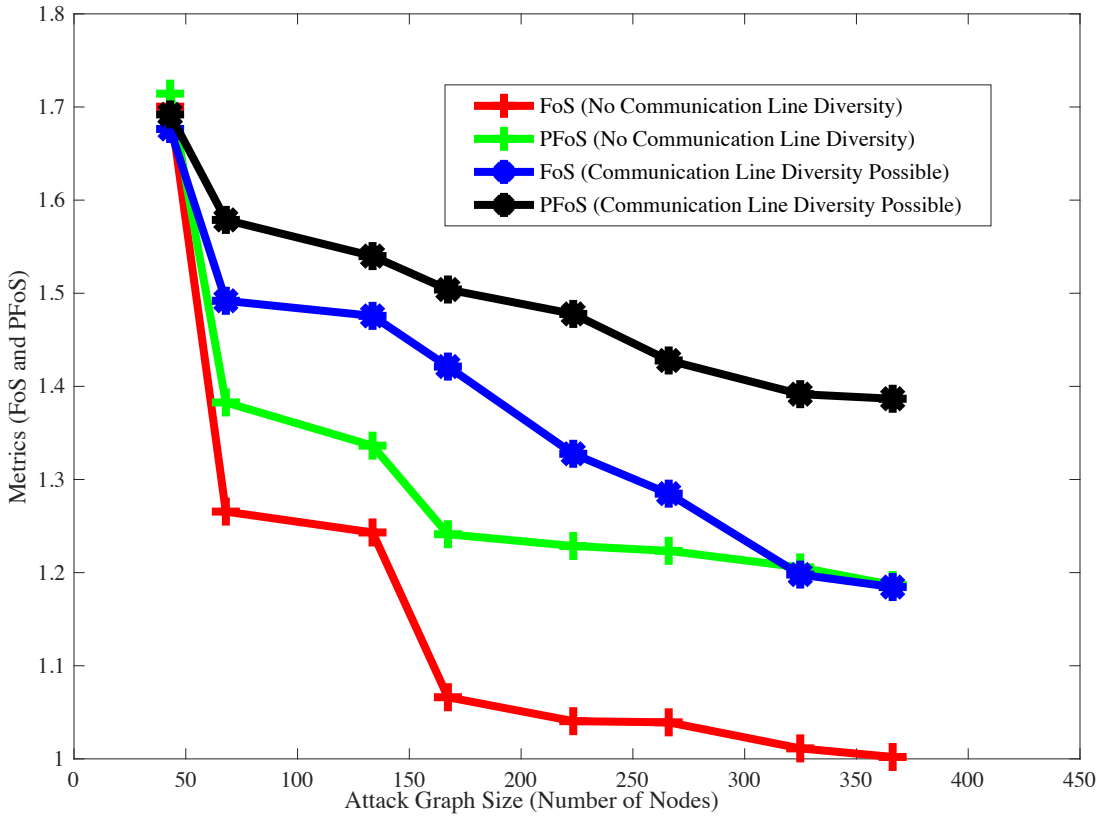
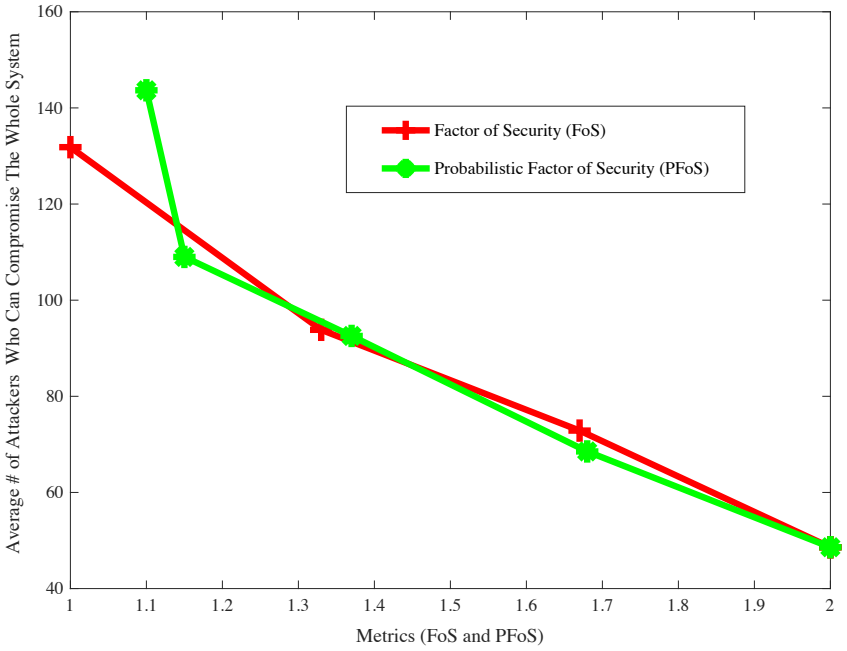


Figure 3.12: FoS and $PFoS$ when Communication Equipment Can(not) be Diversified

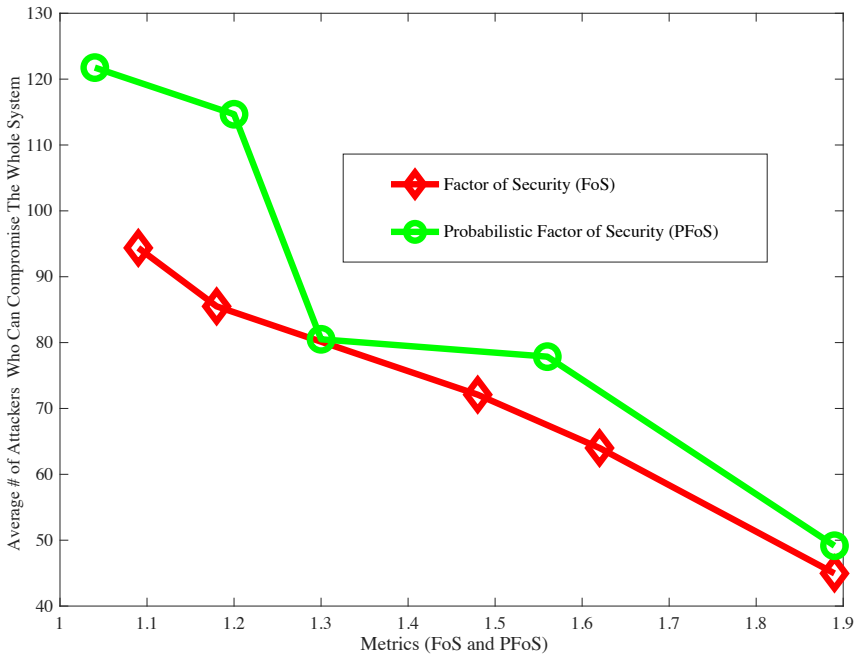
half of the vulnerabilities are zero-day and the other half of vulnerabilities are known. In that case, the metrics diverge slightly due to the increasing uncertainty in the CVSS scores.

Our last set of simulations in this section highlights how the FoS metric behaves when applied to one specific attack in the IEC 61850 substations. The two simulations are performed by generating attack graph models by taking the PTP time delay attack as the seed graph. We focus on the FoS metric, and we fix the $k0d$ value of each subsystem assuming that all vulnerabilities are zero-day.

Results and Implications: As it can be seen in Figs. 3.14a and 3.14b, the attack graphs are much smaller in this case since we are considering one specific attack in a substation. We can observe that the results are quite similar to previous cases, i.e., as the size of substations increases, FoS decreases as the number of successful attackers follows a reversed trend. Also, although larger substations may yield larger $k0d$ values, the size does not have a significant impact on the FoS values. In conclusion, the metrics can be applied with respect to one attack or multiple attacks and

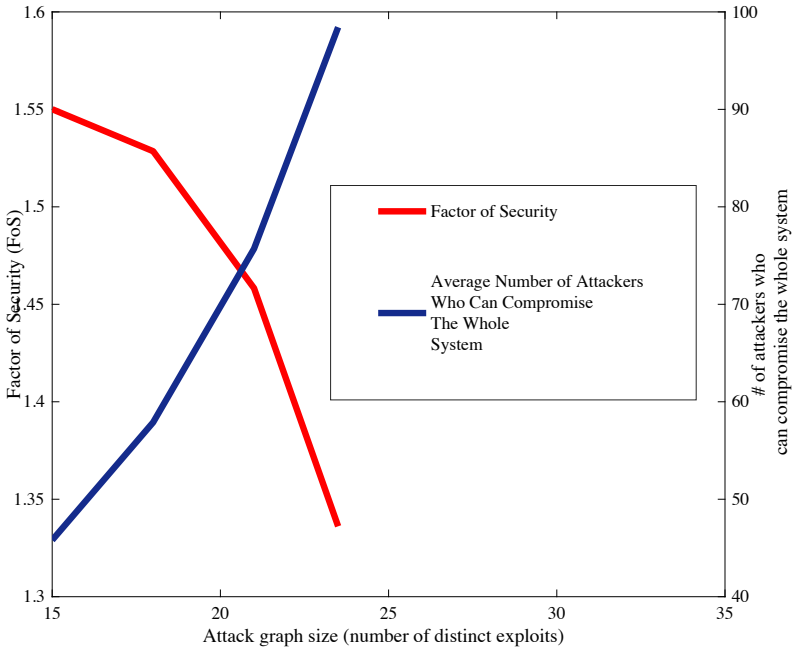


(a) Metrics When All Vulnerabilities Are Zero-Day

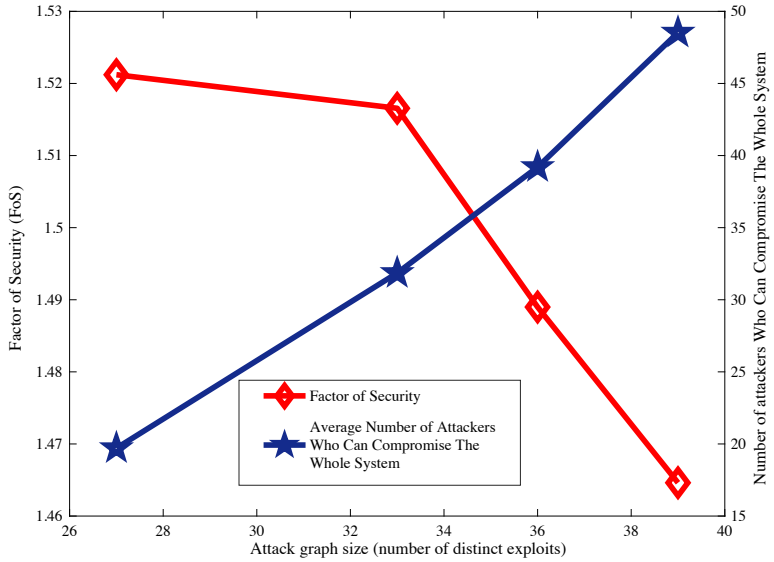


(b) Metrics When Half of Vulnerabilities Are Known

Figure 3.13: FoS and $PFoS$ versus Vulnerabilities



(a) FoS for PTP Attack in IEC 61850 Substation When the $k0d$ Value is 5 in Each Subsystem



(b) FoS for PTP Attack in IEC 61850 Substation When the $k0d$ Value is 10 in Each Subsystem

Figure 3.14: FoS versus Different $k0d$ Values

are applicable to both substations and the distribution domains.

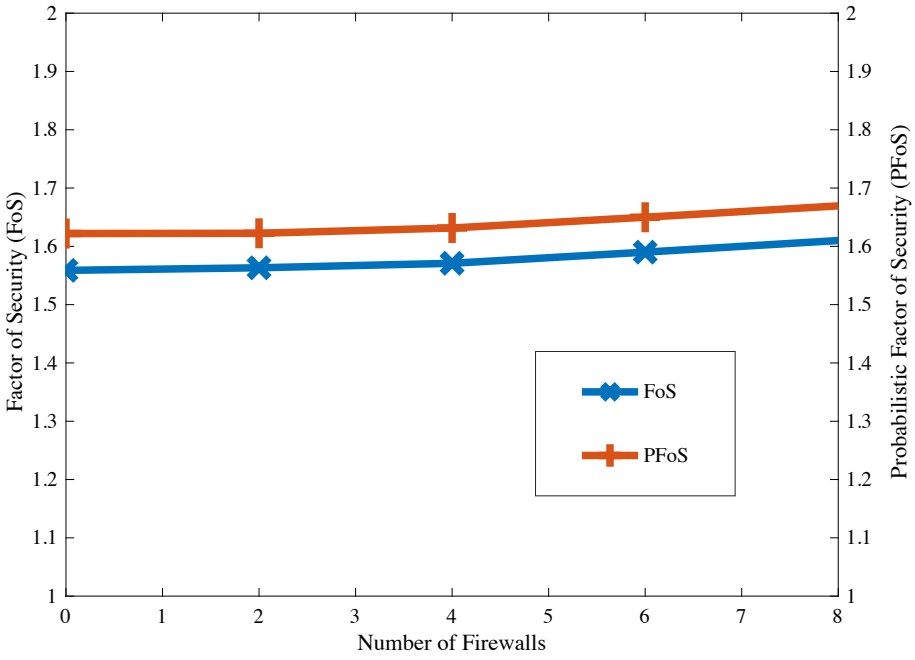
3.5.4 Security Mechanisms and Similarity

Our first two simulations aim to study the effect of an increasing number of security mechanisms, including firewalls and IDSes, on our metrics. In those experiments, we place firewalls and IDS at random locations of each subsystem.

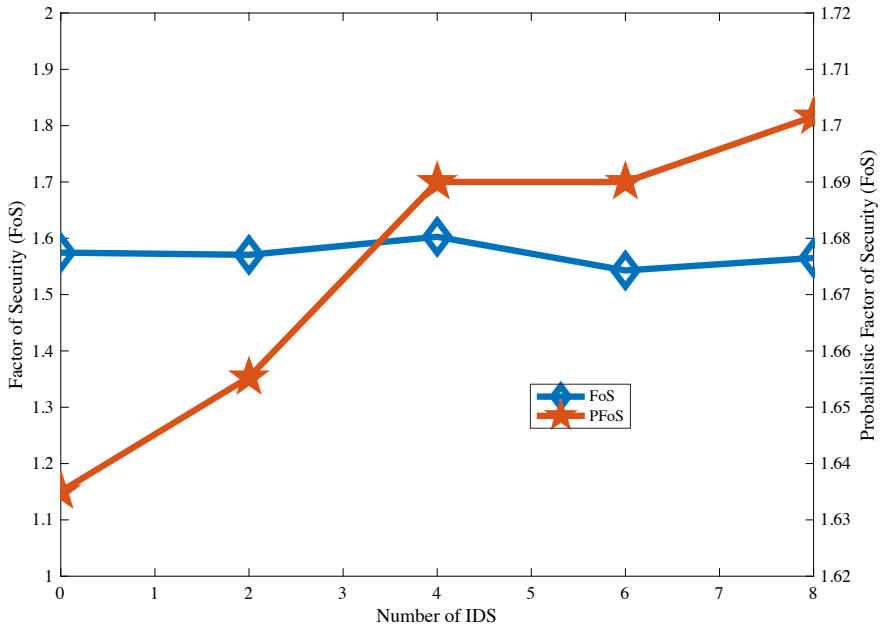
Results and Implications: In Fig. 3.15a, we can see an increase in both metrics as the number of firewalls increases, although the trend is slightly different. This is because adding firewalls can increase the shortest attack path length for the whole system more than it does for the shortest attack path of each subsystem in the case of FoS. For PFoS, adding firewalls increases the length of one or more attack paths. However, as we can see from the results, such an increase is not always significant. In Fig. 3.15b, as it can be seen, adding IDSes does not affect FoS since FoS is only based on the shortest attack path (which will not include detected attacks). On the other hand, PFoS increases as the number of IDSes increases since the probability that the attacker can reach the final goal or any of the subgoals decreases due to the increased detection capability. This difference also shows the importance of considering both FoS and PFoS in practice.

Our third and fourth simulations study the relationship between component similarity and the metrics. We consider two cases as follows. First, in the case where the similarity between components can be fully quantified or estimated, the average similarity score can be used to represent the relative level of similarity between components. Second, in the case where operators favor a more simplistic approach of simply labeling some components as "different", the percentage of similar components can be used. In our simulations, the similarity score of each component is randomly assigned following a uniform distribution between 0.0 and 1.0, and the percentage of similar components is varied from 0.2 to 0.8.

Results and Implications: In Fig. 3.16a, as it can be seen, as the average similarity score increases, FoS is not affected much since the length of the shortest path used in FoS does not consider the similarity between components. On the other hand, PFoS decreases as the average similarity score increases. This is expected since the probability of exploiting the whole system increases with a higher level of similarity between the subsystems' components. In Fig. 3.16b, it can be seen that

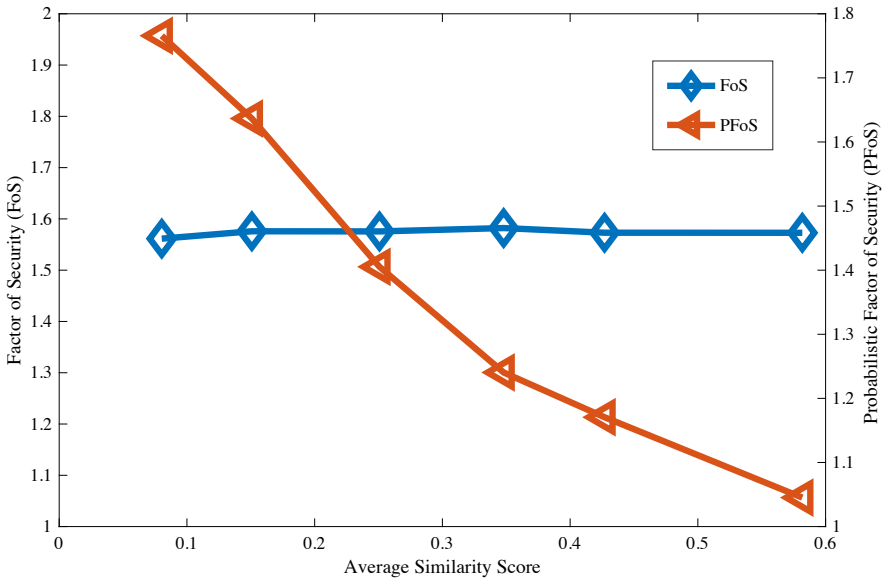


(a) The Number of Firewalls vs Metrics

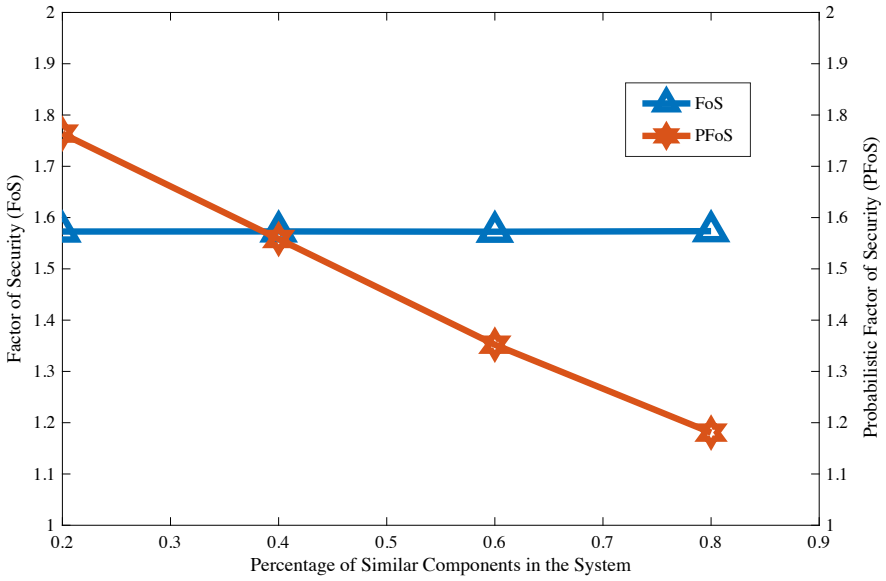


(b) The Number of IDSes vs Metrics

Figure 3.15: Metrics versus Security Mechanisms



(a) The Average Similarity Score vs. Metrics



(b) Percentage of Similar Components vs Metrics

Figure 3.16: Metrics versus Percentage and Similarity

the percentage of similar components has no effect on FoS (the reason is similar to the previous case), while it does affect PFoS with a similar trend as in the previous case, which again shows the importance of taking both metrics into account.

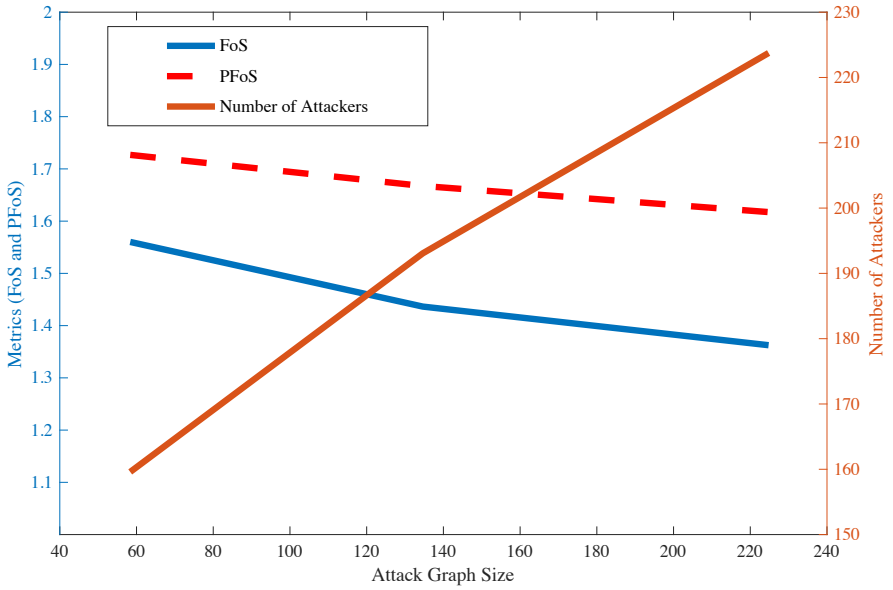
3.5.5 Other Attacks

Our first two simulations aim to analyze how metrics change with respect to attacks against cyber components or attacks against IEDs in the substation.

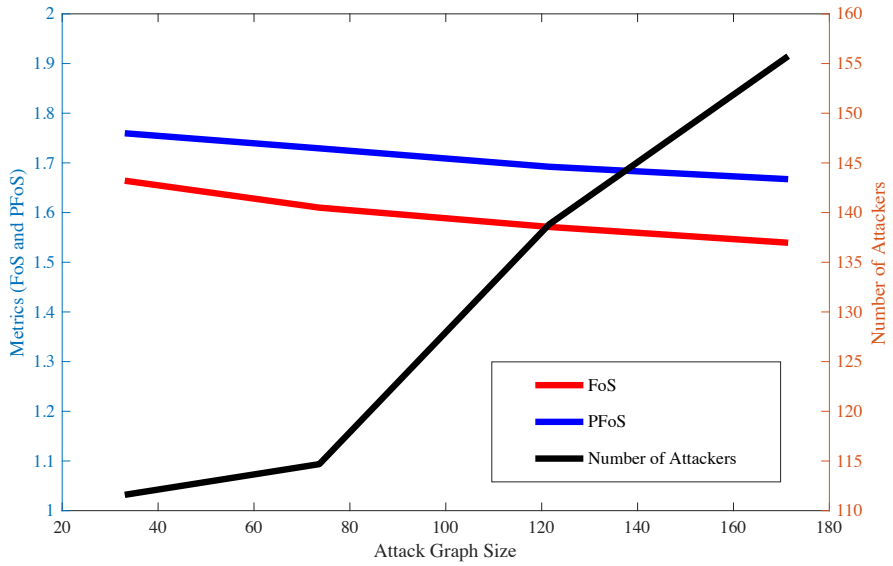
Results and Implications: In Figs. 3.17a and 3.17b, we can see that the metrics and the number of attackers who can succeed show a reverse trend. Metric values (FoS and PFoS) are generally larger in the case of attacks against IEDs since attacking IEDs requires the attacker to compromise the substation HMI or the Workstation. For both attacks against cyber components and IEDs, improving FoS and PFoS metrics enhances the security posture of the substation regarding redundancy. However, in the case of attacks against IEDs, even a small increase in FoS and PFoS causes a large decrease in the number of attackers who can compromise both subsystems compared to attacks against cyber components. This implies that diversifying IEDs, such as using different products from the same supplier for an IED as backups, significantly improves the security posture of the substation with respect to redundancy.

Our third and fourth simulations study combinations of different types of attacks in the substation and different types of attacks in the smart grid. For the case of the substation, we combined attacks against cyber components and attacks against IEDs. For the case of the smart grid, we considered four scenarios which are attacking transmission lines only, attacking communication lines only, attacking substations only, and attacking both substations and communication lines.

Results and Implications: In Fig. 3.18a, it can be seen that even when different types of attacks are combined, the metrics and number of successful attackers show a reverse trend. In Fig. 3.18b, it can be seen that the metric values for attacks involving only transmission lines are higher since most power systems are designed with N-1 reliability criteria, which means that the attacker has to compromise at least two transmission lines in order to compromise the power system. However, the same is not necessarily true for communication lines or substations.

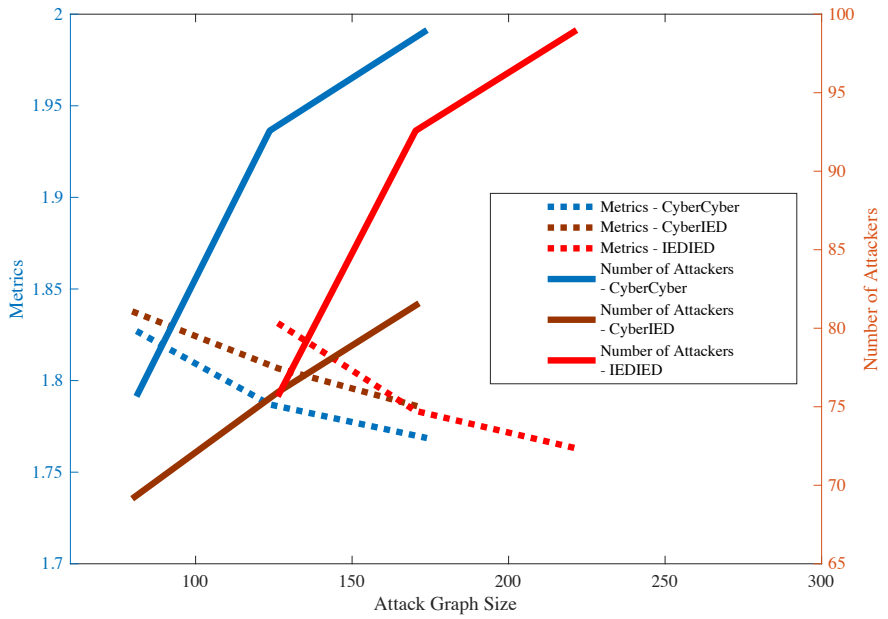


(a) Attacks Against Cyber Components

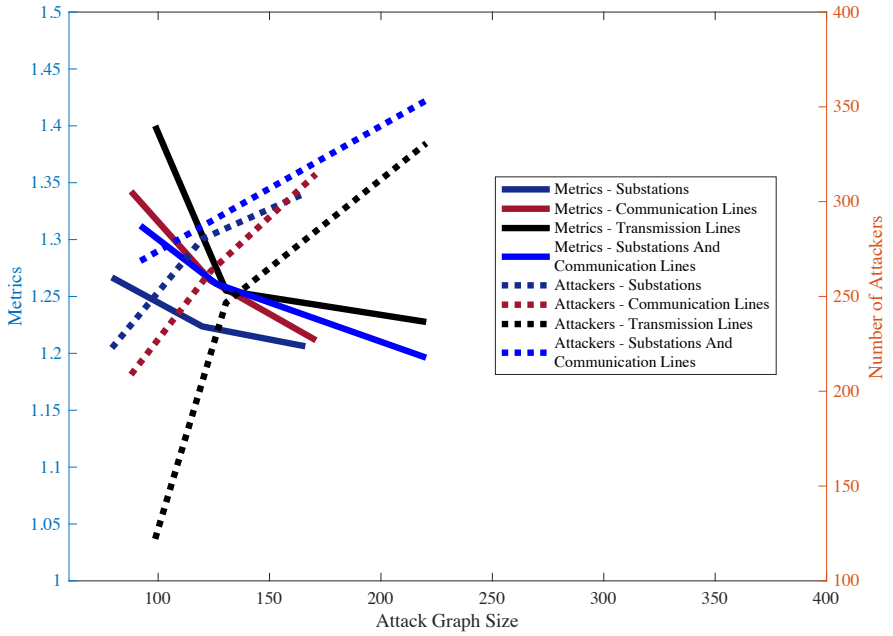


(b) Attacks Against IEDs

Figure 3.17: Metrics versus Different Types of Attacks



(a) Combining Different Types of Attacks in Substations



(b) Combining Different Types of Attacks in the Smart Grid

Figure 3.18: Metrics versus Combining Different Types of Attacks

3.6 Applying FoS and PFoS to Other Critical Systems

In order for those metrics to be utilized for other critical systems, the system in question has to have a redundant design. Redundancy in critical systems can be designed as either active redundancy or cold-standby redundancy [154]. Metrics defined in this chapter are applicable to any critical systems using cold-standby redundancy which implies that only one component is active at a time. In order to apply this work to other critical systems, attack graph models for each subsystem and for the whole system need to be generated according to a different configuration than substations.

3.7 Summary

In this chapter, we have proposed two novel security metrics, namely, the factor of security (FoS) and the probabilistic factor of security (PFoS), to evaluate the security effectiveness of redundant subsystems in smart grids and substations. Specifically, we have provided a concrete design of a substation based on IEC 61850 and standard industry practices. We have adopted the existing factor of safety metric to the context of smart grid security based on the attack graph model in order to define our FoS and PFoS metrics. We have applied the metrics to several attack scenarios in both substations and the smart grid distribution domain. The considered attack scenarios have clearly illustrated that our metrics could allow us to obtain a quantitative understanding of how effective redundant designs are with respect to security resilience. We have performed simulations to study how the metrics would behave in different scenarios, and our results demonstrate how the metrics might be helpful in answering useful questions about security prioritization and planning.

Chapter 4

Measuring the Security Postures of Substations Against Supply Chain Attacks

4.1 Introduction

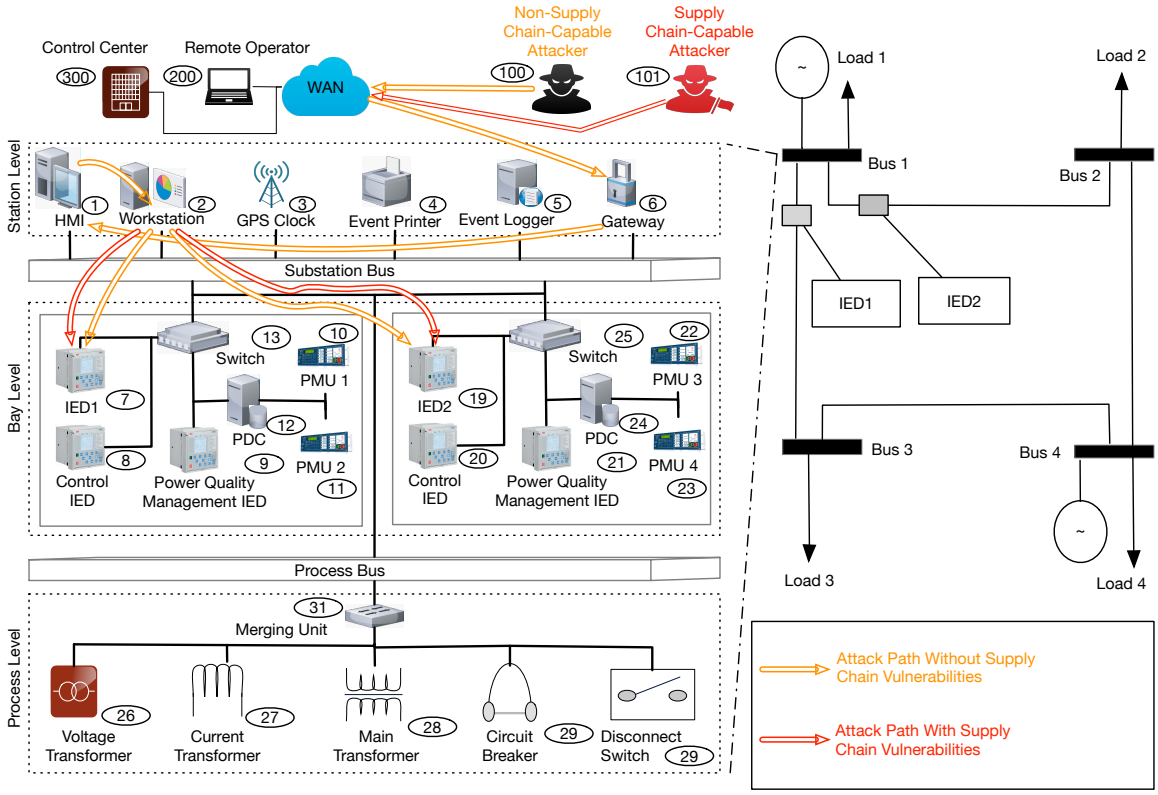
The proprietary nature of smart grid devices and the limited number of vendors entail that a power utility has little choice but to trust all the vendors, even if such a trust is not completely verified. To make things worse, even a trustworthy vendor may become the target of the so-called supply chain hijacking attacks [155]. In those attacks, the vendor's network is infiltrated with malicious codes injected into the developer's copy of firmware or its update, which is then downloaded to hundreds or thousands of devices. Such a threat of supply chain attacks has attracted only limited attention in the smart grid domain. Despite existing efforts in analyzing and detecting supply chain attacks, the development of a metric to quantify the risk of those threats is mostly overlooked in the literature. Significant research gaps remain, such as the lack of means for instantiating the proposed security metrics based on various supply chain-related risk factors and the lack of a model for capturing the potential physical consequences of those attacks.

In this chapter, first, we develop a supply chain risk metric (SCR) to capture several risk factors that jointly determine the likelihood of a vendor containing supply chain vulnerabilities. Second, building upon the proposed SCR, we define two novel security metrics: $kSupply$ and $kSupplier$. Those metrics capture the overall security posture of IEC 61850 substations against supply chain attacks. The $kSupply$ metric measures the security posture regarding the minimum number of supply chain vulnerabilities. Moreover, the $kSupplier$ metric measures the minimum number of devices from malicious vendors that must be compromised to compromise substations. Intuitively, higher values of those metrics would indicate a less vulnerable substation because it is significantly less likely to have many supply chain vulnerabilities or malicious vendors simultaneously and in the same substation. Finally, to capture the potential physical consequences of supply chain attacks, we develop a new model by combining attack graphs with the MDP to differentiate between the shortest attack path and the most critical attack path with respect to potential physical damage. Those models are developed using a realistic IEC 61850 substation (details of the substation can be found in Chapter 3).

In the following, we present a motivating example to emphasize the need for security metrics for IEC 61850 substations against supply chain attacks.

4.1.1 Motivating Example: Tripping Circuit Breakers Remotely

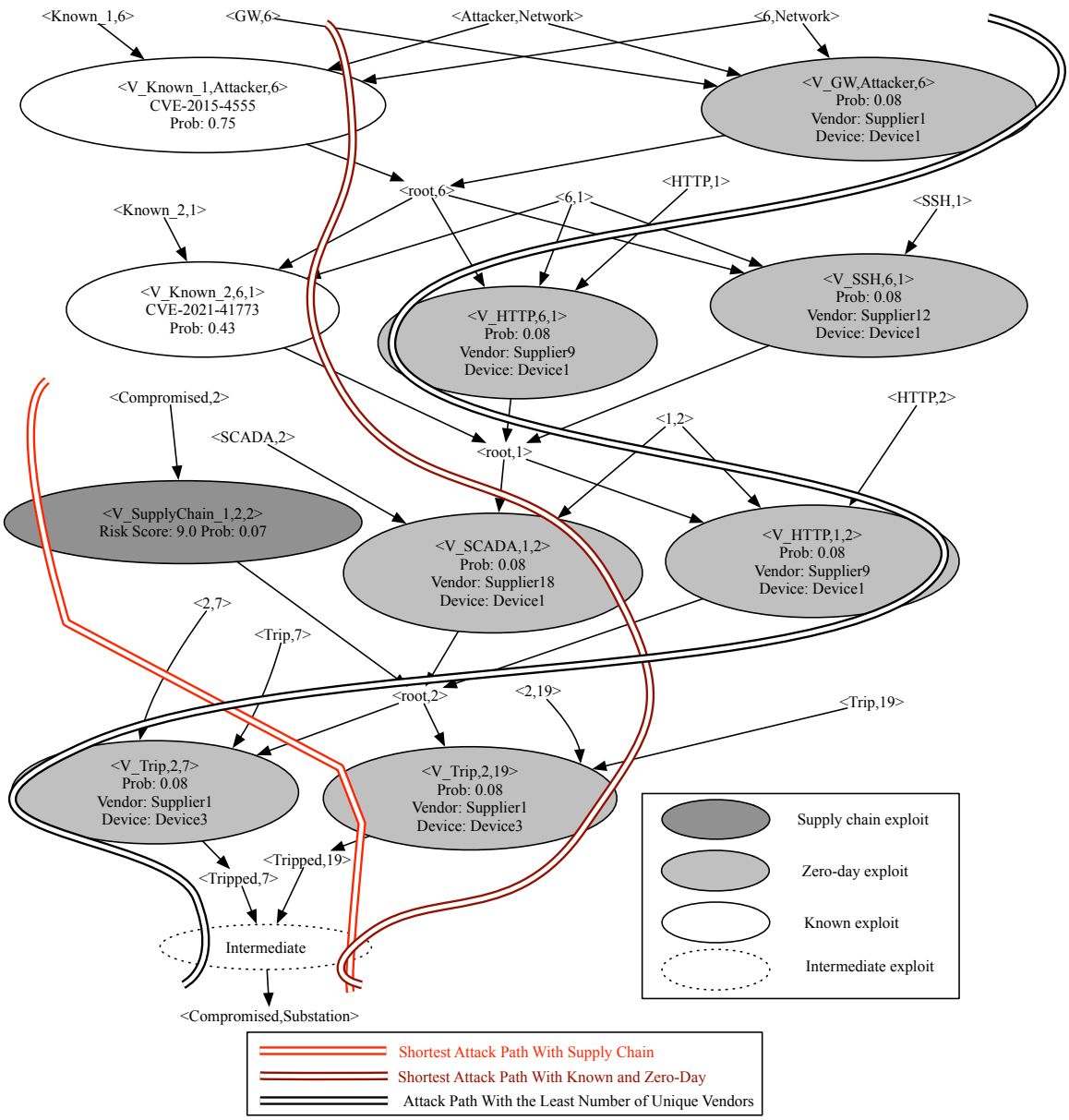
Consider a substation configuration in Fig. 4.1 on the left. Assume that this substation configuration corresponds to Bus 1 of IEEE 4-bus [156] shown in Fig. 4.1 on the right. Also, assume that a cyber adversary who is not capable of exploiting supply chain vulnerabilities (Node 100 in Fig. 4.1) has remote access to the substation network. In order to compromise the substation and start remotely tripping circuit breakers, the adversary needs to follow the attack path, which consists of (in the given order) compromising the gateway (Node 6), the HMI (Node 1), the Workstation (Node 2), and protection IEDs (Nodes 7 and 19). However, an attacker who is capable of injecting supply chain vulnerabilities (Node 101) can inject those into the Workstation (Node 2) by compromising its update channel and then plant a backdoor [157] that allows the attacker to compromise protection IEDs (Nodes 7 and 19) without compromising the gateway (Node 6) and HMI (Node 1). This example shows that supply chain vulnerabilities can make the compromise easier for attackers.



Smart Grid Substation Based on IEC 61850-90-4 (Left) and IEEE 4-Bus System [156] (Right)

Figure 4.1: Motivating Example

As shown in the attack graph in Fig. 4.2, if the attacker is capable of exploiting known and zero-day vulnerabilities, the attacker can follow the attack path: $\langle V\_Known\_1, Attacker, 6\rangle \rightarrow \langle V\_Known\_2, 6, 1\rangle \rightarrow \langle V\_SCADA, 1, 2\rangle \rightarrow \langle V\_Trip, 2, 19\rangle$ (shown using orange double lines in Fig. 4.2). However, a supply chain capable attacker can follow a shorter attack path: $\langle V\_Supply\ Chain\_1, 2, 2\rangle \rightarrow \langle V\_Trip, 2, 19\rangle$ (shown using red double lines in Fig. 4.2). Lastly, the $Interme\ diate$ node is utilized to illustrate the fact that the attacker can compromise either IED1 (Node 7 in Fig. 4.1) or IED2 (Node 19 in Fig. 4.1) or both. The attack graph model motivates us to ask the following questions: *How can we model the likelihood that each device contains a supply chain vulnerability? How can we measure the overall security posture of this substation against potential supply chain attacks? How can we capture the physical consequences of such supply chain attacks, e.g., tripping circuit breakers?* The remainder of this chapter will answer those research questions by developing several security metrics considering supply chain attacks.



A) Tripping Circuit Breakers Using Zero-Day and Supply Chain (Red), B) Known and Zero-Day Only (Orange), C) Containing the Minimum Number of Unique Vendors (Black)

Figure 4.2: Attack Graph Showing Attack Paths

4.2 Supply Chain Attacks

In this section, we first provide background information on supply chain attacks and then discuss their characterization.

4.2.1 Background

Supply chain attacks refer to the exploit of compromised hardware or software in which the vulnerabilities are already present before the hardware/software is deployed and configured [158, 93] or during shipment to the customer [94]. As an example, software update channels of Asus have been compromised by hackers who use supply chain attacks as their main attack vector[1]. Since those attacks are on the rise [1], securing smart grid substations against such attacks is receiving more attention. In order to enhance the security of smart grids against supply chain attacks, four key areas are identified in the U.S. Resilience Project [159], namely, (i) organizational best practices, (ii) supply chain transparency and trust, (iii) supply chain security, and (iv) supply chain integrity. First, organizational best practices involve risk assessment based on the potential impacts of supply chain vulnerabilities for each product. Second, supply chain transparency and trust concerns checking the certifications and trustworthiness of vendors. Third, supply chain security involves tamper detection techniques for hardware and authenticating the assembled parts. Finally, supply chain integrity involves using cryptographic signing for components and authentication of assembled parts. However, despite the existence of such guidelines, supply chain vulnerabilities are still being used as an attack vector. As an example, updates to M.E. Doc financial software in Ukraine contained the NotPetya ransomware [160]. As another supply chain security incident, in the SolarWinds supply chain attack, the attackers compromised the trusted vendor's distribution channel [155]. More specifically, SolarWinds is a company that primarily deals with Network Management Systems (NMS). Orion is a network management software provided by the company. A vulnerability was inserted into updates of the Orion software, which allowed attackers to take control of networks of organizations. These security incidents demonstrate that securing critical infrastructures, such as

[1] A mysterious hacker gang is on a supply-chain hacking spree https://arstechnica.com/information-technology/2019/05/a-mysterious-hacker-gang-is-on-a-supply-chain-hacking-spree/. [Online; Accessed 9-January-2021].

substations, against supply chain attacks is important.

4.2.2 Characteristics

Supply chain vulnerabilities differ from known vulnerabilities in the sense that their existence is not known. They also differ from zero-day [66] vulnerabilities in the sense that they are due to deliberate actions taken by malicious or hijacked vendors rather than developers' unintentional mistakes. In this subsection, we will explain how supply chain vulnerabilities differ from zero-day vulnerabilities, and then we will list unique aspects of supply chain vulnerabilities.

Comparison With Zero-Day Vulnerabilities

The concepts of supply chain and zero-day vulnerabilities are similar since they are not known until the day they are exploited. However, those vulnerabilities are not equivalent. A zero-day vulnerability [66] means the vulnerability remains unknown until the day it is exploited (sometimes a vulnerability without any patch or fix is also called zero-day); since most supply chain vulnerabilities also remain unknown until they are exploited, they are similar to zero-day vulnerabilities. On the other hand, the concept of supply chain vulnerability focuses more on the fact that the vulnerability has been purposefully injected by the (malicious) vendor or by the attacker who has hijacked the vendor's distribution channel. On the contrary, most other (either zero-day or known) vulnerabilities are accidentally brought into systems due to developers' lack of security awareness or simply by mistakes.

Unique Characteristics of Supply Chain Vulnerabilities

The deliberate nature of supply chain vulnerabilities means the attacks can be performed more stealthy and automated, with more severe consequences [161]. Unique characteristics of those vulnerabilities include [12]:

- The (hijacked) vendor may continuously update the vulnerability to ensure it remains stealthy and can evade existing detection and prevention solutions.

- The vulnerability may either take the form of a hidden backdoor that allows external attackers to infiltrate and control the substation or act as malware to spread and activate itself autonomously without the external intervention of attackers.
- The attack by a hijacked vendor may involve many devices across multiple substations to launch a coordinated attack and cause large-scale cyber-physical damages.
- Different devices from the same (hijacked) vendor may be designed to help each other hide their misbehaviors, making detection more difficult.
- The attack may be designed to steal sensitive information, e.g., cryptographic keys or configuration information, and leak it out through existing (covert) channels.
- The attack may be specifically designed to cause long-term damages, e.g., an IED can be shipped with a supply chain vulnerability, which prevents it from reporting overcurrent on the transformer.
- The vulnerability can be harder to remove since the patch may require physical modifications, e.g., changing chipsets.

We have investigated a list of major smart grid vendors producing devices for substations and studied the number of devices they produce. Table 4.1 shows the approximate number of devices from each vendor in each category inside a substation and the relative market shares of each vendor according to their revenues obtained from[2]. Based on Table 4.1, we can conclude that as many as 14 devices may be affected due to one compromised vendor on average.

4.3 Modeling Supply Chain Attacks

In this section, we present concrete models of supply chain attacks under different scenarios in IEC 61850 substations. To this aim, we consider a smart grid substation configuration based on IEC 61850-90-4, as shown in Fig. 4.1 on the left. We will model different supply chain attack

[2] Largest Companies by Market Cap https://companiesmarketcap.com/. [Online; Accessed 9-January-2021]

Table 4.1: Devices and Suppliers in IEC 61850 Substations With Their Relative Market Shares According to Revenues (Names Are Hidden).

Suppliers	Device Types									Vendor Information	
	Gateway	Human Machine Interface (HMI)	Time Synchronization (GPS)	Protection IED	Bay Control	Remote Terminal Unit (RTU)	Merging Unit (MU)	Phasor Measurement Unit (PMU)	Phasor Data Concentrator (PDC)	Total Number of Devices	Vendor Relative Market Share (16 June 2023) [2]
Supplier1	2	3	0	3	2	3	1	1	1	16	16%
Supplier2	1	2	2	5	5	1	1	1	0	18	18%
Supplier3	2	1	2	3	3	0	1	2	2	16	N/A
Supplier4	3	3	5	1	2	1	2	1	1	19	37%
Supplier5	1	1	0	1	1	1	0	1	0	6	9%
Supplier6	1	1	0	1	1	5	0	0	0	9	20%
Average	1.67	1.83	1.50	2.33	2.33	1.83	0.83	1.00	0.67	14.00	

scenarios by extending the widely used attack graph model [15, 16]. In the extended model, grey nodes represent supply chain vulnerabilities, light gray nodes represent zero-day vulnerabilities, and white nodes represent known vulnerabilities (probabilities, vendors, and devices can be ignored for now and will be explained later in Section 4.4.1).

4.3.1 Attack Scenarios

PTP Time Delay Attack

Precision Time Protocol (PTP) is a network time synchronization protocol that is used to synchronize clocks in IEDs of IEC 61850 substations. In the PTP time delay attack [128], the attacker aims to delay PTP messages to deteriorate the time synchronization between devices. In the PTP protocol, there is a master clock and several slave clocks that obtain timing information from the master clock. In Fig. 4.1, node 3 is the master clock. The attacker can inject supply chain vulnerabilities into the master clock through firmware updates [93], similar to the SolarWinds case [155]. This malicious update to the GPS clock can cause a lack of time synchronization, which can lead to the malfunctioning of several processes and applications, e.g., voltage stability monitoring, which depends on timings in Phasor Measurement Units (PMU) [128]. Since the PTP time delay attack can be detected using a guard clock [128], we added detection nodes in the attack graph model. Each component can contain known, zero-day, or supply chain vulnerabilities. Fig. 4.3 shows the attack graph model for the PTP time delay attack.

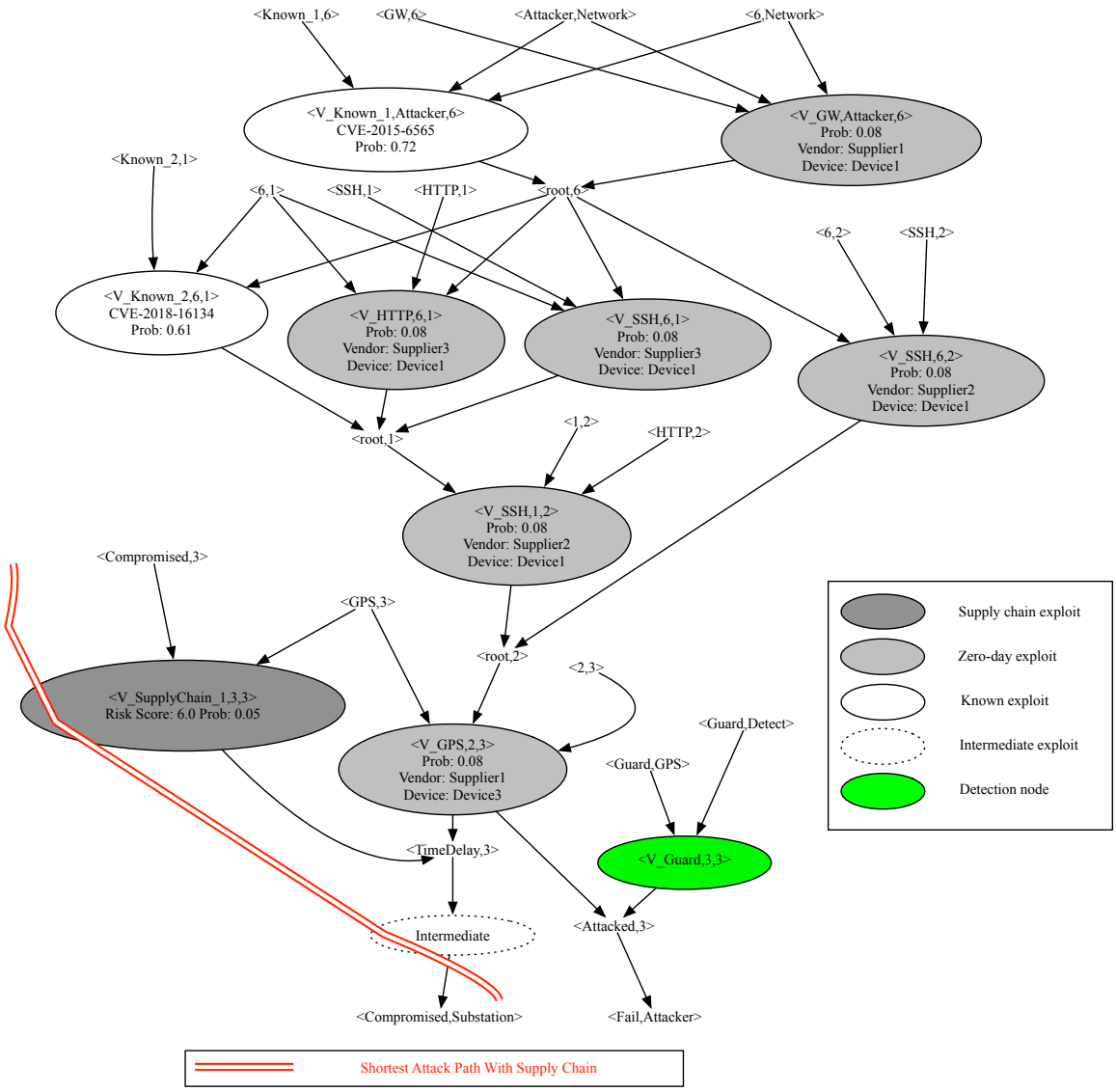


Figure 4.3: Attack Graph for PTP Time Delay Attack

Backdoors in Protection IEDs

In an IEC 61850 substation, typical IEDs include digital relays, fault recorders, controllers for circuit breakers, capacitor banks, and tap changers [126]. As one type of IED, a protection IED (node 7 or 19 in Fig. 4.1) is responsible for tripping circuit breakers due to a physical fault. The protection IED can be shipped with a backdoor which allows attackers to trip circuit breakers remotely. In this scenario, the attacker is assumed to be behind that supply chain exploit, so the

probability of exploiting the supply chain vulnerability is 1.0. In order to perform this attack, the adversary first needs to infiltrate the substation network by compromising the substation gateway (node 6 in Fig. 4.1). After compromising the substation gateway, the attacker can compromise the substation HMI (node 1 in Fig. 4.1) and send fake crafted trip commands to trip circuit breakers maliciously by taking advantage of the backdoor, which is $\langle V\_SupplyChain\_1, 7, 7\rangle$, in the protection IED. The attack graph for tripping circuit breakers using the backdoor is shown in Fig 4.4.

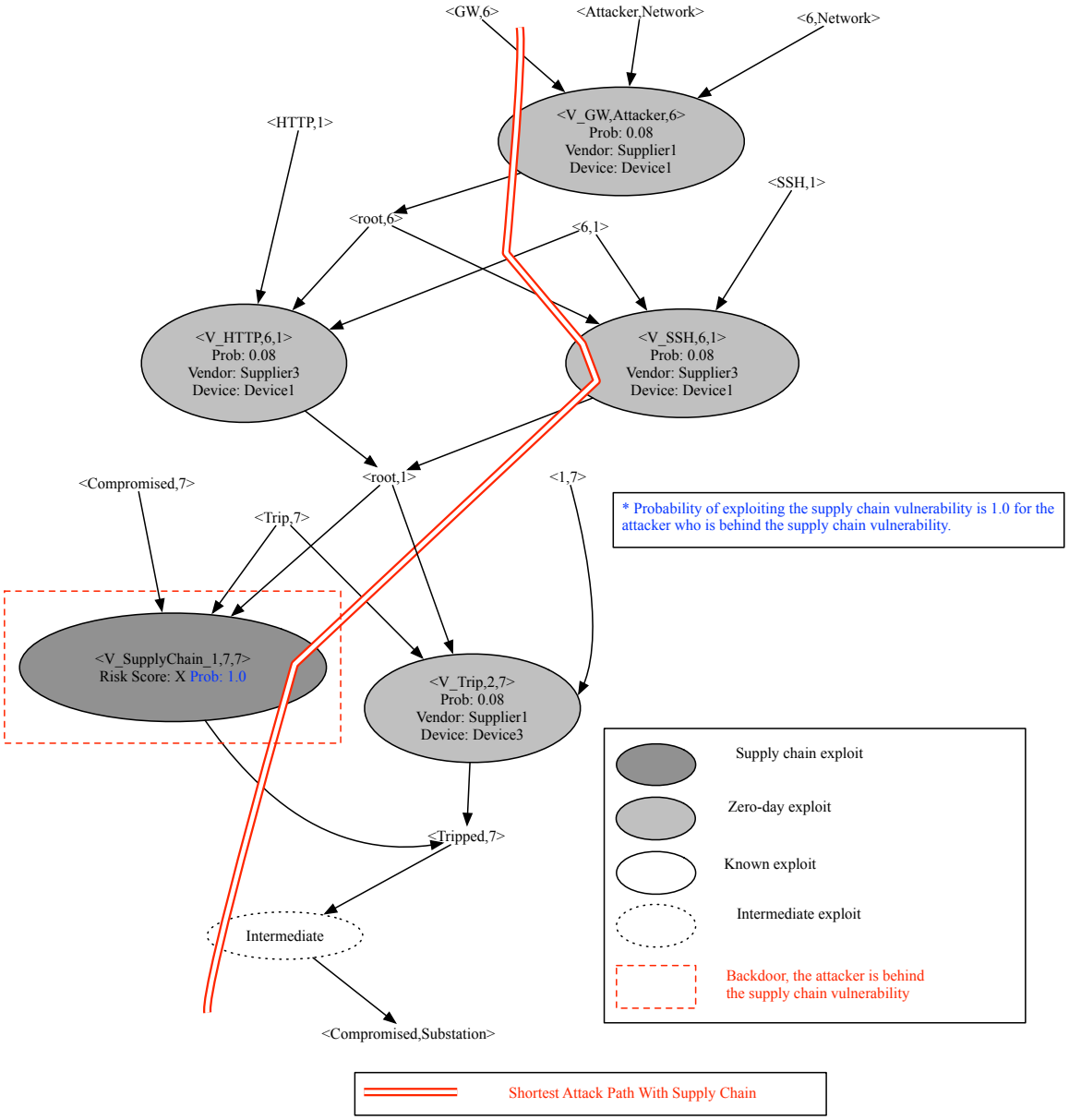


Figure 4.4: Backdoor in the Protection IED

Information Disclosure Using Compromised Devices

IEDs may contain critical information such as cryptographic keys [126] or substation configuration language (SCL) files [162]. IEDs can be accessed remotely using communication channels such as Secure Shell (SSH) or Telnet [163]. An IED containing a supply chain vulnerability can send critical information to attackers via such communication channels or covert channels. Fig. 4.5 shows an attack graph that models the attack scenario in which an IED sends configuration files to the attacker through Telnet.

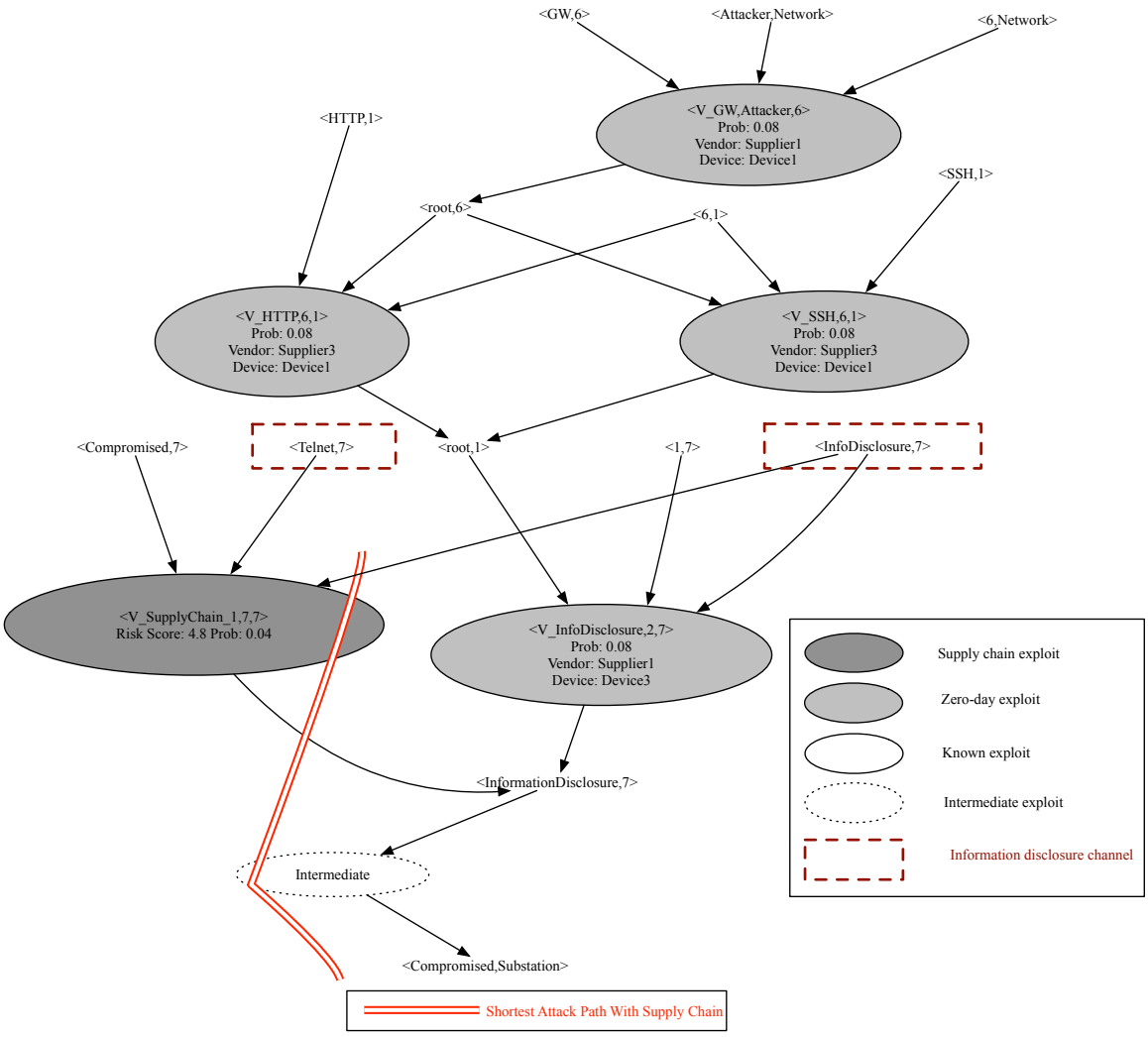


Figure 4.5: Information Disclosure in IEDs

Coordinated Attack

In IEC 61850 substations, merging units (node 31 in Fig. 4.1) are responsible for getting measurements from physical devices. Protection IEDs (node 7 in Fig. 4.1) are responsible for tripping circuit breakers to avoid physical faults in their protection zones. In this attack scenario, the operator decides to replace merging units and protection IEDs with ones from a new vendor. Those devices from the new vendor contain supply chain vulnerabilities making it possible for them to work together to trip circuit breakers maliciously. In this attack scenario, both devices must be compromised for the attack to succeed. The dummy exploit node ($Dummy$ in Fig. 4.6) means that the attacker needs to compromise both the IED and the merging unit to cause malicious tripping. Both devices operate in normal conditions, but their aggregated effect will result in tripping circuit breakers. Coordination between different devices may make detection even more difficult. This scenario shows that using supply chain vulnerabilities, two seemingly ordinary devices from one vendor may become malicious when they coordinate. The attack graph for the coordinated attack scenario is shown in Fig. 4.6.

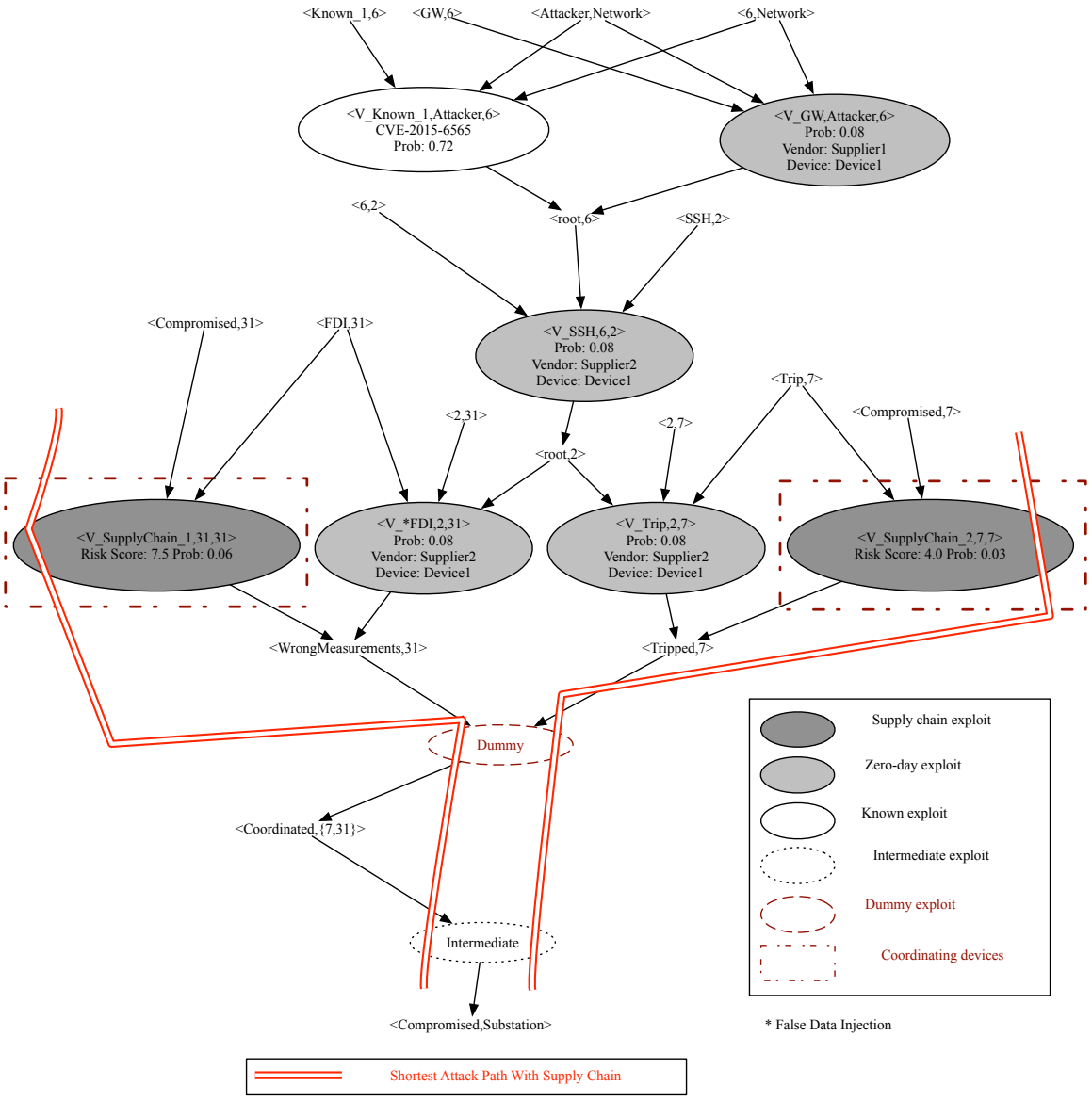


Figure 4.6: Coordinated Attack Involving Merging Units and IEDs

4.4 Supply Chain Security Metrics

In order to measure the security posture of devices against attacks involving supply chain vulnerabilities, we first define several security metrics, which are SCR, $kSupply$, and $kSupplier$. After that, we utilize attack graphs with MDP in order to identify critical substations in large power systems.

4.4.1 Supply Chain Risk Score (SCR)

Background on Supply Chain Risk Score

A substation network may include devices from different vendors. Although it is typically assumed that all those vendors are trustworthy, this may not always be the case. In order to measure the severity of supply chain vulnerabilities and consider the trust levels of vendors, we define the supply chain risk score (SCR) inspired by the CVSS [164]. SCR is defined similarly to CVSS to take lessons from CVSS. For instance, CVSS has an important feature, which is hierarchical scoring. Hierarchical scoring in CVSS means that the base score is first calculated; then, temporal and environmental scores are calculated if (and only if) the necessary information is available. Hierarchical scoring is also applicable in the case of supply chain vulnerabilities since there are risk factors [6] that can be quantified easily, such as the existence of vendors in the Open Trusted Technology Forum [165]. On the contrary, some risk factors [6] may not be quantified easily, such as secure transportation of devices. In addition, risk factors such as outsourcing may be kept confidential. Therefore, we also define the SCR hierarchically such that all the available risk factors can be identified and combined into a metric.

In order to define SCR, we first identify relevant questions in part 4 of NIST Cybersecurity Supply Chain Risk Management for Systems and Organizations [166], to be answered by the power system operator, for risk factors [6]. After that, we map each risk factor to a metric with the same number of possible values in CVSS [164]. Such a mapping allows us to leverage the scoring methodology and aggregation formula of CVSS. CVSS score measures severities of known vulnerabilities based on three metrics which are base, temporal, and environmental [164]. Base metrics define characteristics of vulnerabilities that do not change over time. Temporal metrics are based on factors that change over time. Environmental metrics adjust the base and temporal metrics to a specific computing environment. Risk factors, along with their categories are described in the next section.

Risk Factors for Supply Chain Vulnerabilities

Origin of the Product (O): The country in which the device is produced may be trusted or distrusted. Since the manufacturer's origin is assumed to be known in most cases, this risk factor is

considered in the base score. Questions from the NIST SP-800 [166] related to this risk factor are:

- Table D-13, Section 2, Question 3 from [166]: "Is the product/service manufactured in a geographic location that is considered an area of geopolitical risk for your enterprise based on its primary area of operation (e.g., in the United States)?"
- Table D-13, Section 2, Question 4 from [166]: "Is the product manufactured or developed in a country identified as a foreign adversary or country of special concern?"
- Table D-13, Section 3, Question 7 from [166]: "Do the laws and regulations of any foreign country in which the supplier has headquarters, research development, manufacturing, testing, packaging, distribution, or service facilities or other operations require the sharing of technology or data with that foreign country?"

This risk factor is mapped to the "Attack Vector" base metric in CVSS, which also has 4 possible values: NETWORK (0.85), ADJACENT (0.62), LOCAL (0.55), PHYSICAL (0.20). Based on this, the risk factor **O** for a given device (De) is:

$$O(De) = \begin{cases} \text{HIGH } (0.85), & \text{for 3 YES answers} \\ \text{MEDIUM HIGH } (0.62), & \text{for 2 YES answers} \\ \text{MEDIUM LOW } (0.55), & \text{for 1 YES answer} \\ \text{LOW } (0.20), & \text{for 0 YES answers} \end{cases}$$

Producer of the Product (P): This risk factor depends on the producer of the device. This risk factor is included in the base score, assuming the vendor is known. Questions from the NIST SP-800 [166] related to this risk factor are:

- Table D-13, Section 3, Question 11 from [166]: "Does the supplier have documents that track part numbers to manufacturers?"
- Table D-13, Section 3, Question 18 from [166]: "Does the supplier have procedures for secure maintenance and upgrades following deployment?"

This risk factor is mapped to the "Privileges Required" base metric in CVSS, which has the following 3 possible values: NONE (0.85), LOW (0.62), HIGH (0.27). Based on this, the risk factor **P** for a given device De is:

$$P(De) = \begin{cases} \text{HIGH } (0.85), & \text{for 0 YES answers} \\ \text{MEDIUM } (0.62), & \text{for 1 YES answer} \\ \text{LOW } (0.20), & \text{for 2 YES answers} \end{cases}$$

Inclusion of the Vendor in Open Group Trusted Technology Forum (OTT): The Open Group Trusted Technology Forum (OTTF) [165] develops a global supply chain integrity procedure that provides buyers with a choice of certified partners and vendors. Given the fact that the involvement of the producer in OTTF can be verified from the official website [165], this risk factor is included in the base score. This risk factor can take two possible values: AVAILABLE and NOT AVAILABLE. Therefore, we adopt the scoring of "Attack Complexity" base metric in CVSS [164], which has 2 possible values: LOW (0.77) and HIGH (0.44). Based on this, the risk factor **OTT** for a given device De is:

$$OTT(De) = \begin{cases} \text{AVAILABLE } (0.44), & \text{when the vendor is included in OTTF} \\ \text{NOT AVAILABLE } (0.77), & \text{when the vendor is not included in OTTF} \end{cases}$$

Transparency of Supply Chain Best Practices (BP): This risk factor measures the level of transparency of the vendor in terms of applying supply chain best practices. Since such transparency is assumed to be known in most cases, this risk factor is included in the base score. Questions from the NIST SP-800 [166] related to this risk factor are:

- Table D-13, Section 4, Question 7 from [166]: "Does the supplier use, record, and track risk mitigation activities throughout the life cycle of the product, system, or service?"
- Table D-13, Section 3, Question 10 from [166]: "Does the supplier employ the use of threat scenarios to inform the vetting of sub-tier suppliers?"

- Table D-13, Section 3, Question 16 from [166]: "Does the supplier use industry-standard baselines (e.g., CIS, NES) when purchasing software?"

Similar to the risk factor **O**, this risk factor is mapped to "Attack Vector" in CVSS since they both have 4 possible values. Based on this, the risk factor **BP** for a given device De is:

$$BP(De) = \begin{cases} \text{HIGH } (0.85), & \text{for 0 YES answers} \\ \text{MEDIUM HIGH } (0.62), & \text{for 1 YES answer} \\ \text{MEDIUM LOW } (0.55), & \text{for 2 YES answers} \\ \text{LOW } (0.20), & \text{for 3 YES answers} \end{cases}$$

Usage of Modules From Other Vendors (UV): This risk factor aims to measure the risk related to parts from other vendors used by the vendor. This risk factor is placed in the temporal score since a vendor can utilize parts from different vendors at different times. Questions from the NIST SP-800 [166] related to this risk factor are:

- Table D-13, Section 3, Question 1 from [166]: "Have you identified the supplier's critical suppliers?"
- Table D-13, Section 3, Question 8 from [166]: "Has the supplier declared where replacement components will be purchased from?"
- Table D-13, Section 3, Question 12 from [166]: "Can the supplier provide a list of whom they procure hardware and software from that is utilized in the performance of the contract?"

This risk factor is mapped to "Exploit Code Maturity" in CVSS [164], which is listed as a temporal metric having 4 possible unique values (NOT DEFINED and HIGH have the same value 1.0): HIGH (1.00), FUNCTIONAL (0.97), PROOF OF CONCEPT (0.94), UNPROVEN (0.91). Even though there are other temporal CVSS metrics with 4 possible unique values, Exploit Code Maturity was chosen for mapping since it has the highest range of numeric values (from 0.91 to 1.0). Based on this, the risk factor **UV** for a given device De:

$$UV(De) = \begin{cases} \text{HIGH } (1.00), & \text{for 0 YES answers} \\ \text{MEDIUM HIGH } (0.97), & \text{for 1 YES answer} \\ \text{MEDIUM LOW } (0.94), & \text{for 2 YES answers} \\ \text{LOW } (0.91), & \text{for 3 YES answers} \end{cases}$$

Outsourcing During Production (OU): Outsourcing is the usage of resources of another company during product development. Outsourcing should be done in a way that does not create supply chain security issues. Outsourcing is considered in temporal score since companies to which the vendor outsources may change with time. Quantifying the level at which outsourcing is done in a trusted manner is challenging. The quantification of outsourcing shows how much the end user trusts the vendor in the way the vendor performs outsourcing. Questions from the NIST SP-800 [166] related to this risk factor are:

- Table D-13, Section 3, Question 9 from [166]: "Have the owners and locations of all of the suppliers, subcontractors, and sub-tier suppliers been identified and validated?"
- Table D-13, Section 4, Question 1 from [166]: "Does the supplier have definitive policies and procedures that help minimize supply chain risk, including subsidiary sourcing needs?"

This risk factor is mapped to "Report Confidence" in CVSS [164], which is listed as a temporal metric having 3 possible unique values (NOT DEFINED and CONFIRMED have the same value 1.0): CONFIRMED (1.00), REASONABLE (0.96), and UNKNOWN (0.92). Based on this, the risk factor **OU** for a given device De is:

$$OU(De) = \begin{cases} \text{HIGH } (1.00), & \text{for 0 YES answers} \\ \text{MEDIUM } (0.97), & \text{for 1 YES answer} \\ \text{LOW } (0.91), & \text{for 2 YES answers} \end{cases}$$

Secure Transportation of the Product (ST): Secure transportation means that the product is not tampered with during transportation. Even if the vendor follows supply chain best practices,

supply chain vulnerabilities may be inserted into products during transportation. Secure transportation is considered a temporal risk metric since its value can change with time. Questions from the NIST SP-800 [166] related to this risk factor are:

- Table D-13, Section 5, Question 2 from [166]: "Does the supplier analyze events (environmental or human-made) that could interrupt their supply chain?"
- Table D-13, Section 5, Question 6 from [166]: "Is the supplier's inventory periodically inspected for exposure or tampering?"
- Table D-13, Section 5, Question 8 from [166]: "Is there a documented chain of custody for the deployment of products and systems?"

Similar to the risk factor **UV**, this risk factor is also mapped to "Exploit Code Maturity" in CVSS [164]. Based on this, the risk factor **ST** for a given device De is:

$$ST(De) = \begin{cases} \text{HIGH } (1.00), & \text{for 0 YES answers} \\ \text{MEDIUM HIGH } (0.97), & \text{for 1 YES answer} \\ \text{MEDIUM LOW } (0.94), & \text{for 2 YES answers} \\ \text{LOW } (0.91), & \text{for 3 YES answers} \end{cases}$$

Table 4.2 provides a summary of risk factors.

Defining the Supply Chain Risk Score (SCR)

After defining risk factors, the next step is to combine them into the base and temporal scores. We do this in a similar way as the exploitability base score of CVSS [164], which is defined as:

$$\text{E} = 8.22 \times AV \times AC \times PR \times UI \tag{7}$$

where E is *Exploitability*, AV is *Attack Vector*, AC is *Attack Complexity*, PR is *Privileges Required*, UI is *User Interaction*. Here the constant 8.22 is used to ensure that the base metric values are in the proper range (from 0.0 to 10.0) in CVSS [164]. Similarly, in the developed SCR, we use

Table 4.2: Supply Chain Risk Factors [6] with Their Categories, Possible Values, and Relevant Questions

Risk Factor	**Category**	**Possible Values**	**Relevant Questions from Table D-13 in [166]**
Origin of the Product (O)	BASE SCORE	HIGH: 0.85, MEDIUM HIGH: 0.62, MEDIUM LOW: 0.55, LOW: 0.20	- Sect. 2, Qn. 3 - Sect. 2, Qn. 4 - Sect. 3, Qn. 7
Producer of the Product (P)		HIGH: 0.85,MEDIUM: 0.62, LOW: 0.27	- Sect. 3, Qn. 11 - Sect. 3, Qn. 18
Inclusion of the Product in Open Group Trusted Technology Forum (OTT)		AVAILABLE: 0.44, NOT AVAILABLE: 0.77	N/A
Transparency of Supply Chain Best Practices (BP)		HIGH: 0.85, MEDIUM HIGH: 0.62, MEDIUM LOW: 0.55, LOW: 0.20	- Sect. 4, Qn. 7 - Sect. 3, Qn. 10 - Sect. 3, Qn. 16
Usage of Modules From Other Vendors (UV)	TEMPORAL SCORE	HIGH: 1.0, MEDIUM HIGH: 0.97, MEDIUM LOW: 0.94, LOW: 0.91	- Section 3, Qn. 1 - Section 3, Qn. 8 - Section 3, Qn. 12
Outsourcing During Production (OU)		HIGH: 1.0, MEDIUM : 0.97, LOW: 0.91	- Section 3, Qn. 9 - Section 4, Qn. 1
Secure Transportation of the Product (ST)		HIGH: 1.0, MEDIUM HIGH: 0.97, MEDIUM LOW: 0.94, LOW: 0.91	- Section 5, Qn. 2 - Section 5, Qn. 6 - Section 5, Qn. 8

the constant 32.315 to make sure base metric values are within a proper range. Similar to the CVSS exploitability score, the supply chain base metric ($SCRB$) is defined as:

$$SCRB = 32.315 \times O \times P \times OTT \times BP \tag{8}$$

Additionally, the supply chain risk temporal score ($SCRT$) is defined as follows:

$$SCRT = SCRB \times UV \times OU \times ST \tag{9}$$

After defining $SCRB$ and $SCRT$, SCR for a given device De is formally defined as:

$$SCR(De) = \begin{cases} SCRB, & \text{If factors UV, OU, and ST are not available.} \\ SCRT, & \text{If factors UV, OU, and ST are available.} \end{cases}$$

After defining the SCR, the next step is to define the probability of exploiting a supply chain

vulnerability. The probability of exploiting a given supply chain vulnerability in a given device De is defined as:

$$Pr(De) = \begin{cases} 1.0, & \text{If the attacker is behind that supply chain vulnerability.} \\ 0.08 \times \frac{SCR(De)}{10.0}, & \text{If the attacker is not behind that supply chain vulnerability.} \end{cases}$$

The probability is 1.0 for the attacker behind that supply chain vulnerability. For others, since supply chain vulnerabilities are designed to be stealthy, the probability of exploiting a supply chain vulnerability should be as difficult as even more difficult than exploiting a zero-day vulnerability (with the probability of 0.08 [147]). We use the 0.08 coefficient in $Pr(SE)$ to ensure the proper range of $Pr(SE)$, i.e., between 0.00 and 0.08.

Moreover, to differentiate between the difficulties of exploiting different vulnerabilities, the associated probability is converted to the path length using the approach detailed in Chapter 3. For instance, in Fig. 4.2, the probability of exploiting the supply chain vulnerability ($\langle V\_SupplyChain\_1, 2, 2\rangle$) is $0.08 \times 0.9 = 0.07$. Thus, the path length of exploiting that vulnerability is $log_{0.08}0.07 = 1.05$, which means it takes 1.05 times more effort than a zero-day vulnerability to exploit that supply chain vulnerability. Next, we leverage the supply chain exploit probabilities, which are calculated using SCR, to define the $kSupply$ metric.

4.4.2 Definition of $kSupply$ Security Metric

$kSupply$ metric (inspired by the $k0d$ metric[66]) is defined in this research to measure the security posture of IEC 61850 substations with respect to supply chain vulnerabilities. $k0d$ counts the minimum number of distinct zero-day vulnerabilities required to compromise a critical asset in a given network. Similarly, $kSupply$ counts the minimum number of distinct supply chain vulnerabilities (note that if an attacker is capable of exploiting supply chain vulnerabilities, the same attacker is assumed to be capable of exploiting zero-day and known vulnerabilities as well) in order to compromise the network. As explained in the motivating example in Section 4.1.1, an attacker who is capable of exploiting supply chain vulnerabilities can follow the attack path:

$\langle V\_SupplyChain\_1, 2, 2\rangle \rightarrow \langle V\_Trip, 2, 19\rangle$. This attack path contains one supply chain vulnerability and one zero-day vulnerability. Exploiting the supply chain vulnerability is equivalent to exploiting $\log_{0.08} 0.07 = 1.05$ zero-day vulnerabilities. Therefore, the value of $kSupply$ is 2.05, and the attack path is shown in red in Fig. 4.2. Intuitively, a larger value of the $kSupply$ metric means the substation is more secure since the chance of having a large number of distinct supply chain vulnerabilities all at once in the same substation would be significantly lower. More formally, the extended $kSupply$ metric is defined as:

Definition I: Given a network with a set of hosts H, a set of resources R, a set of services on each host with service mapping $service(.) : H \rightarrow 2^R$, a set of exploits $E = \{\langle r, h_s, h_d\rangle : r \epsilon\ service(h_d), h_s \epsilon\ H, h_d \epsilon\ H\}$, a set of supply chain exploits $S = \{\langle v\_SupplyChain, h_s, h_s\rangle : h_s \epsilon\ H\}$, and a set of detection mechanisms $D = \{\langle d, h_s, h_s\rangle : h_s \epsilon\ H, d \epsilon\ D\}$, an extended attack graph [12] is a directed graph $G(E \cup S \cup D, R_r \cup R_i)$ where $R_r \subseteq C \times (E \cup S \cup D)$ and $R_i \subseteq (E \cup S \cup D) \times C)$ are pre and postconditions, respectively. C is the set of condition nodes which can be connectivity, privilege, vulnerability, information leakage channel, critical information, the existence of a detection mechanism, and detection action. We define the $kSupply$ metric for an attack path P as $kSupply(P) = \sum_{i=1}^{N} log_{0.08} SProb(i, P)$. In this equation, $SProb(i, P)$ is the probability of exploiting the vulnerability i on the attack path P, which is 0.08 for zero-day, $\frac{CVSSScore}{10.0}$ for known, and $Pr(De)$ for supply chain vulnerabilities. $i = 1, 2, \ldots, N$ are identifiers of vulnerabilities on a single attack path leading to the sink condition of G. $kSupply$ for a given attack graph G is calculated by taking the minimum among $kSupply$ values of all attack paths leading to the sink condition. $kSupply(G) = \min_{\forall t=1,\cdots,n}(kSupply(P_t))$, where $P_t, t = 1, \cdots, n$ is an attack path leading to the sink condition of G and there are n such attack paths.

4.4.3 Definition of $kSupplier$ Security Metric

$kSupplier$ metric measures the minimum number of unique vendors required to be targeted by the attacker to compromise the substation. For instance, in Fig. 4.2, the shortest attack path with the minimum number of distinct vendors (shown in black) contains 2 distinct vendors: "Supplier1" and "Supplier9". Thus, $kSupplier$ is 2 for the attack graph in Fig. 4.2. More formally, $kSupplier$ is defined as:

Definition II: For a given attack graph A with n distinct attack paths $P_1, \cdots, P_n$, let an attack path P_i contain k exploits $E_1, \cdots E_k$. Let V be the set of distinct vendors on an attack path P_i. $V(P_i) = \bigcup_{i=1}^{k} \{Vendor(E_i)\}$, where $Vendor(E_i)$ gives the vendor for a given exploit E_i. $kSupplier(A) = \min_{\forall t=1,\cdots,n} \mathbf{card}(V(P_t))$, where $P_t, t = 1, \cdots, n$ represents attack paths on A, and $\mathbf{card}$ is the set cardinality function.

In the $kSupplier$ metric, we also consider services or devices for which the vendor cannot be verified, such as counterfeit devices [98]. For those devices, we cannot ensure the authenticity of the vendor. In this work, we model such devices as devices coming from an "Unknown" vendor. In the metric calculation, we consider an unknown vendor as a "Unknown" which would match any other vendor, and hence unknown vendors on a path will not increase the value of $kSupplier$ metric, and a path including only unknown vendors always has the $kSupplier$ value of 1. As we demonstrate through simulations in Section 4.5, $kSupplier$ and $kSupply$ complement each other in the sense that one of them can be used when the other one cannot be improved.

4.4.4 MDP Generation Based on Attack Graph

$kSupply$ and $kSupplier$ metrics consider cyber attacks in which the attacker attacks the system by exploiting vulnerabilities. The attack may or may not cause physical damage. As an example, in Fig. 4.5, the attacker steals information using covert channels. Even though this attack does not have any direct physical consequences, the attacker may be preparing for a coordinated attack with severe physical consequences. In this chapter, we consider the potential physical consequences of attacks. For this purpose, in addition to finding the shortest attack path for compromising critical hosts in networks, attack graphs can also be utilized for calculating probabilities of compromising each host by converting them into Bayesian networks (as detailed in [46] and [167]). However, attack graphs do not consider the physical damage the attacker can cause to power systems. In order to overcome this limitation, the authors in [74, 78] utilize the MDP model in which the attacker is modeled as an agent aiming to maximize the amount of physical damage to the power system for finding critical states the attacker can visit. States correspond to the set of compromised hosts. However, exploit probabilities in MDP models are calculated without considering exploit relationships in both [74] and [78].

An MDP [168] consists of the following: S is a finite set of states, A is the finite set of actions, $P(s, s', a)$ is the transition probability function, $R(s')$ is the reward function for the state s', $\gamma \in$ [0,1] is the discount factor. In order to generate the MDP model from the attack graph, the first step is to identify the set of states and the set of connections between those states. States (S) are sets of hosts compromised by the attacker in the substation. Actions (A) are vulnerability exploits. Algorithm 1 provides the algorithm for generating MDP for a given attack graph.

Algorithm 1 Generating MDP For a Given Attack Graph

1: **Input:** Attack graph connectivity conditions ($Conn$), and goal host (h_g).
2: **Output:** MDP model generated from the attack graph.
3: - Add the initial state $\{\}$ in which no hosts are compromised to the MDP
4: **while** No more states can be added **do**
5: - Get leaf nodes of the MDP
6: **if** the leaf node is the initial state ($\{\}$) **then**
7: - Find hosts h_i, where $< h_i, Network > \epsilon\, Conn$
8: **for** each host h_i **do**
9: - Add state $\{h_i\}$ to the MDP
10: - Create transition from $\{\}$ to $\{h_i\}$.
11: **end for**
12: **else**
13: **for** each leaf node **do**
14: **if** the leaf node contains the goal host, h_g **then**
15: - Do not add any more states
16: **else**
17: - Let $\{h_1, h_2, \cdots, h_{k-1}, h_k\}$ be the leaf node
18: - Find hosts h_i, for which $< h_k, h_i > \epsilon\, Conn$ in the attack graph,
19: **for** Each host h_i **do**
20: - Add state $\{h_1, h_2, \cdots, h_{k-1}, h_k, h_i\}$ to MDP
21: - Add transition from $\{h_1, h_2, \cdots, h_{k-1}, h_k\}$ to $\{h_1, h_2, \cdots, h_{k-1}, h_k, h_i\}$
22: **end for**
23: **end if**
24: **end for**
25: **end if**
26: **end while**

The first step is to add the empty state, which is the starting state of the attacker, to the MDP model (line 3). After that, the MDP model is updated by getting leaf nodes (line 5) and finding hosts directly reachable from the last hosts in leaf nodes (line 7 when the leaf node is the initial condition and line 17 when the leaf node contains a set of hosts). Consequently, new states are created based on reachable hosts (line 9 for the initial and line 19 when the leaf node contains states), and state

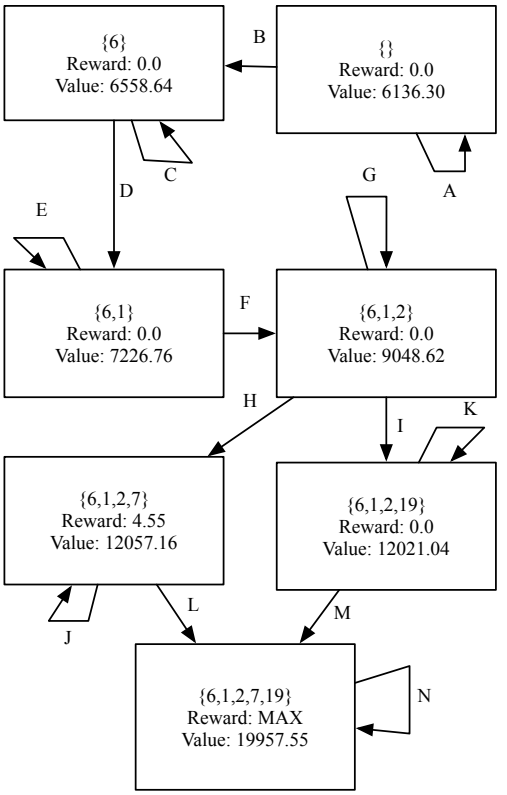
transitions are added (line 10 for the initial and line 20 when the leaf node contains states). This process continues until no states can be added to the MDP (line 4).

State transition probabilities ($P(s, s', a)$) are retrieved from the attack graph by converting it to a Bayesian model and performing inference on each privilege condition. For example, edge D in Fig. 4.7 is the probability that the attacker can compromise HMI (Node 1 in Fig. 4.1) given that the attacker has already compromised the gateway (Node 6 in Fig. 4.1). As a rule for an MDP model, the sum of probabilities of outgoing edges from one state must be equal to 1.0. A vulnerability exploit can either go to the next state (adding one more host to the set of compromised hosts) or stay in the same state (which means the attacker has failed at that attack stage).

The reward function is defined as the sum of overload percentages of transmission lines [74] after tripping a breaker. In pandapower [169], transmission line failures can be simulated, and load percentages can be retrieved using "loading_percent" parameter. The reward function is defined as the sum of loads exceeding the line capacity. As an example, if IED1 in Fig. 4.1 trips the transmission line between Bus1 and Bus2, load percentages for all other transmission lines are 57.60, 41.80, and 104.55 according to pandapower [169]. In this case, the amount of the reward is 4.55. If both IED1 and IED2 trip, it causes a blackout, so the reward is maximum.

The discount factor shows the difference between getting a reward immediately or in the future. In [78], the authors experiment with different discount factors. In this chapter, the discount factor is taken as constant 0.95 (a constant value close to 1.0) to make the value iteration algorithm [168] converge. After defining the MDP, the value iteration function is executed for the MDP to identify state values.

Fig. 4.7 shows the MDP for the attack graph in Fig. 4.1. According to this model, if the attacker has a choice between compromising IED1 (Node 7 in Fig. 4.1) or IED2 (Node 19 in Fig. 4.1), the attacker is more likely to choose IED1 since the value of the state in which IED1 is compromised (state $\{6, 1, 2, 7\}$ in Fig. 4.7) is more than the value of the state in which IED2 is compromised (state $\{6, 1, 2, 19\}$ in Fig. 4.7). As it can be seen, even though attacking both IEDs takes the same effort according to the $kSupply$ metric, their physical impacts are different, i.e., compromising IED2 causes more damage than compromising IED1.



A: Pr(<root,6> | {})' = 0.23
B: Pr(<root,6> | {}) = 0.77
C: Pr(<root,1> | {<root,6>})' = 0.48
D: Pr(<root,1> | {<root,6>}) = 0.52
E: Pr(<root,2> | {<root,6>,<root,1>})' = 0.79
F: Pr(<root,2> | {<root,6>,<root,1>}) = 0.21
G: Pr(<Tripped,7> | {<root,6>,<root,1>,<root,2>})' + Pr(<Tripped,19> | {<root,6>,<root,1>,<root,2>})' = 0.84
H: Pr(<Tripped,7> | {<root,6>,<root,1>,<root,2>}) =0.08
I: Pr(<Tripped,19> | {<root,6>,<root,1>,<root,2>}) =0.08
J: Pr (<Tripped,19> | {<root,6>, <root,1>, <root,2>, <Tripped,7>})' = 0.92
K: Pr (<Tripped,7> | {<root,6>, <root,1>, <root,2>, <Tripped,19>})' = 0.92
L: Pr (<Tripped,19> | {<root,6>, <root,1>, <root,2>, <Tripped,7>}) = 0.08
M: Pr (<Tripped,7> | {<root,6>, <root,1>, <root,2>, <Tripped,19>}) = 0.08
N: Pr (<Compromised,Substation> | {<root,6>, <root,1>, <root,2>, <Tripped,7>,<Tripped,19>}) = 0.08

MDP For The Attack Graph in Fig. 4.2 (Left) and Transition Probabilities (Right)

Figure 4.7: MDP Model and Transition Probabilities

4.4.5 Identifying Critical Substations

After generating attack graphs and MDPs for each substation, $kSupply$ and $kSupplier$ metrics are calculated. After that, the maximum state value is taken from each MDP model. This is done to identify critical substations. Critical substations can be defined as substations with both lower attack resilience and higher attack impacts compared to other substations in the same power system. More formally, for a power system with n substations, the criticality of each substation S_i, $i = 1, \cdots, n$ is a 3-tuple $(kSupply_i, kSupplier_i, max(St_i))$, where St_i is the set of state values [168] for the MDP model generated for S_i. For two substations S_i and S_j, S_i is more critical than S_j if and only if $kSupply_i \leq kSupply_j$ and $kSupplier_i \leq kSupplier_j$ and $max(St_i)) \geq max(St_j))$. Finally, the Pareto front of critical substations is provided to the operator so that those substations can be the focus of security efforts. For example, the 3-tuple for the substation in Fig. 4.1 is $(2.05, 2.00, 19957.55)$.

4.5 Simulations

Simulations for evaluating the effectiveness of $kSupply$ and $kSupplier$ are conducted on MacOS Big Sur 11.2 with a 2.4 GHz $i7$ processor and 16 GB of memory. 4800 attack graphs are generated using the attack graph given in Fig. 4.2, which is based on the substation configuration in Fig. 4.1, as the seed graph and adding new hosts and new exploits (known, zero-day, and supply chain). The scenarios for simulating the effectiveness of the designed security metrics are explained in the following subsections.

4.5.1 Relationship between $kSupply$ And Number of Attackers

In this scenario, we assume the power system operator decides to deploy the $kSupply$ metric for measuring the security posture against different types of attackers that are capable of exploiting the following types of vulnerabilities (i) only known, (ii) known and zero-day, and (iii) known, zero-day, and supply chain. Therefore, our first simulation aims to find the relationship between the $kSupply$ metric and the number of successful attackers.

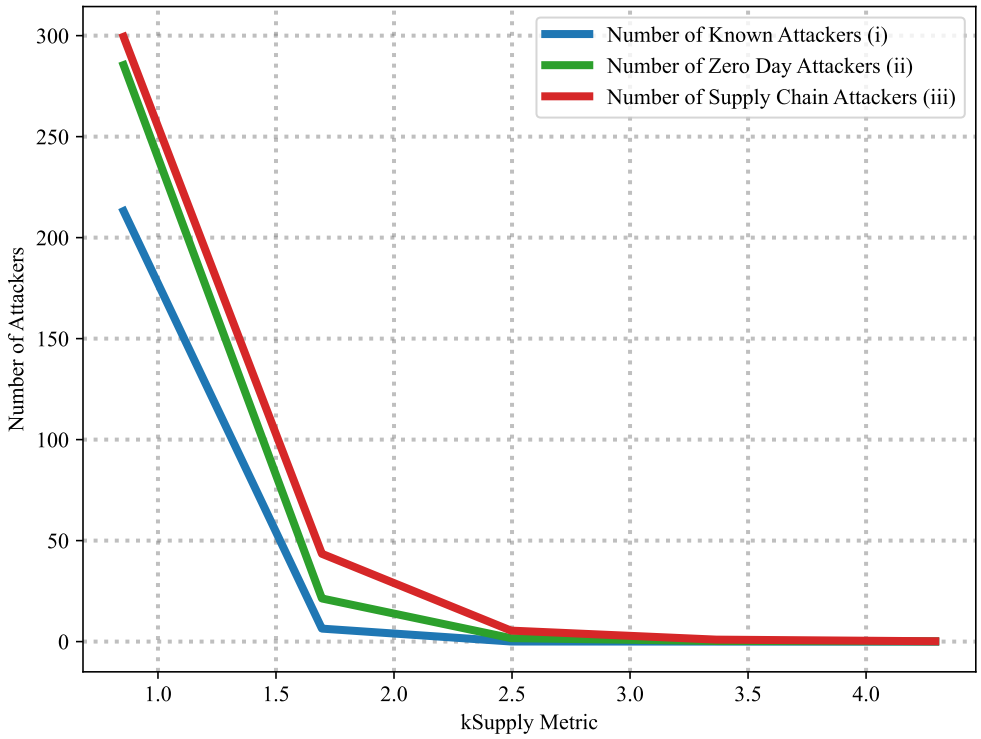


Figure 4.8: Security Metric $kSupply$ versus Attackers

Results and Implications: As it can be seen in Fig. 4.8, the number of category (i) attackers decreases quickly as $kSupply$ increases. After 1.7, only a negligible number of attackers of the category (i) can succeed. The number of category (ii) attackers also decreases sharply; after 2.4, only a negligible number of them can succeed. For category (iii) attackers, after 2.5, only a small number of them can succeed. This result shows the importance of improving the security posture of IEC 61850 substations against attackers of category (iii), which will also lead to its improvement against categories (i) and (ii).

4.5.2 Relationship between $kSupplier$ And Number of Attackers

In the previous scenario, it was shown that the operator could utilize the $kSupply$ metric to measure the security posture of the substation against the three categories of attackers. To overcome

the condition in which the $kSupply$ cannot be improved, another metric, i.e., $kSupplier$, is proposed to the operator. Therefore, our second simulation aims to identify the relationship between the $kSupplier$ metric and the number of attackers for each category to show that the $kSupplier$ metric is also effective.

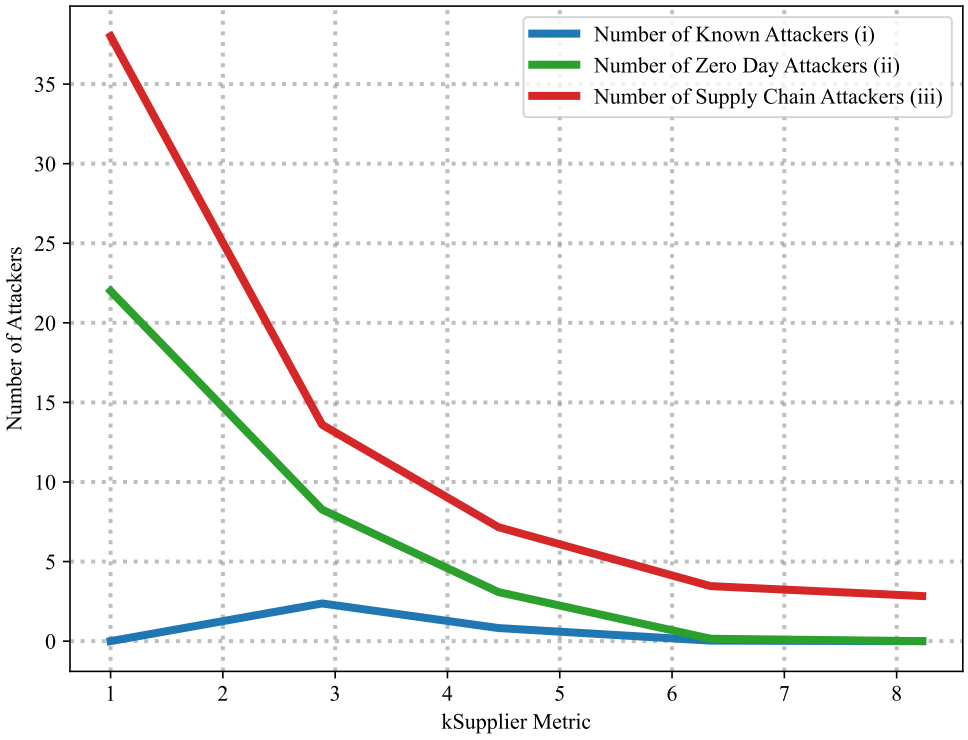


Figure 4.9: Security Metric $kSupplier$ versus Attackers

Results and Implications: As can be seen in Fig. 4.9, the $kSupplier$ metric also represents the difficulty of compromising the network for three categories of attackers. Category (i) attackers become a negligible issue after $kSupplier$ value of 5 and $kSupplier$ is not as effective as $kSupply$ in terms of measuring the security posture against those attackers. The number of category (ii) attackers becomes negligible after the value of 6. The number of category (iii) attackers who can succeed becomes less, but it does not become negligible even with $kSupplier$ value 7. This result is because there can be attack paths with a huge diversity of vendors but autonomous supply chain vulnerabilities close to critical resources.

4.5.3 Relationship between $kSupply$ And $kSupplier$

This simulation aims to determine how $kSupply$ and $kSupplier$ metrics behave differently. A utility may be bound to a limited number of vendors, so it may not always be possible to increase $kSupplier$. Therefore, our third simulation aims to analyze the relationship between $kSupply$ and the number of category (iii) attackers for similar $kSupplier$ values.

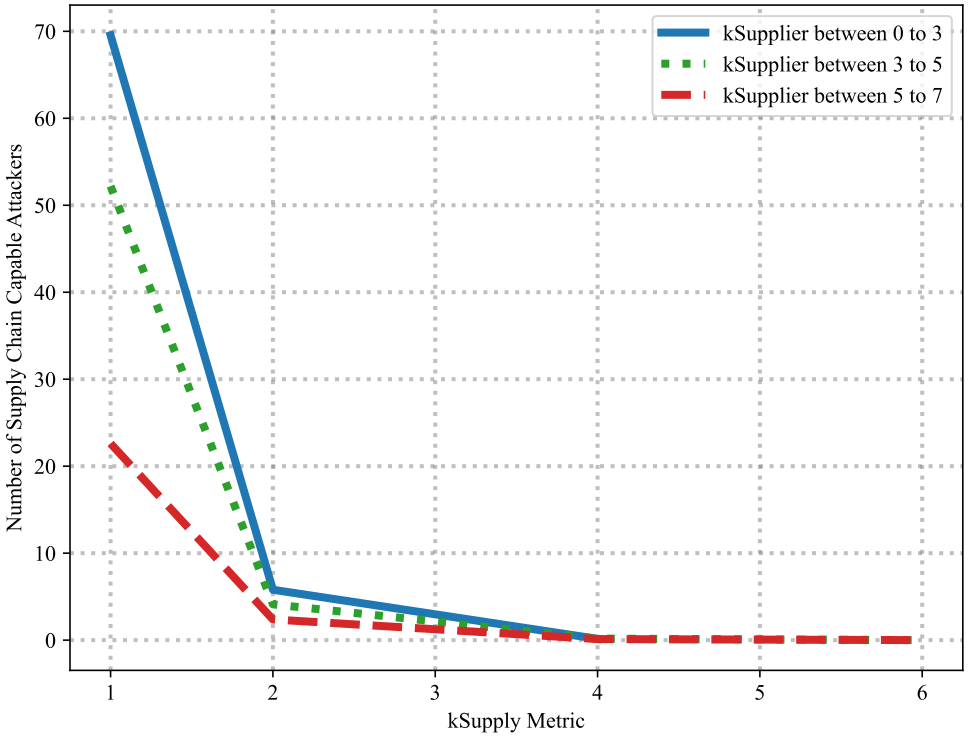


Figure 4.10: Security Metric $kSupply$ for Different $kSupplier$

Results and Implications: As it can be seen in Fig. 4.10, even when the operator is bound to a limited number of vendors, they can eliminate the majority of attackers by increasing $kSupply$. After $kSupply$ value of around 3, category (iii) attackers become negligible even for small $kSupplier$ values. Fig. 4.9 shows that $kSupplier$ is also effective in measuring the security posture, and increased $kSupplier$ leads to more security. However, in cases where $kSupplier$ cannot be improved (such as not being able to diversify vendors), $kSupply$ can be utilized.

4.5.4 Metric Calculation Times for $kSupply$ and $kSupplier$

A power utility may want to increase its substation size and ensure that the calculation of security metrics will not be impacted significantly. Therefore, our fourth simulation aims to determine metric calculation times.

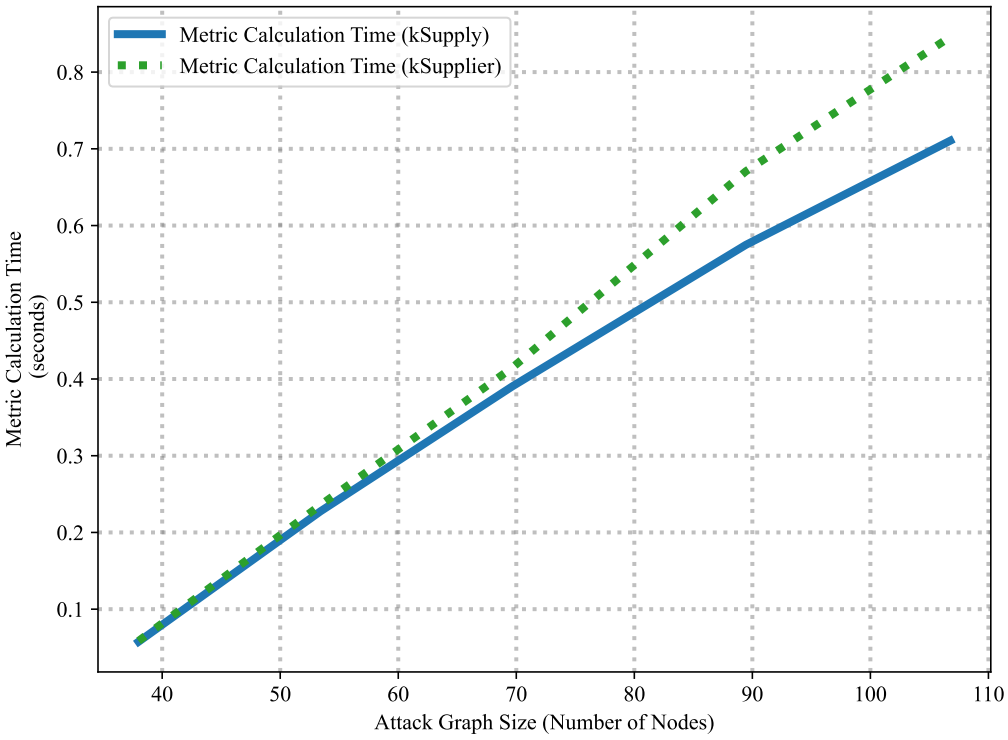


Figure 4.11: Metric Calculation Times

Results and Implications: As it can be seen in Fig. 4.11, even for attack graphs with 100 nodes, the metric calculation takes less than 1 seconds. The calculation time of $kSupply$ is almost linear since heuristics are utilized for efficient calculation (explained in [50]). The calculation time of $kSupplier$ is also almost linear but slightly larger than that of $kSupply$. Based on this, a power utility can utilize $kSupply$ frequently and $kSupplier$ when they deploy devices manufactured by different vendors.

4.5.5 Number of Channels and Coordinating Devices

In this experiment, our goal is to show how the number of coordinating devices affects the $k0d$ metric and $kSupply$ metrics. We generate attack graphs with the number of coordinating devices ranging from 2 to 5, and we take the average metric values for each case.

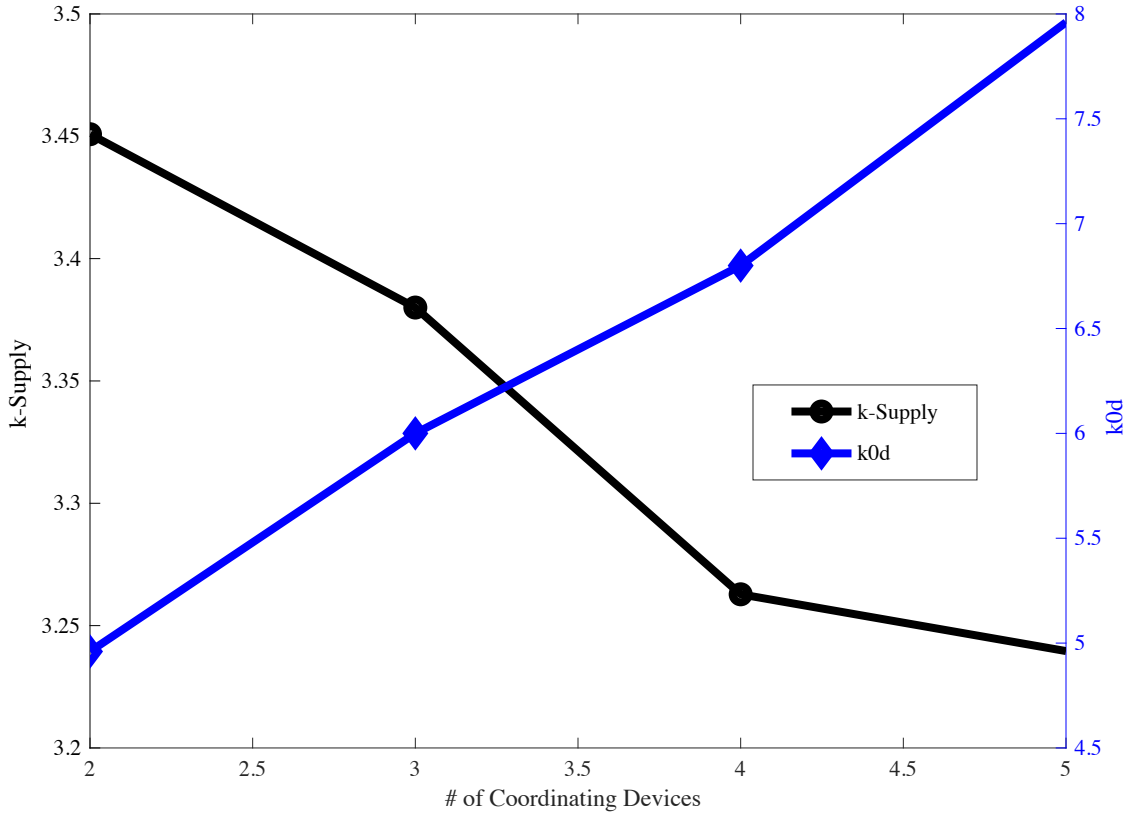


Figure 4.12: Number of Coordinating Devices versus Metrics

Results and Implications: In Fig. 4.12, $kSupply$ decreases as the number of coordinating devices increases. This result is expected since having more devices from a malicious vendor means less security with respect to supply chain vulnerabilities. On the other hand, the $k0d$ metric increases since the attacker has to compromise each device separately without taking advantage of supply chain vulnerabilities.

In this experiment, our goal is to show how the number of communication channels affects the number of attackers who may succeed. For this scenario, we assume that all supply chain

vulnerabilities are autonomous and that one channel is enough to leak critical information. We fix the value of $kSupply$ to 1.

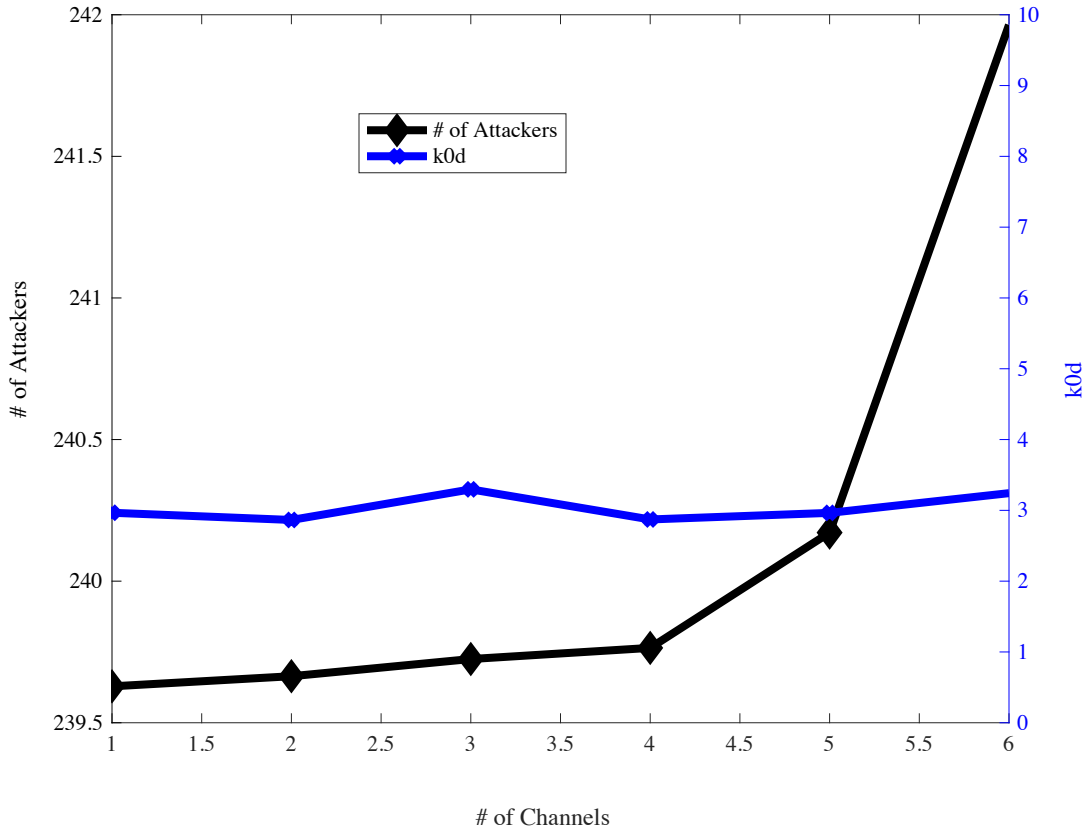


Figure 4.13: Number of Channels versus Number of Attackers

Results and Implications: In Fig. 4.13, the number of successful attackers also increases as the number of channels increases. There is a fast increase from 4 to 5 channels and an even faster increase from 5 to 6. For k0d, the change is negligible since the metric does not consider information leakage. Adding more channels increases the number of paths an attacker may follow, but it usually does not affect the shortest attack path. The implication of this simulation is that a large number of communication channels can be risky, and such channels should be closely examined for abnormal traffic.

4.5.6 The relationship Between $k0d$ and $kSupply$

In this experiment, our goal is to identify how k0d and k-Supply differ as the number of autonomous supply chain vulnerabilities increases.

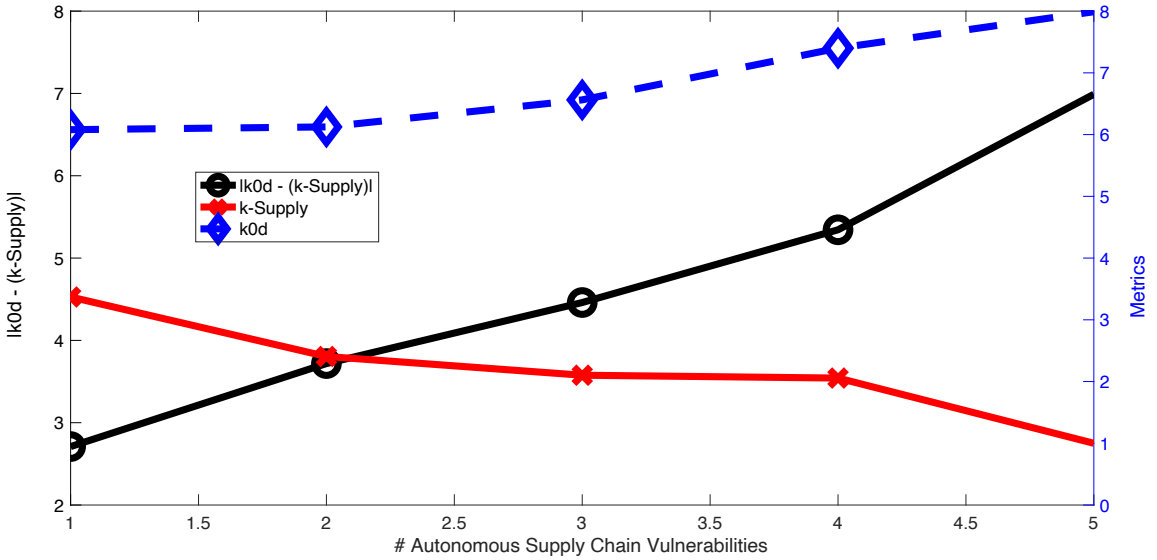


Figure 4.14: The number of autonomous supply chain vulnerabilities vs. the metrics

Results and Implications: In Fig. 4.14, as the number of autonomous supply chain vulnerabilities increases, the average value of k-Supply decreases, whereas the value of k0d does not change as much. This is expected since autonomous vulnerabilities do not need to be activated, which leads to lower k-Supply values.

4.5.7 Supply Chain Risk Factors

In this experiment, our goal is to identify how supply chain risk factors influence the SCR. This simulation analyzes how SCR changes based on supply chain risk factors. In this simulation, we utilize the same 4800 attack graphs, fix each risk factor's values, and randomly assign values of other risk factors.

Results and Implications: As it can be seen in Figs. 4.15 and 4.16, an increase in any of those risk factors causes an increase in the number of type (iii) attackers. Risk factors, O, BP, and P, are the most important ones to focus on since the difference between the numbers of type (iii) attackers

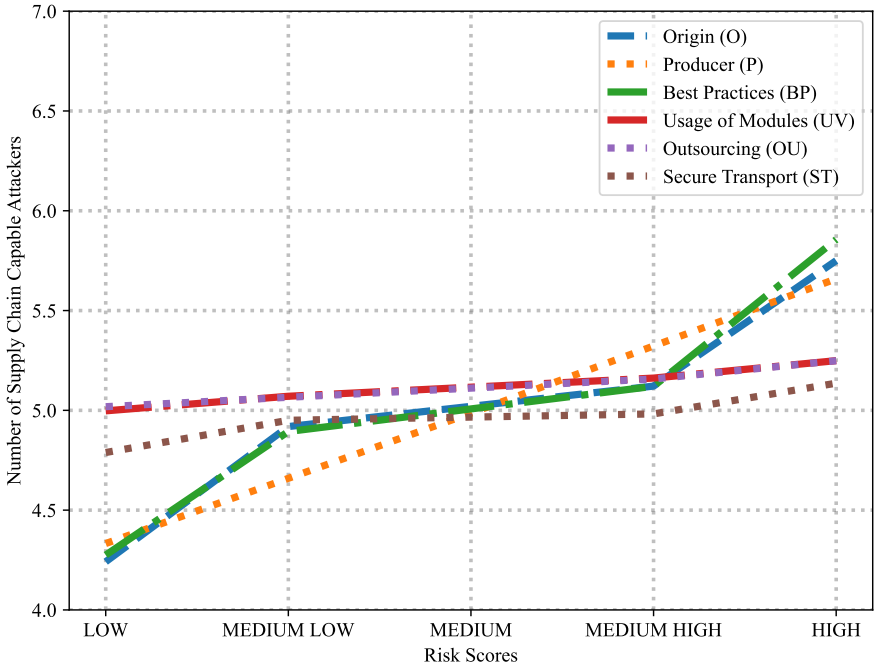


Figure 4.15: Risk Factors O, P, BP, OU, UV, ST

is higher for those. UV, OU, and ST are also important, but their ranges are smaller. For OTT, as it can be seen in Fig. 4.16, the difference between the numbers of types (iii) attackers is significant, so it is essential to choose from vendors included in Open Forum [165] when applicable.

4.5.8 MDP Generation for Different Bus Systems

In this experiment, our goal is to analyze the total times for the MDP and attack graph generation for bus systems with different sizes. For this purpose, we generate attack graphs for each bus in the bus system using a seed graph like the one in Fig.4.2. After that, an MDP model is developed for each bus utilizing the corresponding attack graph models and calculating rewards using pandapower [169]. Experiments are conducted 100 times for each bus system to get the average execution time.

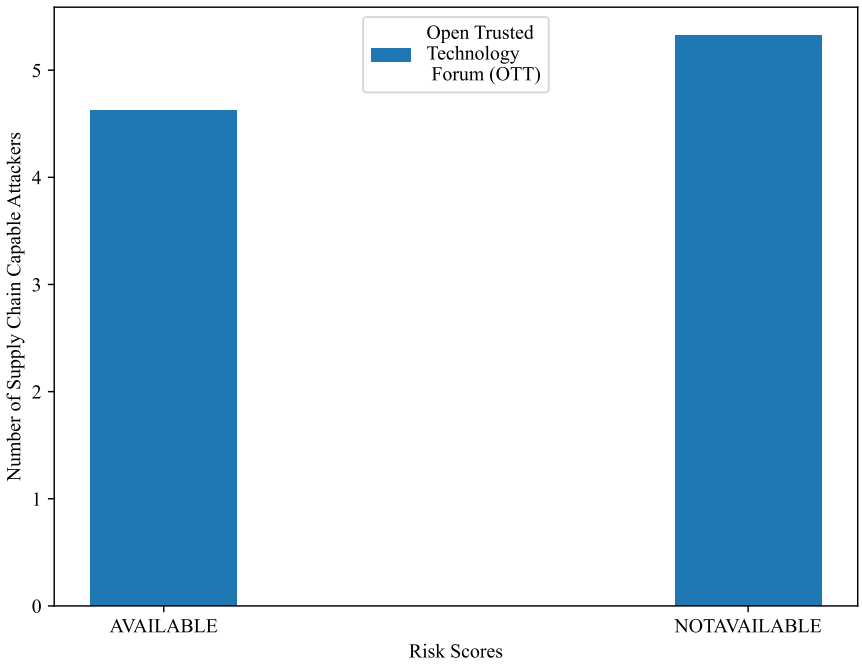


Figure 4.16: Risk Factor OTT

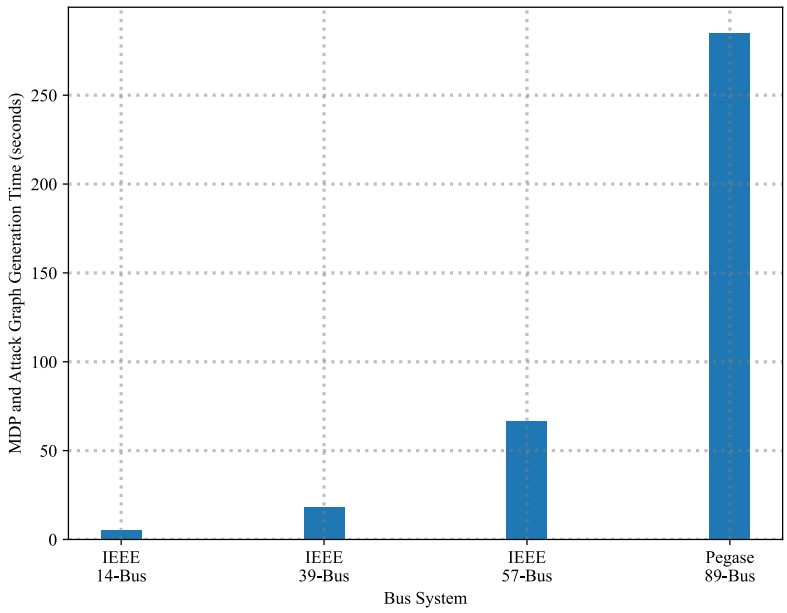


Figure 4.17: MDP Generation Times for Different Bus Systems

Results and Implications: As it can be seen from Fig. 4.17, even for a large system such as Pegase 89-Bus [170], attack graph and MDP generation takes around 260 seconds. These results show that $kSupply$, $kSupplier$, and state criticality values can be deployed to large power systems.

4.5.9 Critical Substations

In this scenario, our goal is to identify critical substations for IEEE 14 and 39-bus systems. For this purpose, we generate attack graphs for each bus in the bus system using a seed graph like Fig.4.2. After that, an MDP model is developed for each bus utilizing the corresponding attack graph models and calculating rewards using pandapower [169]. Consequently, critical substations are identified in each bus system. Experiments are repeated 100 times for each bus system to get the average execution time and critical substations. The number of times each substation in the bus system became critical out of 100 experiments is recorded.

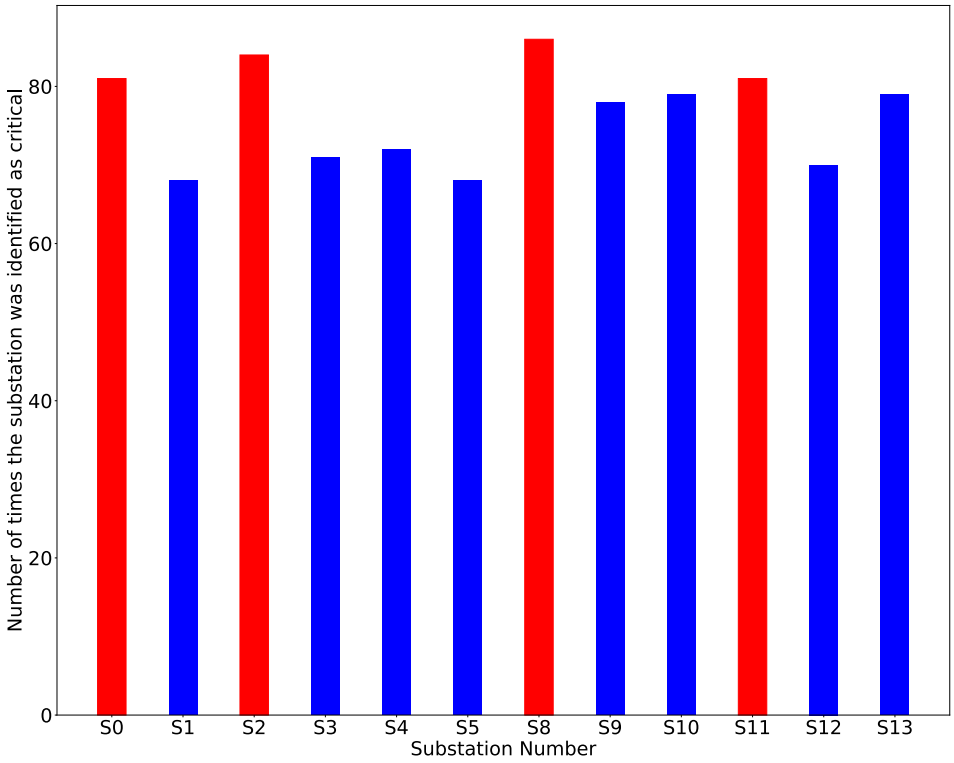


Figure 4.18: Critical Substations in 14-Bus

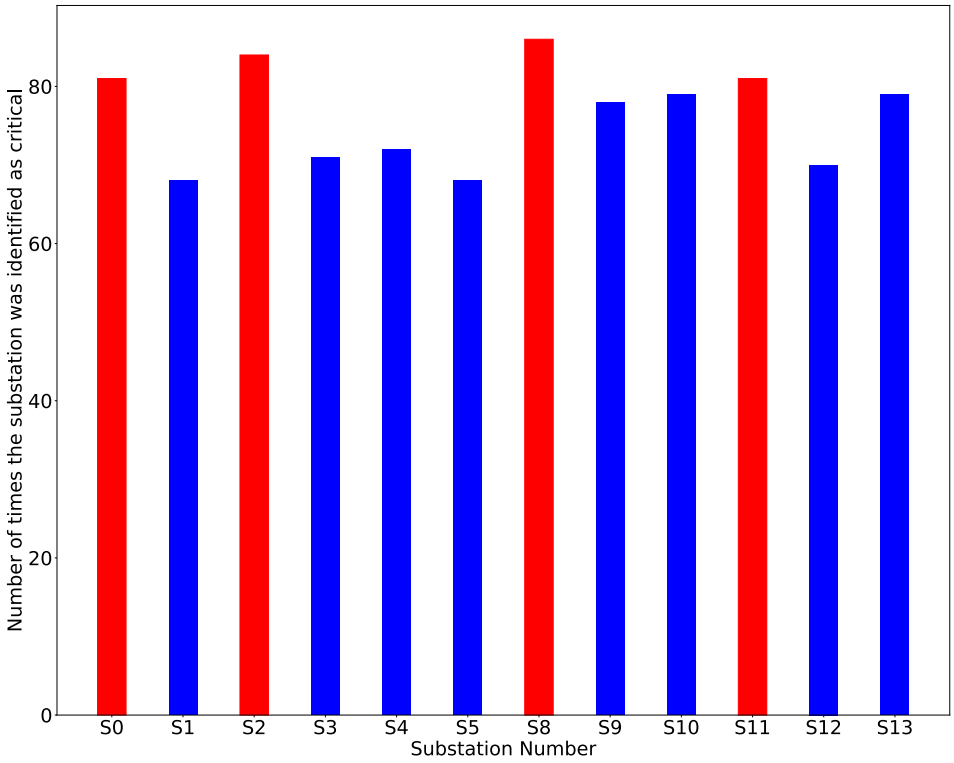


Figure 4.19: Critical Substations in 39-Bus

Results and Implications: As it can be seen in Fig. 4.18, Bus 0, 8 and 11 (note that bus indices are obtained from Pandapower [169]) in the IEEE 14-Bus are identified as critical substations around 80 times among 100 experiments. This shows that those substations should be the focus of security enhancement attempts in the IEEE 14-Bus. Experiment results for identifying critical substations in the IEEE 39-Bus are shown in Fig. 4.19. Critical substations in IEEE 39-Bus are 18 and 35.

4.6 Applying Supply Chain Security Metrics to Other Critical Systems

Security metrics defined in this chapter are applicable to other critical systems as long as those systems deploy devices from different vendors. SCR is a generic risk assessment for supply chain

vulnerabilities. For $kSupply$ and $kSupplier$, only the attack graph model needs to be modified. The MDP model proposed in this chapter can be utilized as long as attacks are cyber with physical consequences. The reward function utilized in the MDP model needs to be modified according to the new system since physical damage can be quantified in different ways for different systems.

4.7 Summary

In this chapter, three new metrics are designed to measure the security posture of IEC 61850 substations against supply chain attacks. The SCR is developed by taking supply chain risk factors into account. After that, we integrated SCR into the $kSupply$ to differentiate between different supply chain vulnerabilities. Moreover, the $kSupplier$, which counts the minimum number of vendors on attack paths, is designed to overcome the cases in which $kSupply$ cannot be improved. Through simulations, it has been shown that the defined metrics effectively measure the security posture of IEC 61850 substations against supply chain attacks.

Chapter 5

Optimal Security Hardening of Substations Against Supply Chain Attacks

5.1 Introduction

Supply chain attacks present a significant threat to the security posture of substations, but they have attracted limited attention in the smart grid research domain. The naïve solution of replacing less trusted vendors may not always be practically feasible due to operational constraints, such as limited availability of devices and prohibitive costs. Most existing works focus on defining supply chain security metrics (Chapter 4), threat reporting among supply chain stakeholders [101], and demonstrating the improvement of the security posture using non-supply chain hardening options [6]. Therefore, there is a need for a systematic solution that identifies hardening options for improving the security posture of IEC 61850 substations against supply chain attacks.

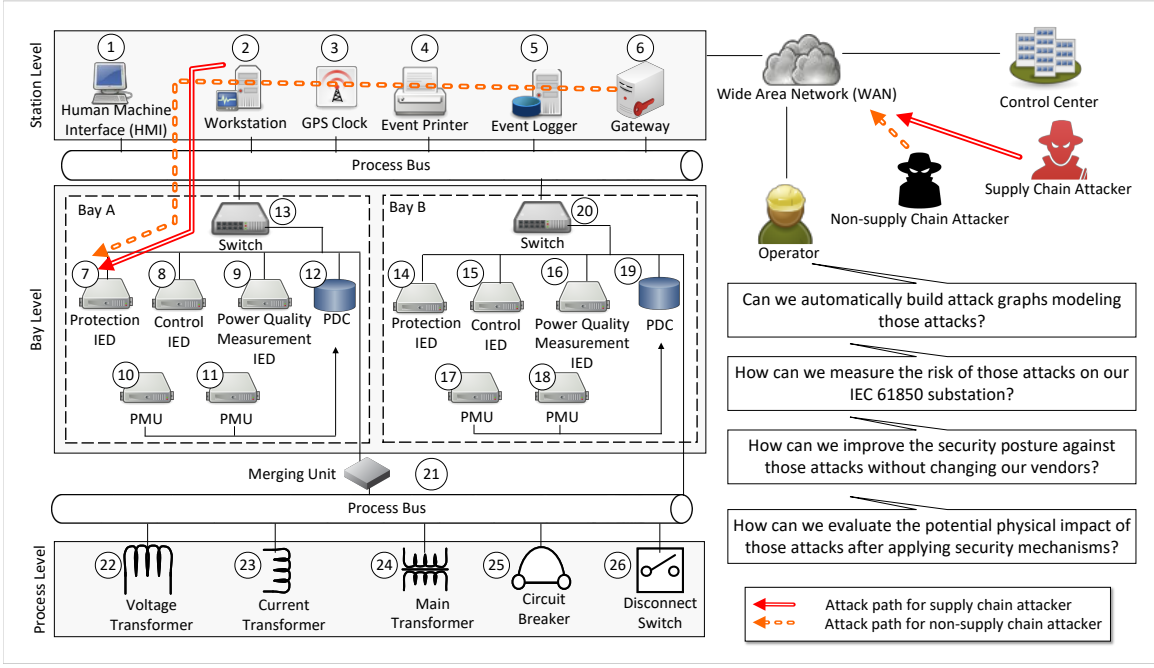
To improve the security posture of substations against supply chain attacks, the operator needs first to establish a model to capture potential vulnerabilities in the substation and how an attacker might combine them to gain access, which can be challenging without automated tools. Moreover,

to identify the optimum hardening options, the operator needs to evaluate and compare the improvement of security metrics that can be achieved by applying different hardening options. Such hardening options must also respect the power utility's limited budget and operational constraints (e.g., certain connections cannot be behind a firewall). However, this can be daunting if done manually, especially considering many hardening options and constraints (detailed in Section 5.2.2). For instance, protecting HMI with a firewall may prevent the non-supply chain attacker, but it does not block the attack path of the supply chain attacker. Moreover, replacing the vendor of the HMI for hardening the substation may not be feasible when there are constraints bounding the operator to a single vendor. Finally, considering the cyber-physical nature of substations, the operator may additionally need to analyze the physical impact of cyber attacks after hardening to identify critical IEDs controlling transmission lines. In the next section, we provide a motivating example based on a real-world substation design, demonstrating challenges faced by operators.

5.1.1 Motivating Example

Fig. 5.1 illustrates an example substation (left) [167] and the challenges faced by an operator (right) to improve the security posture of the substation. Suppose that the operator is concerned mainly about attacks aiming at tripping circuit breakers [50] (as it would lead to overloading transmission lines and their subsequent failure). Assume that two attackers with and without the capability of exploiting supply chain vulnerabilities (i.e., *supply chain attacker* and *non-supply chain attacker*, respectively, in Fig. 5.1) can access the substation through the WAN. To start tripping circuit breakers, the non-supply chain attacker first needs to infiltrate into the substation network by compromising the Gateway ($Node6$). Next, the attacker can compromise the HMI ($Node1$) and the Workstation ($Node2$), and finally, compromise the Protection IED, $Node7$. However, the supply chain attacker can start tripping circuit breakers by simply compromising the Workstation ($Node2$) and the protection IED ($Node7$) through an existing supply chain vulnerability that the attacker has already planted inside the Workstation ($Node2$) to open a backdoor [157] and a zero-day vulnerability, respectively.

To address those challenges, in this chapter, we propose the Hardening Framework for Substations (*HFS*), an interactive system for improving the security posture of substations in the presence



An Example IEC 61850 Substation and the Attack Paths Targeting Protection IEDs (Left); Challenges of Improving the Security Posture of the Substation Against the Supply Chain Attacker and Non-supply Chain Attacker (Right)

Figure 5.1: IEC 61850 Substation and Attack Paths

of budget and operational constraints. Specifically, HFS enables operators to automatically generate attack graphs, calculate security metrics, and reason about different hardening options through a visualization interface. Behind the scenes, HFS hardens substations against supply chain attacks by cost-effectively maximizing the security posture using multi-objective optimization and allows analyzing the potential physical impacts of cyber attacks. To support scenarios where there only exists limited supply chain-related hardening options, we design HFS to mitigate supply chain attacks by applying a mixture of supply chain-related hardening options (e.g., replacing vendors) and non-supply chain-related hardening options (e.g., adding firewalls and relocating services). In the next section, we present the detailed methodology of HFS.

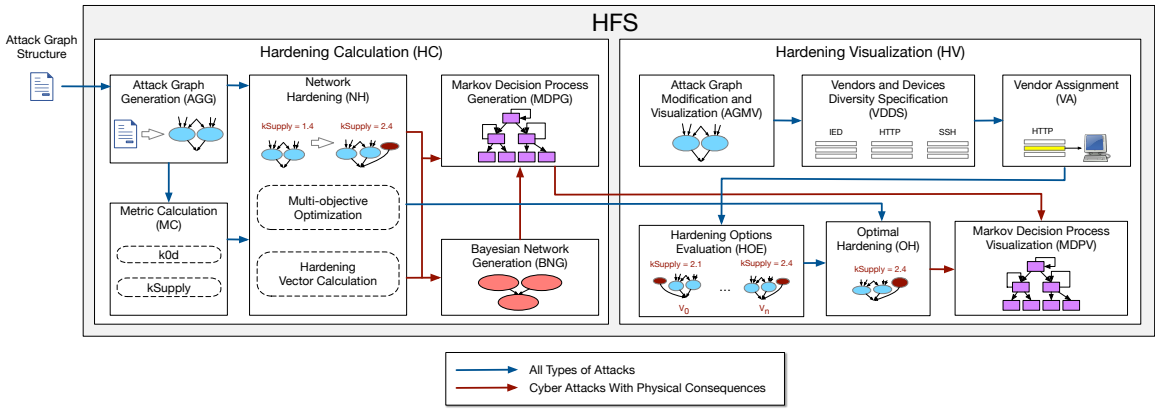


Figure 5.2: Overview of HFS

5.2 Hardening Methodology

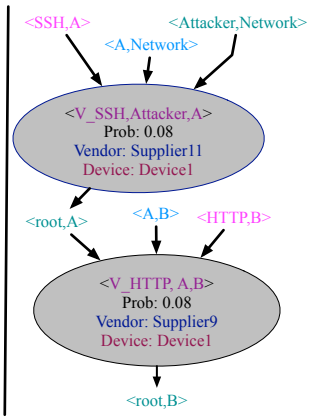
5.2.1 Overview

HFS enables the identification of optimal hardening options based on the operational and cost constraints of the substation network. Fig. 5.2 shows an overview of HFS consisting of two main components: Hardening Computation (HC) and Hardening Visualization (HV).

The HC component (Section 5.2.2) performs the underlying computations required for improving the security of the substation. To this end, it first generates the attack graph corresponding to a given substation network and then calculates the security metrics for the generated attack graph. Next, HC identifies the optimal hardening options based on the computed security metrics. Finally, it leverages the Bayesian Network and Markov Decision Process to analyze the potential physical impact of cyber attacks before and after hardening.

The HV component (Section 5.2.3) enables the operator's interaction with HC and visualizes the hardening results. To this end, HV allows the operator to adjust attack graphs interactively, specify the list of devices for each service, and assign a vendor and a model for services running on each host. The operator can also experiment with different hardening options. Based on the specified values, HV identifies and visualizes the optimal hardening options and demonstrates the effectiveness of hardening against cyber attacks with physical consequences. The rest of this section explains each module of HFS.

```
<attack_graph name="SimpleAG" goal="<root,B>" initial="<Attacker,Network>">
  <exploit_node type="unknown" cveid="CVE-XXXX" source="Attacker" target="A" name="SSH"
    vendor="Supplier11" device="Device1">
    <precondition name="<SSH,A>" type="vulnerability"></precondition>
    <precondition name="<A,Network>" type="connection"></precondition>
    <precondition name="<Attacker,Network>" type="privilege"></precondition>
    <postcondition name="<root,A>" type="privilege"></postcondition>
  </exploit_node>
  <exploit_node type="unknown" cveid="CVE-XXXX" source="A" target="B" name="HTTP"
    vendor="Supplier9" device="Device1">
    <precondition name="<HTTP,B>" type="vulnerability" ></precondition>
    <precondition name="<A,B>" type="connection"></precondition>
    <precondition type="privilege" name="<root,A>"></precondition>
    <postcondition type="privilege" name="<root,B>"></postcondition>
  </exploit_node>
</attack_graph>
```



XML Definition (Left) of a Basic Attack Graph (Right). Semantically Related Items Have the Same Color

Figure 5.3: XML Definition of a Sample Attack Graph

5.2.2 Hardening Computation (HC)

In this section, we discuss how HC handles attack graph generation, metric calculation, and network hardening.

Attack Graph Generation (AGG)

AGG module generates attack graphs that capture known, zero-day, and supply chain vulnerabilities corresponding to an IEC 61850 substation. An example attack graph demonstrating how each attacker (supply chain-capable and non-supply chain-capable) can maliciously trip circuit breakers is shown in Fig. 5.4.

To this end, the AGG module takes its input information in XML format that the operator provides according to the configuration of the substation (i.e., the set of hosts and their interconnections, the services running on each host, etc.). Fig. 5.3 shows an example XML file (left) of a basic attack graph (right). Moreover, to support incremental updates over time, HFS allows the operator to interactively adjust the basic attack graph through the following attack graph operations:

Adding a firewall for a connection Each connectivity condition represented in the attack graph can be placed behind a firewall with a bypass probability that the operator specifies. Fig. 5.5 illustrates the addition of a firewall to protect the connectivity condition $\langle 6, 1 \rangle$ in Fig. 5.4.

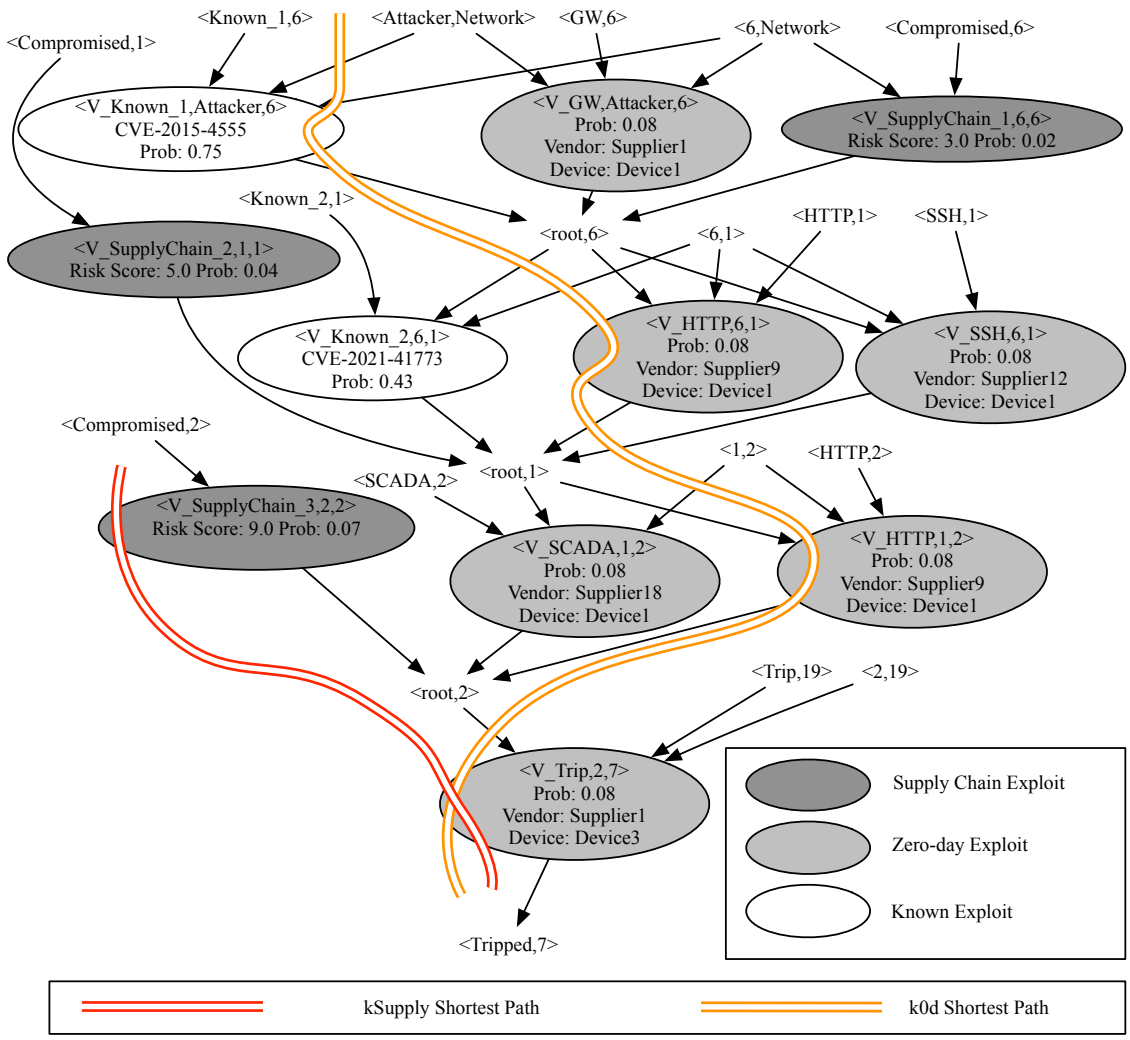
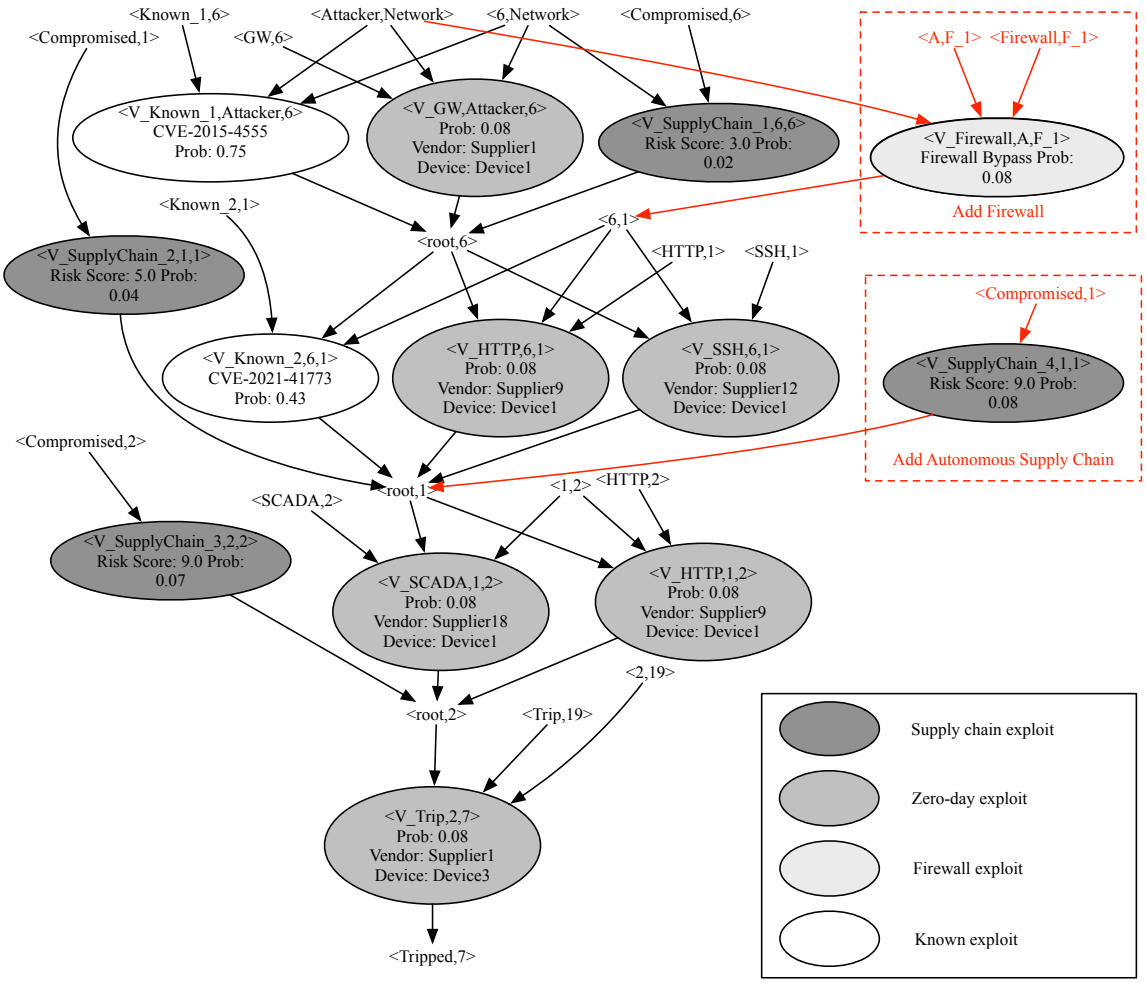


Figure 5.4: Attack Graph for Maliciously Tripping Circuit Breakers

Adding a host between two connected hosts A new host with an assigned service can be added between each pair of directly connected hosts, while the connection between the latter may or may not be retained.

Adding a vulnerability to a host A service with a vulnerability (zero-day, known, or supply chain) can be added to an existing host. Fig. 5.5 illustrates the addition of a supply chain vulnerability with the risk score (Section 4.4.1) of 9.0 to $host1$ in the attack graph shown by Fig. 5.4.

Adding a detection mechanism to a host A new detection mechanism with a particular detection rate can be added to one of the existing hosts.



Attack Graph Resulting From Adding One Firewall and Supply Chain Vulnerability to the Attack Graph in Fig. 5.4 (New Edges Are Shown in Red)

Figure 5.5: Firewall and Supply Chain Vulnerability Addition to Fig. 5.4

In the next section, we detail how the generated attack graph is adopted for measuring the security posture based on the security metrics.

Metric Calculation (MC)

This module is responsible for calculating the security metrics, which are $k0d$ [66] and $kSupply$ (Chapter 4). For calculating those metrics, the MC module first traverses the attack graph and extracts all attack paths leading to the goal condition. Extracting attack paths is typically done through logical reasoning about the conjunctive and disjunctive relationships between nodes [69]. Conjunction means an attack graph node and its postconditions are included in an extracted attack path if all preconditions of that node are satisfied. For instance, in Fig. 5.4, the exploit $\langle V\_GW, Attacker, 6\rangle$ and its postconditions are included in the extracted attack path when all of its preconditions (i.e., $\langle GW, 6\rangle$, $\langle 6, Network\rangle$, and $\langle Attacker, Network\rangle$) are satisfied, which results in an extracted attack path with the following set of nodes: $\{\langle GW, 6\rangle$, $\langle 6, Network\rangle$, $\langle Attacker, Network\rangle$, $\langle V\_GW, Attacker, 6\rangle$, $\langle root, 6\rangle\}$. Disjunction means an attack graph node is included in multiple attack paths, as the node can be satisfied if one of its preconditions is satisfied. For instance, in Fig. 5.4, the node $\langle root, 6\rangle$ can be reached from three different attack paths, so it is added to three attack paths with the following sets of nodes:

- $P1$:$\{\langle GW, 6\rangle$,$\langle 6, Network\rangle$,$\langle Attacker, Network\rangle$, $\langle V\_GW, Attacker, 6\rangle$,$\langle root, 6\rangle\}$
- $P2$:$\{\langle Known\_1, 6\rangle$,$\langle 6, Network\rangle$, $\langle Attacker, Network\rangle$,$\langle V\_Known\_1, Attacker, 6\rangle$, $\langle root, 6\rangle\}$
- $P3$:$\{\langle Compromised, 6\rangle$,$\langle 6, Network\rangle$, $\langle V\_SupplyChain\_1, 6, 6\rangle$,$\langle root, 6\rangle$ $\}$

We recursively perform conjunction and disjunction procedures to extract the final attack paths until the goal condition is included in the extracted attack paths. MC calculates $k0d$ and $kSupply$ metrics based on the extracted attack paths for the attack graph. To reduce the complexity of calculations, we leverage a heuristic method similar to [46], which retains only a certain number (h, which is a parameter determined by the operator to trade-off between efficiency and accuracy [46])) of the shortest paths for each privilege condition. For instance, in Fig. 5.4, the privilege condition $\langle root, 6\rangle$ can be reached through three attack paths, namely $P1$, $P2$, and $P3$, with attack path

lengths of $log_{0.08}(0.08) = 1.0$, $log_{0.08}(0.08) = 0.11$, and $log_{0.08}(0.02) = 1.55$, respectively. The *h* value of two means that the security metrics are calculated based on the two shortest attack paths (i.e., P1 and P2).

Network Hardening (NH)

This module hardens the substation according to the calculated security metrics. In this chapter, we categorize hardening options into supply chain-related and non-supply chain-related options. Specifically, supply chain-related options include:

- **Replacing vendors with more trustworthy ones** This is the most straightforward option to reduce the supply chain risk score. For example, in Fig. 5.4, a suitable candidate for this option is the HMI ($host1$) from a vendor with a risk score of 5.0. However, in some cases, such a replacement is costly and may not effectively improve the network's security posture. For example, replacing the vendor of HMI with a more trustworthy one does not increase the value of $kSupply$ since the supply chain vulnerability in HMI ($\langle V\_SupplyChain\_2, 1, 1\rangle$ in Fig. 5.4) is not on the shortest $kSupply$ attack path.

- **Upgrading the vendor hardware** Compared to a new device, an old device is more likely to contain supply chain vulnerabilities due to its potentially longer exposure to attackers. Upgrading such old products with newer alternatives equipped with more security measures may reduce the risk of supply chain vulnerabilities. For instance, according to[1], new products with stronger hardware security measures can improve the overall security posture of an enterprise by 55%.

- **Upgrading the vendor firmware** Upgrading firmware can sometimes patch existing supply chain vulnerabilities, which reduces the SCR. For instance, according to [2], 60% of security

[1] 63% of organizations face security breaches due to hardware vulnerabilities https://www.techrepublic.com/article/63-of-organizations-face-security-breaches-due-to-hardware-vulnerabilities/ [Online; Accessed 29-May-2021]

[2] The majority of 2019 breaches were the result of unapplied security patches https://www.helpnetsecurity.com/2019/10/30/unapplied-security-patches/ [Online; Accessed 29-May-2021]

breaches are linked to unapplied security patches. However, this option can also be a double-edged sword as applying a firmware upgrade may decrease the security posture (e.g., in the SolarWinds case [155], the update channel of the vendor was hijacked by attackers).

Additionally, HFS leverages non-supply chain-related hardening options. Although those options do not directly improve the SCR (Section 4.4.1) of supply chain vulnerabilities, they may still improve the value of $kSupply$ and thus improve the overall security posture of substations against supply chain attacks. Non-supply chain-related hardening options include:

- **Adding firewalls** Each connectivity condition (e.g., $\langle 6, 1\rangle$ in Fig. 5.4) can be placed behind a firewall, which requires the attacker to circumvent that firewall before attacking the protected services. Adding a firewall makes the attackers' job more difficult since the attacker also needs to bypass the firewall in order to launch successful attacks.

- **Adding detection mechanisms** A detection mechanism, with a particular detection rate, can be added to the privilege conditions of each host to prevent attackers from gaining additional privileges. Even though detection mechanisms do not directly improve the security metrics that are based on the shortest attack paths (e.g., $k0d$ and $kSupply$), they may reduce the probability of the attacker gaining additional privileges as the attacker attempts to evade the detection [167].

- **Patching known vulnerabilities** Known vulnerabilities (e.g., $\langle V\_Known\_1, Attacker, 6\rangle$ in Fig. 5.4) can be removed by applying relevant security patches or upgrading the system [171].

- **Diversifying components** Utilizing a diverse set of vendors or different service instances of the same vendor is an effective hardening option against known, zero-day, and supply chain vulnerabilities. One reason is that different instances are less likely to have shared vulnerabilities that attackers can exploit to compromise multiple hosts running those instances [46].

- **Disabling services** Disabling an unnecessary service may improve the security posture of the substation [171]. To illustrate, in Fig. 5.4, by disabling $HTTP$ service in HMI, we can remove the zero-day vulnerability $\langle V\_HTTP, 6, 1\rangle$. This hardening option can be utilized when it does not affect the overall functionality of the substation, according to the operator.

- **Relocating services between hosts** Relocating services (e.g., moving a critical service to a host that is protected by a firewall) may improve the security posture of the substation [110]. This hardening option can be utilized when it does not affect the overall functionality of the substation, according to the operator.

Each hardening option comes with a monetary cost. An operator can specify the allowed budgets and various costs (e.g., to maintain, replace, and repair devices) associated with hardening options, e.g., based on Gartner's total cost of ownership [172]. Although calculating the exact cost of each hardening option may not be feasible before hardening, even a rough estimate is usually helpful in identifying and choosing the proper hardening options. For example, the operator can estimate that adding a firewall costs around 5300, as purchasing a firewall costs around 5000 dollars[3], and the labor costs around 300.

Additionally, an operator may need to consider the operational constraints. For instance, in some cases, an operator may be bound to a specific vendor, which prevents him/her from using devices from other vendors. By considering hardening options, costs, and constraints, we can formulate the hardening as a multi-objective optimization problem as follows:

[3] Network Firewall Pricing Comparison https://www.trustradius.com/buyer-blog/network-firewall-pricing-comparison [Online; Accessed 29-May-2021]

$$
\begin{aligned}
\max \quad & (\alpha_1 \times k0d) + (\alpha_2 \times kSupply) \\
\min \quad & OC \\
\text{s.t.} \quad & \alpha_1 + \alpha_2 = 1.0 \\
(1) \quad & OC \leq Q \\
(2) \quad & COF \leq Q_f \\
(3) \quad & NOF \leq F_{limit} \\
(4) \quad & \forall f \epsilon F, f \notin CTCF \\
(5) \quad & COD \leq Q_d \\
(6) \quad & NOD \leq D_{limit} \\
(7) \quad & \forall d \epsilon D, d \notin HWCBWD \\
(8) \quad & COP \leq Q_p \\
(9) \quad & NOP \leq P_{limit} \\
(10) \quad & \forall p \epsilon P, p \notin KVCBP \\
(11) \quad & CODIV \leq Q_{div} \\
(12) \quad & NODIV \leq DIV_{limit} \\
(13) \quad & \forall div \epsilon DIV, div \notin STCBD \\
(14) \quad & COS \leq Q_s \\
(15) \quad & NOS \leq S_{limit} \\
(16) \quad & \forall s \epsilon S, s \notin SCVTCM \\
(17) \quad & COM \leq Q_{mov} \\
(18) \quad & CODIS \leq Q_{dis}
\end{aligned}
$$

Specifically, the objective functions are to maximize the weighted average of metrics (specified after the term max in the optimization formula) and minimize the overall cost (OC, specified after the term min in the optimization formula) according to budget and operational constraints where α_1 and α_2 are the weights for $k0d$ and $kSupply$, respectively, and represent the relative importance of each metric in improving the security posture. The optimization formula also provides the following list of constraints. It allows the operator to select those relevant to their substation according to the available budget and operational constraints.

(1) Overall cost (OC) cannot exceed Q.

(2) Cost of the added firewalls (COF) cannot exceed Q_f.

(3) Number of the added firewalls (NOF) cannot exceed F_{limit}.

(4) None of the connectivity conditions that are protected by firewalls (F) can be in the set of connections not intended to be protected by a firewall ($CTCF$).

(5) Cost of the added detection mechanisms (COD) cannot exceed Q_d.

(6) Number of the added detection mechanisms (NOD) cannot exceed D_{limit}.

(7) None of the hosts with the added detection (D) can be in the set of hosts not intended to have a detection mechanism ($HWCBWD$).

(8) Cost of patching (COP) cannot exceed Q_p.

(9) Number of applied patches (NOP) cannot exceed P_{limit}.

(10) None of the known vulnerabilities that are patched (P) can be in the set of known vulnerabilities that do not currently have a patch ($KVCBP$).

(11) Cost of diversity ($CODIV$) cannot exceed Q_{div}.

(12) Number of diversified services ($NODIV$) cannot exceed DIV_{limit}.

(13) None of the diversified services (DIV) can be in the set of services not intended to be diversified ($STCBD$).

(14) Cost of supply chain-related hardening options (COS) cannot exceed Q_s.

(15) Number of applied supply chain-related hardening options (NOS) cannot exceed S_{limit}.

(16) None of the supply chain exploits for which supply chain-related hardening options are applied (S) can be in the set of supply chain exploits that cannot be modified ($SCVTCM$).

(17) Cost of moving services (COM) cannot exceed Q_{mov}.

(18) Cost of disabling services ($CODIS$) cannot exceed Q_{dis}.

The hardening problem is known to be NP-hard [103], so we need to utilize a heuristics-based optimization algorithm to solve this problem rather than a search-based one. To solve this optimization problem, we leverage Pymoo [173] (an open-source Python optimization library) and its implemented Non-dominated Sorting Genetic Algorithm (NSGA)-II. We chose NSGA-II as it is a multi-objective optimization algorithm, and it does not require reference points to start the optimization. NSGA-II provides the Pareto front of the hardening problem as a set of results in which each result is not better than any other results in terms of both metrics and costs. To adopt NSGA-II, we need to represent each hardening solution as a gene (a vector of integers representing a possible solution). To this end, we first represent the set of all possible ways of hardening a substation as a *Hardening Dictionary* [6]. For instance, Table 5.1 shows the Hardening Dictionary for the attack graph in Fig. 5.4 with three connections, four hosts, two known vulnerabilities, six services running on hosts, and three supply chain exploits, the sum of which corresponds to 18 rows of the Hardening Dictionary. A hardening vector [6] refers to one possible solution according to a given Hardening Dictionary. For instance, the hardening vector $[True, 0, 0, True, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$ refers to protecting the connectivity condition $\langle 6, 1\rangle$ by a firewall and adding a detection mechanism to $host6$.

For choosing the coefficients α_1 and α_2, according to Fig. 4.14, as the number of autonomous supply chain vulnerabilities increases, the difference between $k0d$ and $kSupply$ increases. This implies that operators should first estimate the percentage of devices which may contain autonomous supply chain vulnerabilities. If the percentage of those autonomous supply chain vulnerabilities is more than attacker-activated ones, α_2 should be greater than α_1 (i.e., the operator should focus on $kSupply$ more than $k0d$). If the percentage of those autonomous supply chain vulnerabilities is less than attacker-activated ones, α_1 should be larger than α_2 (i.e., the operator should focus on $k0d$ more than $kSupply$). If the operator is not able to estimate the percentages for autonomous and attacker-activated supply chain exploits, they can assign equal weights to α_1 and α_2 (i.e., the operator focuses on both metrics equally).

Index	Explanation	Possible Values	Hardening Option Category
0	Place connection $\langle 6, 1\rangle$ behind a firewall?	True or False	Adding firewalls
1	Place connection $\langle 1, 2\rangle$ behind a firewall?	True or False	
2	Place connection $\langle 2, 7\rangle$ behind a firewall?	True or False	
3	Add detection on $host6$?	True or False	Adding detection mechanisms
4	Add detection on $host1$?	True or False	
5	Add detection on $host2$?	True or False	
6	Add detection on $host7$?	True or False	
7	Patch the known vulnerability $\langle V\_Known\_1, Attacker, 6\rangle$?	True or False	Patching known vulnerabilities
8	Patch the known vulnerability $\langle V\_Known\_2, 6, 1\rangle$?	True or False	
9	Replace GW on $host6$ with which instance (D) from which vendor (S)?	Keep, (D_2, S_1), (D_1, S_2), (D_1, S_3), (D_2, S_3), (D_1, S_4)	Diversifying services
10	Replace $HTTP$ on $host1$ with which instance (D) from which vendor (S)?	Keep, (D_1, S_7), (D_1, S_8), (D_1, S_10)	
11	Replace SSH on $host1$ with which instance (D) from which vendor (S)?	Keep, (D_1, S_11), (D_1, S_13), (D_1, S_14)	
12	Replace $SCADA$ on $host2$ with which instance (D) from which vendor (S)?	Keep, (D_1, S_16), (D_1, S_17), (D_1, S_3)	
13	Replace $HTTP$ on $host2$ with which instance (D) from which vendor (S)?	Keep, (D_1, S_7), (D_1, S_8), (D_1, S_10)	
14	Replace $TRIP$ (Protection IED) on $host7$ with which instance (D) from which vendor (S)?	Keep, (D_2, S_1), (D_1, S_2), (D_1, S_3), (D_2, S_3), (D_1, S_4)	
15	Apply which supply chain-related hardening option for $\langle V\_SupplyChain\_1, 6, 6\rangle$?	None, Hardware Update, Firmware Update	Supply Chain-Related Hardening Options
16	Apply which supply chain-related hardening option for $\langle V\_SupplyChain\_2, 1, 1\rangle$?	None, Hardware Update, Firmware Update	
17	Apply which supply chain-related hardening option for $\langle V\_SupplyChain\_3, 2, 2\rangle$?	None, Hardware Update, Firmware Update	

Table 5.1: Hardening Dictionary for the Attack Graph in Fig. 5.4

Bayesian Network Generation (BNG)

In addition to finding the shortest attack paths, attack graphs can be leveraged to calculate the probability of compromising each host [46]. To this end, the BNG module first builds a Bayesian network with the same structure as a given attack graph (Chapter 4). Next, it calculates the probability of compromising each host based on the obtained Bayesian network.

Markov Decision Process Generation (MDPG)

Although attack graphs can aid in improving the security posture by identifying the shortest attack paths leading to compromising critical hosts, they do not consider the physical damage an attacker can cause. To address this, MDPG leverages MDP [74, 1, 78] to model the attacker as an agent aiming to maximize the potential physical damage. A typical MDP consists of sets of states,

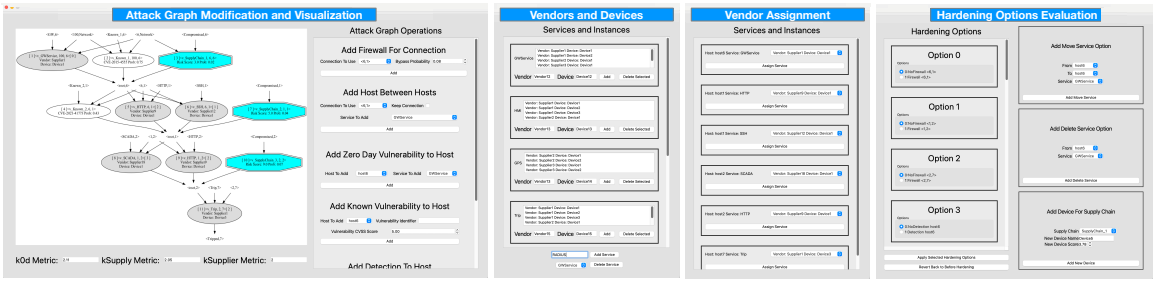


Figure 5.6: AGMV, VDDS, VA, and HOE Modules of HFS

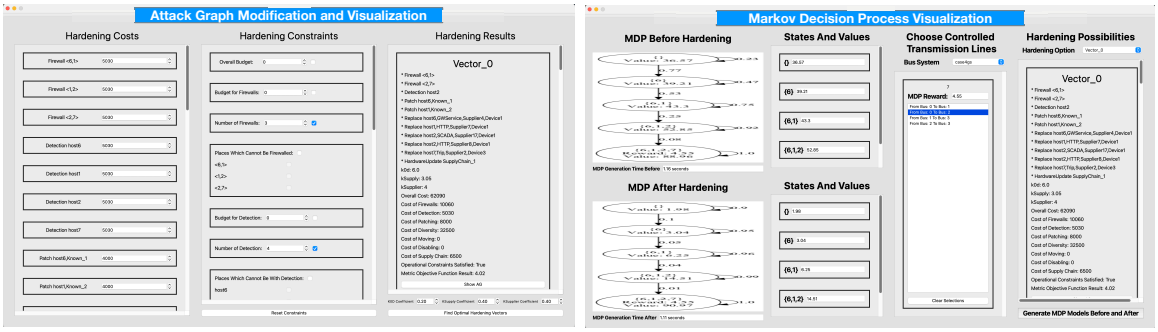


Figure 5.7: OH and MDPV Modules of HFS

actions, probability of transitions between states, rewards for each state, and a discount factor [168]. In order to identify critical security states the attacker can be in, the MDPG module first generates an MDP based on the attack graph, where states represent sets of compromised hosts and transitions represent compromising one more host in addition to previously compromised hosts by exploiting vulnerabilities. Transition probabilities are obtained by converting the attack graph into a Bayesian network and performing Bayesian inference. Next, MDPG calculates the rewards based on the total load on transmission lines using a reward function [74]. To analyze the impact of hardening on potential physical damages, the operator can evaluate the MDP model generated before and after hardening for different optimal hardening vectors previously identified by the NH module.

5.2.3 Hardening Visualization (HV)

In this section, we discuss how the Hardening Visualization (HV) module enables an operator to interact with HFS.

Attack Graph Modification and Visualization (AGMV)

This module allows the operator to load a basic attack graph (e.g., Fig. 5.3) and extend it to model and visualize potential attacks against their substation. We show a screenshot of the AGMV interface in Fig. 5.6. Through AGMV, the operator first provides an XML file as the input for generating a basic attack graph model (e.g., the XML file in Fig. 5.3 (left)) and then interactively modifies this basic configuration using attack graph operations described in Section 5.2.2.

Vendors and Devices Diversity Specification (VDDS)

This module allows the operator to manage the list of possible instances and vendors for each service. Such a list is used by the hardening option *Diversifying Components*. We show a screenshot of the VDDS interface in Fig. 5.6. Through VDDS, the operator can add a new service, add a new vendor and instance for an existing service, delete an existing instance of a service, and delete an existing service.

Vendor Assignment (VA)

This module allows the operator to assign different service instances to each host based on the list of vendors and devices provided to the VDDS module. We show a screenshot of the VA interface in Fig. 5.6. After assigning a new service instance to a host, VA automatically triggers the MC module to update the values of security metrics. For instance, once the operator changes the instance ($Supplier9$, $Device1$) of $HTTP$ service in $host1$ to instance ($Supplier7$, $Device1$), the value of $k0d$ increases from 2.11 to 3.11.

Hardening Options Evaluation (HOE)

This module allows the operator to interactively apply hardening vectors according to the Hardening Dictionary generated by the NH module (such as Table 5.1). We show a screenshot of HOE in Fig. 5.6. The operator can select and apply any hardening vector in the Hardening Dictionary and evaluate its impact on the value of security metrics. For instance, to evaluate the impact of the hardening vector $[True, 0, 0, True, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$, the operator selects *Firewall*

$\langle 6, 1 \rangle$ and *Detection* $host6$ from the list of hardening options. In addition to possible hardening options, the operator can add other options for disabling a service on a host, moving a service from one host to another, and adding more potential vendors for the supply chain-related option, replacing vendors.

Optimal Hardening (OH)

This module allows the operator to specify the cost of different hardening options and constraints (e.g., the operator may specify a total available budget in dollars) and the weights for different security metrics as mentioned in Section 5.2.2. We show a screenshot of the OH module in Fig. 5.7. The Optimal Hardening (OH) module identifies the optimal hardening vectors based on the specified values and visualizes them along with their corresponding hardening options and the cost of the hardening vectors and their hardening options. Additionally, OH shows the calculated values of security metrics for optimal hardening vectors and whether the hardening option satisfies the specified constraints.

Markov Decision Process Visualization (MDPV)

This module enables evaluating the effectiveness of hardening against potential physical damages. We show a screenshot of MDPV in Fig. 5.7. Through MDPV, the operator selects the bus system for calculating the load on transmission lines [169]. Next, the operator specifies the set of transmission lines controlled by the sink host of the attack graph (e.g., $host7$ in Fig. 5.4) and selects any of the identified optimal hardening vectors. Accordingly, MDPV visualizes the MDP models generated before and after hardening.

5.3 Hardening Use Cases

In this section, we illustrate how HFS can be applied to harden substations under six scenarios with different budget and operational constraints. For all scenarios, HFS can effectively assist the analyst in finding optimal hardening vectors for the substation.

5.3.1 No Constraints

In this scenario, an operator with a sufficient budget and no particular operational constraints aims to improve the substation's security posture against attackers capable of exploiting supply chain vulnerabilities (i.e., supply chain attackers). The operator first loads the attack graph using the AGMV module, as shown in Fig. 5.4, and subsequently applies the OH module to harden the substation against supply chain attackers. Suppose the operator is more concerned about supply chain attackers than non-supply chain attackers, so he specifies 0.20 for α_1 (weight of $k0d$) and 0.80 for α_2 (weight of $kSupply$). NH identifies the Pareto front of the hardening problem in which no solution is better than others regarding the metric objective and the overall cost. Among those solutions, suppose the operator selects the one providing the highest possible value of the metric objective function, despite its potentially higher cost since the operator is more concerned about the security posture at this stage. As such, the operator chooses the following hardening vector:

- Protect connections $\langle 6, 1\rangle$, $\langle 1, 2\rangle$, $\langle 2, 7\rangle$ by firewalls.
- Patch known vulnerabilities $\langle V\_Known\_1, Attacker, 6\rangle$ and $\langle V\_Known\_2, 6, 1\rangle$.
- Replace the $HTTP$ service instance on $host2$ with another service instance.
- Apply firmware update on $\langle V\_SupplyChain\_3, 2, 2\rangle$.

Such a hardening vector increases $k0d$ and $kSuppy$ from 2.11 to 7.0 and from 2.05 to 3.11, respectively. Fig. 5.8 shows the attack graph after applying this hardening vector and the shortest attack paths.

5.3.2 Overall Cost Constraint

In this scenario, an operator with a limited budget aims to improve the security posture against supply chain attackers. Assuming that the cost of protecting each network connection by a firewall is around $5,300$ and applying a firmware update costs around $6,500$ dollars[4], the operator specifies the overall allowed cost to be $13,000$ dollars, which is slightly greater than the cost of protecting

[4] How much does a server cost for a small business? https://www.serverpronto.com/spu/2019/04/how-much-does-a-server-cost-for-a-small-business/ [Online; Accessed 29-May-2021]

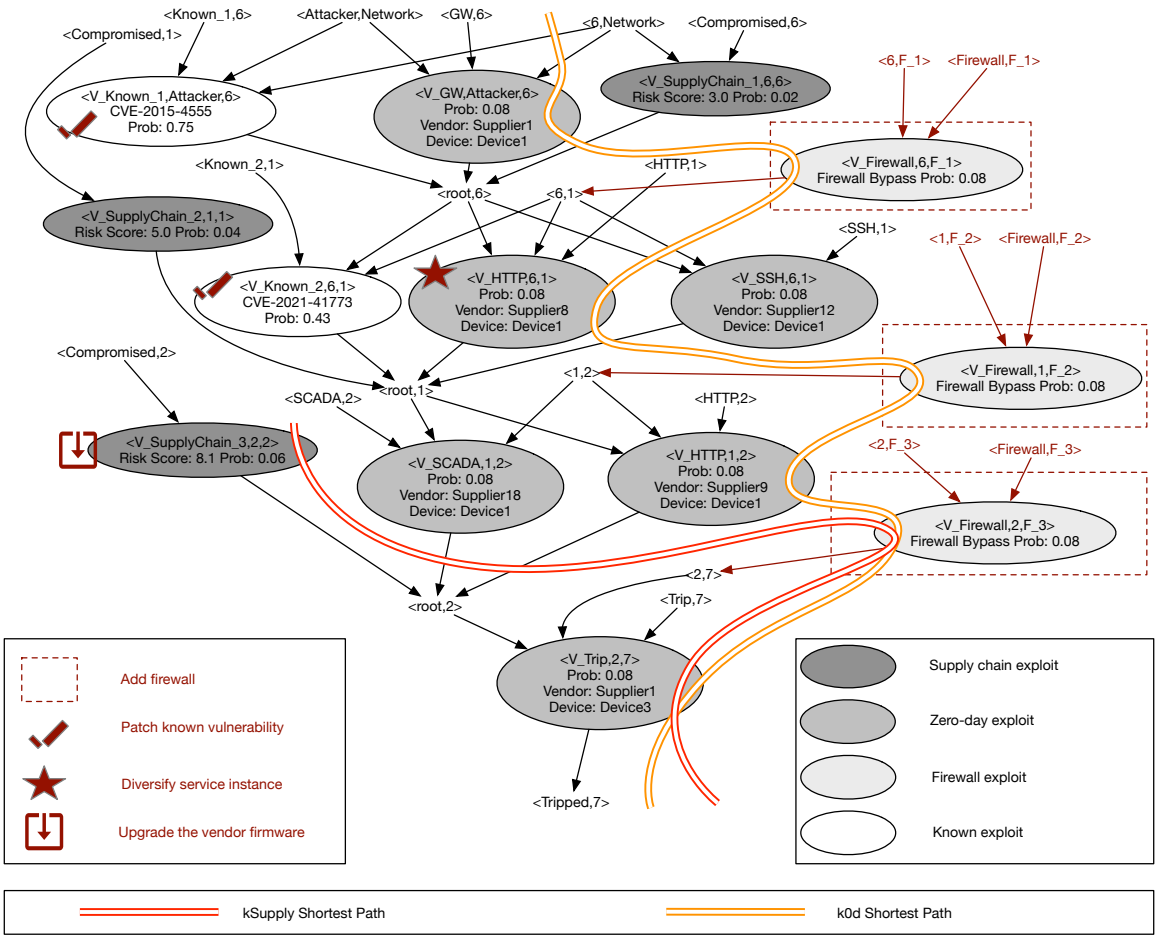


Figure 5.8: Hardening for the Attack Graph in Fig. 5.4 without Constraints

two network connections by firewalls. Accordingly, NH calculates the following hardening vector as the solution:

- Protect the connection $\langle 1, 2\rangle$ by a firewall.
- Apply firmware update on $\langle V\_SupplyChain\_3, 2, 2\rangle$.

Applying such a hardening vector increases $k0d$ and $kSupply$ from 2.11 to 3.11 and from 2.05 to 2.11, respectively. Considering the cost of adding a firewall and updating the firmware, the cost of applying this hardening vector is $11,800$ dollars, which is below the overall budget that the operator specifies. Fig. 5.9 shows the attack graph after applying this hardening vector with the shortest attack paths.

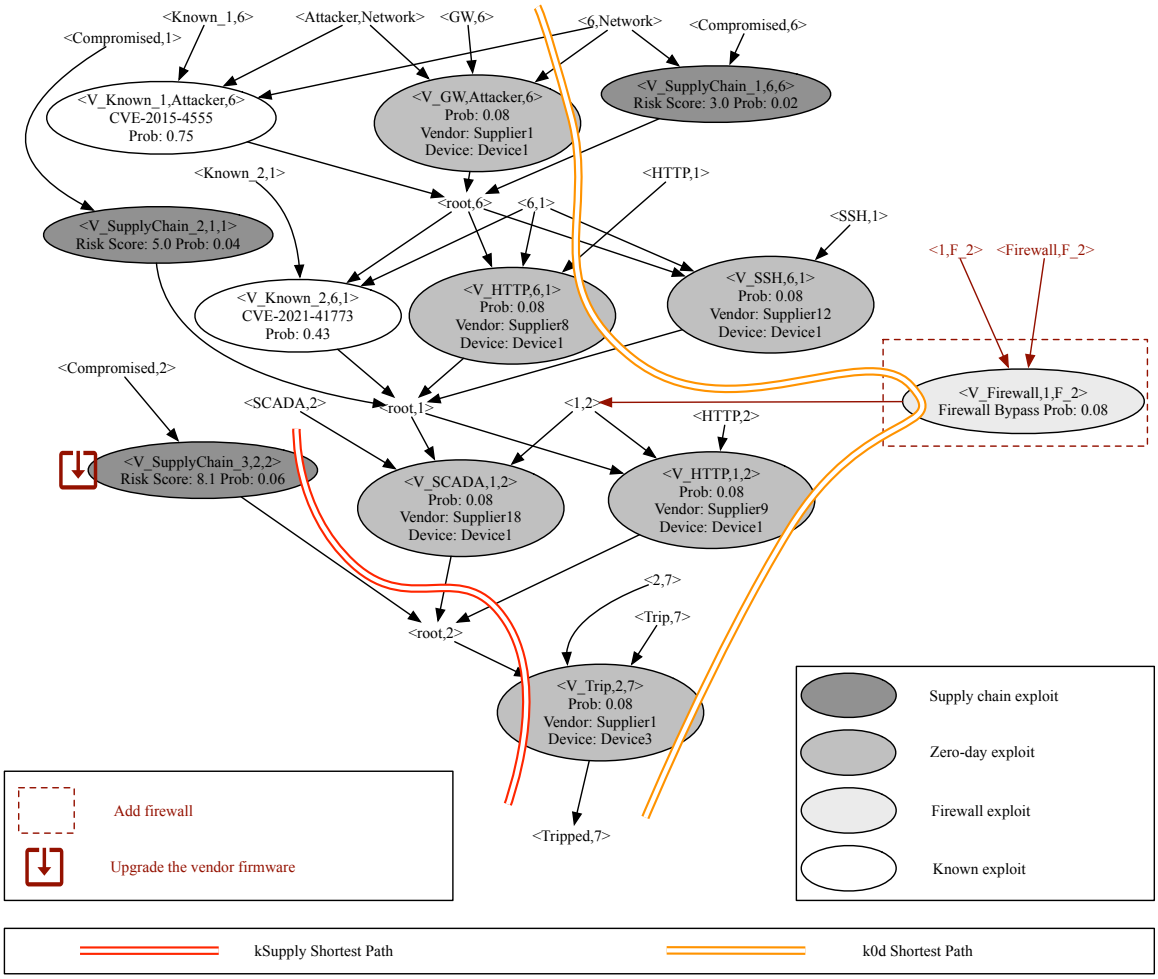


Figure 5.9: Hardening for the Attack Graph in Fig. 5.4 with Overall Cost Constraint

5.3.3 Limited Number of Firewalls

In this scenario, the operator aims to increase the security posture against supply chain attackers with the constraint that only one connection can be protected by a firewall due to the limited availability of technicians capable of installing and maintaining firewalls. The operator does not input any cost constraints. Therefore, he/she specifies the number of firewalls (i.e., one) as a constraint through the OH module. Accordingly, NH identifies the following hardening vector:

- Protect the connection $\langle 2, 7 \rangle$ by a firewall.
- Patch known vulnerabilities $\langle V\_Known\_1, Attacker, 6 \rangle$ and $\langle V\_Known\_2, 6, 1 \rangle$.
- Replace HTTP service instance on $host2$ instance with another service instance.
- Apply firmware update on $\langle V\_SupplyChain\_3, 2, 2 \rangle$.

Applying such a hardening vector increases $k0d$ and $kSupply$ from 2.11 to 5.0, and from 2.05 to 3.11, respectively. Fig. 5.10 shows the attack graph after applying this hardening vector and the shortest attack paths. In this scenario, the NH module successfully identifies the most critical connection (i.e., $\langle 2, 7 \rangle$) that a firewall should protect, and thus, an attacker that exploits the supply chain vulnerability $\langle V\_SupplyChain\_3, 2, 2 \rangle$ can no longer trip the circuit breakers (i.e., the goal condition $\langle Tripped, 7 \rangle$ in Fig. 5.10) through the connection $\langle 2, 7 \rangle$.

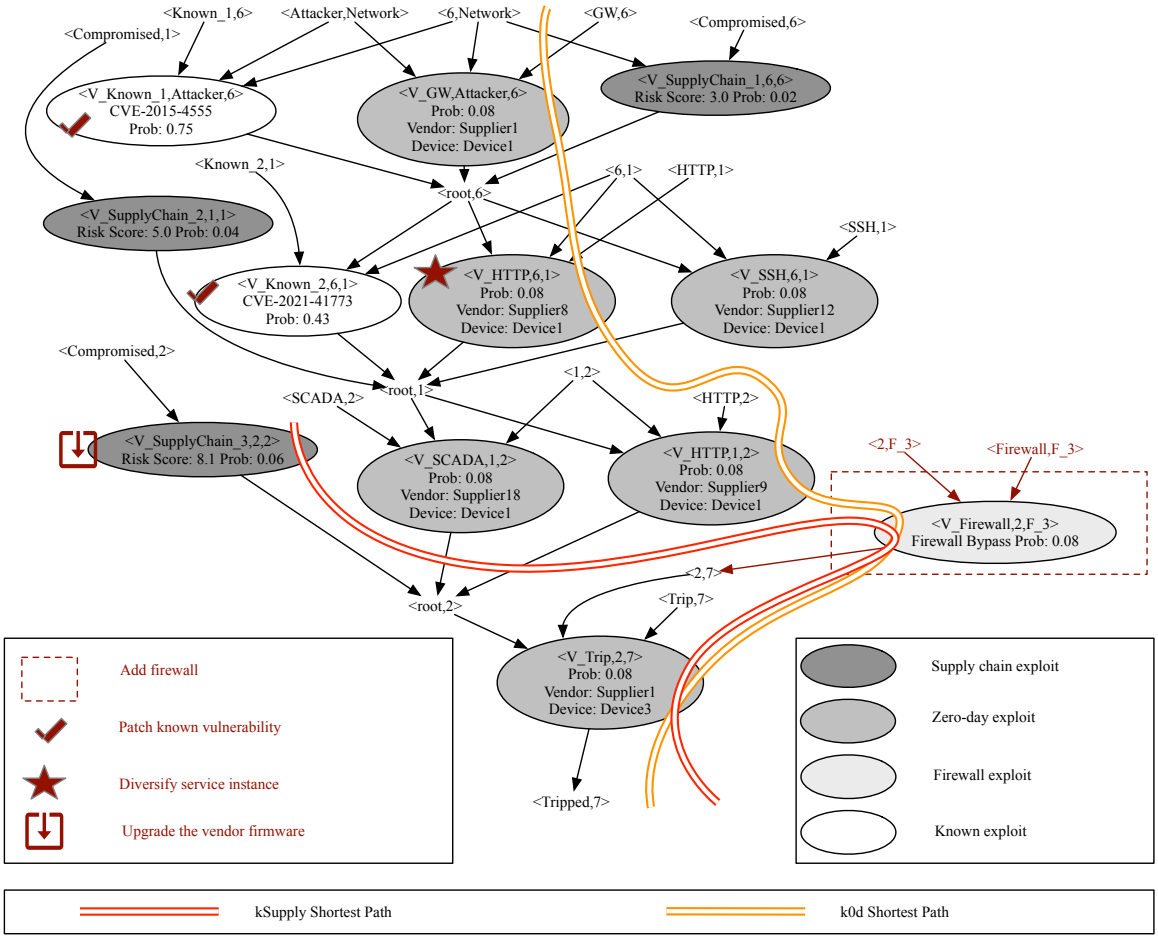


Figure 5.10: Hardening for the Attack Graph in Fig. 5.4 with One Firewall

5.3.4 Limited Budget for Diversifying Services

In this scenario, the operator would like to adopt different HTTP services from different vendors on $host1$ and $host2$ to improve the security posture in terms of the $k0d$ metric. However, suppose there are several vendors providing the same service, namely $HTTP$, but with different costs. The operator applies the NH module to identify the most optimal hardening vector in this case. Assuming that the costs are 6400, 6300, and 6200 dollars (the average cost is around $6,500$ [4]) for replacing the vendor of HTTP service running on $host2$ with $Supplier7$, $Supplier8$, and $Supplier10$, respectively. Next, the operator specifies the available budget for hardening (i.e., $6,400$ dollars) as a hardening constraint. Accordingly, NH identifies the following hardening vector:

- Protect connections $\langle 6, 1\rangle$, $\langle 1, 2\rangle$, and $\langle 2, 7\rangle$ by firewalls.

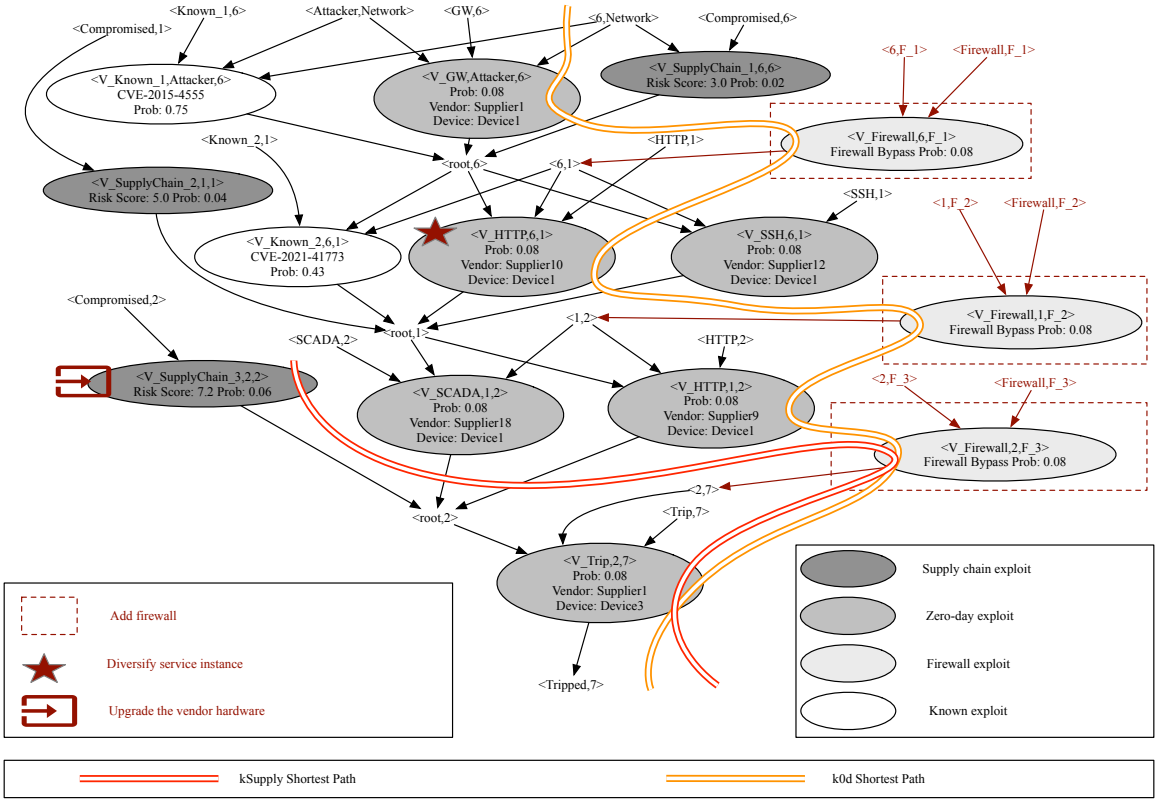


Figure 5.11: Hardening for the Attack Graph in Fig. 5.4 with Limited Diversity Budget

- Replace HTTP service instance on $host2$ with the instance $Device1$ from $Supplier10$.
- Apply firmware update on $\langle V\_SupplyChain\_3, 2, 2\rangle$.

Applying such a hardening vector increases $k0d$ and $kSupply$ from 2.11 to 5.45 and from 2.05 to 3.11, respectively. Fig. 5.11 shows the attack graph after applying this hardening vector and the shortest attack paths.

5.3.5 No Available Supply Chain-Related Hardening Options

In this scenario, the operator aims to improve the security posture of a substation against supply chain attackers (i.e., improving the value of kSupply) while there is no available supply chain-related hardening options due to operational constraints. To this end, the operator first specifies α_1

(weight of $k0d$) as 0.0 and α_2 (weight of $kSupply$) as 1.0. Next, the operator specifies the number of available supply chain-related hardening options (i.e., none) as a constraint. This implies that only non-supply chain-related hardening options can be applied to the substation. Accordingly, NH identifies the following hardening vector:

- Protect the connection $\langle 2, 7 \rangle$ by a firewall.

Applying such a hardening vector increases $kSupply$ from 2.05 to 3.05. Fig. 5.12 shows the attack graph after applying this hardening vector and the shortest attack paths.

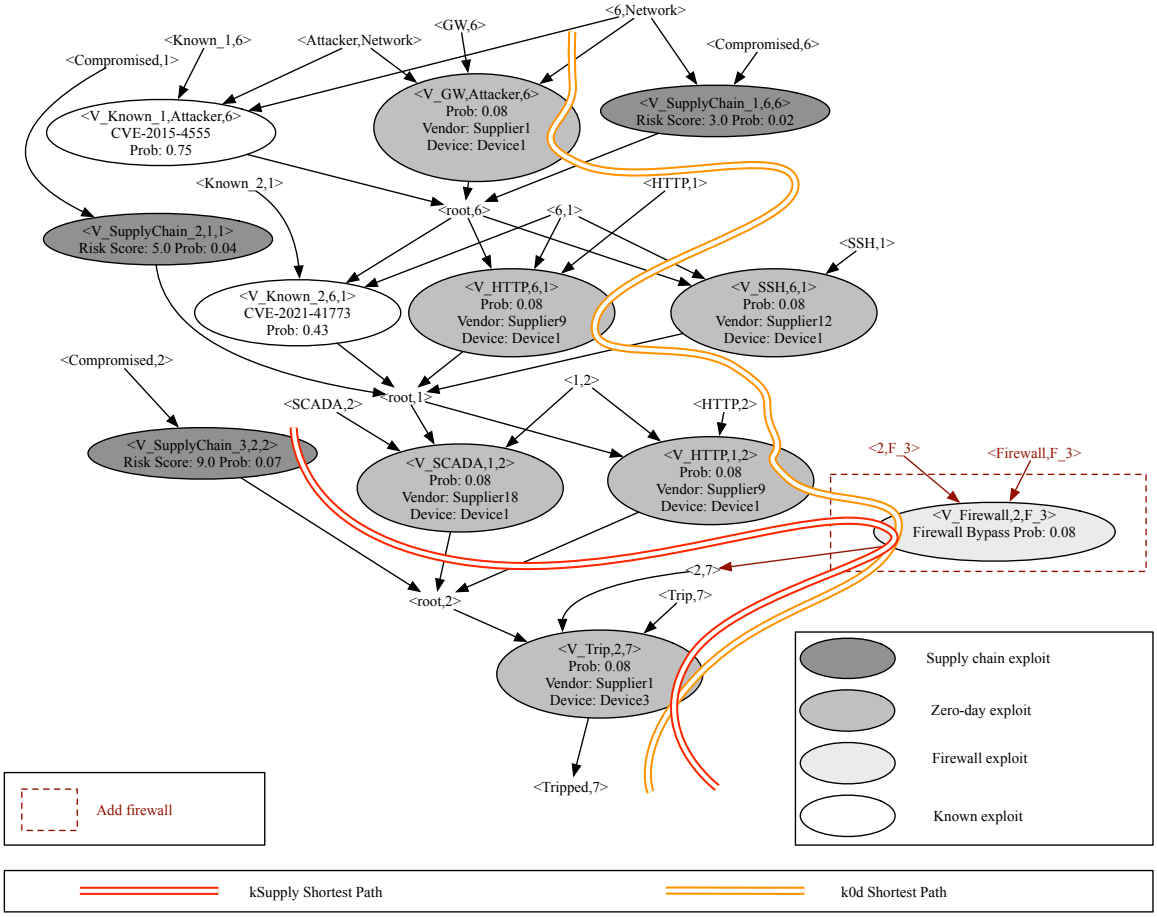


Figure 5.12: Hardening for the Attack Graph in Fig. 5.4 without Supply Chain Options

5.3.6 Limited Supply Chain-Related Hardening Options

In this scenario, the operator aims to improve the security posture only against supply chain attackers (α_1 is 0.0 and α_2 is 1.0). At the same time, he/she realizes that only one supply chain-related hardening option can be utilized due to operational constraints. Therefore, the operator specifies the number of supply chain-related hardening options (i.e., one) as a constraint through the OH module. Accordingly, NH identifies the following hardening vector.

- Protect the connection $\langle 2, 7 \rangle$ by a firewall.
- Apply firmware update on $\langle V\_SupplyChain\_3, 2, 2 \rangle$.

Applying such a hardening vector increases $kSupply$ from 2.05 to 3.11. Fig. 5.13 shows the attack graph after applying this hardening vector and the shortest attack paths.

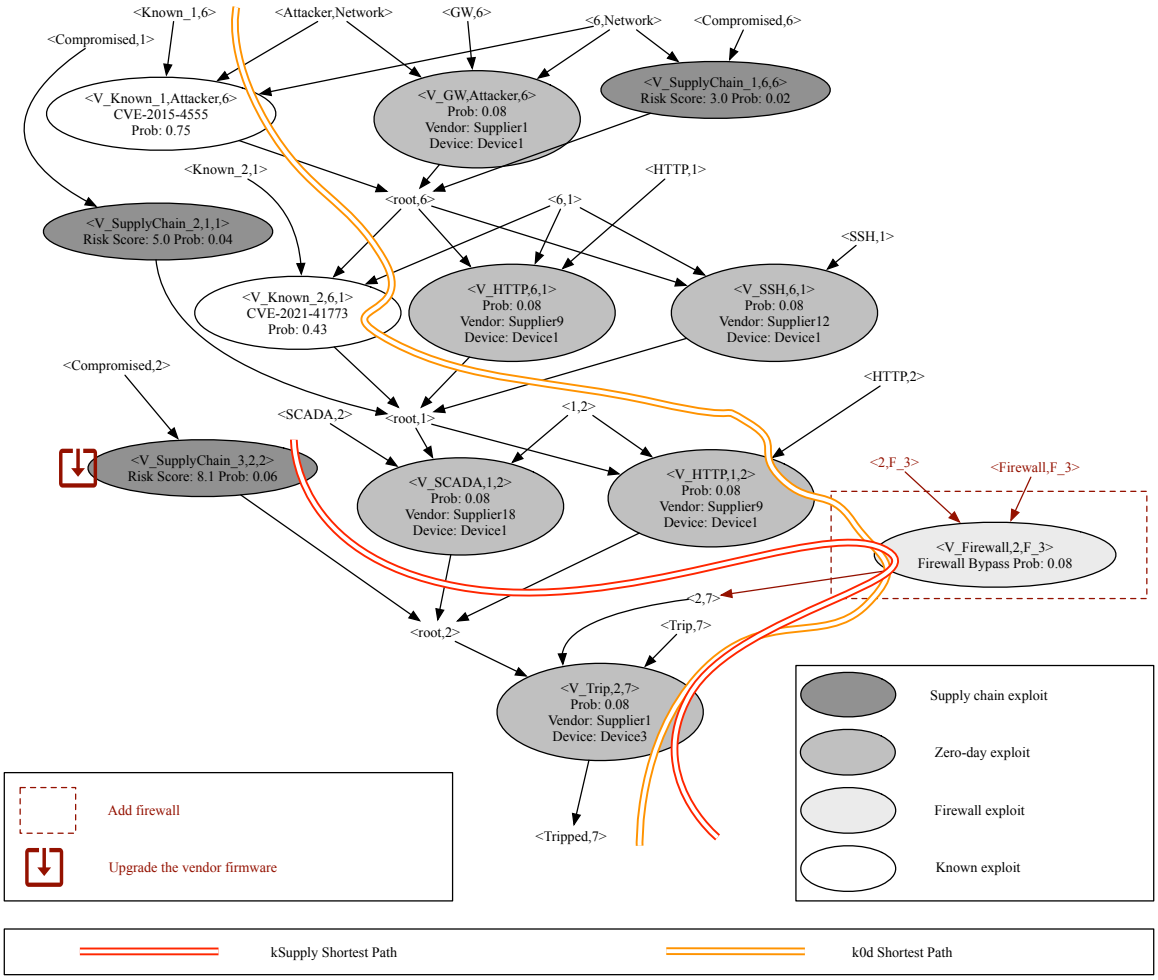


Figure 5.13: Hardening for the Attack Graph in Fig. 5.4 with One Supply Chain Option

5.4 Simulations

To evaluate HFS, we seek to answer the following research questions:

- RQ1: How effective is HFS in improving the security posture of substations?
- RQ2: What are the optimum parameters for NSGA-II multi-objective optimization in HFS?
- RQ3: How scalable is HFS in terms of the latency introduced by its different modules?

Experimental setup

We perform all simulations on a server with 64 cores and 755 GB memory equipped with Intel (R) Xeon(R) Gold 5218 processors @2.30 GHz. To simulate the effectiveness of HFS, we provide

the attack graph in Fig. 5.4 as the seed graph and generate 2400 attack graphs using attack graph operations in the AGG module (Section 5.2.2). Since a typical substation may contain up to 30 hosts according to the substation design provided by the authors in [167], we simulated up to 30 hosts. After that, we report the average results.

5.4.1 Effectiveness Evaluation

To answer RQ1, we evaluate the effectiveness of our approach based on the scenarios detailed in Section 5.3.

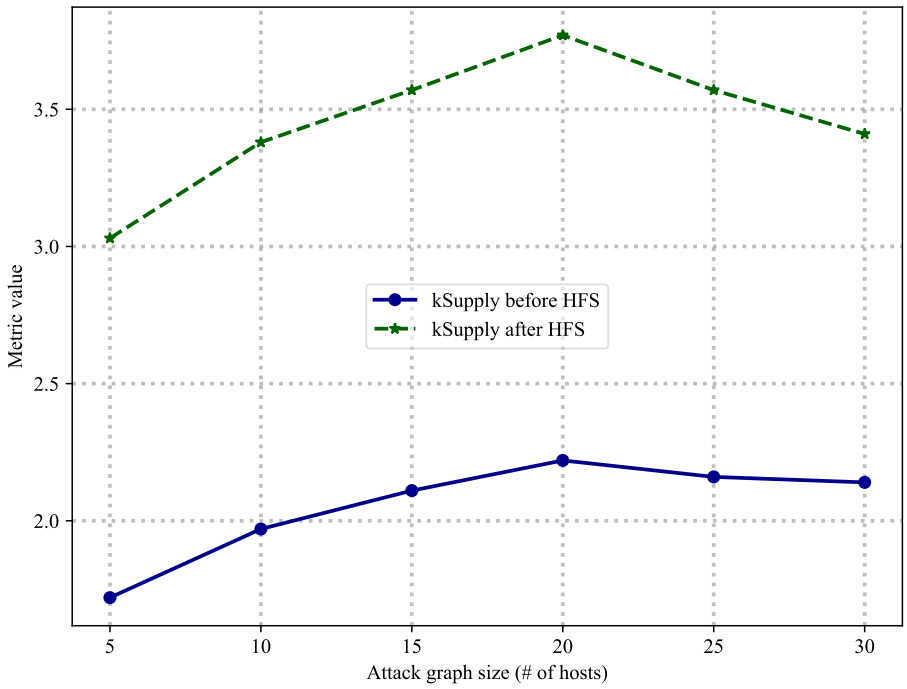


Figure 5.14: Effectiveness of NH in Scenarios without Constraints

No constraints Fig. 5.14 shows that HFS can significantly improve the value of the $kSupply$ metric when there are no constraints. $kSupply$ is greater than 3.0 for all sizes of attack graphs after applying HFS (compared to the initial value of around 2.30). We can also see a minor variation in the value of $kSupply$ as the size of the attack graphs increases. This variation can be explained by the possibility that the number and location of autonomous supply chain vulnerabilities do not depend on the size of substations.

Overall cost constraint To measure the effectiveness of our approach under overall cost constraint (OC), we first perform the simulation without constraints and record the average cost of all solutions in the Pareto front as the overall cost constraint. This allows us to specify a relevant cost constraint for each substation since considering the same cost constraint for substations of different sizes may not be realistic.

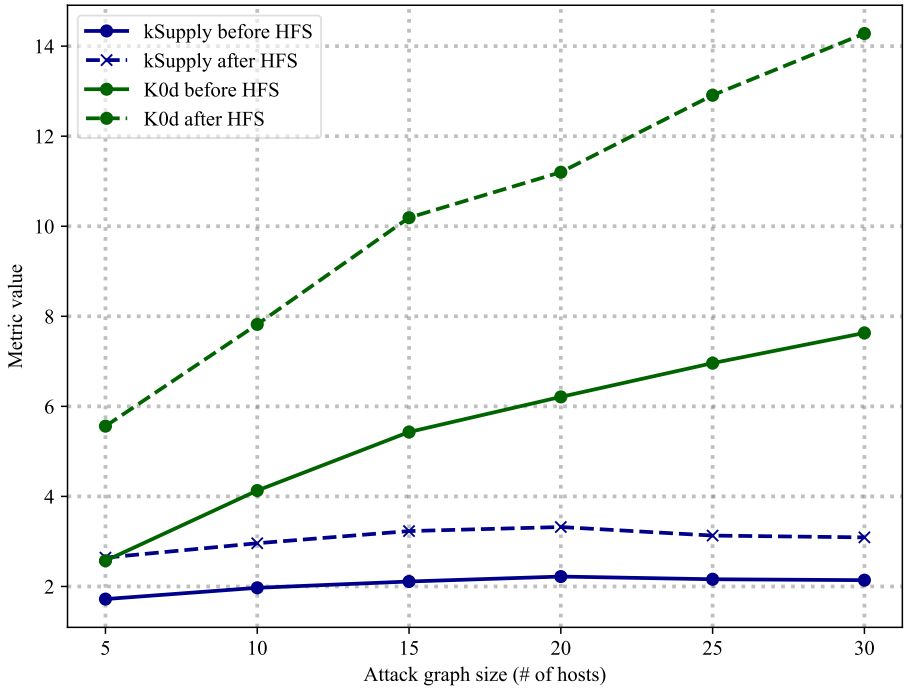


Figure 5.15: Effectiveness of NH in Scenarios with Overall Cost Constraint

Fig. 5.15 shows that HFS improves both $k0d$ and $kSupply$ values, while there is a minor variation in the value of $kSupply$ with the size of the attack graph. $k0d$ improves more significantly with the size of the attack graph due to more possible hardening vectors in larger substations.

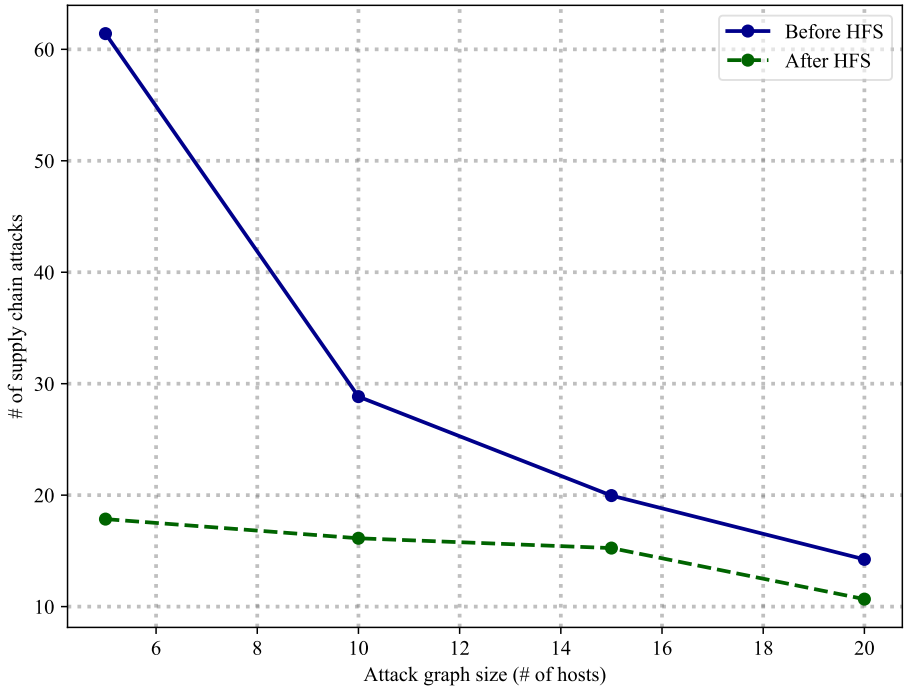


Figure 5.16: Effectiveness of NH in Scenarios with One Available Firewall

Limited number of firewalls Fig. 5.16 demonstrates that our approach is effective against supply chain attackers even when only one connectivity condition can be protected by a firewall. The improvement is more significant for small substations (such as ones with up to 10 hosts). The less significant improvement for larger substations can be explained by the possibility that larger substations require more firewalls to improve their security posture significantly.

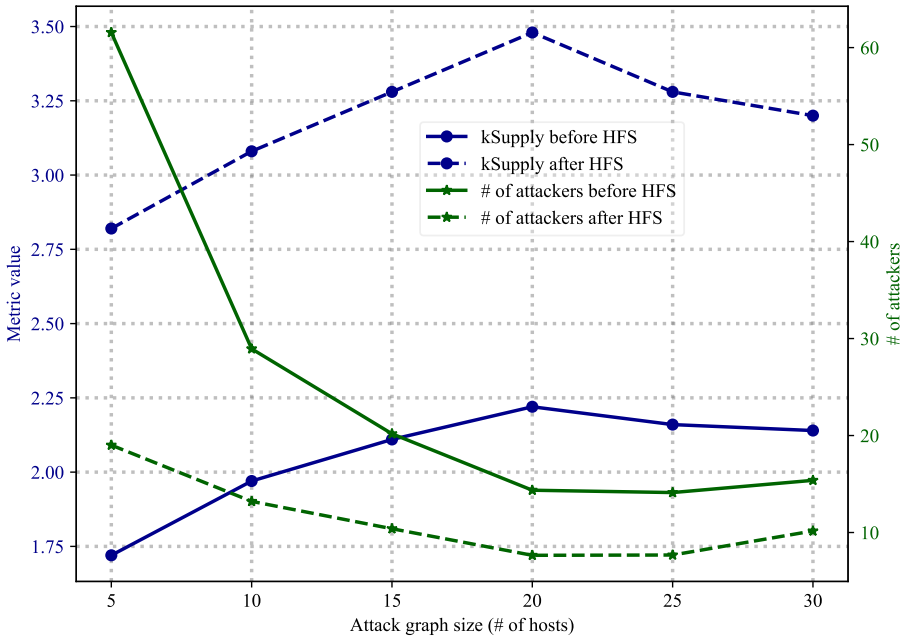
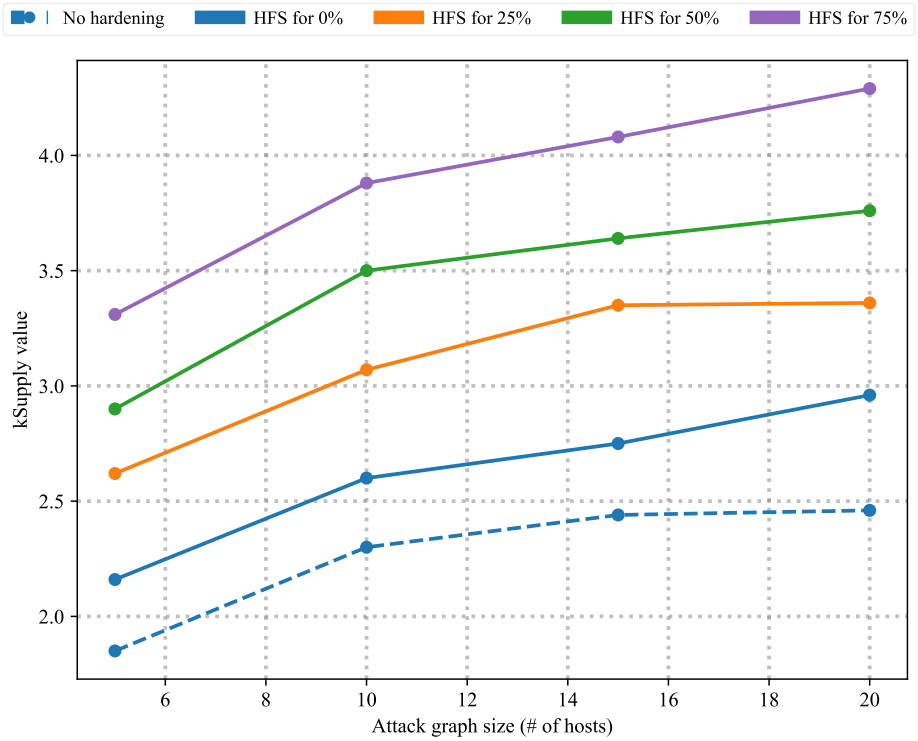
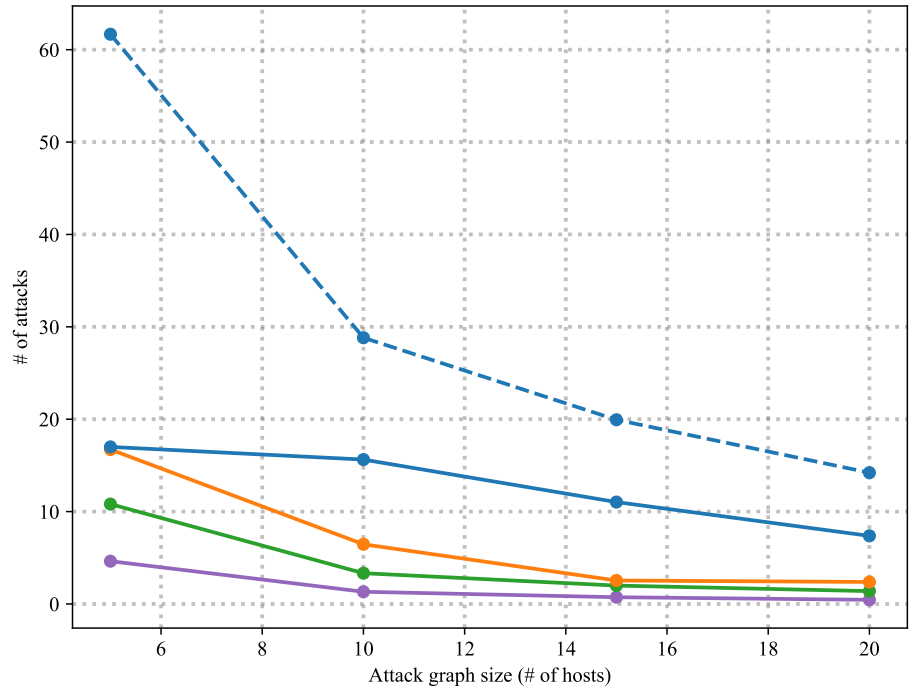


Figure 5.17: Effectiveness of NH in Scenarios with Limited Budget for Diversity

Limited budget for diversifying services. Fig. 5.17 demonstrates that our approach significantly improves the value of $kSupply$ even with a limited available budget for diversifying services. The value of $kSupply$ is greater than 3.0 for substations with more than 10 hosts. Our approach is more effective against supply chain attackers in smaller substations (reducing the number of successful attacks from 60 to 20). Although the need for diversifying more services in larger substations does not allow an equally significant reduction, HFS still prevents around half of supply chain attackers (i.e., reducing the number of successful attacks from around 20 to 10 in the largest substation).



(a) Effectiveness of NH in Scenarios With Different Percentages of Supply Chain-Related Exploits That Can Be Addressed in Terms of Metric Values



(b) Effectiveness of NH in Scenarios With Different Percentages of Supply Chain-Related Exploits That Can Be Addressed in Terms of the Number of Attackers

Figure 5.18: Percentage Experiments

Limited supply chain-related hardening options Fig. 5.18a shows that HFS improves the value of $kSupply$ with a varying percentage of supply chain exploits that can be used in hardening. As expected, our approach is more effective when hardening options are available for a high percentage of supply chain-related exploits. For instance, $kSupply$ increases from the average value of around 2.6 to 3.9 by addressing 75% of supply chain-related exploits, demonstrating the importance of available supply chain-related hardening options. Moreover, our results demonstrate the applicability of HFS to scenarios where there is no available supply chain-related hardening option by increasing the average $kSupply$ value from around 2.2 to 2.55. Note that the value of $kSupply$ increases with the size of the attack graph, as achieving the attacker's goal condition in larger substations typically requires meeting more conditions in longer attack paths of their corresponding attack graphs. Fig. 5.18b shows that HFS prevents around two-thirds (i.e., 40 out of 60) and half (i.e., 10 out of 20) of supply chain attackers in small and large substations, respectively, even with no available supply chain-related hardening options. In summary, although our approach is more effective under larger percentages of available supply chain-related hardening options, the results generally show a clear reduction of attacks in most cases.

5.4.2 Identifying Optimization Parameters for NSGA-II

To answer RQ2, we measure the average security metrics and cost objectives achieved by NSGA-II optimization (Section 5.2.2) for different values of its parameters (i.e., population size, number of generations, crossover probability, and ETA [4]).

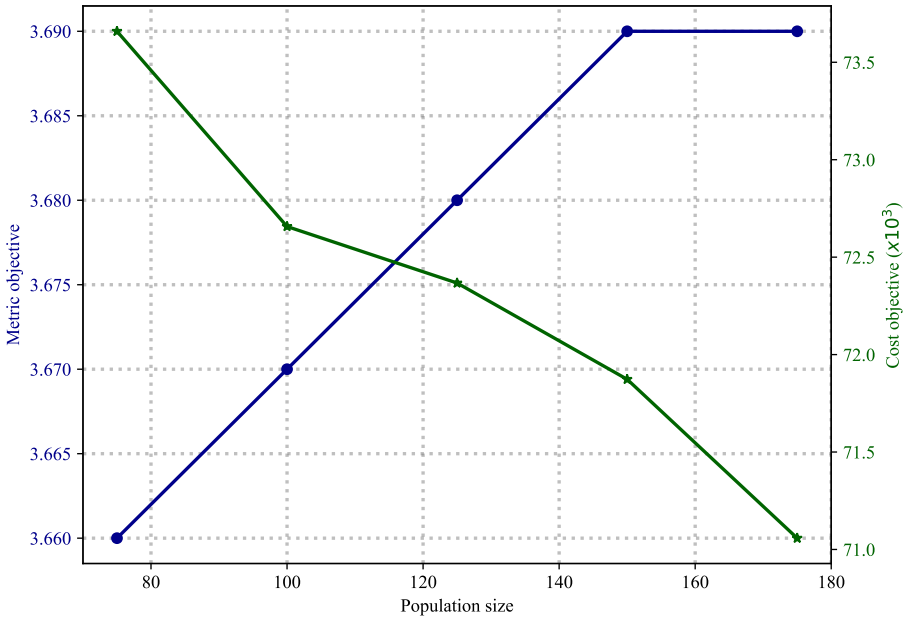


Figure 5.19: Metric and Cost Objectives versus Different Population Sizes

Population size Fig. 5.19 shows that the average metric objective function remains almost similar, and the cost objective function decreases for only around 1 percent (from 71800 to 71000) with the population size of greater than 150. Therefore, we specify 150 as the population size for the NH module.

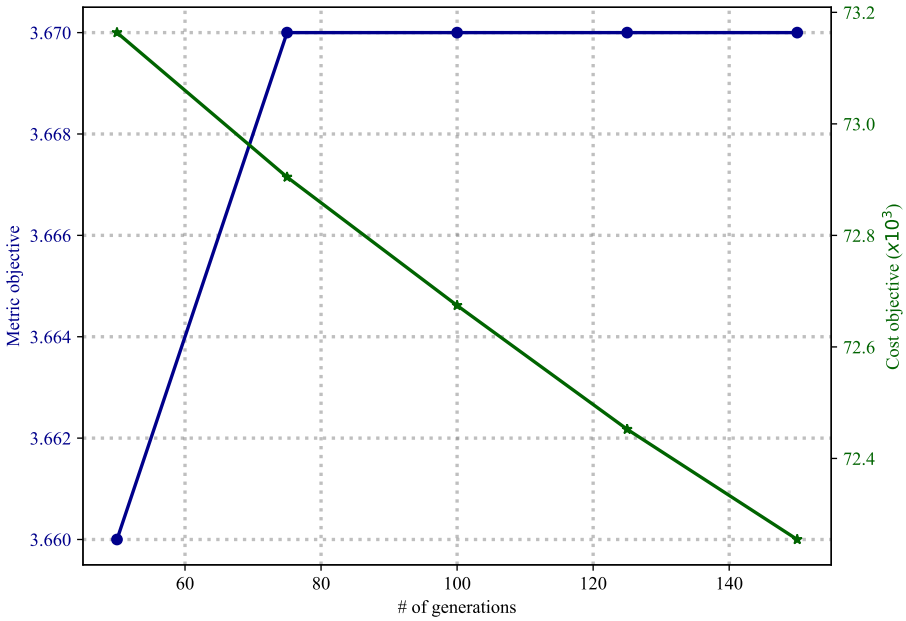


Figure 5.20: Metrics and Cost Objectives versus Number of Generations

Number of generations Fig. 5.20 shows that the average metric objective remains almost similar after 75 generations, and the average cost objective slightly decreases (around 3 percent, from 73600 to 71000) after 75 generations. Therefore, we specify 75 as the number of generations for the NH module.

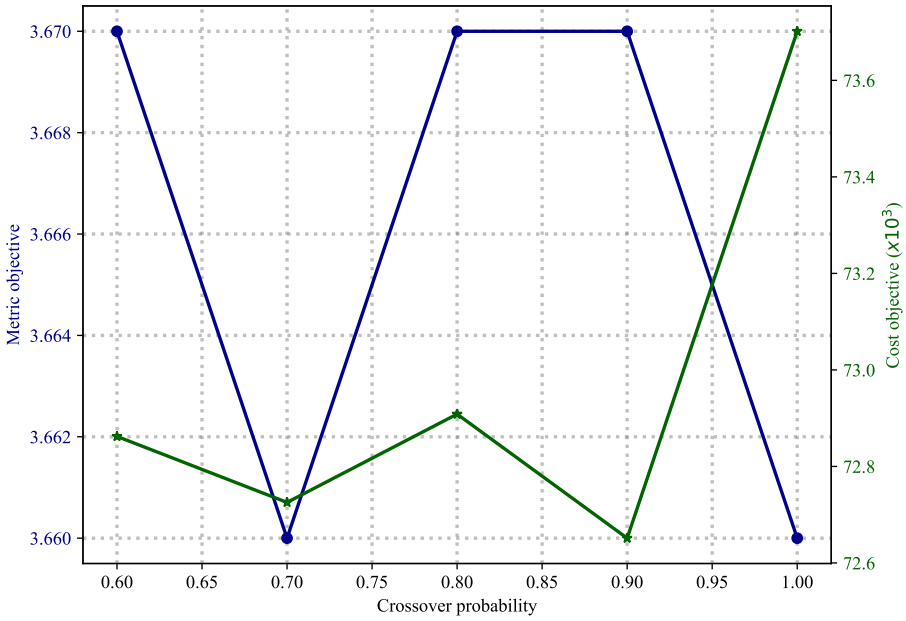


Figure 5.21: Metric and Cost Objectives versus Crossover Probabilities

Crossover probability Fig. 5.21 shows that HFS achieves the highest average metric objective and the lowest cost objective functions with the crossover probability value of 0.9, which we apply in our approach.

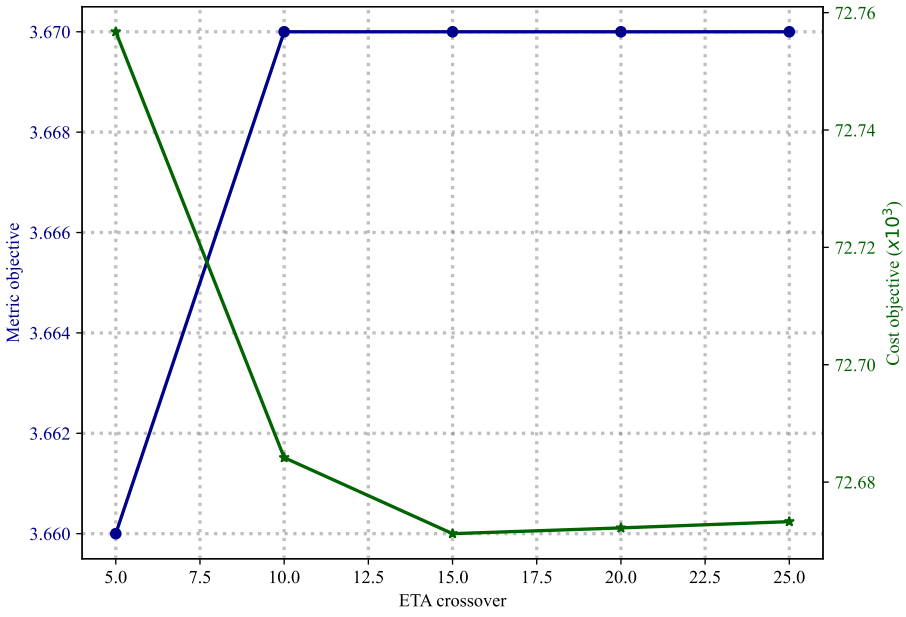


Figure 5.22: Evaluating Metrics and Cost Objectives versus Different ETA [4]

ETA Fig. 5.22 shows that the average metric objective and cost objective remain almost similar for the ETA value greater than 10. Therefore, the ETA value of 10 is chosen for NH.

5.4.3 Execution Time

To answer RQ3, we measure the time required by different modules of HFS.

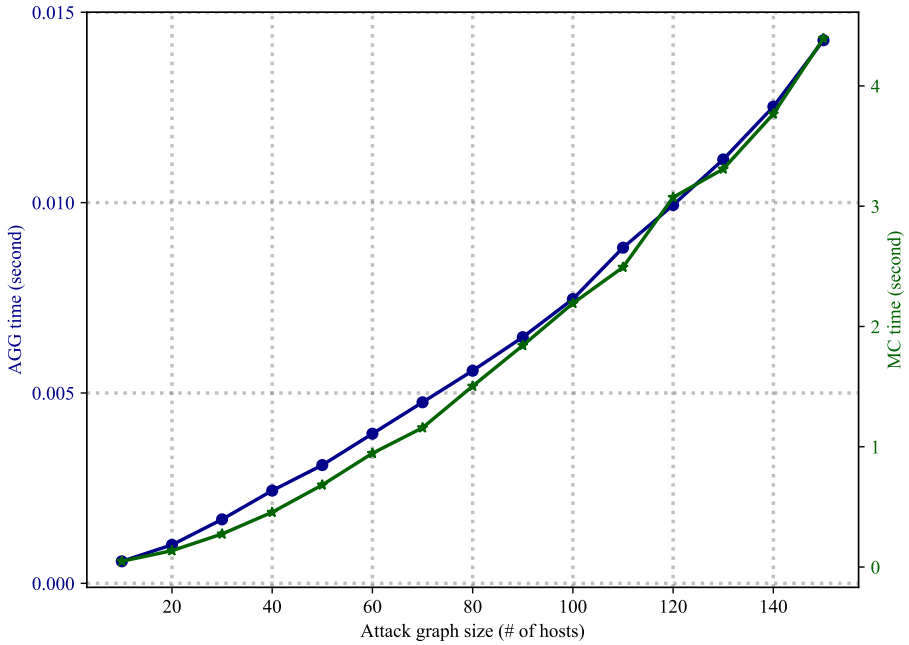


Figure 5.23: Execution Time for Attack Graph Generation and Metric Calculation

AGG and MC execution time Fig. 5.23 shows that although the time required by AGG increases almost exponentially with the number of hosts, generating the attack graph of a substation with 140 hosts (seven times larger than a typical substation [167]) incurs a negligible overhead of around 0.14 seconds. Similarly, although the time required by the MC module increases almost exponentially as the number of hosts grows, it incurs around four seconds even for a substation with 140 hosts. The execution time of AGG and MC has a similar trend as both processes consist of scanning all attack graph nodes.

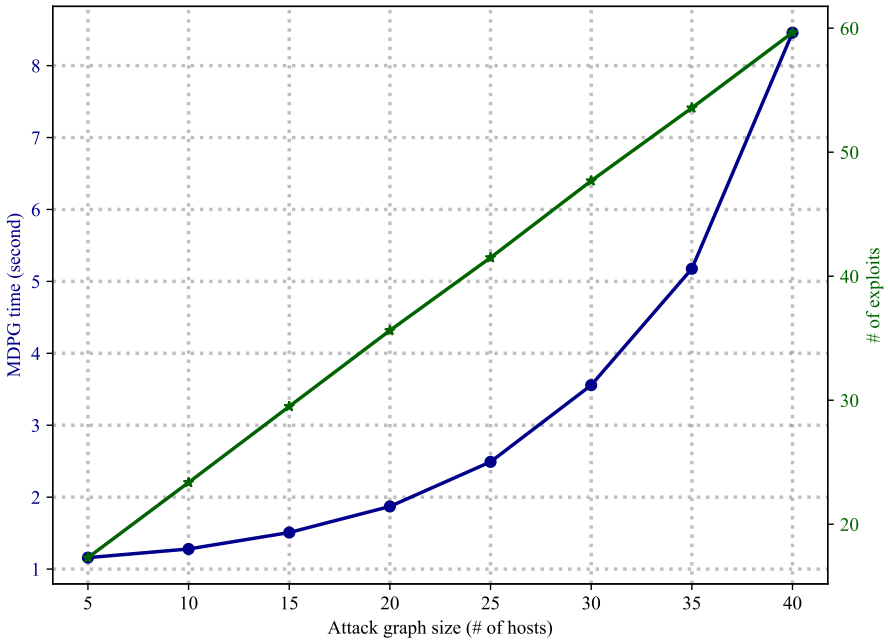


Figure 5.24: Execution Time for MDP Generation

MDPG execution time Fig. 5.24 shows that although the time required by MDPG grows almost exponentially with the size of the attack graph, it is around 8 seconds for attack graphs with 40 hosts, which is twice the number of hosts in typical substations [167]. One reason for such an exponential increase is that MDPG traverses the attack graph to build the structure of the MDP model and performs many Bayesian queries to calculate the probabilities of the MDP model. Moreover, the number of exploit nodes in the attack graph grows linearly as the size of the attack graph increases since each added host adds a certain number of exploits to the substation (at least one zero-day and a limited number of known exploits).

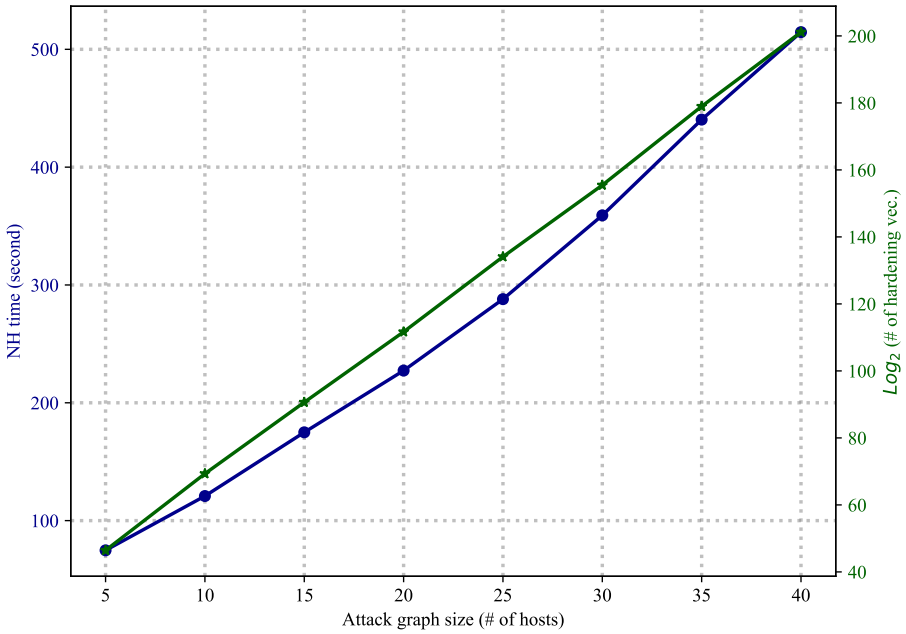


Figure 5.25: Execution Time for Network Hardening

NH execution time Fig. 5.25 shows that although the time required by the NH module grows almost linearly with the number of hosts, it remains around 500 seconds for attack graphs with 40 hosts, which demonstrates the applicability of our approach to cases where there is a need for timely hardening decisions. Moreover, the number of hardening possibilities grows exponentially with the size of the attack graph (the right y-axis shows in logarithmic scale the number of possible hardening vectors for the sake of readability). For instance, even for substations with as few as five hosts, the number of possible hardening vectors is around 2^{46}, and therefore identifying the optimal hardening using brute force is impractical in a real-world substation, which demonstrates the usefulness of an efficient optimization algorithm such as NSGA-II in HFS.

5.5 Utilizing HFS for Other Critical Systems

HFS is designed in a way that new metric calculation functions can be added to the framework. If a new attack graph-based security metric is proposed, it can be added to HFS. In addition, HFS makes it easy for operators to generate attack graphs. The attack graph generation module is not dependent on the system being analyzed. This makes HFS extensible to other systems.

5.6 Summary

In this chapter, we presented a hardening system for improving the security posture of smart grid substations against supply chain attacks. We combine supply chain-related and non-supply-chain-related hardening options under the same framework, making our approach applicable to cases where it is infeasible to address supply chain vulnerabilities. Using multi-objective optimization, we formulated and solved the problem of hardening substations under operational and cost constraints. Our simulation results demonstrate that HFS can effectively improve the security posture of substations against supply chain attacks under different constraints, and it has an acceptable execution time even for large substations.

Chapter 6

SecMonS: A Security Monitoring Framework for Substations Based on Configuration Files and Logs

6.1 Introduction

In order to protect substations against cyber-physical attacks, it is critical for the operators to identify possible attacker strategies. Typically, threat modeling can be performed to get an overview of possible attacks and security threats [14]. Attack graphs, which can visualize different ways to compromise critical assets, are used in modeling threats [15, 16]. However, generating attack graphs requires extensive expert knowledge [17], which remains an open problem. Most existing tools for attack graph generation in the literature, such as MulVAL [17] and CAULDRON [174], are utilized to generate attack graphs based on the static network configuration. In addition, attack graph models generally do not contain the physical consequences of attacks which is a crucial aspect in the case of cyber attacks against substations. Security operation centers receive millions of alerts about ongoing attacks or false alarms, which can lead to difficulty differentiating between real alerts and noises [1]. In [118], the authors tackle this problem by generating attack graphs dynamically based on alerts,

[1] Survey: 27 Percent of IT professionals receive more than 1 million security alerts daily https://www.imperva.com/blog/27-percent-of-it-professionals-receive-more-than-1-million-security-alerts-daily [Online; Accessed 3-July-2023]

but they do not consider cyber-physical attacks. Therefore, it is crucial to generate threat models considering both dynamic log events and physical aspects of attacks for substations. In this chapter, we provide a framework for security monitoring in substations. The next section builds the intuition through a motivating example.

6.1.1 Motivating Example

Fig. 6.1 illustrates the operator's challenges in monitoring the substations' security posture. Firstly, SCL files do not tell operators how IEDs can be attacked, and manually generating attack graphs is an error-prone task. Secondly, attack graphs are generated based on the static configuration, and detailed information about the ongoing attack instance (such as GOOSE [5] parameters currently being attacked) is unavailable from the static configuration. Thirdly, a large power system may contain many transmission lines, and the sizes of models for contingency analysis (analyzing what-if scenarios in power systems [74]) can quickly explode.

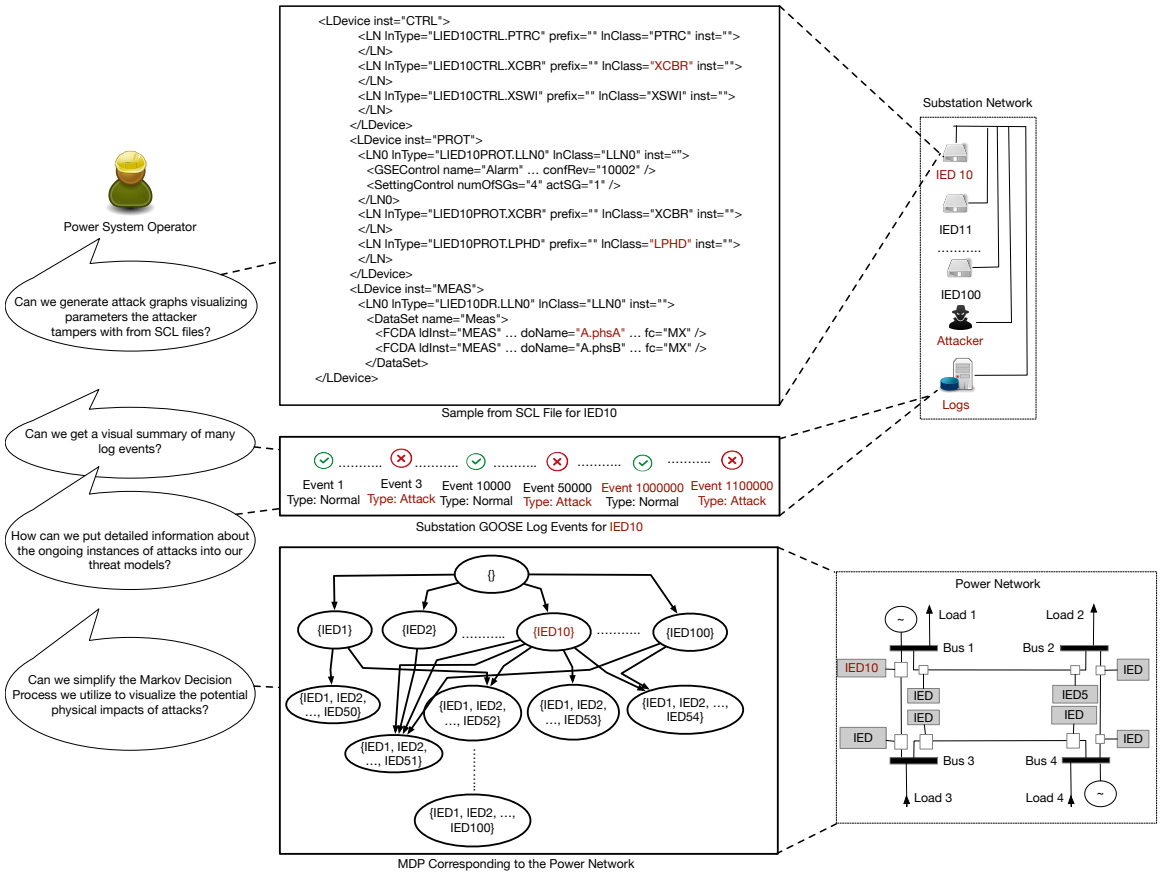


Figure 6.1: Motivating Example

To address those challenges, in this chapter, we provide a framework called SecMonS which provides security monitoring for substations. First, SecMonS generates attack graphs from SCL files. After that, it generates automaton models, which model the system behavior from a set of events extracted from logs. Then, it enhances the SCL-based attack graph with the automaton model. Finally, the framework generates MDP models for power system contingency analysis from the starting state of the attacker in order to avoid the state explosion problem MDP models face.

6.2 Background

6.2.1 Substation Configuration Language (SCL)

Digital substations are essential components of the smart grid. A typical digital substation consists of three levels: substation, bay, and process. The substation level consists of cyber components

such as the substation gateway and HMI. The bay level consists of IEDs, which act as an interface between cyber components at the substation level and physical components at the process level. IEC 61850 is an essential standard for enabling communication between IEDs from different manufacturers. An important part of IEC 61850 is substation configuration language (SCL) [175].

SCL files are utilized for information exchange between IED vendors and power utilities [59]. There are four types of SCL files [175], which are: system specification description (SSD), substation configuration description (SCD), IED capability description (ICD), and configured IED description (CID). SSD files describe the entire power system, including many substations. SCD files contain the description of one substation. ICD files contain descriptions of IEDs inside one substation. CID files contain a description of a single IED.

Given the importance of SCL files, there is existing literature on threat model generation based on SCL files [59, 121]. In [59], the authors define a meta attack language (MAL) [176] based SCL-lang for defining attacks against substations based on steps taken by the attacker. In [121], the authors define another threat modeling language called substation automation systems (SAS)-lang, which is used to generate attack graphs according to attacker steps. In this chapter, we define SCL-based attack graphs demonstrating how the attacker can perform cyber attacks by tampering with communication parameters defined in SCL files.

6.2.2 Generic Object-Oriented Substation Event (GOOSE) Protocol

In IEC 61850 substations, GOOSE protocol is utilized for protection, control, and automation in substations [177]. Several cyber-attacks target the GOOSE protocol [178]. For example, the attacker can delay or modify time-critical trip commands sent using GOOSE messages. In addition, since GOOSE messages are exchanged between IEDs in multicast mode, a message sent by one IED is received by all other IEDs in the same network. Data values in a GOOSE message are contained in the Protocol Data Unit (PDU). Fig. 6.2 shows example GOOSE PDUs for each type of GOOSE messages obtained from the dataset provided in [5].

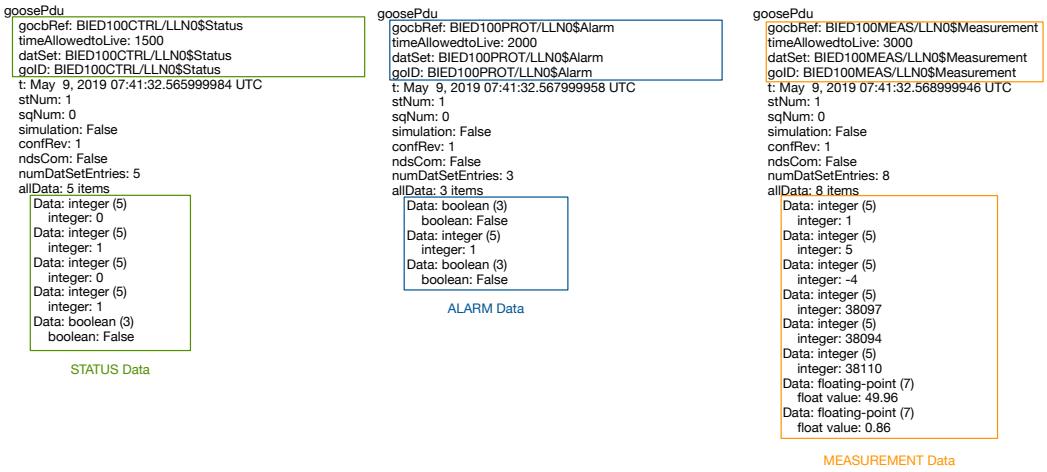


Figure 6.2: A Sample GOOSE Protocol Data Unit [5]

A typical GOOSE PDU contains a sequence number, state number, timestamp, message type, and data values. In order to attack the GOOSE protocol, the attacker needs to gain full access to IEDs and be able to tamper with messages sent in the substation network. In [5], the authors mention four types of attacks against the GOOSE protocol: denial of service, message suppression, data manipulation, and control manipulation. In the denial of service (DoS) attack, a targeted IED is prevented from receiving legitimate GOOSE packets by being flooded. In message suppression, the attacker modifies GOOSE headers, such as timestamps, to prevent IEDs from receiving legitimate GOOSE packets. In data manipulation, the attacker modifies data values such as measurements. In control manipulation, control messages are tampered with by the attacker.

6.2.3 GOOSE Attack Dataset and Dataset Generation

In [5], the authors generate a GOOSE attack dataset for different attack scenarios. Different attack scenarios are simulated using the testbeds described in [179] and [180]. Firstly, packet captures of the normal behavior with and without varying load is provided along with SCL files. After that, packet captures for attack scenarios are provided. As a follow-up work for [5], the authors also develop a packet capture generation tool in which the user defines SCL parameters and attack steps and generates traces [181]. In this chapter, we leverage the GOOSE dataset provided in [5] for simulations.

6.2.4 Automaton Generation

Flexfringe is an automaton generation tool developed to analyze software or protocol behavior [182]. Analyzing the behavior of a given software can aid in its maintenance and testing. Flexfringe is utilized to generate attack graphs from logs in [118]. In [118], the authors generate attack graphs from alert logs without requiring expert input or knowledge to analyze frequently used attacker strategies. In this chapter, we utilize Flexfringe [182] to analyze GOOSE protocol behavior under the normal scenario and attack scenarios to enhance static SCL-based attack graphs with events from GOOSE packet captures.

6.3 Proposed Framework

In this chapter, we provide an online framework named "Security Monitoring for Substations" (SecMonS). SecMons is utilized for monitoring the security posture of substations based on static and dynamic information. Static information includes SCL files for IEDs, and dynamic information includes log files and power system dynamics, such as loads on transmission lines. By taking those into account, SecMonS first generates SCL-based attack graphs (Section 6.3.1). Static attack graphs are generated after every SCL configuration change, so they do not change often. After that, it generates automaton models from log files according to GOOSE[2] parameters sent by IEDs for the normal and attack behavior of the system. Those models are generated with a certain frequency, such as once a minute. After being generated by SecMonS, those models are then used to identify GOOSE parameter changes without going through log files manually (Section 6.3.2). After that, the SCL-based attack graph is enhanced based on the log-based automaton model (Section 6.3.3). As the last step, SecMonS generates the MDP model [74, 78] corresponding to the substation starting from compromised IEDs to analyze the physical impacts of cyber attacks. Inputs to the MDP generation include probabilities obtained from the enhanced attack graph, transmission line capacities, and loads on transmission lines after each IED compromise (Section 6.3.4). Fig. 6.3 shows the overview of the framework. This section details each module of SecMonS.

[2] Although we focus on attacks against GOOSE protocol, this monitoring framework can be applied to other attacks.

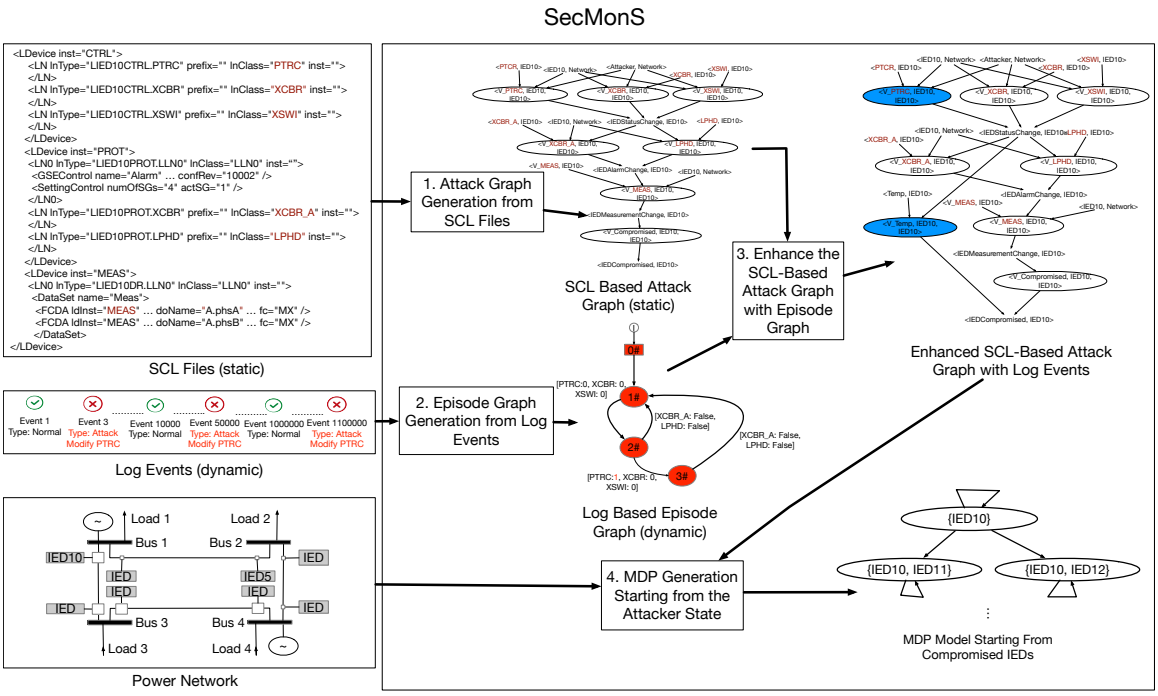


Figure 6.3: Overview of SecMonS

6.3.1 SCL-based Attack Graph Generation

The initial phase in the SecMonS involves generating attack graphs from SCL files. It is very challenging for the operations to identify the attack paths for substations from raw SCL files; attack graphs can help to visualize this task. In [59], the authors generate attack graphs based on SCL files showing assets targeted by the attacker without going through details, such as GOOSE parameters tampered with by the attacker. In this chapter, we generate SCL-based attack graphs visualizing how the GOOSE protocol can be attacked.

We focus on data manipulation attacks based on the GOOSE protocol. When such an attack happens, the attacker modifies GOOSE parameters, such as current measurements on transmission lines, parameters showing the status of each IED, and any alarms generated by IEDs. In the SCL-based attack graph, each GOOSE parameter is modeled as an exploit, showing the attacker's action of modifying that GOOSE parameter. In [5], GOOSE parameters are categorized as STATUS, ALARM, and MEASUREMENT. Fig. 6.4 shows the SCL file and the resulting attack graph. In Fig. 6.4, each GOOSE parameter is represented as an attack graph exploit which represents the fact

that the attacker attacks the GOOSE protocol by tampering with those parameters. In the generation of the static SCL-based attack graph, it is assumed that the attacker first tampers with any of the STATUS parameters (e.g., XCBR, XSWI, and PTRC in Fig. 6.4), after that, ALARM parameters (e.g., PIOC, XCBR, and LPHD in Fig. 6.4), and lastly MEASUREMENT parameters (e.g., MMXU in Fig. 6.4). After the attacker has tampered with all those three types of parameters (i.e., STATUS, ALARM, and MEASUREMENT), the attacker is assumed to compromise the IED sending GOOSE messages (e.g., represented as $\langle Compromised, IED100, IED100 \rangle$ in Fig. 6.4). Generating an SCL-based attack graph allows us to visualize different ways the attacker can attack the GOOSE protocol, and it can help with hardening [13] the substation by identifying critical parameters. In this chapter, the SCL-based attack graph is the starting point for threat modeling, and the initial attack graph is updated according to attack instances.

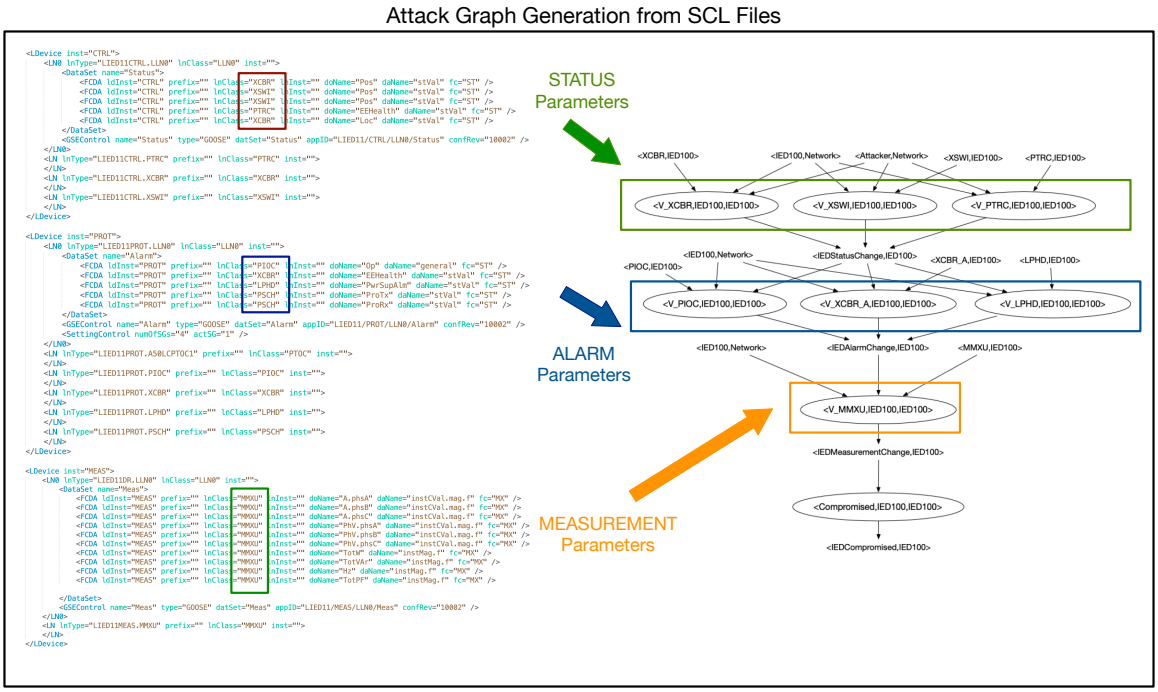


Figure 6.4: Static Attack Graph Generation from SCL File in [5]

6.3.2 Log-based Automaton Model

In order to generate a visual summary of log events, we utilize an automaton generator named Flexfringe [182]. Flexfringe provides an automaton model based on given lists of events. Automaton models generated by Flexfringe depend on differences between lists of events, so those models get larger as the differences between lists of events increase not as the number of events increases. As an example, we can have millions of lists of events with the same behavior and in this case, the automaton model does not get larger. To illustrate, in the case of substation logs, the list of events refers to GOOSE parameter values. According to the data set in [5], GOOSE parameters do not change under normal behavior. Fig. 6.5 shows the automaton model generated using Flexfringe for the normal behavior without varying load for the data set [5]. The automaton model in Fig. 6.5 is generated for 600 events, as shown in states $1\#600$ and $2\#600$ where 1 and 2 are state identifiers.

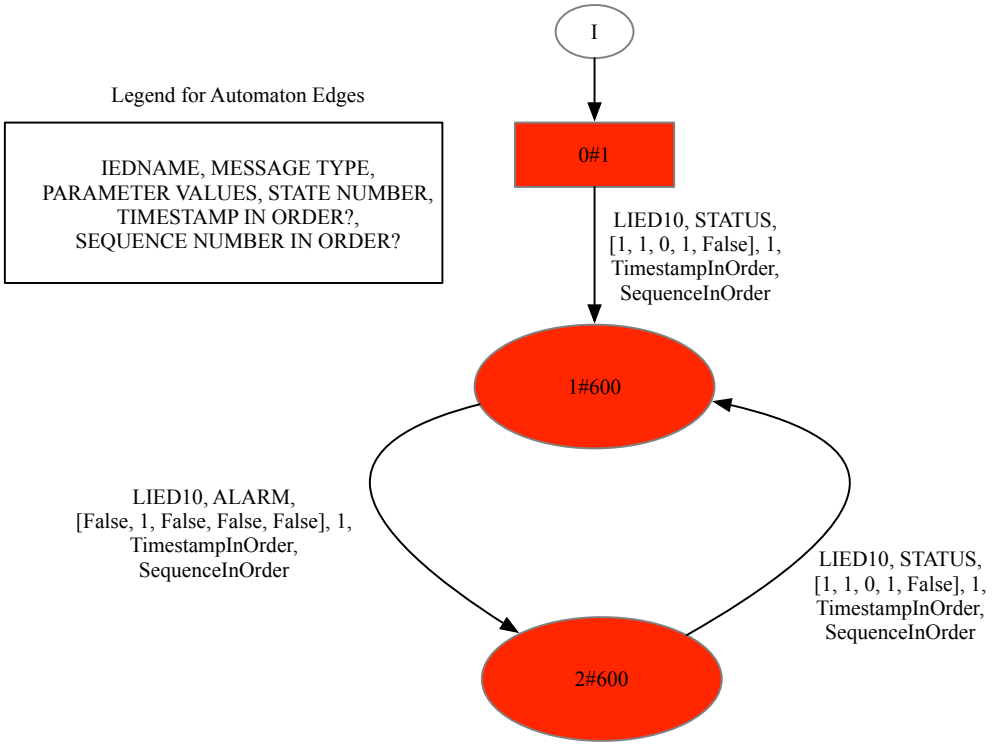


Figure 6.5: Automaton Model for the Normal Behavior

As shown in Fig. 6.5, during the normal behavior without load variation, GOOSE Status values sent are $\{1, 1, 0, 1, False\}$, GOOSE Alarm values sent are $\{False, 1, False, False, False\}$. After generating the automaton model for the normal behavior, the next step is to analyze the attack dataset

to identify GOOSE data values sent by IEDs during an attack. Fig. 6.6 shows the automaton model resulting from one change in the alarm parameters as $\{True, 1, False, False, False\}$.

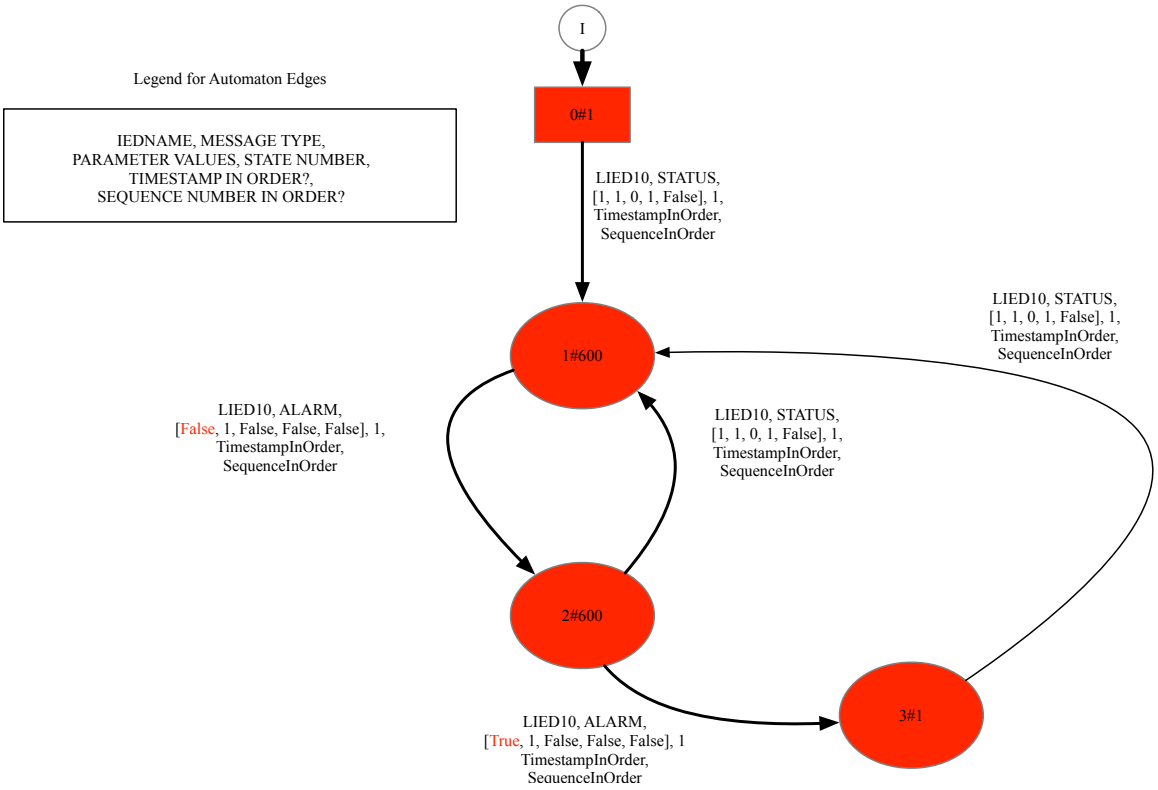


Figure 6.6: Automaton Model for Alarm Parameter Change

6.3.3 Enhancing the SCL-Based Attack Graph with Automaton Model

After building automaton models for the normal and attack behaviors, the next step is to enhance the SCL-based attack graph with parameters changed by the attacker. The SCL-based attack graph assumes that the attacker can change any GOOSE parameters equally likely. However, this is not a realistic assumption. Based on that, the log-based automaton model needs to enhance the SCL-based attack graph.

All of those exploits that refer to GOOSE parameters are assumed to be zero-day [66] in the SCL-based attack graph since the vulnerability the attacker utilizes to change those parameters is unknown. Log-based automaton model for the attack shows which parameters the attacker has changed during the attack. Every GOOSE parameter change is assumed to be performed utilizing a zero-day vulnerability until the attacker uses that parameter in her attack path. In that case, the

exploit becomes known, and its probability becomes 0.58 since the average CVSS score of known vulnerabilities is 5.8[3]. Those exploit nodes are marked as blue. In addition to the standard attack graph models utilized in the literature (such as [46, 66]), we add three more types of exploits: Temporary, Compromised, and Intermediate. Temporary exploits have an exploit probability of 1.0 and are used to connect two privileges. Compromised exploits, also with the probability of 1.0, are utilized to add attack paths in which the attacker does not have to compromise each type of parameter (STATUS, MEASUREMENT, ALARM) to reach the goal. Intermediate exploits also have the probability of exploiting 1.0, and they are used to merge attack graphs for many IEDs, which represents the fact that the attacker needs to compromise all IEDs in the substation in order to compromise the substation. Taking all those into account and according to the possible combinations of types of parameters (STATUS, ALARM, MEASUREMENT), we model ways to enhance the SCL-based attack graph according to possible combinations of parameter types the attacker can change. There are 7 possible cases for generating enhanced attack graphs, which are detailed as:

- **Change in MEASUREMENT Values:** If the attacker tampers with MEASUREMENT values, temporary nodes (attack graph exploits with the probability 1.0) are added to STATUS and ALARM to demonstrate that just tampering with MEASUREMENT values can lead to compromising the IED.

- **Change in ALARM Values:** If the attacker tampers with an ALARM parameter, a temporary node is added from the beginning to STATUS change, and another node is added from the beginning of ALARM change to Compromised to demonstrate that just tampering with ALARM value is an attack path leading to compromising the IED. Fig. 6.7 shows an example merged graph generated according to changed ALARM parameters.

- **Change in ALARM and MEASUREMENT Values:** If the attacker tampers with ALARM and MEASUREMENT values, a temporary node is added from the beginning to STATUS change to demonstrate that just tampering with ALARM and MEASUREMENT values is an attack path leading to compromising the IED.

[3] Current CVSS Score Distribution For All Vulnerabilities https://www.cvedetails.com/cvss-score-distribution.php [Online; Accessed 29-May-2023].

- **Change in STATUS Values:** If the attacker tampers with a STATUS parameter, a temporary node is added from STATUS change to a newly generated Compromised exploit (which also acts as a temporary exploit with the probability of 1.0) in order to demonstrate that just tampering with a STATUS value is a valid attack path.

- **Change in STATUS and MEASUREMENT Values:** If the attacker tampers with STATUS and MEASUREMENT values, a temporary node is added from STATUS change to ALARM change in order to demonstrate that tampering with STATUS and MEASUREMENT values is a valid attack path.

- **Change in STATUS and ALARM Values:** If the attacker tampers with STATUS and ALARM values, a temporary node is added from ALARM change to Compromised in order to demonstrate that just tampering with STATUS and ALARM values is a valid attack path.

- **Change in STATUS, ALARM, and MEASUREMENT Values:** If the attacker tampers with STATUS, ALARM, and MEASUREMENT values, no temporary node needs to be added since one exploit is going to be marked in each level.

According to log-based automaton models for the normal behavior (Fig. 6.5) and for the attack which shows the attacker tampering with ALARM values (Fig. 6.6), the merged attack graph is shown in Fig. 6.7.

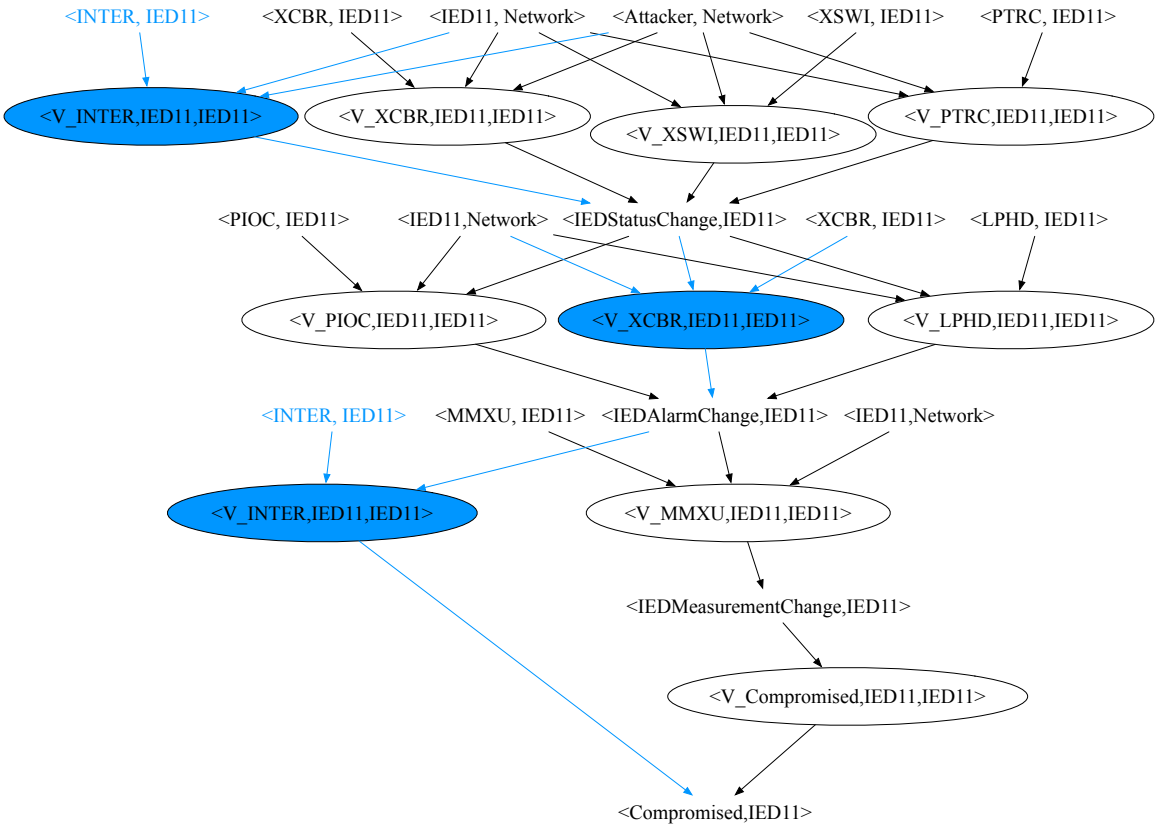


Figure 6.7: Merged Attack Graph According to SCL Files and Logs

6.3.4 Generating MDP Model

After generating the merged attack graph, the next step is to generate an MDP model where the attacker is modeled as an agent aiming to maximize the physical damage. A typical MDP model [168] consists of the following entities:

- S is a finite set of states.
- A is a finite set of actions.
- $P(s, s', a)$ is the transition probability function.
- $R(s')$ is the reward function for the state s'.
- $\gamma \epsilon [0, 1]$ is the discount factor.

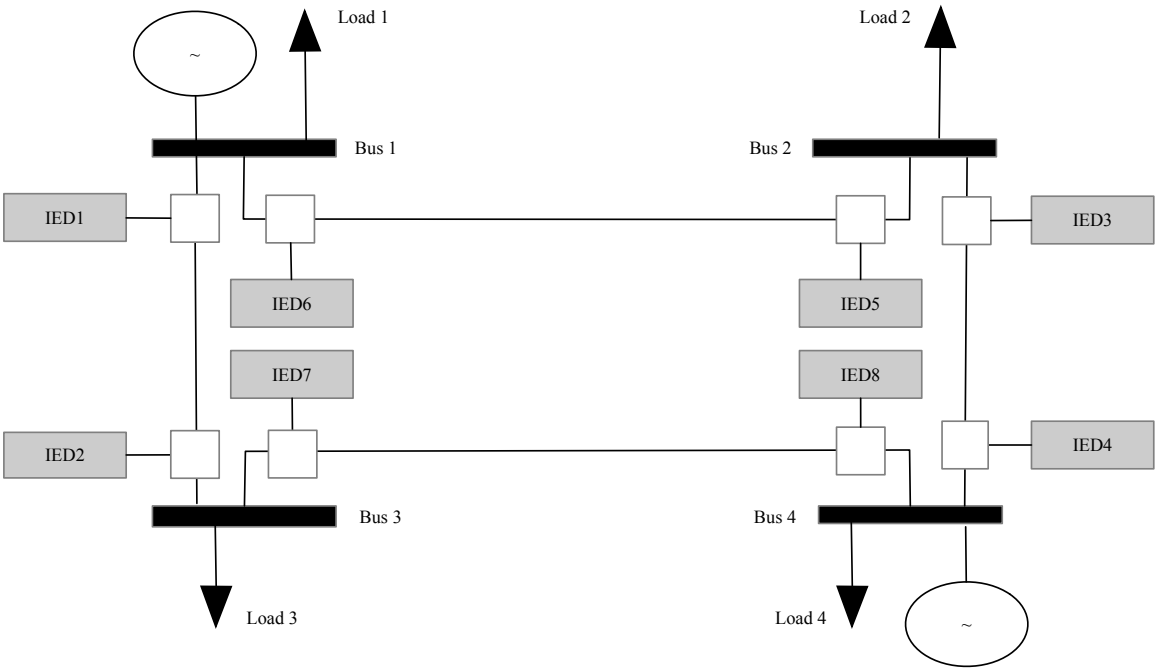


Figure 6.8: IEEE 4-Bus System with IEDs

In SecMonS, states correspond to the set of compromised IEDs. Actions correspond to the attacker compromising an IED by tampering with GOOSE messages. Probabilities are obtained by converting the enhanced attack graph to a Bayesian network and performing Bayesian inferences (an interested reader is referred to [167] for examples). The reward function is the sum of transmission line overloads which is the reward function utilized in [74]. The discount factor refers to the difference between obtaining the reward in the present and the future. The MDP model has been utilized to identify the potential physical damage in [74, 1, 78]. Different from those, we generate the MDP model by starting from the state the attacker is in rather than the state in which none of the IEDs are compromised.

6.4 Case Study: IEEE 4-Bus

In this section, we demonstrate how SecMonS can be leveraged for security monitoring in the IEEE 4-Bus system obtained from pandapower as case4gs [169]. Each bus corresponds to a substation, and each transmission line contains circuit breakers at the beginning and end, controlled by IEDs. Fig. 6.8 shows such a configuration.

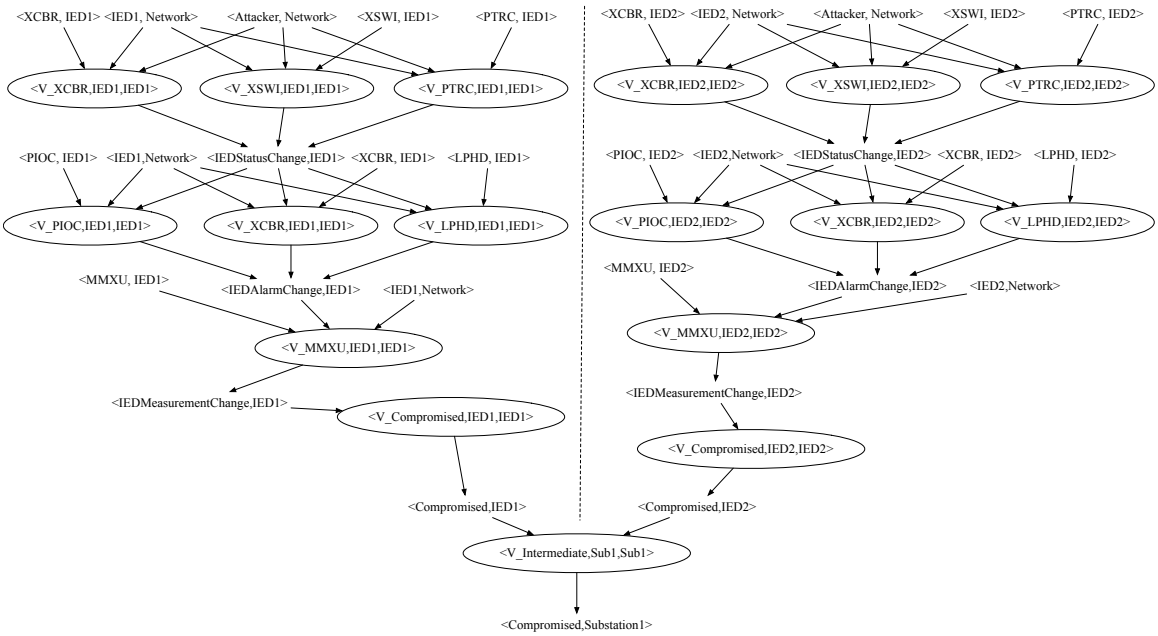


Figure 6.9: SCL-Based Attack Graph for Substation 1

Assume that each IED has the same configuration as IED10 in the dataset [5]. The first step is to generate SCL-based attack graphs for each bus system by merging SCL-based attack graphs generated for IEDs connected to the Bus. Fig. 6.9 shows the SCL-based attack graph for Bus 1 (which contains Substation 1), including SCL-based attack graphs for both IED1 and IED2. SCL-based attack graphs for other buses are generated similarly.

The next step is to generate log-based automaton models for each IED to determine changed parameters. After determining the changed parameters, the next step is to generate the enhanced attack graph. Assuming that IED1 has the STATUS parameter modified, and IED6 has the STATUS and ALARM parameters modified, the enhanced attack graph for substation 1 is shown in Fig. 6.10.

The next step is to generate MDP models for each substation after generating the SCL-based attack graphs and the enhanced attack graphs for each substation. In this section, we explain the MDP generation for Bus 1. MDP models for other buses can be generated in a similar manner. For MDP generation, a starting point (which is the set of compromised IEDs) is also given as input by the operator. Probabilities are calculated by converting the enhanced attack graph into a Bayesian network such that probabilities of all white exploits are 0.08 [167], assuming that those exploits are zero-day. The probabilities of blue exploits are 0.58 since those represent exploits in attack

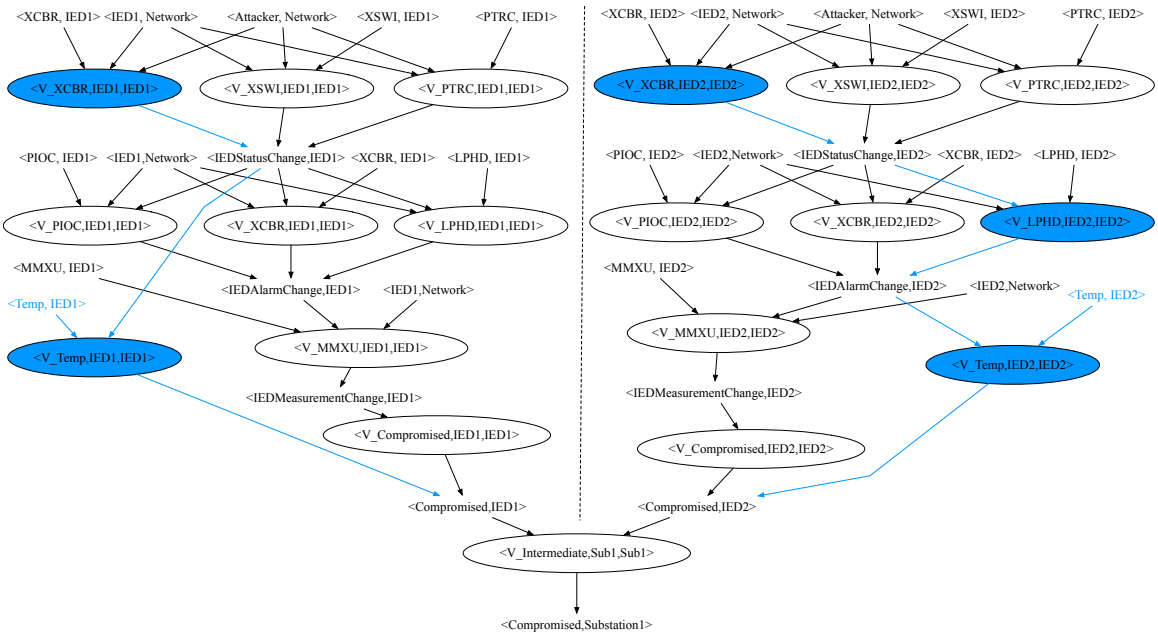


Figure 6.10: Enhanced Attack Graph for Substation 1

paths the attacker takes. Initial conditions (such as $\langle IED1, Network\rangle$ in Fig. 6.10), which are conditions without preconditions, are assumed to be always true. Exploit nodes are true when all of their preconditions are true (i.e., $\langle V\_MMXU, IED1, IED1\rangle$ is true when $\langle IED1, Network\rangle$, $\langle IEDAlarmChange, IED1\rangle$ and $\langle MMXU, IED1\rangle$ are all true in Fig. 6.10). Conditions that are not initial (such as $\langle Compromised, IED1\rangle$ in Fig. 6.10) are true when either preconditions are true. After converting the attack graph in Fig. 6.10 into a Bayesian network, the following probabilities are obtained:

- Pr($\langle Comp., IED1\rangle \mid \{\}$) = 0.64
- Pr($\langle Comp., IED6\rangle \mid \{\}$) = 0.42
- Pr($\langle Comp., Substation1\rangle \mid \{\langle Comp., IED1\rangle\}$) = 0.42
- Pr($\langle Comp., Substation1\rangle \mid \{\langle Comp., IED6\rangle\}$) = 0.64

These probabilities are utilized for state transitions in the MDP model. After that, each state's

rewards are calculated using the power system analysis tool named Pandapower [169]. In Pandapower, each transmission line can be disabled by setting the parameter "in_service" to False before calculating loads on each transmission line. Loads on transmission lines are calculated during the power flow and provided as the parameter "loading_percent" in Pandapower [169]. In order to calculate the reward function, we calculate the sum of loads that exceed the capacity on each transmission line. As an example, if IED1 in Fig. 6.8 trips, the following load percentages are obtained for each transmission line, calculated using Pandapower [169]:

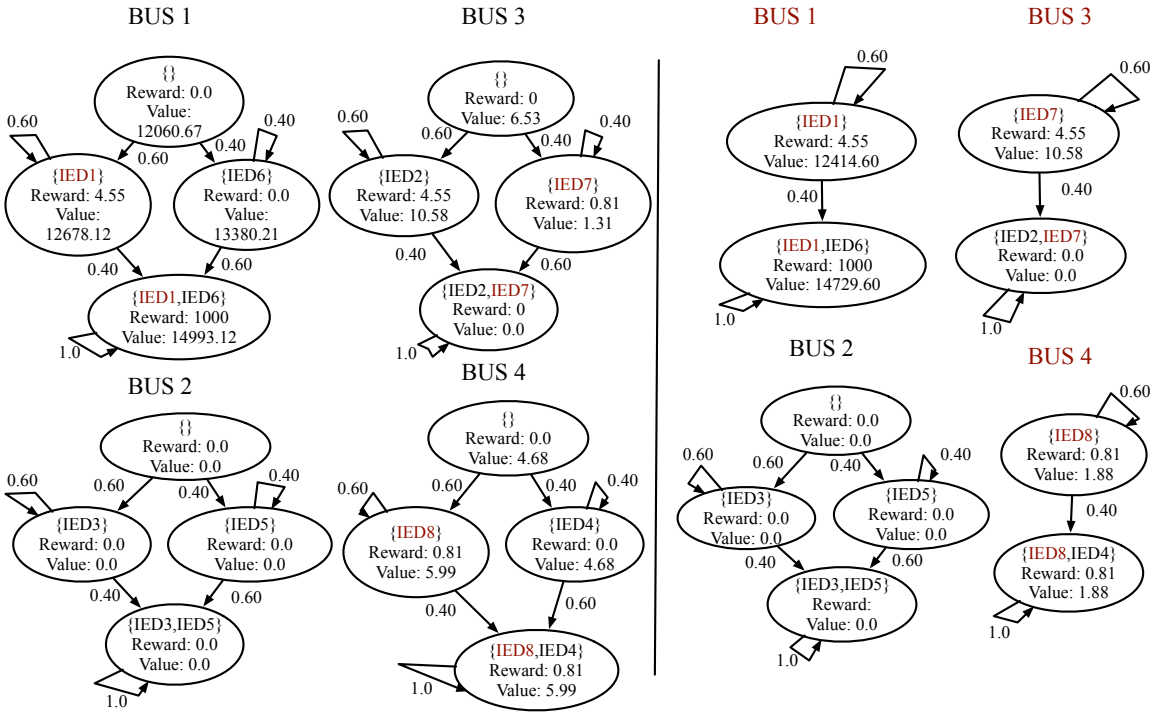
- Transmission Line From Bus1 to Bus2 : 57.61%
- Transmission Line From Bus1 to Bus3: 0.0% since this transmission line is disabled.
- Transmission Line From Bus2 to Bus4: 41.80%
- Transmission Line From Bus3 to Bus4: 104.55%

Based on these calculations, the amount of reward when IED1 is maliciously tripped is 4.55 since it causes 4.55% overload on transmission lines. Rewards for each state are calculated as follows:

- R($\{\}$) = 0.0
- R($\langle Comp., IED1 \rangle$) = 4.55
- R($\langle Comp., IED6 \rangle$) = 0.0
- R($\langle Comp., Substation1 \rangle$) = 1000, which is the maximum reward since if the attacker is in this state, he/she can cause a blackout.

By taking the probabilities and rewards, Fig. 6.11 provides the MDP model starting from the state in which none of the IEDs are in the power system are compromised on the left, and the MDP model starting from the state in which IED1, IED7, and IED8 are compromised on the right.

In order to solve the MDP model, the value iteration function is utilized with a discount factor of 0.95 (the authors in [78] experimented with varying discount factors. In this chapter, we assume that the difference between obtaining the reward at present or in the future does not have a huge



MDP Models for Each Bus Starting From the Beginning (Left) and MDP Models Considering IED1, IED7, and IED8 Are Compromised (Right)

Figure 6.11: MDP Models Generated by SecMonS

difference from the attacker's perspective). State values calculated by the value iteration function are also shown in Fig. 6.11. As it can be seen, even though attacking IED1 gives a non-zero immediate reward to the attacker, attacking IED6 has more value in terms of the future reward (value function 13380.21 vs. value function 12678.12).

6.5 Simulation

In order to evaluate the practicality of SecMonS, our simulations aim to estimate the execution time for SCL-based attack graph generation, log-based episode graph generation, merged graph generation, and MDP generation for different bus systems and different percentages of compromised IEDs. Simulations are conducted on MacOS Ventura 13.4 with an Apple M1 Max processor and 64 GB of memory. In simulations, both the dataset [5] and simulated data are utilized.

Our first simulation aims to determine if SecMonS can be leveraged for large substations with many IEDs. For that purpose, we simulate the SCL-based graph generation according to the number

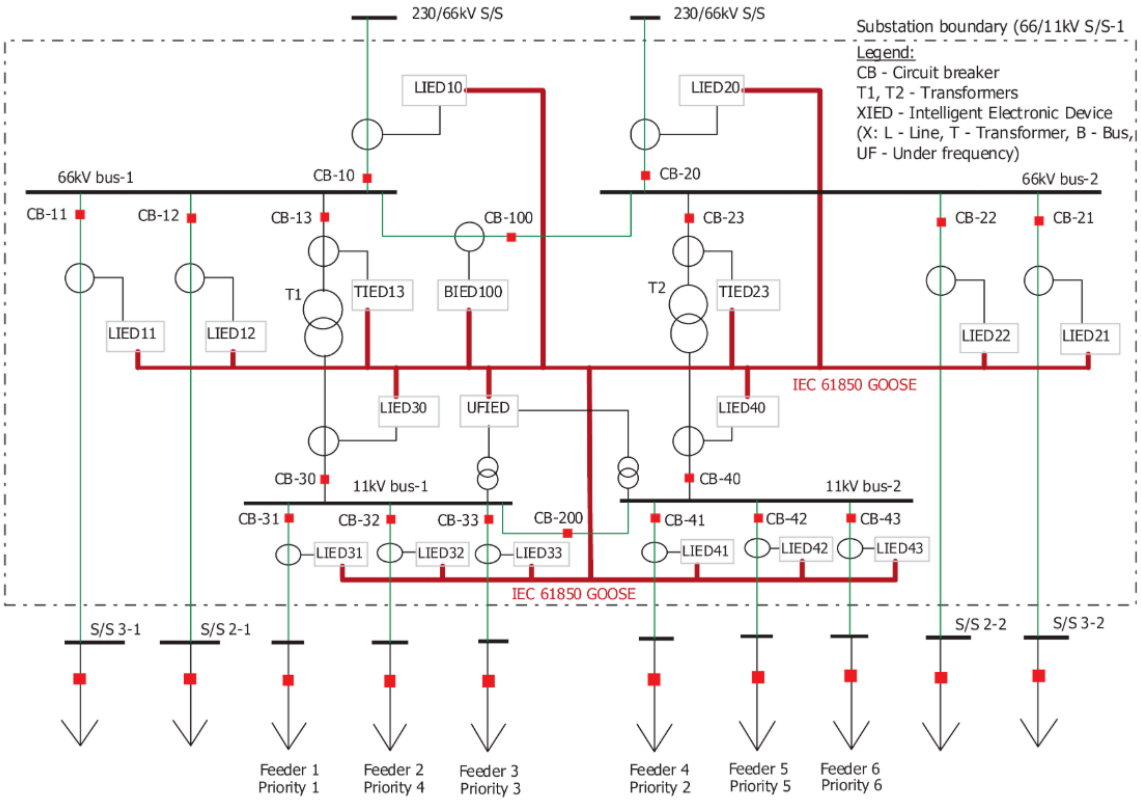


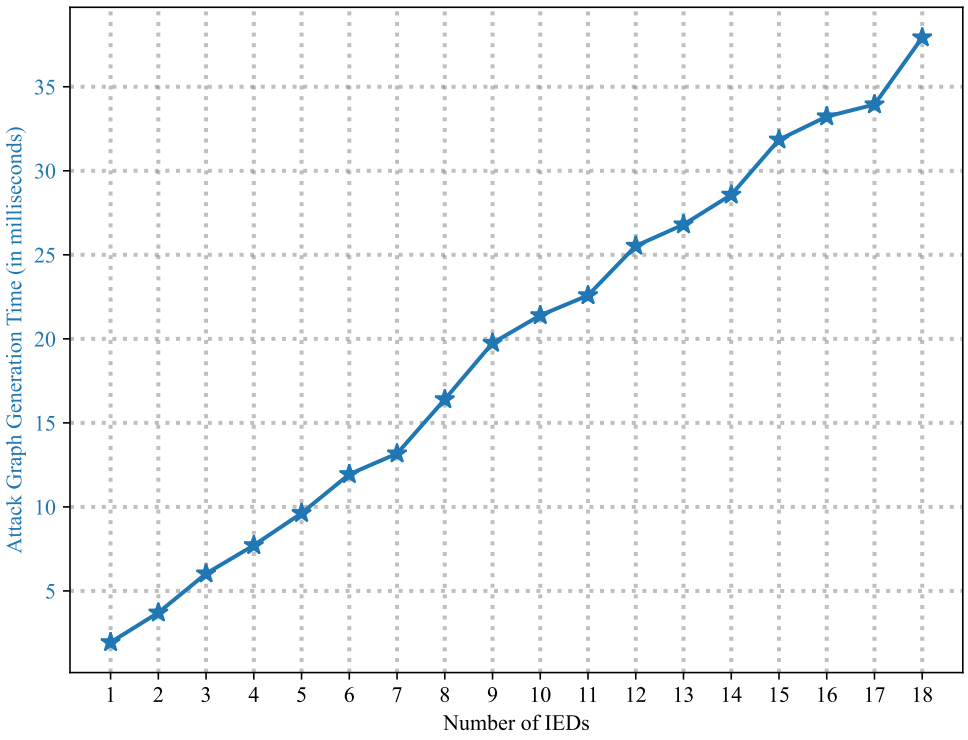
Figure 6.12: Substation Configuration Used to Generate the Dataset in [5]

of IEDs by using SCL files in the dataset provided by the authors in [5]. The dataset [5] contains 18 SCL files. Fig. 6.12 shows the configuration used to generate the dataset in [5].

In the first simulation, we vary the number of IEDs from 1 to 18. For each number of IEDs, we choose a random set of IEDs according to the number of IEDs in that experiment and perform 100 experiments for each number of IEDs and calculate the average execution time and the average attack graph size, measured as the number of nodes. Results of this simulation are provided in Fig. 6.13.

Results and Implications: As it can be seen from Fig. 6.13, both the attack graph size and the execution time increase linearly as the number of IEDs increases. Even for 18 IEDs, the execution time is around 40 milliseconds. This result implies that SecMonS can be utilized for substations with many IEDs. For 18 IEDs, the attack graph size is around 360 nodes, implying that SecMonS is scalable for larger systems.

Our second simulation aims to identify the relationship between the number of log entries and



SCL-Based Attack Graph Generation Time versus Number of Nodes Using SCL Files Provided in [5]

Figure 6.13: SCL-Based Attack Graph Generation Time

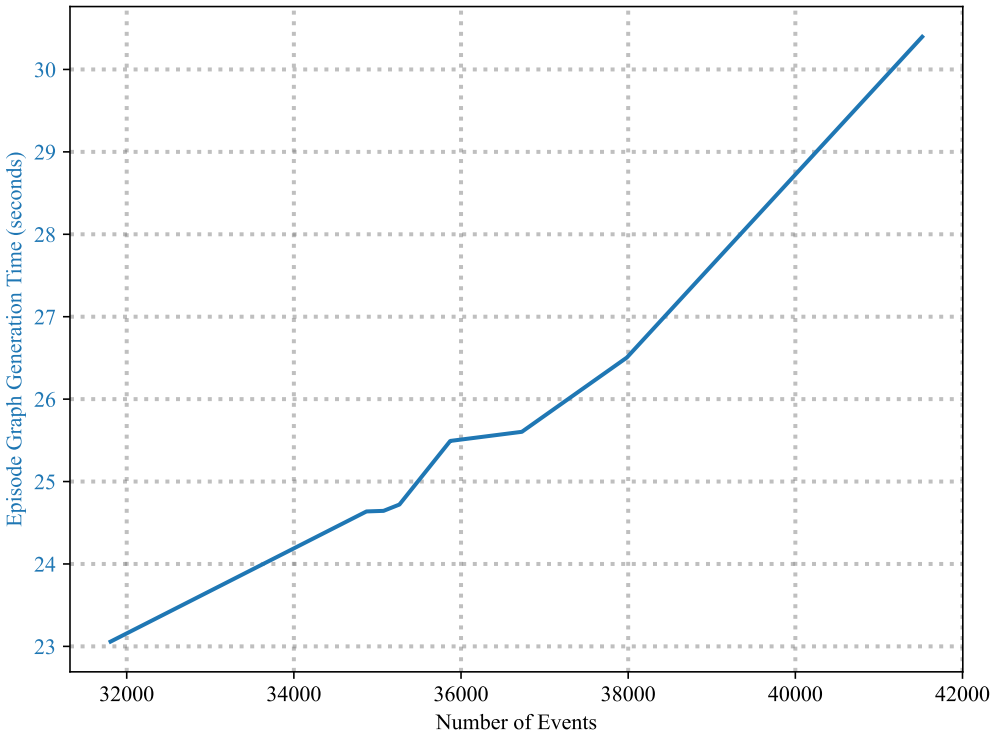


Figure 6.14: Number of Log Entries versus Automaton Generation Time for [5]

the episode graph generation time. According to a survey [1], 27 percent of information technology professionals receive more than 1 million security alerts daily. Given that the number of security alerts can grow, it is essential to have a reasonable automaton model generation time. We simulate log-based automaton generation time using captures provided in [5]. Results of this simulation are provided in Fig. 6.14.

Results and Implications: As seen in Fig. 6.14, episode graph generation time grows linearly with the number of events. There are small variations around $36,000$ events since logs containing attacks require different processing times based on the number of parameters maliciously modified by the attacker during the attack.

Our third simulation aims to identify the relationship between the number of parameter changes and execution time for simulated data. The goal is to identify how the generation time for the log-based automaton model changes according to the number of changed parameters to identify

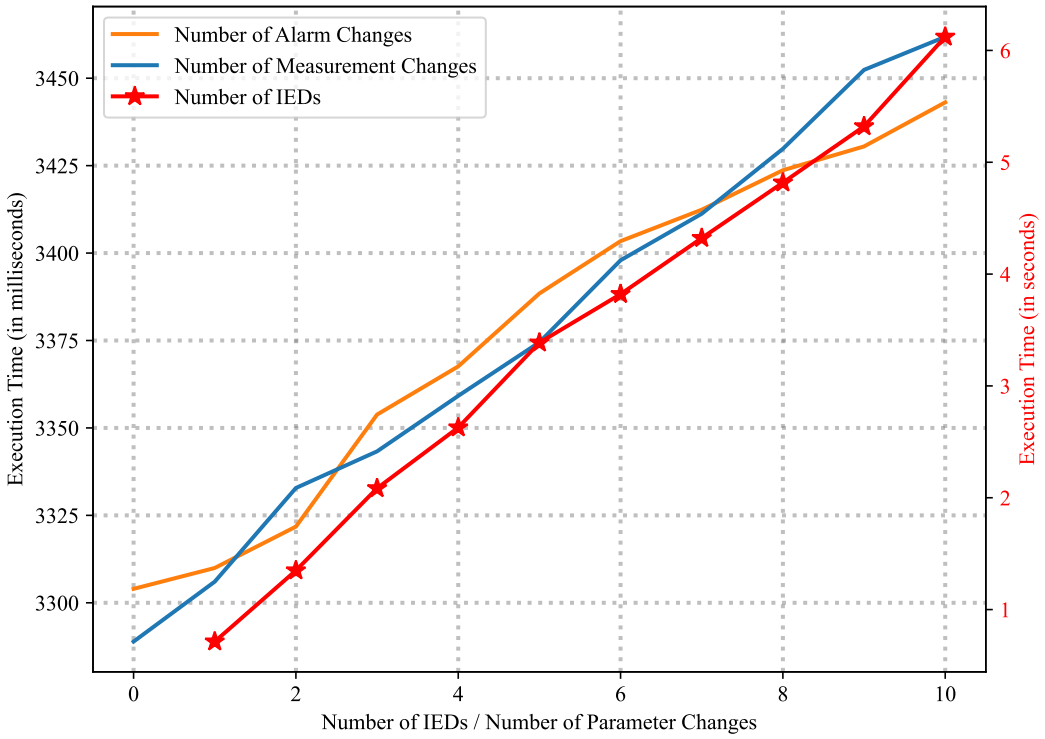


Figure 6.15: IEDs and Parameter Changes versus Automaton Generation Time

the scalability of SecMonS for coordinated attacks in which many parameters may be changed concurrently. Results of this simulation are provided in Fig. 6.15.

Results and Implications: As seen in Fig. 6.15, as the number of IEDs increases, episode graph generation time increases linearly. Even for a substation with 10 IEDs, the episode graph can be generated in 6 seconds. As the number of tampered parameters increases, episode graph generation time increases linearly. This result shows that SecMonS can be utilized for coordinated attacks in which many GOOSE parameters can be tampered with by attackers simultaneously.

Our fourth simulation aims to determine the merged graph generation time according to the number of IEDs and changed parameters. Merged graph generation should be done more frequently compared to SCL-based graph generation since merged graphs utilize dynamic information. Based on this, merged graph generation should be done in a reasonable time. Results of this simulation are provided in Fig. 6.16.

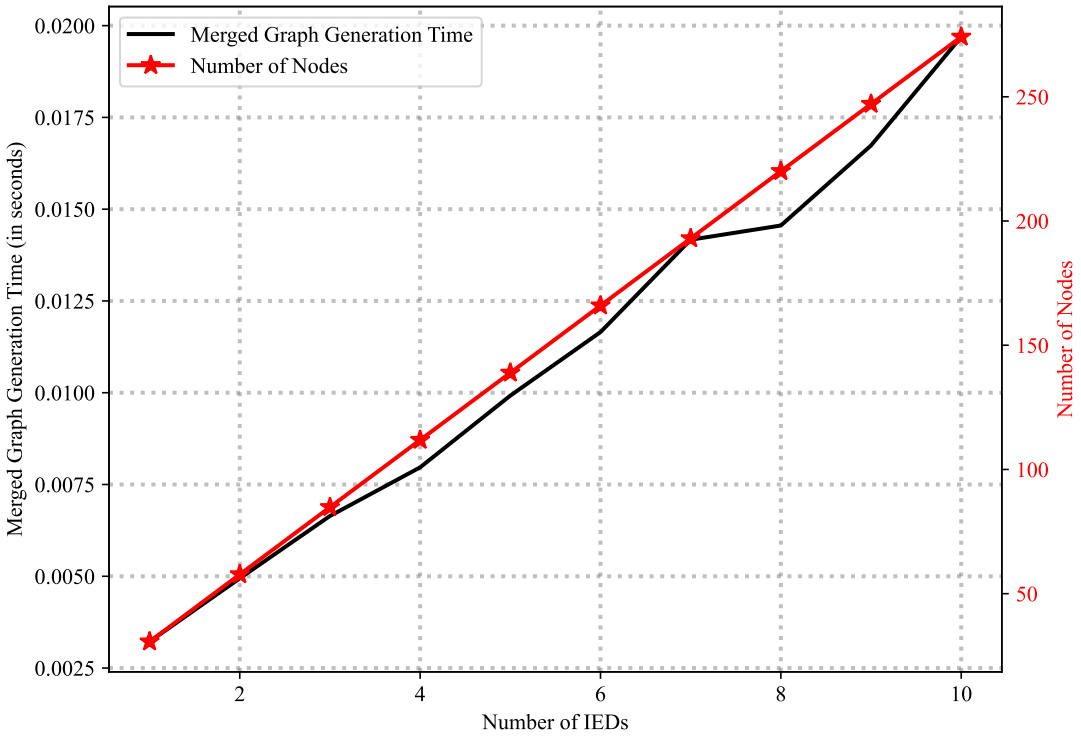


Figure 6.16: IEDs versus Merged Graph Generation Time and Number of Nodes

Results and Implications: As seen in Fig. 6.16, merged graph generation time and size grow linearly according to the number of IEDs. This result shows that merged graphs can be generated in a reasonable time for monitoring the security posture of IEC 61850 substations.

Our last simulation aims to determine MDP generation time and MDP number of states for different percentages of compromised IEDs for IEEE 9-Bus, 33-Bus, and 39-Bus from pandapower [169]. Fig. 6.17 shows MDP generation time and MDP number of states with respect to the percentage of compromised IEDs for 9-Bus. Fig. 6.18 shows MDP generation time and MDP number of states with respect to the percentage of compromised IEDs for 33-Bus. Fig. 6.19 shows MDP generation time and MDP number of states with respect to the percentage of compromised IEDs for the 39-Bus system.

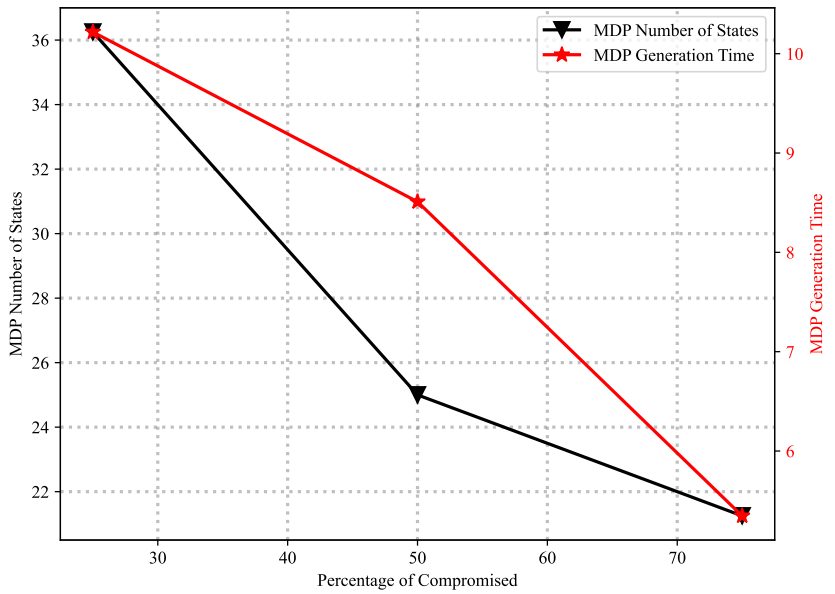


Figure 6.17: Percentage of Compromised IEDs vs MDP Number of States and MDP Generation Time IEEE 9Bus

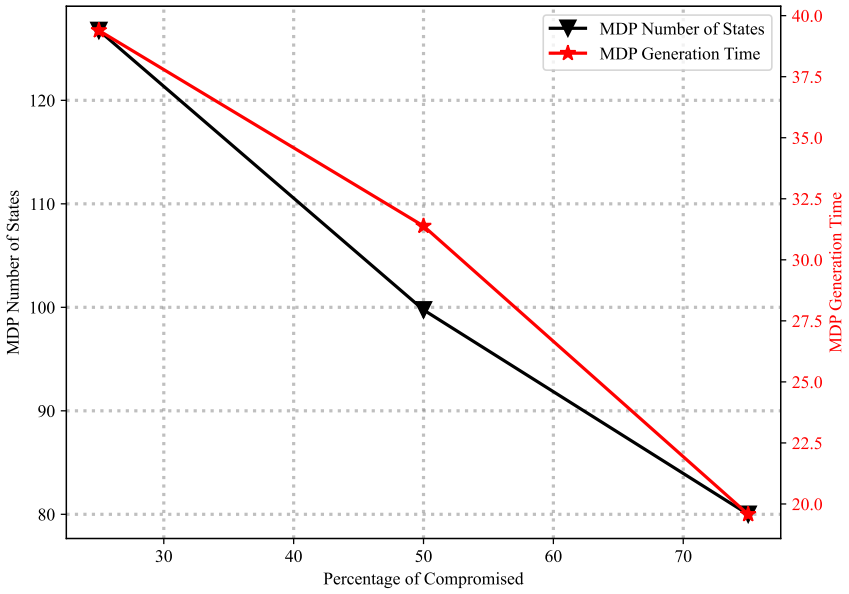


Figure 6.18: Percentage of Compromised IEDs vs MDP Number of States and MDP Generation Time 33Bus

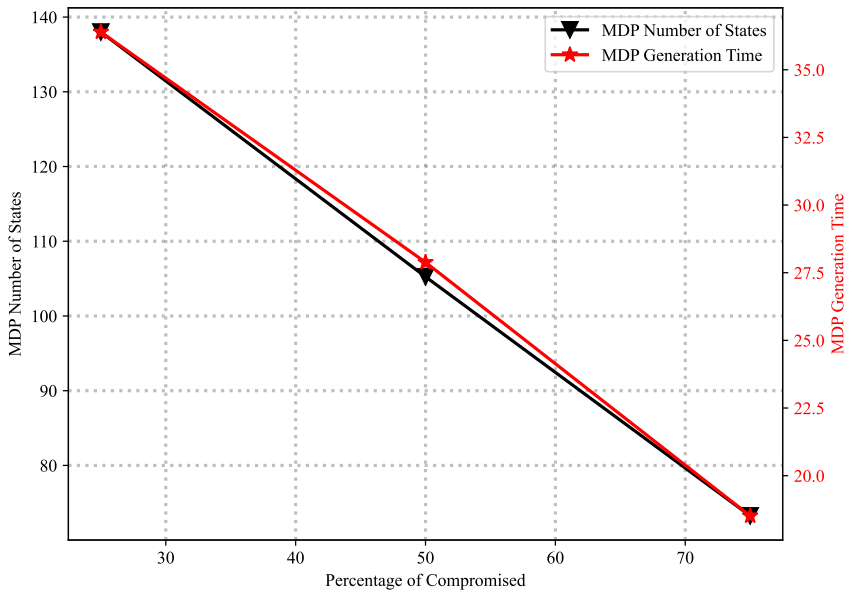


Figure 6.19: Percentage of Compromised IEDs vs MDP Number of States and MDP Generation Time 39Bus

Results and Implications: As seen in Fig. 6.17, Fig. 6.18, and Fig. 6.19, MDP generation time decreases exponentially with respect to the percentage of compromised IEDs. This result implies that starting the MDP model from IEDs, which the attacker has already compromised, can reduce the execution time no matter how big the power system is. In addition, the decrease in MDP number of states and MDP generation time becomes almost linear as the power system size increases. This shows the importance of generating MDP models from the starting state of the attacker for large power systems.

6.6 Utilizing SecMonS for Other Critical Systems

For utilizing SecMonS for critical systems other than substations, we need both static configuration and event logs matching the static configuration. For other attacks in substations, SecMonS can be used as long as we identify parameters the attacker tampers with. SecMonS was developed considering attacks against GOOSE protocol in substations and utilizing it for other critical systems is not a trivial problem.

6.7 Summary

In this chapter, we have developed a security monitoring framework, SecMonS, that utilizes static and dynamic information. This framework monitors the security of power systems concerning cyber attacks with physical consequences. Our simulation results demonstrate that SecMonS can be deployed on large power networks.

Chapter 7

Conclusion

IEC 61850 substations are critical components in the smart grid since compromising them can lead to significant consequences, such as blackouts. Securing substations is different than securing cyber networks since cyber attacks with major physical consequences are possible in the case of substations. In this thesis, we have tackled the important problem of measuring and improving the security postures of IEC 61850 substations. Several security metrics are designed, a hardening system is developed, and an online security monitoring framework is presented in this thesis. Simulations reflecting real-life scenarios are performed in order to demonstrate the usage of designed security metrics, the hardening system, and the security monitoring framework. The main challenge was to come up with those metrics and show that those metrics reflect the security posture utilizing simulations reflecting real-life scenarios.

According to works done in this thesis, we have shown that security metrics are important for evaluating the effectiveness of security enhancement efforts, such as adding firewalls, and diversifying devices. Focusing on secure deployment in substations can prevent a majority of potential attackers. In this regard, this thesis shows some valuable insights. First, redundancy, which is commonly used in substations, should be designed in a secure manner since if all subsystems of an IEC 61850 substation are identical, that can make it possible for the attacker to easily compromise all subsystems after compromising one of them. Second, supply chain vulnerabilities pose an important security risk to substations since substations may contain devices from different vendors, and not all of those vendors may be trustworthy. By taking this into account, we have designed supply

chain security metrics and demonstrated that supply chain vulnerabilities can make substations an easy target for attackers. Therefore, measuring the security posture of IEC 61850 substations according to supply chain vulnerabilities is important. Third, we developed a hardening system and showed that it is possible to harden substations against supply chain attacks without utilizing supply chain-related hardening options. Lastly, we developed a security monitoring framework and showed that it is possible and convenient to develop threat models from that state the attacker is currently in, rather than enumerating all possible attacker states. This thesis shows that securing substations before they are compromised is important since successful cyber attacks targeting substations can lead to major consequences, such as blackouts.

The contributions of this thesis can be summarized and related as follows. First, security metrics for measuring the effectiveness of redundancy from a security perspective are designed and their effectiveness is demonstrated through simulations. Second, security metrics for measuring the security postures of substations considering supply chain attacks are designed. Third, we have developed a hardening system that considers both supply chain-related and non-supply chain-related hardening options. We have shown that this system is effective in hardening substations against supply chain attacks according to supply chain security metrics developed in the second chapter. Lastly, a security monitoring framework is developed for substations which generates attack graphs based on the static substation configuration and then enhances the generated attack graph according to real-time log events. Given the criticality of substations in the smart grid, this thesis makes major contributions to measuring and improving the security postures of substations. In addition, this thesis demonstrates that rather than trusting our intuitions for security, we should always quantify security utilizing security metrics in order to prevent the majority of different types of attacks from happening. The following can be done as a future work to extend this thesis:

- $kSupply$ and $kSupplier$ metrics are going to be extended for the whole power system rather than just substations.
- There can be cases in which supply chain vulnerabilities may pretend to be zero-day vulnerabilities. Such cases are also going to be considered for the extended versions of $kSupply$ and $kSupplier$.

- We are going to utilize blockchain in the SCR metric in order to get more accurate values for risk scores since blockchain improves integrity (e.g., the authenticity of the origin can be verified utilizing blockchain).

- Security metrics are going to be defined based on threat models modeling human factors since lots of cyber attacks happen due to misbehaving employees or human factor errors [183].

- The time required to generate attack graphs, calculate metrics, and generate MDP models must be improved to identify critical substations in huge power systems.

- SecMonS is verified using a public dataset since power system operators are reluctant to share the contents of their log files, however, SecMonS is going to be deployed to a testbed or to a real substation in order to extend SecMonS as a tool that is ready to be used by power system operators.

- Metrics calculating the security postures of substations according to physical aspects of attacks will be integrated into the hardening framework.

- The hardening framework is going to be modified to utilize enhanced attack graphs defined in Section 6.

Bibliography

[1] K. R. Davis, C. M. Davis, S. A. Zonouz, R. B. Bobba, R. Berthier, L. Garcia, and P. W. Sauer, "A cyber-physical modeling and assessment framework for power grid infrastructures," *IEEE Transactions on smart grid*, vol. 6, no. 5, pp. 2464–2475, 2015.

[2] P. Akaber, B. Moussa, M. Debbabi, and C. Assi, "Cascading link failure analysis in interdependent networks for maximal outages in smart grid," in *Smart Grid Communications (SmartGridComm), 2016 IEEE International Conference on*, pp. 429–434, IEEE, 2016.

[3] Y. Zhu, J. Yan, Y. Tang, Y. L. Sun, and H. He, "Coordinated attacks against substations and transmission lines in power grids," in *Global Communications Conference (GLOBECOM), 2014 IEEE*, pp. 655–661, IEEE, 2014.

[4] K. Deb, K. Sindhya, and T. Okabe, "Self-adaptive simulated binary crossover for real-parameter optimization," in *Proceedings of the 9th annual conference on genetic and evolutionary computation*, pp. 1187–1194, 2007.

[5] P. P. Biswas, H. C. Tan, Q. Zhu, Y. Li, D. Mashima, and B. Chen, "A synthesized dataset for cybersecurity study of IEC 61850 based substation," in *2019 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, pp. 1–7, IEEE, 2019.

[6] O. Duman, L. Wang, M. Au, M. Kassouf, and M. Debbabi, "Hardening Substations against Supply Chain Attacks Under Operational Constraints," in *2022 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1–5, IEEE, 2022.

[7] W. Wang and Z. Lu, "Cyber security in the smart grid: Survey and challenges," *Computer Networks*, vol. 57, no. 5, pp. 1344–1371, 2013.

[8] G. Lu, D. De, and W.-Z. Song, "Smartgridlab: A laboratory-based smart grid testbed," in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, pp. 143–148, IEEE, 2010.

[9] X. Fang, S. Misra, G. Xue, and D. Yang, "Smart grid—the new and improved power grid: A survey," *IEEE communications surveys & tutorials*, vol. 14, no. 4, pp. 944–980, 2011.

[10] "Attack on Nine Substations Could Take Down U.S. Grid." urlhttps://spectrum.ieee.org/attack-on-nine-substations-could-take-down-us-grid. [Online; accessed 23-March-2018].

[11] I. Standard, "Network Engineering Guideline for Communication Networks and Systems in Substations," tech. rep., IEC 61850-90-4.

[12] O. Duman, M. Ghafouri, M. Kassouf, R. Atallah, L. Wang, and M. Debbabi, "Modeling Supply Chain Attacks in IEC 61850 Substations," in *2019 IEEE Int. Conf. on Commun., Control, and Computing Technol. for Smart Grids (SmartGridComm)*, pp. 1–6, IEEE, 2019.

[13] D. Borbor, L. Wang, S. Jajodia, and A. Singhal, "Optimizing the network diversity to improve the resilience of networks against unknown attacks," *Computer Communications*, vol. 145, pp. 96–112, 2019.

[14] E. Ling, R. Lagerström, and M. Ekstedt, "A systematic literature review of information sources for threat modeling in the power systems domain," in *Critical Information Infrastructures Security: 15th International Conference, CRITIS 2020, Bristol, UK, September 2–3, 2020, Proceedings*, pp. 47–58, Springer, 2020.

[15] P. Ammann, D. Wijesekera, and S. Kaushik, "Scalable, graph-based network vulnerability analysis," in *Proceedings of the 9th ACM Conference on Computer and Communications Security*, pp. 217–224, ACM, 2002.

[16] O. Sheyner, J. Haines, S. Jha, R. Lippmann, and J. M. Wing, "Automated generation and analysis of attack graphs," in *Security and privacy, 2002. Proceedings. 2002 IEEE Symposium on*, pp. 273–284, IEEE, 2002.

[17] X. Ou, S. Govindavajhala, and A. W. Appel, "MulVAL: A Logic-based Network Security Analyzer," in *USENIX security symposium*, vol. 8, pp. 113–128, Baltimore, MD, 2005.

[18] A. Ramos, M. Lazar, R. Holanda Filho, and J. J. Rodrigues, "Model-Based Quantitative Network Security Metrics: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2704–2734, 2017.

[19] K. Kaynar, "A taxonomy for attack graph generation and usage in network security," *Journal of Information Security and Applications*, vol. 29, pp. 27–56, 2016.

[20] H. Hu, J. Liu, Y. Zhang, Y. Liu, X. Xu, and J. Tan, "Attack scenario reconstruction approach using attack graph and alert data mining," *Journal of Information Security and Applications*, vol. 54, p. 102522, 2020.

[21] H. Huang, S. Zhang, X. Ou, A. Prakash, and K. Sakallah, "Distilling critical attack graph surface iteratively through minimum-cost sat solving," in *Proceedings of the 27th Annual Computer Security Applications Conference*, pp. 31–40, 2011.

[22] M. Ryan and M. Ryan, "Ransomware Case Studies," *Ransomware Revolution: The Rise of a Prodigious Cyber Threat*, pp. 65–91, 2021.

[23] A. V. Uzunov and E. B. Fernandez, "An extensible pattern-based library and taxonomy of security threats for distributed systems," *Computer Standards & Interfaces*, vol. 36, no. 4, pp. 734–747, 2014.

[24] B. Schneier, "Attack trees: Modeling security threats." https://www.schneier.com/academic/archives/1999/12/attack_trees.html, 1999.

[25] C.-W. Ten, C.-C. Liu, and M. Govindarasu, "Vulnerability assessment of cybersecurity for SCADA systems using attack trees," in *2007 IEEE Power Engineering Society General Meeting*, pp. 1–8, IEEE, 2007.

[26] C.-W. Ten, G. Manimaran, and C.-C. Liu, "Cybersecurity for critical infrastructures: Attack and defense modeling," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 40, no. 4, pp. 853–865, 2010.

[27] E. G. da Silva, L. A. D. Knob, J. A. Wickboldt, L. P. Gaspary, L. Z. Granville, and A. Schaeffer-Filho, "Capitalizing on SDN-based SCADA systems: An anti-eavesdropping case-study," in *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pp. 165–173, IEEE, 2015.

[28] W. Li, J. Huang, and W. You, "Attack modeling for electric power information networks," in *2010 International Conference on Power System Technology*, pp. 1–5, IEEE, 2010.

[29] P. McDaniel and S. McLaughlin, "Structured security testing in the smart grid," in *2012 5th International Symposium on Communications, Control and Signal Processing*, pp. 1–4, IEEE, 2012.

[30] Y. Ru, Y. Wang, J. Li, J. Liu, G. Yang, K. Yuan, and K. Liu, "Risk assessment of cyber attacks in ECPS based on attack tree and AHP," in *2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, pp. 465–470, IEEE, 2016.

[31] N. Chakraborty and E. Kalaimannan, "Minimum cost security measurements for attack tree based threat models in smart grid," in *2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*, pp. 614–618, IEEE, 2017.

[32] S. Paudel, P. Smith, and T. Zseby, "Attack models for advanced persistent threats in smart grid wide area monitoring," in *Proceedings of the 2nd Workshop on Cyber-Physical Security and Resilience in Smart Grids*, pp. 61–66, 2017.

[33] G. Falco, A. Viswanathan, C. Caldera, and H. Shrobe, "A master attack methodology for an AI-based automated attack planner for smart cities," *IEEE Access*, vol. 6, pp. 48360–48373, 2018.

[34] Y. Yude and L. Shuilan, "Optimal prevention and control strategy against preconceive faults in electric cyber-physical system," in *2018 2nd IEEE Conference on Energy Internet and Energy System Integration (EI2)*, pp. 1–9, IEEE, 2018.

[35] Z. Tian, W. Wu, S. Li, X. Li, Y. Sun, and Z. Chen, "A security model of SCADA system based on attack tree," in *2019 IEEE 3rd Conference on Energy Internet and Energy System Integration (EI2)*, pp. 2653–2658, IEEE, 2019.

[36] R. Meyur, "A Bayesian attack tree based approach to assess cyber-physical security of power system," in *2020 IEEE Texas Power and Energy Conference (TPEC)*, pp. 1–6, IEEE, 2020.

[37] M. A. McQueen, W. F. Boyer, M. A. Flynn, and G. A. Beitel, "Time-to-compromise model for cyber risk reduction estimation," in *Quality of Protection: Security Measurements and Metrics*, pp. 49–64, Springer, 2006.

[38] A. Sadu, M. Stevic, N. Wirtz, and A. Monti, "A stochastic assessment of attacks based on continuous-time Markov chains," in *2020 6th IEEE International Energy Conference (ENERGYCon)*, pp. 11–16, IEEE, 2020.

[39] T. R. B. Kushal, Z. Gao, J. Wang, and M. S. Illindala, "Causal chain of time delay attack on synchronous generator control," in *2020 IEEE Power & Energy Society General Meeting (PESGM)*, pp. 1–5, IEEE, 2020.

[40] Z. Kai-lun, T. Quan, and Z. Hong-gang, "Vulnerability Assessment of WAMS Communication System in the Energy Internet Based on Attack Tree Model," in *2020 IEEE Sustainable Power and Energy Conference (iSPEC)*, pp. 2640–2645, IEEE, 2020.

[41] J. Fei, K. Chen, Q. Yao, Q. Guo, and X. Wang, "Security Vulnerability Assessment of Power IoT based on Business Security," in *Proceedings of the 2020 1st International Conference on Control, Robotics and Intelligent System*, pp. 128–135, 2020.

[42] B. Wen and P. Li, "Risk Assessment of Security and Stability Control System against Cyber Attacks," in *2021 IEEE 2nd China International Youth Conference on Electrical Engineering (CIYCEE)*, pp. 1–5, IEEE, 2021.

[43] M. Silveira, D. Dolezilek, S. Wenke, and J. Yellajosula, "Attack tree analysis of a digital secondary system in an electrical substation," in *16th International Conference on Developments in Power System Protection (DPSP 2022)*, vol. 2022, pp. 152–157, IET, 2022.

[44] X. Gao, T. Shang, D. Li, and J. Liu, "Quantitative risk assessment of threats on scada systems using attack countermeasure tree," in *2022 19th Annual International Conference on Privacy, Security & Trust (PST)*, pp. 1–5, IEEE, 2022.

[45] A. Rahiminejad, O. Duman, M. Ghafouri, R. Atallah, A. Mohammadi, and M. Debbabi, "A Resilience Quantitative Framework for Wide Area Damping Control Against Cyberattacks," in *2023 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1–5, IEEE, 2023.

[46] M. Zhang, L. Wang, S. Jajodia, A. Singhal, and M. Albanese, "Network diversity: a security metric for evaluating the resilience of networks against zero-day attacks," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 5, pp. 1071–1086, 2016.

[47] 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*, pp. 161–164, IEEE, 2012.

[48] Y. Zhang, Y. Xiang, and L. Wang, "Reliability analysis of power grids with cyber vulnerability in SCADA system," in *2014 IEEE PES general meeting— conference & exposition*, pp. 1–5, IEEE, 2014.

[49] N. Vijayamohanan Pillai, "Loss of load probability of a power system," *J Fundamentals of Renewable Energy Applications*, vol. 5, pp. 1–9, 2014.

[50] Y. Zhang, L. Wang, Y. Xiang, and C.-W. Ten, "Power system reliability evaluation with SCADA cybersecurity considerations," *IEEE Transactions on Smart Grid*, vol. 6, no. 4, pp. 1707–1721, 2015.

[51] P. S. Patapanchala, C. Huo, R. B. Bobba, and E. Cotilla-Sanchez, "Exploring security metrics for electric grid infrastructures leveraging attack graphs," in *Technologies for Sustainability (SusTech), 2016 IEEE Conference on*, pp. 89–95, IEEE, 2016.

[52] Y. Zhang, Y. Xiang, and L. Wang, "Power system reliability assessment incorporating cyber attacks against wind farm energy management systems," *IEEE transactions on smart grid*, vol. 8, no. 5, pp. 2343–2357, 2016.

[53] Q. Dai, L. Shi, and Y. Ni, "Risk assessment for cyberattack in active distribution systems considering the role of feeder automation," *IEEE Transactions on Power Systems*, vol. 34, no. 4, pp. 3230–3240, 2019.

[54] M. Ibrahim and A. Alsheikh, "Determining resiliency using attack graphs," *Recent developments on industrial control systems resilience*, pp. 117–133, 2020.

[55] A. Sahu and K. Davis, "Structural learning techniques for Bayesian attack graphs in cyber physical power systems," in *2021 IEEE Texas Power and Energy Conference (TPEC)*, pp. 1–6, IEEE, 2021.

[56] G. F. Cooper and E. Herskovits, "A bayesian method for constructing bayesian belief networks from databases," in *Uncertainty Proceedings 1991*, pp. 86–94, Elsevier, 1991.

[57] A. Umunnakwe, A. Sahu, and K. Davis, "Multi-Component Risk Assessment Using Cyber-Physical Betweenness Centrality," in *2021 IEEE Madrid PowerTech*, pp. 1–6, IEEE, 2021.

[58] G. A. Weaver, K. Davis, C. M. Davis, E. J. Rogers, R. B. Bobba, S. Zonouz, R. Berthier, P. W. Sauer, and D. M. Nicol, "Cyber-physical models for power grid security analysis: 8-substation case," in *2016 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pp. 140–146, IEEE, 2016.

[59] E. Rencelj Ling and M. Ekstedt, "Generating threat models and attack graphs based on the IEC 61850 system configuration description language," in *Proceedings of the 2021 ACM Workshop on Secure and Trustworthy Cyber-Physical Systems*, pp. 98–103, 2021.

[60] A. Umunnakwe, A. Sahu, M. R. Narimani, K. Davis, and S. Zonouz, "Cyber-physical component ranking for risk sensitivity analysis using betweenness centrality," *IET Cyber-Physical Systems: Theory & Applications*, vol. 6, no. 3, pp. 139–150, 2021.

[61] A. Dutta, S. Purohit, A. Bhattacharya, and O. Bel, "Cyber attack sequences generation for electric power grid," in *2022 10th Workshop on Modelling and Simulation of Cyber-Physical Energy Systems (MSCPES)*, pp. 1–6, IEEE, 2022.

[62] M. Li, H. Xu, J. Xu, Z. Ding, Q. Liu, and Z. Yin, "Risk Assessment of Cyber Physical Power System considering Attack Model," in *2022 IEEE 5th International Electrical and Energy Conference (CIEEC)*, pp. 4650–4655, IEEE, 2022.

[63] E. Rencelj Ling, J. E. Urrea Cabus, I. Butun, R. Lagerström, and J. Olegard, "Securing Communication and Identifying Threats in RTUs: A Vulnerability Analysis," in *Proceedings of the 17th International Conference on Availability, Reliability and Security*, pp. 1–7, 2022.

[64] A. Presekal, A. Ştefanov, V. S. Rajkumar, and P. Palensky, "Attack graph model for cyber-physical power systems using hybrid deep learning," *IEEE Transactions on Smart Grid*, 2023.

[65] A. Sahu and K. Davis, "Inferring adversarial behaviour in cyber-physical power systems using a Bayesian attack graph approach," *IET Cyber-Physical Systems: Theory & Applications*, 2023.

[66] L. Wang, S. Jajodia, A. Singhal, P. Cheng, and S. Noel, "k-zero day safety: A network security metric for measuring the risk of unknown vulnerabilities," *IEEE Transactions on Dependable and Secure Computing*, vol. 11, no. 1, pp. 30–44, 2014.

[67] R. Lippmann, K. Ingols, C. Scott, K. Piwowarski, K. Kratkiewicz, M. Artz, and R. Cunningham, "Validating and restoring defense in depth using attack graphs," in *MILCOM 2006-2006 IEEE Military Communications Conference*, pp. 1–10, IEEE, 2006.

[68] J. Pamula, S. Jajodia, P. Ammann, and V. Swarup, "A weakest-adversary security metric for network configuration security analysis," in *Proceedings of the 2nd ACM workshop on Quality of protection*, pp. 31–38, 2006.

[69] L. Wang, T. Islam, T. Long, A. Singhal, and S. Jajodia, "An attack graph-based probabilistic security metric," in *Data and Applications Security XXII: 22nd Annual IFIP WG 11.3 Working Conference on Data and Applications Security London, UK, July 13-16, 2008 Proceedings 22*, pp. 283–296, Springer, 2008.

[70] U. Premaratne, J. Samarabandu, T. Sidhu, R. Beresh, and J.-C. Tan, "Security analysis and auditing of IEC61850-based automated substations," *IEEE Transactions on Power Delivery*, vol. 25, no. 4, pp. 2346–2355, 2010.

[71] E. LeMay, M. D. Ford, K. Keefe, W. H. Sanders, and C. Muehrcke, "Model-based security metrics using adversary view security evaluation (advise)," in *2011 Eighth International Conference on Quantitative Evaluation of SysTems*, pp. 191–200, IEEE, 2011.

[72] S. Zonouz and P. Haghani, "Cyber-physical security metric inference in smart grid critical infrastructures based on system administrators' responsive behavior," *Computers & security*, vol. 39, pp. 190–200, 2013.

[73] J. Filar and K. Vrieze, *Competitive Markov decision processes*. Springer Science & Business Media, 2012.

[74] S. Zonouz, C. M. Davis, K. R. Davis, R. Berthier, R. B. Bobba, and W. H. Sanders, "SOCCA: A security-oriented cyber-physical contingency analysis in power infrastructures," *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 3–13, 2014.

[75] M. Khanabadi, H. Ghasemi, and M. Doostizadeh, "Optimal transmission switching considering voltage security and N-1 contingency analysis," *IEEE Transactions on Power Systems*, vol. 28, no. 1, pp. 542–550, 2012.

[76] C. Vellaithurai, A. Srivastava, S. Zonouz, and R. Berthier, "CPIndex: Cyber-physical vulnerability assessment for power-grid infrastructures," *IEEE Transactions on Smart Grid*, vol. 6, no. 2, pp. 566–575, 2014.

[77] X. Liu, J. Zhang, and P. Zhu, "Dependence analysis based cyber-physical security assessment for critical infrastructure networks," in *2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, pp. 1–7, IEEE, 2016.

[78] P. Akaber, B. Moussa, M. Ghafouri, R. Atallah, B. L. Agba, C. Assi, and M. Debbabi, "CASeS: concurrent contingency analysis-based security metric deployment for the smart grid," *IEEE Transactions on Smart Grid*, vol. 11, no. 3, pp. 2676–2687, 2019.

[79] V. Venkataramanan, A. Hahn, and A. Srivastava, "CP-SAM: Cyber-physical security assessment metric for monitoring microgrid resiliency," *IEEE Transactions on Smart Grid*, vol. 11, no. 2, pp. 1055–1065, 2019.

[80] J. Ospina, V. Venkataramanan, and C. Konstantinou, "CPES-QSM: A Quantitative Method Toward the Secure Operation of Cyber–Physical Energy Systems," *IEEE Internet of Things Journal*, vol. 10, no. 9, pp. 7577–7590, 2022.

[81] A. Sahu, K. Davis, H. Huang, A. Umunnakwe, S. Zonouz, and A. Goulart, "Design of next-generation cyber-physical energy management systems: Monitoring to mitigation," *IEEE Open Access Journal of Power and Energy*, vol. 10, pp. 151–163, 2023.

[82] C. Alcaraz and J. Lopez, "Analysis of requirements for critical control systems," *International journal of critical infrastructure protection*, vol. 5, no. 3-4, pp. 137–145, 2012.

[83] C. Alcaraz and J. Lopez, "Addressing situational awareness in critical domains of a smart grid," in *Network and System Security: 6th International Conference, NSS 2012, Wuyishan, Fujian, China, November 21-23, 2012. Proceedings 6*, pp. 58–71, Springer, 2012.

[84] C. Alcaraz, E. E. Miciolino, and S. Wolthusen, "Structural controllability of networks for non-interactive adversarial vertex removal," in *International Workshop on Critical Information Infrastructures Security*, pp. 120–132, Springer, 2013.

[85] C. Alcaraz and S. Wolthusen, "Recovery of structural controllability for control systems," in *International Conference on Critical Infrastructure Protection*, pp. 47–63, Springer, 2014.

[86] C. Alcaraz, E. E. Miciolino, and S. Wolthusen, "Multi-round attacks on structural controllability properties for non-complete random graphs," in *Information Security*, pp. 140–151, Springer, 2015.

[87] C. Alcaraz, J. Lopez, and S. Wolthusen, "Policy enforcement system for secure interoperable control in distributed smart grid systems," *Journal of Network and Computer Applications*, vol. 59, pp. 301–314, 2016.

[88] C. Alcaraz and J. Lopez, "Safeguarding structural controllability in cyber-physical control systems," in *European Symposium on Research in Computer Security*, pp. 471–489, Springer, 2016.

[89] J. E. Rubio, C. Alcaraz, and J. Lopez, "Preventing advanced persistent threats in complex control networks," in *European Symposium on Research in Computer Security*, pp. 402–418, Springer, 2017.

[90] C. Alcaraz, J. Lopez, and K.-K. R. Choo, "Resilient interconnection in cyber-physical control systems," *Computers & Security*, vol. 71, pp. 2–14, 2017.

[91] C. Alcaraz and J. Lopez, "A cyber-physical systems-based checkpoint model for structural controllability," *IEEE Systems Journal*, vol. 12, no. 4, pp. 3543–3554, 2017.

[92] C. Alcaraz, J. E. Rubio, and J. Lopez, "Blockchain-assisted access for federated smart grid domains: Coupling and features," *Journal of Parallel and Distributed Computing*, vol. 144, pp. 124–135, 2020.

[93] E. D. Knapp and R. Samani, *Applied cyber security and the smart grid: implementing security controls into the modern power infrastructure*. Newnes, 2013.

[94] Y. Mo, T. H.-J. Kim, K. Brancik, D. Dickinson, H. Lee, A. Perrig, and B. Sinopoli, "Cyber–physical security of a smart grid infrastructure," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 195–209, 2011.

[95] "Special Report: The State of Software Supply Chain Security 2023 ." https://www.reversinglabs.com/reports/the-state-of-software-supply-chain-security. [Online; Accessed 11-June-2023].

[96] S. Fuloria, R. Anderson, F. Alvarez, and K. McGrath, "Key management for substations: Symmetric keys, public keys or no keys?," in *2011 IEEE/PES Power Systems Conference and Exposition*, pp. 1–6, IEEE, 2011.

[97] J. H. Lambert, J. M. Keisler, W. E. Wheeler, Z. A. Collier, and I. Linkov, "Multiscale approach to the security of hardware supply chains for energy systems," *Environment Systems and Decisions*, vol. 33, no. 3, pp. 326–334, 2013.

[98] L. Babun, H. Aksu, and A. S. Uluagac, "Identifying counterfeit smart grid devices: A lightweight system level framework," in *2017 IEEE International Conference on Communications (ICC)*, pp. 1–6, IEEE, 2017.

[99] M. Mylrea and S. N. G. Gourisetti, "Blockchain for supply chain cybersecurity, optimization and compliance," in *2018 resilience week (RWS)*, pp. 70–76, IEEE, 2018.

[100] A. Yeboah-Ofori, S. Islam, and A. Brimicombe, "Detecting Cyber Supply Chain Attacks on Cyber Physical Systems Using Bayesian Belief Network," in *2019 International Conference on Cyber Security and Internet of Things (ICSIoT)*, pp. 37–42, IEEE, 2019.

[101] A. Yeboah-Ofori and S. Islam, "Cyber security threat modeling for supply chain organizational environments," *future internet*, vol. 11, no. 3, p. 63, 2019.

[102] A. S. Sani, D. Yuan, K. Meng, and Z. Y. Dong, "Idenx: A Blockchain-based Identity Management System for Supply Chain Attacks Mitigation in Smart Grids," in *2020 IEEE Power & Energy Society General Meeting (PESGM)*, pp. 1–5, IEEE, 2020.

[103] L. Wang, S. Noel, and S. Jajodia, "Minimum-cost network hardening using attack graphs," *Computer Communications*, vol. 29, no. 18, pp. 3812–3824, 2006.

[104] F. Chen, L. Wang, and J. Su, "An efficient approach to minimum-cost network hardening using attack graphs," in *2008 The Fourth International Conference on Information Assurance and Security*, pp. 209–212, IEEE, 2008.

[105] T. Islam and L. Wang, "A heuristic approach to minimum-cost network hardening using attack graph," in *2008 New Technologies, Mobility and Security*, pp. 1–5, IEEE, 2008.

[106] N. C. Idika, B. H. Marshall, and B. K. Bhargava, "Maximizing network security given a limited budget," in *the Fifth Richard Tapia Celebration of Diversity in Computing Conference: intellect, initiatives, insight, and innovations*, pp. 12–17, 2009.

[107] M. Albanese, S. Jajodia, and S. Noel, "Time-efficient and cost-effective network hardening using attack graphs," in *IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2012)*, pp. 1–12, IEEE, 2012.

[108] B. Yigit, G. Gür, and F. Alagöz, "Cost-aware network hardening with limited budget using compact attack graphs," in *2014 IEEE Military Communications Conference*, pp. 152–157, IEEE, 2014.

[109] D. Borbor, L. Wang, S. Jajodia, and A. Singhal, "Diversifying network services under cost constraints for better resilience against unknown attacks," in *IFIP Annual Conference on Data and Applications Security and Privacy*, pp. 295–312, Springer, 2016.

[110] D. Borbor, L. Wang, S. Jajodia, and A. Singhal, "Securing networks against unpatchable and unknown vulnerabilities using heterogeneous hardening options," in *IFIP Annual Conference on Data and Applications Security and Privacy*, pp. 509–528, Springer, 2017.

[111] B. Yiğit, G. Gür, F. Alagöz, and B. Tellenbach, "Cost-aware securing of iot systems using attack graphs," *Ad Hoc Networks*, vol. 86, pp. 23–35, 2019.

[112] K. Zenitani, "A multi-objective cost–benefit optimization algorithm for network hardening," *International Journal of Information Security*, vol. 21, no. 4, pp. 813–832, 2022.

[113] S. Y. Enoch, J. Mendonça, J. B. Hong, M. Ge, and D. S. Kim, "An integrated security hardening optimization for dynamic networks using security and availability modeling with multi-objective algorithm," *Computer Networks*, vol. 208, p. 108864, 2022.

[114] D. Levshun and I. Kotenko, "A survey on artificial intelligence techniques for security event correlation: models, challenges, and opportunities," *Artificial Intelligence Review*, pp. 1–44, 2023.

[115] S. Moskal, S. J. Yang, and M. E. Kuhl, "Extracting and evaluating similar and unique cyber attack strategies from intrusion alerts," in *2018 IEEE International Conference on Intelligence and Security Informatics (ISI)*, pp. 49–54, IEEE, 2018.

[116] A. Nadeem, S. Verwer, S. Moskal, and S. J. Yang, "Enabling visual analytics via alert-driven attack graphs," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pp. 2420–2422, 2021.

[117] A. Nadeem, S. Verwer, and S. J. Yang, "Sage: Intrusion alert-driven attack graph extractor," in *2021 IEEE Symposium on Visualization for Cyber Security (VizSec)*, pp. 36–41, IEEE, 2021.

[118] A. Nadeem, S. Verwer, S. Moskal, and S. J. Yang, "Alert-driven attack graph generation using s-pdfa," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 2, pp. 731–746, 2021.

[119] B. Bhattarai and H. Huang, "SteinerLog: prize collecting the audit logs for threat hunting on enterprise network," in *Proceedings of the 2022 ACM on Asia Conference on Computer and Communications Security*, pp. 97–108, 2022.

[120] T. Li, Y. Jiang, C. Lin, M. S. Obaidat, Y. Shen, and J. Ma, "Deepag: Attack graph construction and threats prediction with bi-directional deep learning," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 1, pp. 740–757, 2022.

[121] E. R. Ling and M. Ekstedt, "A threat modeling language for generating attack graphs of substation automation systems," *International Journal of Critical Infrastructure Protection*, vol. 41, p. 100601, 2023.

[122] M. Strobel, N. Wiedermann, and C. Eckert, "Novel weaknesses in IEC 62351 protected Smart Grid control systems," in *Smart Grid Communications (SmartGridComm), 2016 IEEE International Conference on*, pp. 266–270, IEEE, 2016.

[123] T. Phinney, "IEC 62443: Industrial network and system security," *Last accessed July*, vol. 29, 2013.

[124] C. W. Johnson, "Why We Cannot (Yet) Ensure the Cybersecurity of Safety-Critical Systems," 2016.

[125] Y. Yang, H.-Q. Xu, L. Gao, Y.-B. Yuan, K. McLaughlin, and S. Sezer, "Multidimensional intrusion detection system for IEC 61850-based SCADA networks," *IEEE Transactions on Power Delivery*, vol. 32, no. 2, pp. 1068–1078, 2016.

[126] J. Wang and D. Shi, "Cyber-Attacks Related to Intelligent Electronic Devices and Their Countermeasures: A Review," in *2018 53rd International Universities Power Engineering Conference (UPEC)*, pp. 1–6, IEEE, 2018.

[127] O. Duman, M. Zhang, L. Wang, and M. Debbabi, "Measuring the security posture of IEC 61850 substations with redundancy against zero day attacks," in *2017 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pp. 108–114, IEEE, 2017.

[128] B. Moussa, M. Debbabi, and C. Assi, "A detection and mitigation model for PTP delay attack in an IEC 61850 substation," *IEEE Transactions on Smart Grid*, 2016.

[129] K. Huang, C. Zhou, Y.-C. Tian, W. Tu, and Y. Peng, "Application of Bayesian network to data-driven cyber-security risk assessment in SCADA networks," in *Telecommunication Networks and Applications Conference (ITNAC), 2017 27th International*, pp. 1–6, IEEE, 2017.

[130] N. Falliere, L. O. Murchu, and E. Chien, "W32. stuxnet dossier," *White paper, Symantec Corp., Security Response*, vol. 5, no. 6, p. 29, 2011.

[131] A. Ashok, M. Govindarasu, and V. Ajjarapu, "Online detection of stealthy false data injection attacks in power system state estimation," *IEEE Transactions on Smart Grid*, vol. 9, no. 3, pp. 1636–1646, 2016.

[132] Y. Wu, L. Nordström, and D. E. Bakken, "Effects of bursty event traffic on synchrophasor delays in IEEE C37. 118, IEC61850, and IEC60870," in *Smart Grid Communications (SmartGridComm), 2015 IEEE International Conference on*, pp. 478–484, IEEE, 2015.

[133] Y. Zhang, Y. Xiang, and L. Wang, "Power system reliability assessment incorporating cyber attacks against wind farm energy management systems," *IEEE transactions on smart grid*, vol. 8, no. 5, pp. 2343–2357, 2017.

[134] A. Clark and S. Zonouz, "Cyber-Physical Resilience: Definition and Assessment Metric," *IEEE Transactions on Smart Grid*, 2017.

[135] Z. Zhang, S. Gong, A. D. Dimitrovski, and H. Li, "Time synchronization attack in smart grid: Impact and analysis," *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 87–98, 2013.

[136] D.-Y. Yu, A. Ranganathan, T. Locher, S. Capkun, and D. Basin, "Short paper: detection of GPS spoofing attacks in power grids," in *Proceedings of the 2014 ACM conference on Security and privacy in wireless & mobile networks*, pp. 99–104, ACM, 2014.

[137] J. Northcote-Green and R. G. Wilson, *Control and automation of electrical power distribution systems*. CRC press, 2017.

[138] Y. Liu, R. Zivanovic, S. Al-Sarawi, C. Marinescu, and R. Cochran, "A synchronized event logger for substation topology processing," in *2009 Australasian Universities Power Engineering Conference*, pp. 1–6, IEEE, 2009.

[139] P. Castello, C. Muscas, P. A. Pegoraro, and S. Sulis, "Active Phasor Data Concentrator performing adaptive management of latency," *Sustainable Energy, Grids and Networks*, vol. 16, pp. 270–277, 2018.

[140] S. Liu, B. Chen, T. Zourntos, D. Kundur, and K. Butler-Purry, "A coordinated multi-switch attack for cascading failures in smart grid," *IEEE Transactions on Smart Grid*, vol. 5, no. 3, pp. 1183–1195, 2014.

[141] "Analysis of the Cyber Attack on the Ukrainian Power Grid." "https://ics.sans.org/media/E-ISAC SANS Ukraine DUC 5.pdf.". [Online; Accessed 21-March-2017].

[142] C.-C. Sun, J. Hong, and C.-C. Liu, "A co-simulation environment for integrated cyber and power systems," in *2015 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pp. 133–138, IEEE, 2015.

[143] C.-C. Sun, A. Hahn, and C.-C. Liu, "Cyber security of a power grid: State-of-the-art," *International Journal of Electrical Power & Energy Systems*, vol. 99, pp. 45–56, 2018.

[144] D.-T. May and M. Massoud, "On the Relation Between the Factor of Safety and Reliability," *ASME Journal ofEngineering for Industry*, pp. 852–857, 1974.

[145] L. Bilge and T. Dumitraş, "Before we knew it: an empirical study of zero-day attacks in the real world," in *Proceedings of the 2012 ACM conference on Computer and communications security*, pp. 833–844, 2012.

[146] P. Giannopoulos and R. C. Veltkamp, "A pseudo-metric for weighted point sets," in *European Conference on Computer Vision*, pp. 715–730, Springer, 2002.

[147] W. Nzoukou, L. Wang, S. Jajodia, and A. Singhal, "A unified framework for measuring a network's mean time-to-compromise," in *Reliable Distributed Systems (SRDS), 2013 IEEE 32nd International Symposium on*, pp. 215–224, IEEE, 2013.

[148] "CVSS v2 Calculator." 'https://nvd.nist.gov/vuln-metrics/cvss/v2-calculator" [Online; Accessed 07-March-2018].

[149] P. Mell, K. Scarfone, and S. Romanosky, "Common vulnerability scoring system," *IEEE Security & Privacy*, vol. 4, no. 6, pp. 85–89, 2006.

[150] "Circl CVE Search." https://cve.circl.lu/. [Online; Accessed 17-August-2019].

[151] "Download CVE List." https://cve.mitre.org/data/downloads/index.html. [Online; Accessed 18-Dec-2023].

[152] "The 2003 Northeast Blackout–Five Years Later." https://www.scientificamerican.com/article/2003-blackout-five-years-later/. [Online; Accessed 13-April-2017].

[153] "IEEE 14-Bus System." https://icseg.iti.illinois.edu/ieee-14-bus-system/. [Online; Accessed 9-February-2020].

[154] X. Li and W. Ding, "On allocation of active redundancies to systems: a brief review," *Stochastic orders in reliability and risk: In honor of Professor Moshe Shaked*, pp. 235–254, 2013.

[155] "SolarWinds Supply-Chain Attack." https://www.sans.org/blog/what-you-need-to-know-about-the-solarwinds-supply-chain-attack/. [Online; Accessed 9-January-2021].

[156] "4-bus, 2-generator transmission network model." https://egriddata.org/dataset/4-bus-2-generator-transmission-network-model. [Online; Accessed 24-December-2022].

[157] S. L. Thomas and A. Francillon, "Backdoors: Definition, deniability and detection," in *Research in Attacks, Intrusions, and Defenses: 21st International Symposium, RAID 2018, Heraklion, Crete, Greece, September 10-12, 2018, Proceedings 21*, pp. 92–113, Springer, 2018.

[158] K. Tazi, F. Abdi, and M. F. Abbou, "Review on cyber-physical security of the smart grid: Attacks and defense mechanisms," in *2015 3rd International Renewable and Sustainable Energy Conference (IRSEC)*, pp. 1–6, IEEE, 2015.

[159] "Supply Chain Solutions for Smart Grid Security: Building on Business Best Practices." https://usresilienceproject.org/wp-content/uploads/2014/09/report-Supply_Chain_Solutions_for_Smart_Grid_Security.pdf. [Online; Accessed 9-January-2021].

[160] S. Y. A. Fayi, "What Petya/NotPetya ransomware is and what its remidiations are," in *Information Technology-New Generations*, pp. 93–100, Springer, 2018.

[161] D. Tychalas, A. Keliris, and M. Maniatakos, "LED alert: Supply chain threats for stealthy data exfiltration in industrial control systems," in *2019 IEEE 25th International Symposium on On-Line Testing and Robust System Design (IOLTS)*, pp. 194–199, IEEE, 2019.

[162] E. Molina, E. Jacob, J. Matias, N. Moreira, and A. Astarloa, "Using software defined networking to manage and control IEC 61850-based systems," *Computers & Electrical Engineering*, vol. 43, pp. 142–154, 2015.

[163] T. Bartman and K. Carson, "Securing critical industrial systems with SEL solutions," *Puttman, Washington*, 2015.

[164] "CVSS Version 3.1 Release." https://www.first.org/cvss/specification-document. [Online; Accessed 23-January-2021].

[165] "The Open Group Trusted Technology Forum." https://www.opengroup.org/membership/forums/trusted-technology-forum/trusted. [Online; Accessed 9-January-2021].

[166] J. Boyens, A. Smith, N. Bartol, K. Winkler, A. Holbrook, and M. Fallon, "Cybersecurity supply chain risk management practices for systems and organizations," tech. rep., NIST, 2022.

[167] O. Duman, M. Zhang, L. Wang, M. Debbabi, R. Atallah, and B. Lebel, "Factor of Security (FoS): Quantifying the Security Effectiveness of Redundant Smart Grid Subsystems," *IEEE Trans. on Dependable and Secure Computing*, 2020.

[168] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[169] L. Thurner, A. Scheidler, F. Schafer, J. H. Menke, J. Dollichon, F. Meier, S. Meinecke, and M. Braun, "pandapower - an Open Source Python Tool for Convenient Modeling, Analysis and Optimization of Electric Power Systems," *IEEE Trans. on Power Systems*, 2018.

[170] S. Fliscounakis, P. Panciatici, F. Capitanescu, and L. Wehenkel, "Contingency ranking with respect to overloads in very large power systems taking into account uncertainty, preventive, and corrective actions," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4909–4917, 2013.

[171] R. Dewri, N. Poolsappasit, I. Ray, and D. Whitley, "Optimal security hardening using multi-objective optimization on attack tree models of networks," in *Proceedings of the 14th ACM conference on Computer and communications security*, pp. 204–213, 2007.

[172] L. Mieritz and B. Kirwin, "Defining Gartner total cost of ownership," *L. Mieritz, B. Kirwin*, 2005.

[173] J. Blank and K. Deb, "pymoo: Multi-Objective Optimization in Python," *IEEE Access*, vol. 8, pp. 89497–89509, 2020.

[174] S. Jajodia, S. Noel, P. Kalapa, M. Albanese, and J. Williams, "Cauldron mission-centric cyber situational awareness with defense in depth," in *2011-MILCOM 2011 Military Communications Conference*, pp. 1339–1344, IEEE, 2011.

[175] P. CODE, "Communication networks and systems for power utility automation–Part 6: Configuration description language for communication in electrical substations related to IEDs," 2010.

[176] P. Johnson, R. Lagerström, and M. Ekstedt, "A meta language for threat modeling and attack simulations," in *Proceedings of the 13th International Conference on Availability, Reliability and Security*, pp. 1–8, 2018.

[177] P. Silveira, E. F. Silva, A. Galletta, and Y. Lopes, "Security analysis of digitized substations: A systematic review of GOOSE messages," *Internet of Things*, p. 100760, 2023.

[178] J. Hoyos, M. Dehus, and T. X. Brown, "Exploiting the GOOSE protocol: A practical attack on cyber-infrastructure," in *2012 IEEE Globecom Workshops*, pp. 1508–1513, IEEE, 2012.

[179] A. NETRESEC, "Capture files from 4SICS Geek Lounge," *Retrieved January*, vol. 14, p. 2017, 2015.

[180] S. Adepu, N. K. Kandasamy, and A. Mathur, "Epic: An electric power testbed for research and training in cyber physical systems security," in *Computer Security: ESORICS 2018 International Workshops, CyberICPS 2018 and SECPRE 2018, Barcelona, Spain, September 6–7, 2018, Revised Selected Papers 2*, pp. 37–52, Springer, 2019.

[181] P. P. Biswas, Y. Li, H. C. Tan, D. Mashima, and B. Chen, "An attack-trace generating toolchain for cybersecurity study of IEC61850 based substations," in *2020 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, pp. 1–7, IEEE, 2020.

[182] S. Verwer and C. A. Hammerschmidt, "Flexfringe: a passive automaton learning package," in *2017 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pp. 638–642, IEEE, 2017.

[183] A. Reeves, P. Delfabbro, and D. Calic, "Encouraging employee engagement with cybersecurity: How to tackle cyber fatigue," *SAGE open*, vol. 11, no. 1, p. 21582440211000049, 2021.