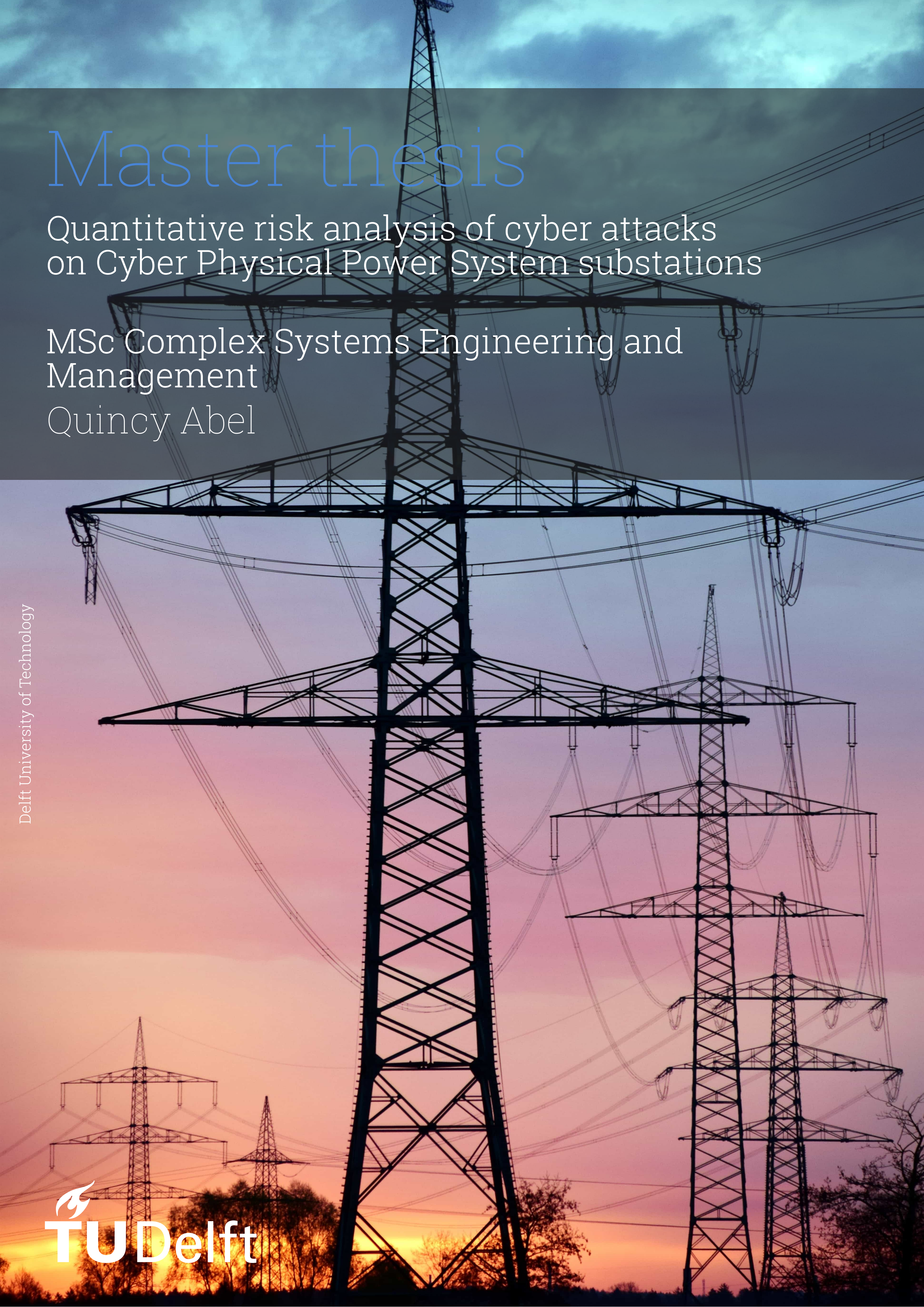


Master thesis

Quantitative risk analysis of cyber attacks on Cyber Physical Power System substations

MSc Complex Systems Engineering and Management

Quincy Abel

Delft University of Technology

Master thesis

Quantitative risk analysis of cyber attacks on Cyber Physical Power System substations

by

Quincy Abel

submitted to Delft University of Technology in partial fulfilment of the requirements
to obtain the degree of Master of Science
in Complex Systems Engineering and Management
at the Delft University of Technology,
Faculty of Technology Policy and Management,
to be defended publicly on Wednesday July 5, 2023 at 15:00.

Student number:	5411769	
Project duration:	February 1, 2023 – July 5, 2023	
Thesis committee:	Prof. dr. ir. P. H. A. J. M. van Gelder,	TU Delft, First supervisor
	Dr. ir. R. van Bergem,	TU Delft, Second supervisor
	Dr. A. I. Stefanov,	TU Delft, Advisor
	Ir. I. Semertzis,	TU Delft, Advisor

Management Summary

The modern power grid is becoming more susceptible to cyber-attacks due to an increase in digitalization, leading to a larger attack surface for malicious actors to attack. Such attacks on critical infrastructure could lead to partial power outages, minor societal disruption, or in the worst-case scenario, a rolling black-out in which the entire country has no access to electricity. Electrical utility companies can decrease the likelihood of a successful cyber-attack on the Cyber Physical Power System (CPPS) – consisting of the physical power grid, and vulnerable Information Technology (IT) and Operational Technology (OT)- by implementing cyber security interventions. Investing in these cyber security mechanisms is not cheap, which is why it is expected to have a certain return on investment. However, it is hard to quantify the effects of prospective cyber security investments. The main research question of this study is: *"To what extent can cyber security measures decrease the risk of cyber attacks on CPPS substations?"* This research question is answered by means of an implicitly mixed research approach that uses computer-assisted attack tree modelling and Monte Carlo simulation. The model is based on the publicly available technical system information of known suppliers of relevant substation components and other documentation acquired by means of multiple literature studies and document analyses. The change in likelihood and subsequent risk has been studied by extensively modelling the possible attack paths of a digital substation. This has been combined with financial analysis in the form of a societal cost-benefit analysis. As a result, potential cyber security investments can be evaluated on their merits in the form of risk reduction and their required costs as expressed in dollars. The contribution of the performed research to science is the elaboration of existing models to more accurately represent reality, and simultaneously provide the cyber security decision-making process with a tool that provides guiding Key Performance Indicators (KPIs).

This study has shown that suggested measures from the quantified model are able to increase the TTC_{avg} needed by malicious actors to reach their intended target, and therefore cause a decrease in likelihood and subsequent risk of the studied scenarios. An important finding of this study emphasizes the need for extensive attack path modelling. This finding was the fact that the application of some well-intended countermeasure (such as remote-attestation), might have no significant effect on the likelihood and risk of a certain scenario at all, but only changes the dominant attack path. While the constructed quantified model, as proposed in this study, is able to provide quantified insights into the effects of proposed cyber security investments, it is merely a simplified tool that should be expanded upon to generate more accurate insights.

Besides the aforementioned there have been additional findings from this study. Such as a list of weaknesses in the current state of digital substation cyber security. This list has been created by an extensive document analysis of over 40 sources. Also, an overview of 23 different possible cyber security interventions has been compiled by a systemic literature review of over 16 sources.

According to the quantified model, a reduction (between 21.8% and 93%) in the total risk of certain attack scenarios against digital substation by malicious actors can be achieved. The costs for these possible risk reductions range between $28 thousand for a honeypot deception system and $413 thousand for a combination of all the simulated countermeasures. These countermeasures could, in comparison to a base case with no protection, potentially reduce the total risk by an amount between $3.7 billion and $15.9 billion. According to the general societal cost-benefit analyses, the best Retun-on-Investment (ROI)/cost-effectiveness of investment is the investment in a honeypot (scenario 5) which has an ROI of 247,390, and the least cost-effective is the investment in remote attestation (scenario 4), which has an ROI of -2,066.

Altogether, this study has shown that there is added value in using a simplified quantified model to aid in decision-making for digital substation cyber security investments aimed at risk reduction.

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

Abbreviation	Definition
CIA	Confidentiality Integrity and Availability
CI	Critical Infrastructure
CPPS	Cyber-Physical Power System
CVE	Common Vulnerabilities and Exposures
DNP3	Distributed Network Protocol 3
DoS	Denial of Service
EEMCS	Electrical Engineering, Math and Computer Science
EI	Electromagnetic Interference
DMZ	Demillitarized Zone
FDIA	False Data Injection Attack
GIS	Gas Insulated Switchgear
GPS	Global Positioning System
HMI	Human Machine Interface
IDS	Intrusion Detection System
IED	Intelligent Electronic Devices
IEPG	Intelligent Electrical Power Grids
IDS	Intrusion Detection System
I/O	Input/Output
IT	Information Technology
KPI	Key Performance Indicator
KSF	Kill-chain Scoping Factor
MFA	Multi-Factor Authentication
MTTD	Mean Time To Detect
MTTR	Mean Time To Restore
MU	Merging Unit
NVD	National Vulnerability Database
NGO	Non-Governmental Organisation
OT	Operational Technology
PLC	Programmable Logic Controller
PRISMA	Preferred Reporting Items for Systematic reviews and Meta-Analyses
QoS	Quality of Service
RA	Remote Atttestation
RAS	Remedial Action Scheme
RMTTD	Relative Mean Time To Detect
ROI	Return-on-Investment
RTU	Remote Terminal Unit
SCADA	Supervisory Control And Data Acquisition
(S)CBA	(Societal) Cost Benefit Analysis
SCEDM	Supply Chain and External Dependencies Management
SoC	System on Chip
TCP/IP	Transmission Control Protocol - Internet Protocol
TSO	Transmission System Operator
TTC	Time To Compromise

Abbreviation	Definition
TTE	Time To Exploit
VoLL	Value of Lost Load
WAN	Wide Area Network
WLS	Weighted Least Square

1 Problem introduction

Charging your phone at night before you go to sleep, keeping your laptop powered during lectures, and monitoring a patient's heartbeat in a hospital. All these processes are taken as given in first-world countries, but all rely on a functioning electricity grid. When the power supply is interrupted in a country, the society in that country seizes to function as is. Therefore, it is not surprising that the energy grid is considered a critical infrastructure (CI). The European Union has defined CI as follows: *"An asset, system or part thereof located in Member States which is essential for the maintenance of vital societal functions, health, safety, security, economic or social well-being of people, and the disruption or destruction of which would have a significant impact in a Member State as a result of the failure to maintain those functions."* [1]. Modern CI is not merely controlled offline, but makes use of smart applications such as Intelligent Electronic Devices (IEDs) which are controlled by Operational Technology (OT) in order to supply their critical services [2]. The electricity grid consists of multiple layers: the physical grid, the OT systems, and the IT systems. Together they form a system-of-systems. OT systems are susceptible to cyber-attacks by malicious actors which leads to disastrous outcomes. In the winter of 2015, a blackout occurred in Ukraine as a result of a successful cyber-attack on three electric power plants, leaving 225 thousand people without power for over five hours. This was the first documented power outage as a consequence of a cyber-attack [3]. According to [4] the impact of cyber-security incidents will reach over $50 billion by 2023.

Utility companies can decrease the likelihood of a successful cyber-attack by implementing cyber security interventions such as Multi-Factor Authentication (MFA), firewalls, Intrusion Detection Systems (IDS), and organizational protocols. Investing in the aforementioned costs large amounts of money, which is expected to have a certain return on investment. The return can be in either monetary form or in the form of increased security. But estimating and quantifying the effects of cyber security investment is hard, as there are many uncertainties, and quantification of qualitative indicators might be hard to execute. Investing in cyber security for the electricity grid is made even more complex considering the fact that most utility companies in the European Union are private companies that are (partly) subsidized by their respective states, so under- or overinvestment in cyber security measures by these companies will lead to costs to be paid by the taxpayer [5]. Current cyber security literature is mostly focused on either the qualitative or quantitative aspects of the problem. The aim of this study is to combine both qualitative and quantitative aspects of cyber security (investing) while keeping relevancy for real-world applications by utility companies.

The fact that this problem crosses the electrical, ICT, cyber security, and energy policy domains, combined with a complex system-of-systems environment, makes this a suitable CoSEM thesis. The goal is to quantify the change in risk as a result of investments in policy interventions. This quantification could help organizations guide and/or justify their possible investments in cyber security interventions. Timely and adequate investments in the cyber security of the cyber-physical system of the electricity grid are needed to protect the critical infrastructure from malicious actors.

In Chapter 2, State-of-the-art, the research question will be constructed from the resulting knowledge gap. Chapters 3 and 4 discuss the research approach and methodology respectively. In chapter 5 the digital substation is examined. Chapter 6 describes the model setup and relevant calculations. The simulation scenarios are given in Chapter 7, and the results are discussed in Chapter 8. Finally, the report ends with a conclusion.

2
State-of-the-art

The aim of this chapter is to identify a knowledge gap in academic papers regarding cyber security of CPPSs. Ensuring that there is a knowledge gap is required in order to guarantee that the research of the thesis yields a significant contribution to science and that one does not reinvent the wheel. This knowledge gap is identified by means of a PRISMA literature review. The results of the literature review are used to construct the main research question.

2.1. Literature review methodology

The literature review has been executed according to the PRISMA methodology. The methodology itself, the execution, and the results are further discussed in this section.

2.1.1. PRSIMA literature review

The methodology used for the literature review is the Preferred Reporting Items for Systematic reviews and Meta-Analyses (PRISMA) method. This method for literature reviewing is used internationally and serves as a framework that permits the readers to determine the legitimacy of the outcome that is produced by the literature review [6]. The framework contains four steps: Identification, screening, eligibility, and inclusion. All of these steps are shown in Figure 1, which displays the outcomes of every previously mentioned step. All potentially pertinent records (i.e. articles, papers, etc.) found through database searches and/or other means, are gathered and documented during the identification step. Duplicate records are deleted when encountered. The remaining records will then be examined and, if deemed irrelevant, are expunged. Subsequently, a full-text eligibility assessment of the remainder of the records is executed, followed by the rejection of expendable entries.

2.1.2. Execution of PRISMA method

In the first step, identification, 14 papers from two databases (Google Scholar and WorldCat), but three queries (see Table 2.1) have been identified, based on first impressions of the many generated query results. One record has been obtained from expert consultation and one duplicate has been deleted after finding little relevance to the main research question. This resulted in the same number of entries being screened in the screening step, which led to the emission of one paper due to limited affiliation with the subject under study. The records that were left – 13 to be precise – have been studied in detail and prompted the rejection of two papers. In the end, 11 papers have been included in this meta-analysis via database searches. A graphic overview of the PRISMA framework and the number of records used is displayed in Figure 2.1. For this literature review, three similar queries have been used. The queries are shown in Table 2.1.

2.1.3. Search method

Besides these 11 hits through database searches and expert consultation, the snowballing method has been applied to the paper by [7]. Resulting in an additional three hits. Ultimately this gives 14 records to be used in the meta-analysis in total.

Table 2.1: Used search queries PRISMA literature review

Query	Database	First picks
cyber AND attack AND electricity AND grid AND modelling	Google Scholar	6
cyber AND attack AND electricity AND grid AND modelling AND ”substation”	Google Scholar	3
cyber physical power system AND cyber attack AND substation AND modelling	Worldcat TU Delft	5

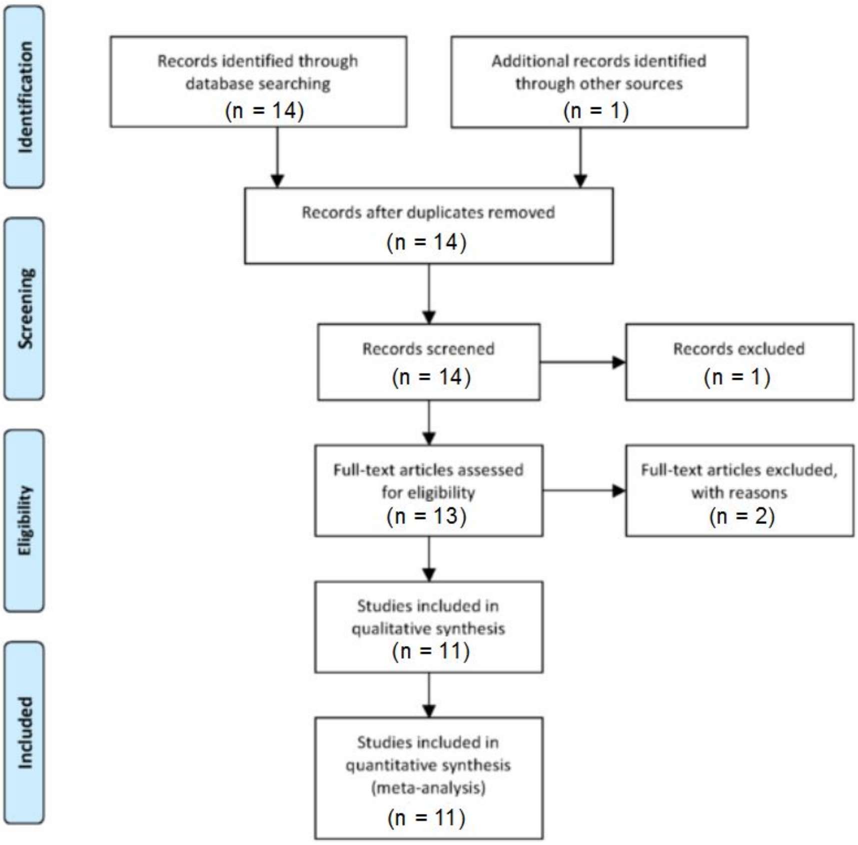

Identification
Screening
Eligibility
Included
Records identified through database searching (n = 14)
Additional records identified through other sources (n = 1)
Records after duplicates removed (n = 14)
Records screened (n = 14)
Records excluded (n = 1)
Full-text articles assessed for eligibility (n = 13)
Full-text articles excluded, with reasons (n = 2)
Studies included in qualitative synthesis (n = 11)
Studies included in quantitative synthesis (meta-analysis) (n = 11)

Figure 2.1: PRISMA steps results and schematic overview

2.2. Results

An overview of the literature review findings is given in Table 2.2. This table shows the relevant summarized (read: key takeaways) findings for this thesis and the corresponding author(s). Now that the findings of every record are summarized, they can be analyzed in order to distil the knowledge gap(s).

2.2.1. Imperfect models

The literature review exposed an academic knowledge gap in the modelling of CPPSs. In recent literature, a common conception became apparent: in order to correctly study cyber security of CPPSs an extensive model is required [8], which should keep up with the growing number and intensity of CPPSs and the further adoption of the IEC 61850 protocol [9][10]. This model should capture all the complex interactions and dynamics of the real-life CPPS [11] [12] [13], as the current models, which mostly use N-1 contingency analysis, are not suitable for this task [7]. Current models lack a detailed description of assets and their corresponding protective mechanisms [14], complex attacker behaviour, and the costs of the defensive mechanisms [15] [16].

2.2.2. System-of-system needs

Crafting extensive models requires combining the work of different people who focus on a specific part of the system-of-systems to eventually reach a model that captures all the needed characteristics. Substations are an important part of the CPPS system-of-systems, and much work can be done in further modelling these subsystems, such as more realistic model assumptions [17], increased attack graph detail, and metric assessment [18]. Providing these would aid in fulfilling the need for cyber security formats for CPPS experts [17] [19].

Table 2.2: Findings of the PRISMA literature review

Author:	Findings:	Deficiencies:
[8]	Impact analysis framework focused on model synthesis stage.	Need for large dynamical systems framework model.
[9]	Overview of barriers and policy drivers in CPPS security.	Simulation and modelling techniques should keep up with the growing speed of CPS to aid in increasing grid resilience.
[7]	Comprehensive review of state-of-the-art simulation, modelling and analysis methods for CPPS. Division of modelling techniques into three broad classes: interaction, interdependent and interconnection modelling.	Need for more financial analysis (at the national level if possible) and a more interconnected model analysis for the communication network of generation, transmission and distribution. N-1 contingency analysis is not suitable enough for CPPS, focus should be on stochastic contingency analysis.
[7] snowballing:[11]	Hybrid systems are used to combine/capture the behaviour of continuous and discrete variables in CPPS.	N.A.
[7] snowballing:[13]	modelling the physical impact of cyber-contingencies, so the effect of cyber on the physical.	Need more large scale model testing.
[7] snowballing:[12]	modelling dependency of cyber on physical and vice-versa (through degree-betweenness)	N.A.
[17]	Substation topology model of substation defence.	Need for a common format for experts, realistic models and realistic assumptions
[14]	Multistate (Markov) model of substation CS with consequences	limited defence mechanisms (IDS, password models and firewall)
[15]	Detailed Markov decision Process (MDP) models of substations are used to obtain optimal action policies for attackers and defenders.	Costs of defensive measures and complicated evasions of sophisticated attackers are left out
[18]	Methods for modelling CPS cascading failure by cyber-attacks and modelling attack graphs of digital substations for security assessment, combined with Time To Compromise (TTC) calculation algorithm.	Limited detail in attack graph and metrics with regards to cyber security related impact (loss of data and information theft)
[20]	Detailed description of how TTC is calculated and the benefits of using TTC for risk reduction estimations.	Limitations in distribution of vulnerabilities and vulnerability dependency, due to lack of detail of constructed attack graphs.
[10]	Describes the weaknesses of IEC 61850 SV and GOOSE protocols and how these can be negated by IDS for substations.	Stresses the commercial unavailability of such detectors/tools and increased attention on cyber security as IEC 61850 adoption increases.
[19]	Combines steady-state probabilities for switching (spoof) attacks with potential electricity losses in a generalized stochastic Petri net for various substation contingencies.	Need for further research into switching attacks for e.g. estimating cyber insurance premiums.
[16]	Indicates that parts of current literature propose a power system security framework. Uses a Petri net to model protection processes of spoof attacks on substations.	Only focused on qualitative aspects.

2.2.3. Focus: Substation cyber security

The paper regarding quantitative risk assessment of cyber-attacks on CPPSs using attack graphs by [18] provides a good model for analyzing the cyber security of digital substations. This model consists of a novel Time To Compromise (TTC) calculation algorithm that uses attack graphs for its calculations. The work by [18] contains some room for improvement, which stems from the limitations of the author's study. These limitations, and thus the fillable knowledge gap, are: how and where to apply security measures such as firewalls, IDS, MFA, organizational, enhancing the impact assessment on the communications layer of the CPPS, and lastly updating the TTC algorithm. In this thesis, the contribution to science will be the further elaboration on (one of) these academic caveats.

The identified academic literature gap and the studied actual concepts have led to the following main research question:

“To what extent can cyber security measures decrease the risk of cyber-attacks on CPPS substations?”

3
Research approach

This chapter describes the structure of the research that is executed during the thesis research period. Both the chosen research approach and the sub-questions, which follow from the main research question, are discussed.

3.1. Main research question and sub-questions

The main research question of this thesis, as constructed in the last paragraph per literature review, goes as follows: *"To what extent can cyber security measures decrease the risk of cyber-attacks on CPPS substations?"*

Answering this main research question is not an easy task. There are many facets and factors that have to be known and taken into account before an answer could be provided. However, this also makes the subject of this study a perfect subject for a CoSEM master thesis study. The engineering aspects of securing a physical power system (with an added communication layer) with cyber security measures in an economical manner, require different perspectives to be combined. The human behaviour of attackers and defenders, as well as the system behaviour, form the socio-technical part, while the distribution of benefits and liabilities lies in the institutional economics domain. The system itself is owned by businesses, but they provide an essential public service, creating a tangent plane where public and private interests intersect. This myriad of elements makes the challenge that faces this system-of-systems difficult to tackle. In order to make the main research question more manageable, it is divided into the following sub-questions:

1. What is the current state of substation cyber security and what are the weaknesses?
   (a) How is the (sub)system set up?
   (b) What is the level of interconnectedness?
   (c) Who are the relevant actors/stakeholders?
   (d) What are the most common attack types?
2. What are possible cyber security policies/interventions that can be implemented to decrease the likelihood of a successful attack on substations?
   (a) Looking at technological, organizational and/or hybrid policies
3. How do the identified policies/interventions influence the likelihood of a successful attack on a substation?
4. What are the (financial) consequences of a successful substation attack on the whole grid?

The first sub-question provides a comprehensive overview of the system-of-systems under study and its environment, which is required for the analysis of the system. The second sub-question aids in exploring the possible solution space for the given challenge. Answering the third sub-question yields

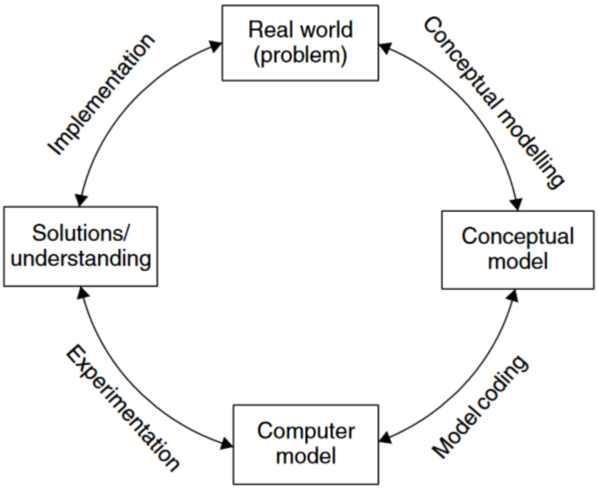


Figure 3.1: modelling outline for simulation studies (adapted by [24])

information about the effectiveness of the designed policies and/or interventions on the likelihood of a successful attack on a substation. Finally, the last sub-question can provide information about the impact of a successful attack on the whole grid. The sum of the answers to these sub-question gives an answer to the main research question.

3.2. The (implicitly mixed) research approach

Studying the dynamic behaviour of a complex system such as a real-life operational substation – which is part of a system-of-systems – would take an enormous amount of resources such as time and money. Besides that, being able to control the experimental conditions when comparing alternatives and their outcomes is very important [21]. [22] Argues that besides prediction, demonstrating trade-offs and/or suggesting efficiencies is another reason to build a model. For these reasons alone, it is necessary to use modelling within this study to answer the main research question. A simulation study consists of many different steps or processes and sub-processes. As mentioned by [21] many authors have written about these key processes but their common denominator is the fact that they outline a certain set of processes that have to be performed. For this study, the (slightly) modified approach by [23] is used. This modelling outline (Figure 3.1) consists of four main stages and four processes that allow for movement between these stages. The main stages (shown in the boxes) are also important deliverables.

- The *conceptual model:* this is a descriptive version of that which is going to be modelled;
- The *computer model:* the model which is simulated by the computer;
- *Solutions and/or understanding:* useful results which are derived from the experimentation;
- *Improvement for the real world (problem):* is acquired by the implementation of the obtained solutions and/or understanding

Important processes are situated between the stages. These are discussed in the following paragraphs.

3.2.1. Conceptual modelling

In this process, the nature of the (real world) problem is tried to be understood and a model which is able to tackle the problem is proposed. It consists of four sub-processes [21]:

- Understanding the problem situation;
- Determining the objective of the model;
- Designing the conceptual model: model content, inputs and outputs;
- Collecting and analyzing the data required to develop the model

Collecting and analyzing the data for the conceptual modelling, as well as understanding the problem situation and determining the objective of the model requires qualitative research. So within the modelling approach, there is an implied qualitative approach.

3.2.2. Model coding

The model coding process consists of translating the conceptual model into a computer model by means of some sort of coding.

3.2.3. Experimentation

After the model is complete, simulation experiments are executed to gain insights into real-world solutions and help answer the main research question. Experimentation requires certain experiments to be set up in advance which can be executed to get these results. In this process, one should consider that the obtained results are appropriately accurate, the robustness is examined (sensitivity analysis) and a rigorous exploration of the potential solution space is performed (solution space search). The experimentation process implies a certain quantitative aspect in the modelling approach[21].

3.2.4. Implementation

Implementing the results is the last process. In a real-life scenario, it would entail the implementation of the found solutions into practice. However, in this case, the implementation comes in the form of policy advice which could consist of some helpful framework to be put into practice by professionals[21].

3.2.5. Iterative nature

Throughout all and alongside the different processes, verification and validation activities are performed to ensure model validity [24]. The modelling and validation processes cannot be separated and could be seen as a single modelling-validating process [23]. So the study follows a modelling, approach with an implicit quantitative and qualitative approach.

4 Methodology

Answering the sub-questions, which are discussed in Chapter 3, requires data. This data is gathered by means of research methods. Choosing the right method is of great importance as wrong research methods could lead to unusable data for answering the main research question. In this paragraph, the research methods that have been used during the research are discussed and arguments are given for why the given research method is suitable for answering the relevant sub-question. First, the type of information/data that is required for each sub-question is discussed. Subsequently, the information/data types are matched with a suitable research method. In the end, the research approach, sub-questions, research methods, and data analysis tools are combined into a single research flow diagram which shows the relations between the before-mentioned parts. An overview of the datatypes, research methods, and analysis tools is also given in Appendix A.

4.1. Sub-question 1

The first sub-question of the research goes as follows:

What is the current state of substation cyber security and what are the weaknesses?

4.1.1. Datatype

This sub-question consists of multiple aspects. In order to analyze the current state of substation cyber security the setup of such a system has to be clear. For this reason, the first sub-question aspect has to answer the question: “How is the (sub)system set up?”. This requires technical data about what components are in a substation and what vulnerabilities these components have.

The substation, consisting of its components, is not operating in isolation but is connected to components of other subsystems, such as connection to OT and IT. These connections should be mapped, as these connections are also entry points for malicious actors to gain access through [25]. This is another sub-question aspect that should answer the question: “What is the level of interconnectedness?”. Acquiring both technical and organizational data is required for answering this aspect. Technical data is required about the components themselves, but the connection is of a more techno-organizational nature.

The interconnectedness allows for interaction with other actors – such as the previously mentioned malicious actors -, but other stakeholders might also influence the system under study. In order to understand the bigger picture of this complex system, these actors and stakeholders and their drivers have to be identified. The sub-question layer “Who are the relevant actors/stakeholders?” covers this aspect. Analyzing this aspect requires organizational data.

In order to model the defence of the substations against certain attacks, the types of cyber attacks have to be researched and defined. This is covered in the sub-question layer “What are the most common attack types?”. This layer requires quantitative information about the distribution of attack types on substations and qualitative information about how such attacks are executed. The answers to this sub-question serve as input for the conceptual model.

4.1.2. Document analysis, and Mendeley Reference Manager

All of the aspects of this sub-question can be answered by means of desk research. For this sub-question, just one type of desk research will be used, namely: document analysis. Document analysis is used to attain understanding, elicit meaning and finally develop empirical knowledge about a certain subject, through examining and interpreting documents [26]. Technical and organizational data can be acquired through document analysis of technical documents of component providers or certain white papers and possible research papers. The data will be analyzed with aid from the Mendeley Reference Manager which keeps track of all the found documents. This research method is constrained by the availability of the documents needed for the study. Also, a drawback of this method is that there might be too many different types of documents. If that is the case a select subsection of the documents should be analyzed to keep the research method feasible within the given timeframe.

4.2. Sub-question 2

The second sub-question of the research goes as follows:

What are possible cyber security policies/interventions that can be implemented to decrease the likelihood of a successful attack on substations?

4.2.1. Datatype

Answering this sub-question requires information about the different types of policies that could be implemented. These policies could be technical, organizational, cultural, or hybrid in nature [27]. Subsequently, the type of information required to answer this sub-question is of similar essence. The information obtained in this sub-question serves as input for the conceptual model.

4.2.2. Document analysis, literature review, and Mendeley Reference Manager

The second sub-question can also be answered by means of desk research. Two types of desk research will be used, namely: document analysis and a literature review. Document analysis has been briefly explained in the paragraph regarding the research method of sub-question 1, and the workings of a literature review are discussed in paragraph 2.1.1. The literature review, in this case, is performed to gain insight into what kind of cyber security policies/interventions can be used for decreasing the likelihood of a successful attack on a substation. Most likely the literature review will focus on published research papers, as these discuss the latest developments in cyber security, but usually also contain detailed information on how these policies/interventions could be modelled. The data is analyzed with aid from the Mendeley Reference Manager which keeps track of all the found documents and the findings of the literature review will be represented in a clear table. This research method deals with the same constraints and drawbacks as the research method of sub-question 1.

4.3. Sub-question 3

The third sub-question of the research goes as follows:

How do the identified policies/interventions influence the likelihood of a successful attack on a substation?

4.3.1. Datatype

Assessing the influence of identified policies/interventions on the likelihood of a cyber-attack requires building attack graphs and applying probabilistic theory to said graphs. These attack graphs are partly based on the information which is acquired in the previous sub-questions. The data which has to be acquired in this sub-question is data, consisting of theory-based formulas and system characteristics, to put in the attack graph model.

4.3.2. Modeling (attack graph method), and Python analysis

Sub-question 3 is a clear modeling sub-question. The goal is to discover how different policies/interventions affect the likelihood of a successful attack on a substation. For this reason, a modeling research method is used. In this study, the attack graph method is chosen, because the authors [28], upon whose work this study is going to elaborate, also used this method. In order to ensure a good fit, a similar method must be used. Besides the aforementioned reason, the attack graph method is suitable because it can visually represent dependencies (illustrated as edges) between components (illustrated as nodes). These graphs show possible attack paths that a malicious actor could use [29]. The data of the model is analyzed by means of a Python model with NetworkX and other libraries, as will be further discussed in the data analysis section of the last sub-question. A constraint of this method is that the attack graph could become too large to configure manually within the given time frame. The drawbacks of this method are that it might require such a high level of expertise, that it is not realistic that the author could model what was intended to be modeled. In both cases, simplifications could offer some alleviation.

4.4. Sub-question 4

The last sub-question of the research goes as follows:

What are the (financial) consequences of a successful substation attack on the grid?

4.4.1. Datatype

The required data for this research question consists of simulation results. These results are generated by running multiple model scenarios in a simulation. These quantitative results have to be analyzed.

4.4.2. Lost load analysis and (financial) impact calculation

The last sub-question will be answered by simulation of certain constructed scenarios. First, a combination of attack graph scenarios (defence policy-attacker scenarios) of the modelled system in two Python models will provide the needed likelihood values, this will be complemented by the DIgSILENT Powerfactory and Mininet Software results that have been generated in the work of [18], and is used in this work as well. The work by [18] provides the needed impact figures (measured in lost load). These software packages, implicitly serve as the data analysis tool, and allow an analysis of the consequences of a successful attack on the grid via a substation attack path. Ultimately, the financial consequences can be estimated by calculating the results of the loss of load after a successful cyber attack. [28] Has shown that this research method is feasible and yields verifiable results. This research method would deal with the same drawback as the research method of sub-question 3. However, this drawback is overcome by not using the DIgSILENT Powerfactory and Mininet Software directly, but only using published results which are obtained through this software.

4.5. Research flow diagram

In Figure 4.1 the sequential research steps are shown (design, theoretical framework, data collection, analysis and results, conclusion and recommendations) and their corresponding activities. Some of the activities are also deliverables. The main research question and the sub-questions are also coupled with these activities and deliverables. On the right side, a time frame (white-blueish) is visible which shows the timeline of the study and next to that, four circles can be seen. These are milestone points that are important for the research. These milestones are: Start, Kick-Off Meeting, GreenLight meeting, and defence. A Gantt chart which is based on Figure 4.1 can be found in Appendix A (this chart is better readable in print as A3).

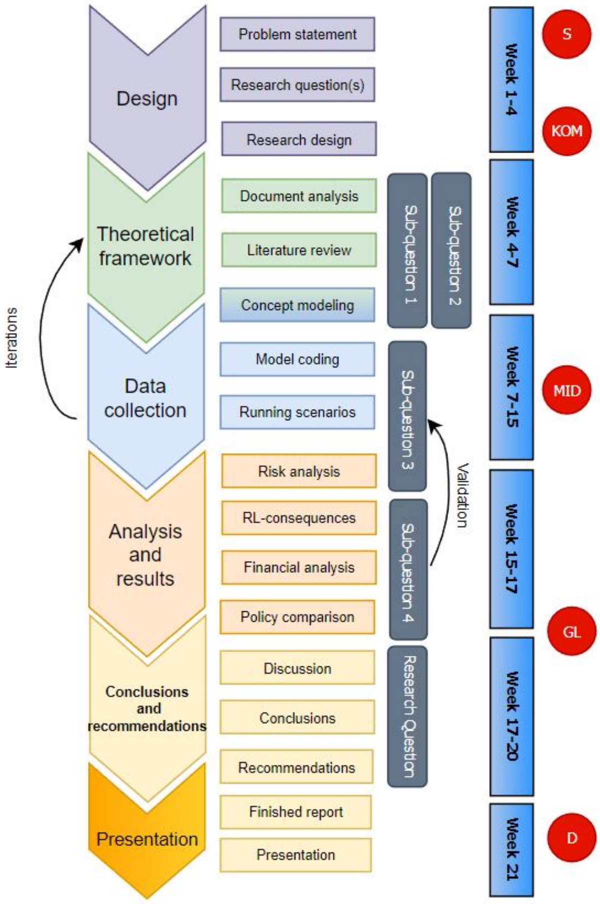


Figure 4.1: Research Flow Diagram

5

Digital substation structure, threats and defenses

This chapter discusses the structure of a digital substation, what threats the substation faces, and how the substation can be protected against these threats. First, the physical and communication layer of the substation is laid out, together with the relevant stakeholders and actors in the environment of the system. This is followed by a description of the different possible threat vectors that can be targeted by malicious actors to compromise the system. Once the threat vectors are clear, the types of attacks that can be launched on those vectors are discussed, followed by a description of the most common attack types in the utility sector. Ultimately, this chapter ends with an overview of current possible countermeasures against cyber attacks on digital substations, as found per the literature review.

A substation is a site within the electricity network wherein distribution feeders and transmission cables are linked together by means of switches through busbars and transformers or by circuit breakers. Substations enable the power flows in the network to be controlled and allow operations to be switched for the purpose of system maintenance [30].

5.1. Physical layer

Traditional substation hardware consists of a bay cubicle that is connected to the Gas Insulated Switchgear (GIS), distribution protection, and control systems via copper cables. These conventional substations were controlled locally and had an in-house Supervisory Control and Data Acquisition (SCADA) system and control room. More recent substations, called digital substations, are controlled from a central remote control point and use an optical communication network (optical fibre) to communicate and control the substation. Older substations are often partly retrofitted to be more compatible with modern systems, but some legacy components can still be found within these substations [31]. Within the bay cubicle, IEDs are used to control the power-switching devices. The GIS in the digital substation is now also equipped with sensors and actuators [32][33]. The connection between the sensors/actuators and the IEDs in the bay cubicle is called the process bus. In the digital substation, there are two more important components, namely the Human Machine Interface (HMI) and the gateway. The HMI is a graphical interface that can be used by an on-site human operator to communicate and interact with the system. The gateway is a network component that enables the flow of data from the substation network to e.g. the central control system of the grid. The connection between the IEDs and the HMI and gateway is called the station bus [34].

5.2. Communication layer

The communication between the IEDs and the TSO control center server (via the gateway) is made possible by a Remote Terminal Unit (RTU), which couples objects in the physical world to the digital SCADA system [35]. Both buses (station and process) and the IEDs in between, are governed by a communication protocol called IEC 61850, which enables data to be transferred as TCP/IP packets [36]. Substations thus make use of communication based on TCP/IP to allow the main control center (and external users such as suppliers) to supervise and maintain the equipment. In order to isolate the private communication from external traffic, firewalls and routers are used which enable connections between the substation and off-site users [36]. The use of TCP/IP has a downside and an upside. It allows for effortless data collection and controlling of IEDs, but it exposes IEDs to the dangers of the internet (WAN) via the gateway. This connection to the WAN is needed to communicate with the SCADA system of the TSO and subsequently with the corporate network of the TSO. ICT and OT networks are clearly separated by firewalls to increase cyber security [25].

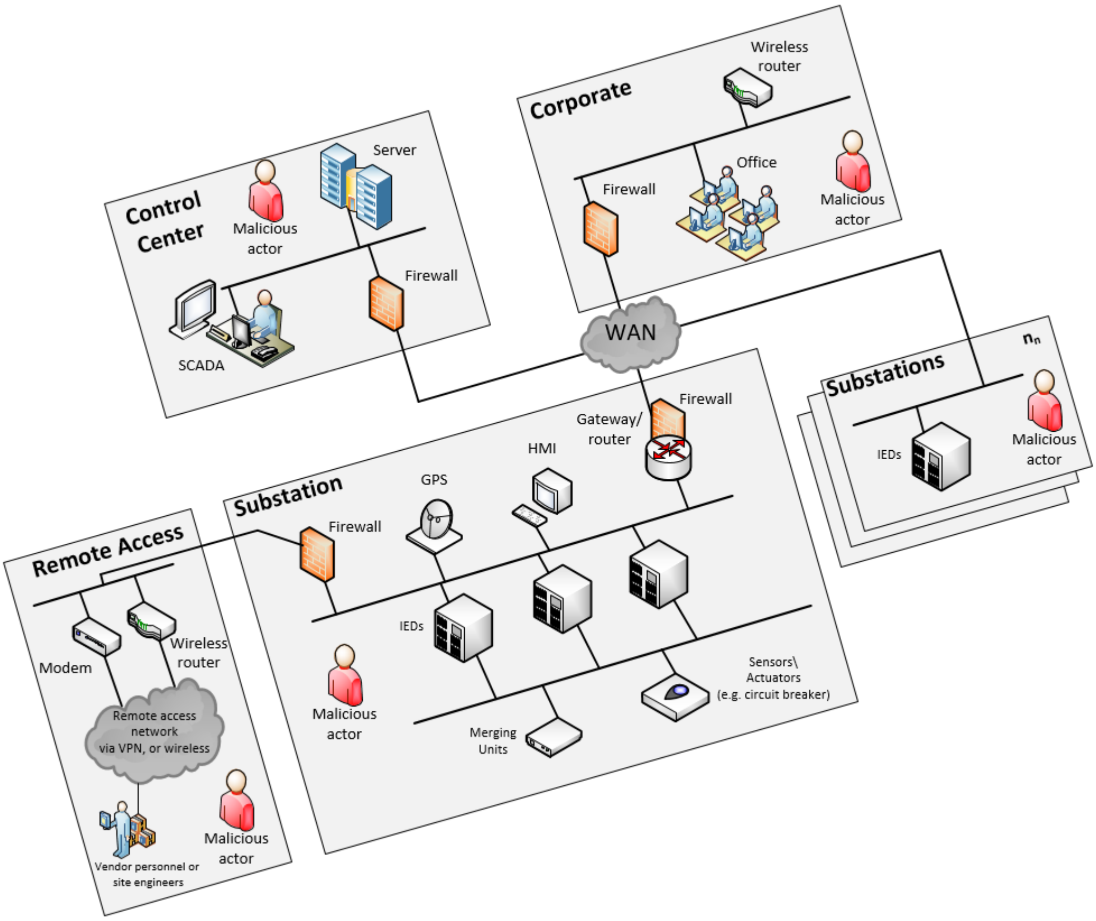


Figure 5.1: Overview of substation cyber and physical environment (adaptation of [37])

5.3. Stakeholders and actors

Stakeholders are internal and external organizations or people with an interest in the target system data and functions [38]. Actors are stakeholders who actively influence the specific (sub)system and the processes under study. According to the integrated Annual Report of a large TSO, their stakeholders are: Employees, NGOs, Governments and policy-makers, customers, suppliers, regulators, shareholders, and energy market participants [39]. Because substations are owned by, and thus part of the TSO, it is assumed that these stakeholders are equal for substations. This stakeholder list excludes malicious actors, which in this case should be added to the list of stakeholders. These aforementioned stakeholders are relevant for obvious reasons, nevertheless, a couple of these stakeholders will be examined further due, to their crucial role in the cyber security of substations. An extensive list of the different stakeholders and actors is given in Appendix C.

As mentioned by [40] there is a strong need to define the collective responsibilities which span across all interdependent organizations connected to the substation. According to [41] this concept is called Supply Chain and External Dependencies Management (SCEDM) and has the following purpose: To develop and maintain cyber security controls that are proportionate to the risk to critical infrastructure and organizational goals for assets and services that are dependent on external parties.

5.4. Possible threat vectors

A digital substation contains a large array of different possible threat vectors. Threat vectors are certain points in a system where an attack can be manifested [42]. The different attack vectors can be grouped into four categories, namely: Software related, firmware related, hardware related, and communication related threat vector [36].

5.4.1. Software related threat vectors

Software related threat vectors are threat vectors that depend on vulnerabilities in the software which is used by components in the substation.

Ransomware attack A ransomware attack works by infecting devices with malware that thoroughly encrypts the data located on the device. The device can only be encrypted by paying an enclosed amount of money to the culprit. If the demanded ransom is not paid, the data on the device is erased. According to [43] ransomware attacks on Programmable Logic Controllers (PLC) in industrial control systems have already occurred. PLCs are similar to IEDs, so this threat vector should be taken into consideration [36].

HMI attack IEDs are often connected to HMIs. These HMIs usually consist of physical operation panels or online client-based or internet-based entry points. The software of such HMIs is a source of vulnerabilities such as willfully designed back-doors to alleviate debugging workload or plain mistakes in programming which can be exploited by a malicious actor [44]. These exploits can range from interruption of communication to complete shutdown of the devices [36].

Payload attack A PLC, being a much less advanced and capable IED, consists of two main software components, namely: Firmware and control logic. The latter can be considered as the application layer of the PLC and controls the behaviour of the PLC, while the former acts as the operating system (OS) which allows the control logic to read and write lines of code [45]. If the control logic is compromised by a malicious actor, it has full control over all the devices that fall under the PLC. The control logic is also called the payload [46]. Attacks on these payloads (payload attacks) can be hard to detect and are able to linger within a device for a long period of time [47]. The danger of the payload attack is that compromised IEDs can prevent outputs from being sent, which might lead to a failure in the system, or it can divert data to the malicious actor for use in a later attack. It has also been reported that this type of attack has the ability to spread itself to other IEDs in the SCADA network of the compromised component [48][36].

5.4.2. Firmware related threat vectors

Firmware related threat vectors are threat vectors that depend on vulnerabilities in the firmware which is used by components in the substation. Firmware and software are related, but the difference lies in the fact that firmware is created to serve as an intermediary between hardware and software, but can also be used to operate an embedded system with a sole purpose (e.g. a printer) [49]. Firmware is thus a specific form of software.

Pin control attack The input/output (I/O) interfaces of IEDs are commonly governed by means of a System on Chip (SoC). Usually, these SoCs utilize a large number of pins that connect them to the electric circuit [36]. A pin controller manages these SoC pins, allowing the controller to configure pin multiplexing (programming a pin in such a way that it has multiple functionalities [50]) or configure the I/O mode of the pins by adjusting the data in a set of registers. Malicious actors could change or stop the IED from performing a certain action without the operating system being aware of the fact that the IED is malfunctioning. This makes a pin control attack very hard to detect and could have a big impact on the physical system which is controlled by the compromised pin controller of the IED [51].

Firmware update modification Component vendors regularly circulate updates of their firmware from their respective websites. By means of reverse engineering, it is possible for malicious actors to force manipulated firmware updates upon devices [52]. These manipulated updates lead to compromised devices which may have altered inputs and outputs or can be completely stopped from functioning by causing a reload death spiral (forcing a device to keep updating endlessly, rendering it useless) [53][36].

Boot process hijack Boot process hijacking takes advantage of the fact that sophisticated cyber security procedures are not able to execute while the IED is in the boot process [54]. The boot process is a start-up process that is triggered when a device, such as an IED, is started up. In this process, the IED loads a sequence of trusted programs which is stored locally within the IED. The boot process hijack aims to interrupt the regular booting order and install manipulated payload or firmware to the IED [55]. Once the device is compromised it can be forced to malfunction [36].

5.4.3. Hardware related threat vectors

Hardware related threat vectors are threat vectors that depend on vulnerabilities in the hardware which is part of the components in the substation.

Electromagentic field interference Under normal circumstances it is unavoidable for IEDs to be unaffected by some level of natural or artificial electromagnetic interference (EI) fields. The IEDs can be affected through the joined connection of signal, ground, and power or more directly from radiative sources themselves [56]. Besides these operational non-disrupting EI there is also Intentional Electromagnetical Interference (IEMI). These IEMIs aim to intentionally disrupt the standard operation of IEDS [57]. Malicious actors could use IEMI to corrupt analogue or digital signals at the substation process level to feed the IED with corrupted data or corrupt stored data in the register of the IED.[58]. Computers within substations and IEDs often communicate via wireless communication (e.g. ZIgBee and Wi-Fi). These wireless communication methods are even more vulnerable to IE than their wired counterparts [59][36].

Physical tampering Physically tampering with IEDs is the most straightforward and impactful danger to IEDs within a substation. While most of the IEDs are placed within secured substation buildings, it is still possible for people with the right clearance to enter these sites [60]. A malicious actor could make use of social-engineering or other hacking tools (e.g. keyloggers or keyboard emulators) to obtain credentials and take control of IEDs [61]. Once the IED is compromised it is possible for the malicious actor to make changes in the device or feed the IED false data to trigger actuators, causing the system to operate irregularly [62]. Physical tampering largely takes place in the physical world and thus is harder to detect for traditional IDS [36].

Jamming and GPS spoofing Jamming and GPS spoofing are threats that specifically target IEDs. In digital substation communication, it is common to use GPS for unique time-stamping for separate IEDs. This is used for waveform data and internal logic event time-stamping. IEDs also often have phasor measurement capabilities embedded in their system that depend on the correct reference of time to support the functioning of protection, monitoring, and control across the grid [36]. Spoofing of GPS data aims to mislead GPS receivers through the transmission of fabricated GPS signals or via repetition of legitimate signals which were collected at a different time and location [63]. Closely related to GPS spoofing is GPS jamming. This method broadcasts signals with high power to disrupt an IED from receiving the real GPS signals. If this is done successfully the time synchronization of the IED will be scrambled and unreliable [64] [36].

Cyptography key theft Communication of data is often encrypted via cryptographic keys. These keys are mostly stored in the flash memory of IEDs. The storage of the kets in this flash memory has long been deemed secure, but recent research has shown the opposite [65]. Once the key has been compromised the malicious actor can access all data which is encrypted with that key. Cryptographic key theft can be done even easier when malicious actors get a hold of scrapped IEDs that still have the keys in their chips. This valuable information is often neglected to be cleared from the device before disposing of it[36].

5.4.4. Communication related threat vectors

Communication related threat vectors are threat vectors that depend on vulnerabilities in the internal and external communication within the substation.

Communication media sabotage A large part of IEDs allow the use of remote access through private WAN. Because communication media is subject to a large geographical distribution it is prone to sabotage (e.g. line cutting or installation of tapping devices) [36]. The integrity of the lines and the data that flows through them is critical for IEDs. This stems from the fact that IEDs require timely and reliable communication for the execution of pilot differential protection (comparing the current entering at one end of a line with the current leaving the other end of the line [66]).

Grey and black holes Grey and black holes are nodes within a network that have been compromised. In a SCADA-based network that runs the DNP3 protocol (Distributed Network Protocol: set of communication protocols for interaction between components in process automation systems [67]) unrequested messages are disclosed to system control as an anomaly. When such a network has grey or black holes then a part or all of the packets that travel through the network will be lost. The dropping of data by these holes is hard to detect for a central controller and a well-coordinated effort could completely isolate parts of the network from central control [68].

Vulnerable communication protocols The vulnerabilities in nonproprietary smart grid protocols of SCADA networks with IEDs such as: IEC 61850, IEC 61400-25, IEC 60807-5, and DNP3 are a threat to grid integrity [36]. The source of these vulnerabilities comes from the dearth of encryption, integrity, authentication, confidentiality, authorization, and availability [69].

Rogue nodes These are congruent devices that have been placed by malicious actors to pose as genuine nodes. These rogue nodes are able to send instructions to actuators and read all traffic that passes through the compromised node [36]. An example of such a rogue node is the installation of a wireless device on a SCADA network to flood the communication channels with junk packets to prevent other devices to communicate legitimate messages to the command centre [70].

5.5. Cyber attack categories

The previous section discussed the different vectors that could be used by malicious actors to enter the substation system. Once control (of a part of the system) is established there are different types of attacks that can be launched. In this section, the types of attacks are shortly described.

Direct actuator control attack Once a malicious actor has access to the substation specifications the plainest option is to disguise themselves as a legitimate system operator and transmit instructions to IED actuators (e.g. circuit breakers) [36]. As shown in section 5.3, there are plenty of options to initiate this type of attack. Once actuators are taken over, the malicious actor can induce a power outage which could lead to a cascading failure, which is when a line failure (due to exceedance of a certain threshold) makes that line unusable and subsequently leads to a change in network topology, that further strengthens this effect and ultimately causes a system blackout due to a negative feedback loop [71]. This happens when post-fault mechanisms are not effective enough. The Ukrainian blackout is an example of this type of attack [3].

Manipulation of measurements attack The safe and reliable operation of the energy grid is assured by certain control processes such as automatic voltage regulation, transient stability assessment, automatic generation control and maintaining optimal power flow. All these processes depend on timely and accurate measurements which are collected by IEDs and sent to the central command for monitoring and control purposes. If undesired control actions are taken as a result of manipulated IED measurements, then the power supply of the system becomes jeopardized [37][72][73]. Two types

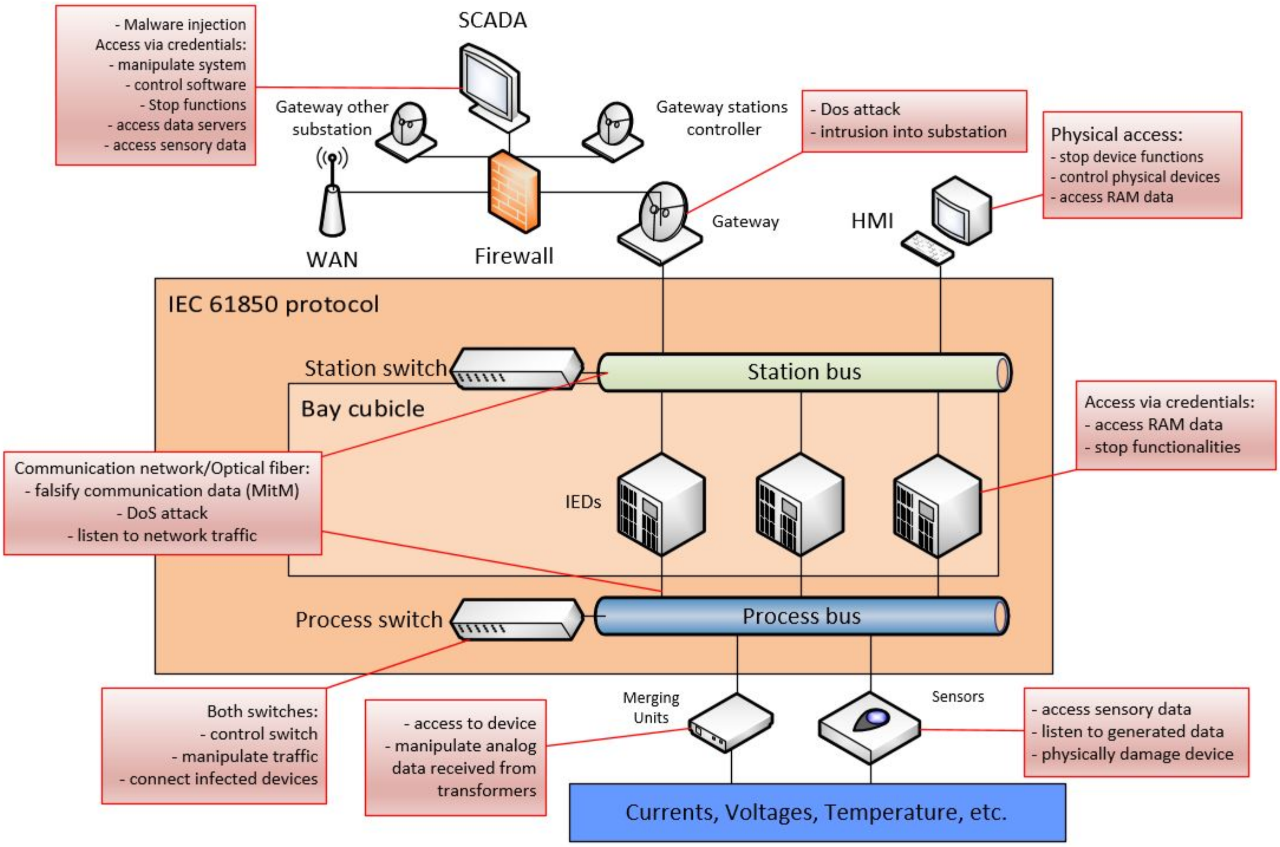


Figure 5.2: Overview of substation components and potential cyber attacks per component (adaptation of [34])

of measurement manipulation attacks are common, namely replay attack [74] and false data injection attack (FDIA) [62]. FDIAs mostly focus on disturbing Weighted-least-square-based (WLS-based) state estimation. This state estimation uses a set of non-linear equations concerning measurements and power system states (i.e. phase angle and bus voltage level) to fine-tune the power system by minimizing the sum of square residuals [75]. If malicious actors compromise strategically chosen measurements, then these actors are able to circumvent faulty data detection in order to inject bad data into the estimates [36].

A replay attack presumes that the malicious actor is able to seize control over sensors and copy their readings during a certain period and then replay this recording afterwards. By sending normal measurements during abnormal events the human operator is not aware and thus unable to act if a fault is occurring in the system. This could have large impacts on the grid stability [36].

Depletion of resources and delay of response attack It is also possible for a malicious actor to focus their attacks on the IEDs' computational resources and communication. This is done by transmitting floods of junk packets to the IEDs, this way the malicious actor does not only overcrowd the communication channel but also drains the computing power of affected devices [76]. Preventing legitimate packets from arriving at the recipient is called a Denial of Service (DoS) attack. Power systems are often susceptible to latency (delayed arrival of packets). This is due to the fact that Remedial Action Schemes (RASs), which automatically perform a corrective measure such as generation or load shedding when a threshold is exceeded, is time sensitive. For example, fast protection protocols have to be executed within 4 ms for 60 HZ networks [77]. Therefore, when the data conveyance is delayed it could severely influence the safety and stability of the grid.

Time Sync attack As discussed before in this chapter, a large number of processes in the power system are dependent on accurate and timely information such as event localization, fault detection, and voltage stability monitoring [36]. Through the use of GPS across the grid, different IEDs are able to work together. If malicious actors compromise the time-stamp the previously described processes will not be able to function properly anymore [64].

5.6. Most common attack types

Finding a quantification of the number of cyber attack types or the total number of cyber attacks is not an easy task. There is little to no publicly available data on the number of cyber attacks, and even less data regarding critical infrastructure. The absence of a historical cyber attack database appears to be caused by the aversion to cyber incident disclosure due to fear of reputation loss, liability, and negative effects on their competitive advantage [78] [79]. However, despite the previously mentioned, some sources have been found which provide a glimpse of the attack type distribution. Even the sources acknowledge that the cited figures are but a fraction of the presumed real figures of cyber attacks. According to [80] over 60% of the cyber attacks in the Mining, Quarrying, and Oil & Gas Extraction Utility industry are phishing-based attacks (which can act as a prelude to an HMI attack or attack via SCADA). The distribution of other attacks is as follows: approximately 13% stolen credentials (for physical tampering or prelude for other attacks), 10% ransomware (ransomware attack), 8% command and control (vulnerable communication protocols and rogue nodes), 5% exploit vulnerability (firmware update modification, boot process hijack, or attack on HMI firmware vulnerabilities) and the remaining percentages are classified as other. Another source [79] gives the following figures: 38% unknown (organization declared a compromise, but forensics could not find the method used due to lack of detection and monitoring capabilities), 9% weak authentication (used for HMI attack), network scanning/probing 22% (misusing vulnerable communication protocols), removable media 2%, brute force intrusion 1% (physical tampering), abuse of access authority 4% (social engineering), phishing 17% (which can act as a prelude to an HMI attack or attack via SCADA), SQL injection 2%, and 9% miscellaneous. These figures are summed over all industries. According to [79] 32% of the reported incidents originate from the energy industry. These figures are far from exact, but give an overall view of the types of cyber attacks that have happened (and have been reported).

5.7. Main security objectives

It is the goal of each system design to ensure security. This entails that the three main security objectives will have to be satisfied. These main security objectives are confidentiality, integrity, and availability (CIA), also known as the CIA-triad [81] [82]. With confidentiality, the unauthorized release of information is meant. This occurs when an unauthorized entity is able to read and exploits information that is stored in a computer (system). Observing the patterns of information use, called traffic analysis, also falls under confidentiality. Integrity refers to information that is not modified by unauthorized entities. Whenever information is stored it should only be modified by authorized personnel or actors. The availability objective aims to ensure that unauthorized denial of the use of information does not occur [82].

5.8. Digital substation cyber security interventions

There are many different policies and interventions that combat the threat of cyber security attacks. Many of the novel, and now traditional interventions, can be thoroughly found in published papers. To ensure that the search for these papers is done systematically, the PRISMA-method will be applied once again for the execution of a literature review. For more information about this methodology revisit paragraph 2.1.1.

For this literature review, three queries have been used. The queries are shown in Table 5.1.

Table 5.1: Used search queries second PRISMA literature review

Query	Database	First picks
substation AND cyber security AND technical AND policy AND defence AND countermeasures	Google Scholar	8
substation AND cyber security AND organizational AND policy AND defence AND countermeasures	Google Scholar	10
substation AND cyber security AND hybrid AND policy AND defence AND countermeasures	Google Scholar	10

An overview of the outcomes per step is placed in Appendix D to improve the readability of the text. The findings of this literature review are shown in table 5.2 below. During the search for hybrid defensive policies, there were no explicitly hybrid countermeasures found. Instead, there were papers that advocated for both technical and organizational policies. So, the found policies were added to their respective stand-alone category in table 5.2.

Table 5.2: Findings of the second PRISMA literature review

Findings:	**Author:**
Technical	
IDS (signature- and anomaly based)	[83] [84] [85] [15] [36]
Remote attestation (software, hardware hardening and verification)	[83] [86] [85] [36] [81]
Deception technology	[83] [36]
Secure communication protocols	[83] [86] [87] [88]
Firewalls	[84] [86] [15] [36]
Router configuration blocking illegitimate traffic	[84] [86] [87]
Bad data detection	[84] [36]
Unobservability prevention	[84] [36]
VPN	[84]
Antivirus	[84] [85] [36]
Redundant design	[86] [85]
Smart tokens for authentication	[85] [87] [36] [81]
Organizational	
Zoning	[83] [86]
Incident response	[83]
Cyber forensics	[84] [83]
Network identity and authentication	[84] [83]
Situational cyber awareness training	[84] [89] [90] [91] [92]
Access management (user rights, role segregation)	[86] [85] [87] [89] [91]
Whitelisting	[86] [89]
Backup and restore	[86] [89] [91]
cyber security management (patch- and security event logging, audits NERC CIP, NIST(IR), ISO, BP1-SPNI)	[86] [85] [87] [36] [89] [90] [93] [94] [91] [88] [92] [95] [83]
Strict password policies (and other personal authentication methods)	[85] [89] [91] [81]
Vendor security requirements	[36] [89] [91] [81]

6 Model setup and calculations

As discussed in the chapters research approach, and methodology, the main research question will be answered by means of an (implicitly mixed) research approach that makes use of modelling. In this chapter, the calculations that are used within the model are discussed, as well as the overall setup for the model and the (financial) calculations that are done with the model results. Firstly the base case testbed is described, which serves as a baseline model, whereupon later additions will be built to see the change in risk per added countermeasure. Afterwards, the main calculation theory, Time-to-Compromise, is elaborated, followed by the methodology for (total) risk assessment, a description of the Value of Lost Load (VoLL) measure as well as other measures describing the physical impact, and also a description of the simulated countermeasures within the model is given. This chapter also covers the Societal Cost-Benefit Analysis (SCBA).

6.1. Python-based models

As of May 2022 Forseeti, the company that owns the SecuriCAD software, has been acquired by Google. The result of this takeover is that SecuriCAD is no longer free to use for researchers. Since the modelling part of the research was heavily based on the use of SecuriCAD, an alternative free-to-use software package had to be found. Most alternatives such as IriusRisk and CAIRIS are predominantly qualitative in nature. The qualitative essence of these packages doesn't align with the quantification goals of this research. Therefore, the choice has been made to continue the research with a self-made model. This model will be built in Python and makes use of, among others, the NetworkX library.

6.1.1. Base case testbed

The core of the model is the graph structure, which is based upon NetworkX, that represents the possible attack paths, consisting of attack steps, that the malicious actor can take to compromise its intended target. The graph structure is shown in Figure 6.1. The nodes represent the individual attack steps and the edges (lines) represent the connection between the attack steps, which together form an attack path. In Appendix E a combined graphic of Figure 6.1 and 5.2 can be found. Within the edges, the TTC of the following attack step is stored.

6.1.2. Component-oriented Python model

The nodes are given certain attributes such as an id and color. The edges carry the attribute which defines the node that it connects to, such as the TTC of the subsequent node. The TTC is calculated according to equation 6.8. The calculation of the TTC distribution per edge is done in the first part of the model. The TTC is computed by running a ten thousand-sample Monte-Carlo simulation of the TTC formula, Equation 6.8. The results of the Monte-Carlo simulation are plotted in a histogram and the distribution of the TTC is fitted, so it can be used in the Graph-oriented Python model, which calculates the average TTC of the attack paths. This component-oriented Python model consists of a large part of code written by [18] who used this code to check in a white box manner what SecuriCAD calculated in a black box way. The code for this Python model can be found in Appendix H.

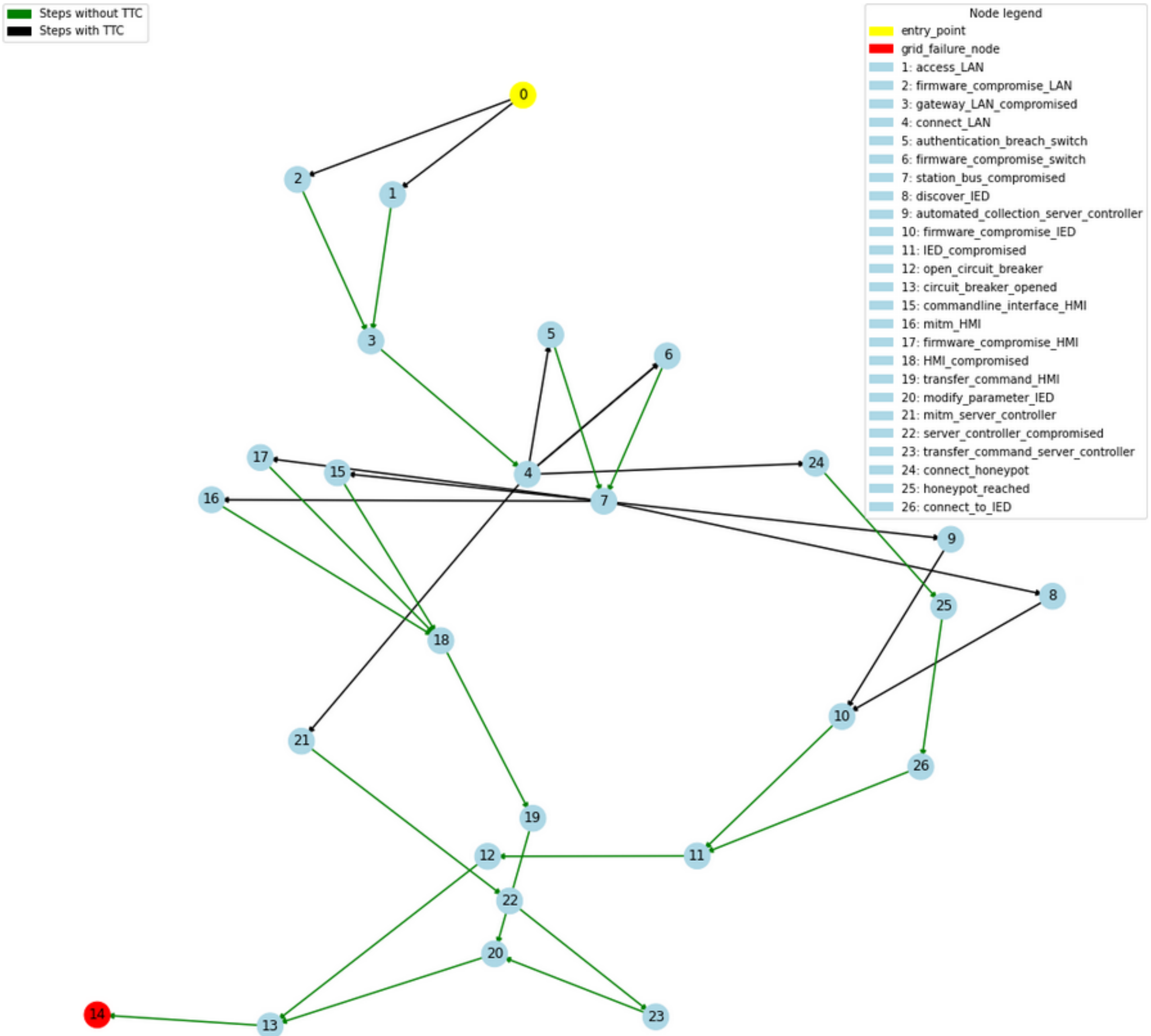

Steps without TTC
Steps with TTC
Node legend
entry_point
grid_failure_node
1: access_LAN
2: firmware_compromise_LAN
3: gateway_LAN_compromised
4: connect_LAN
5: authentication_breach_switch
6: firmware_compromise_switch
7: station_bus_compromised
8: discover_IED
9: automated_collection_server_controller
10: firmware_compromise_IED
11: IED_compromised
12: open_circuit_breaker
13: circuit_breaker_opened
15: commandline_interface_HMI
16: mitm_HMI
17: firmware_compromise_HMI
18: HMI_compromised
19: transfer_command_HMI
20: modify_parameter_IED
21: mitm_server_controller
22: server_controller_compromised
23: transfer_command_server_controller
24: connect_honeypot
25: honeypot_reached
26: connect_to_IED

Figure 6.1: Graph representation of substation attack paths within Python (NetworkX)

6.1.3. Graph-oriented Python model

In order to calculate the total TTC of the attack path for reaching the goal node (which is reached after the circuit breaker, indicated in red, is opened) in the substation, the Dijkstra's Single-Source shortest path algorithm was initially used [96]. This algorithm finds the shortest path from one node to all nodes in a graph. However, due to limitations of the algorithm in the ability to take into account updated TTC values, the algorithm was substituted by the Depth-First-Search algorithm. This algorithm is a graph traversal algorithm that searches all branches of a given graph G [97]. The algorithm has been adjusted so that it starts at a given source node ”entry_point” and searches for the quickest path to the target node ”goal_node”. This adjusted Depth-First-Search algorithm provided more convenience and flexibility in terms of calculating and adjusting the shortest path during runs in the Monte-Carlo simulation. The weight of the distance in this case is equal to the calculated TTC of the first part of the model. The code for this Python model can be found in Appendix I.

The shortest route (read: attack path) to the goal node (marked red in Figure 6.1) is calculated from the entry-point node (marked yellow in Figure 6.1) in the base case testbed to get a baseline measurement of a standard configuration. For performing the (base) attack simulation the Monte-Carlo method will be used. Thus, this second model also performs a Monte Carlo simulation, but this time with fifteen thousand samples in total. Per sample, the TTC for all edges in the graph is calculated, based on the given distribution, and the shortest path to all other nodes in the graph is computed. This shortest path represents the minimum time that a malicious actor spends in the substation, beginning

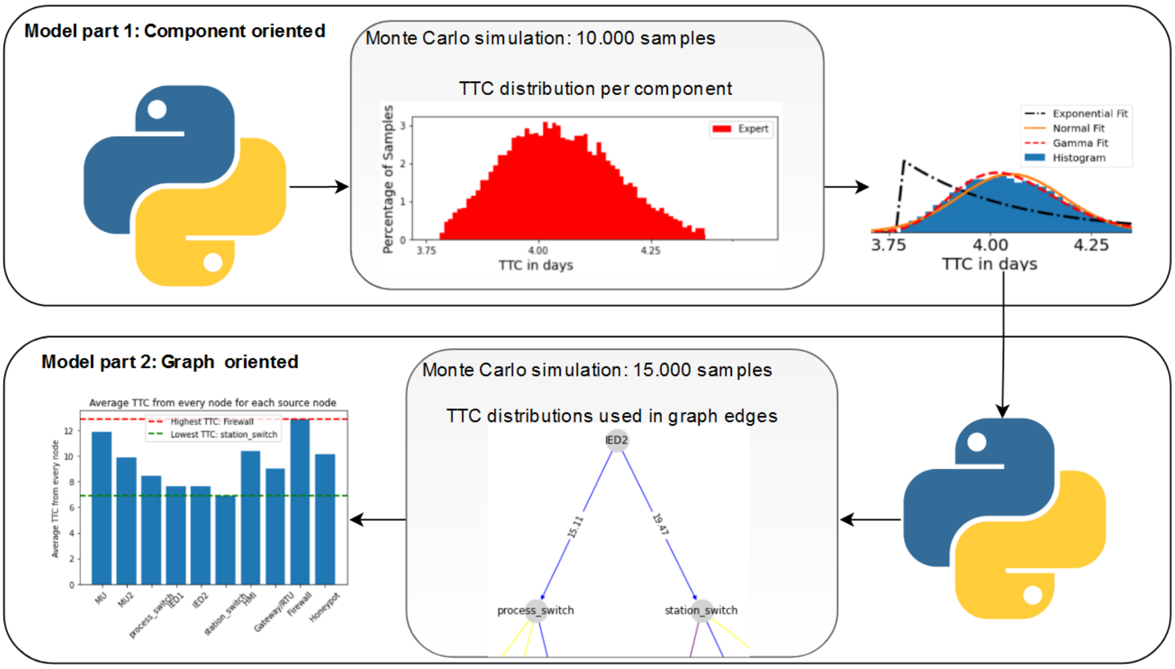

Model part 1: Component oriented
Monte Carlo simulation: 10.000 samples
TTC distribution per component
Percentage of Samples
Expert
0
1
2
3
3.75
4.00
4.25
TTC in days
Exponential Fit
Normal Fit
Gamma Fit
Histogram
3.75
4.00
4.25
TTC in days
Model part 2: Graph oriented
Average TTC from every node for each source node
Highest TTC: Firewall
Lowest TTC: station_switch
Average TTC from every node
0
2
4
6
8
10
12
MU
MU2
process_switch
IED1
IED2
station_switch
HMI
Gateway/RTU
Firewall
Honeypot
Monte Carlo simulation: 15.000 samples
TTC distributions used in graph edges
IED2
15.11
19.47
process_switch
station_switch

Figure 6.2: Connection between component- and graph-oriented Python models

from a set point of entry and advancing through the substations nodes via attack steps (the edges). Due to the probabilistic nature of the TTC distribution that governs the attack steps, the outcome of each different attempt leads to some extent to a varied outcome. Ultimately, the target of the malicious actors is compromised with success within diverse stretches of time, which are characterized by the distribution of the probability of the target assets and their connections. Figure 6.2 gives an overview of how the two Python model parts are connected.

6.1.4. Components base case

The base case consists of five components: Gateway/RTU, HMI, Station bus, IED, and the server controller. For each component, the most popular vendor on Google, that supplied the technical document sheet, has been chosen. For each chosen component the amount of Common Vulnerabilities and Exposures (CVE) in the National Vulnerability Database (NVD) has been documented in order to be used in the TTC calculation. Table 6.1 shows the amounts of CVEs per component.

Table 6.1: CVEs per component of base case testbed

Component	Vendor(s) + type	Vulnerabilities
IED (relay)	Siemens SIPROTEC-4	18
Station bus	Cisco IE9300	16
HMI	Starter Kit WinCC V16 SIMATIC HMI	2
Server controller	Siemens SICAM PAS	7
Gateway	Kalkitech SYNC3000	1

6.2. Time-To-Compromise

In order to quantify and compare the risk reduction of different systems a unified metric has to be used. In this thesis, the methodology and subsequent metric of [20] has been used, because the predecessor on whom this research is based has also used this metric, but also due to the fact that this metric is easy to explain and understand for a layman. [20] Coined the term Time-To-Compromise and defined it as follows: "the time needed for an attacker to gain some level of privilege *p* on some system component *i*". This metric is similar to the Time to exploit (TTE) metric by [98] which has been defined as the time needed for a malicious actor to exploit a given vulnerability. The latter has been less substantiated than

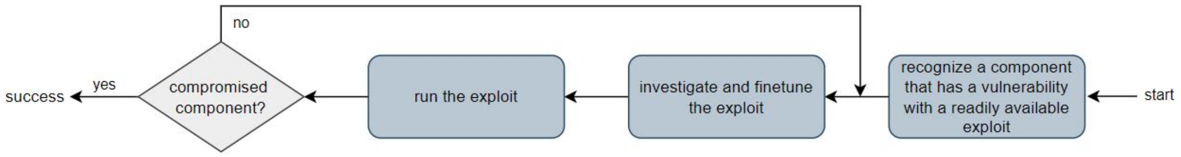

no
yes
success
compromised component?
run the exploit
investigate and finetune the exploit
recognize a component that has a vulnerability with a readily available exploit
start

Figure 6.3: Flowchart outlining Time to Compromise subprocess 1 (adaptation of [20])

the former, so with quantifiability as a goal of this research the former metric will be maintained.

6.2.1. TTC model processes

The TTC (T_{comp}) is dependent on two variables: the characteristics of a certain vulnerability (V_c) and the skill level of the attacker (A_s). The estimation of the TTC has been modelled as a random process and consists of three subprocesses of the attacker [20]:

- **1st process** is applicable when there is a minimum of one vulnerability of component *i* known, that leads to a gained level of privilege, *p*, and the malicious actor also has a minimum of one exploit ready which can be successfully deployed versus (one of) the known vulnerability.
- **2nd process** happens whenever there is a minimum of one vulnerability on component *i* familiar that leads to a gained level of privilege, *p*, however, there is not a readily available exploit for the malicious actor which could be deployed versus (one of) the known vulnerability
- **3rd process** consists of the discovery of novel vulnerabilities and exploits. The third process runs parallel to processes 1 and 2, and runs continuously in the background. The malicious actor could only utilize the results of the third process, but may also be an active party in this process. In other words, the malicious actor could stand by until a novel exploit/vulnerability is identified and shared, or personally probe for a novel one.

Every single one of the aforementioned processes has its own distinct failure probability distribution. While the first and second processes are mutually exclusive, the third process continues in parallel to the other processes.

6.2.2. Subprocess 1

The model of subprocess 1 is depicted in Figure 6.3 as a flow chart. This process always ends in completion. The model consists of two parts: Firstly, the estimation probability that the malicious actor has a readily available exploit that can be deployed versus the vulnerability of the component. This corresponds to the probability that a malicious actor is in fact in process 1. Secondly, the estimated time of process 1.

Probability estimation of subprocess 1 Calculating the probability that the malicious actor is in fact in process 1, is done via the search theory which operates in a manner comparable to the physical security systems by [99]. The equation below employs the assumption that available exploits are distributed uniformly over all the vulnerabilities:

$$P_1 = 1 - exp\left(-\frac{VM}{V_{tot}}\right) \tag{6.1}$$

where P_1 is the probability that there is a readily available exploit in possession of the malicious actor that compromises the component, *V* is the amount of vulnerabilities of the targeted component, *M* is the amount of exploits that are readily available to the malicious actor, and V_{tot} is equal to the total amount of vulnerabilities. In this study, the value of V_{tot} is 7,000 and is designated as the total amount of vulnerabilities, which is an assumption proposed by [29]. However, due to limitations in the first Python model, this value had to be downscaled to 6,100 to ensure usable results.

The value of variable *M* is the function of the skill level of the malicious actor. The skill level has been divided into the following four levels: novice, beginner, intermediate, and expert. These skill level categories correspond to the values 50, 150, 250, and 450, respectively. The skill level of a novice malicious actor is equal to 50, because according to the website Metasploit, there are 50 trivial-to-use exploits. The superior skill levels have been based on a postulated exponential growth in exploits that are readily available as a function of the skill level [20]. These figures (and the method) have also been used by [29] for calculating power system reliability.

Time estimation of subprocess 1 In accordance with the findings of [20] the mean time to perform a successful attack, assuming that the malicious actor is in fact in process 1, is presumed to be t_1 = *1 day*

6.2.3. Subprocess 2

The model of subprocess 2 is depicted in figure 6.4 as a flow chart.

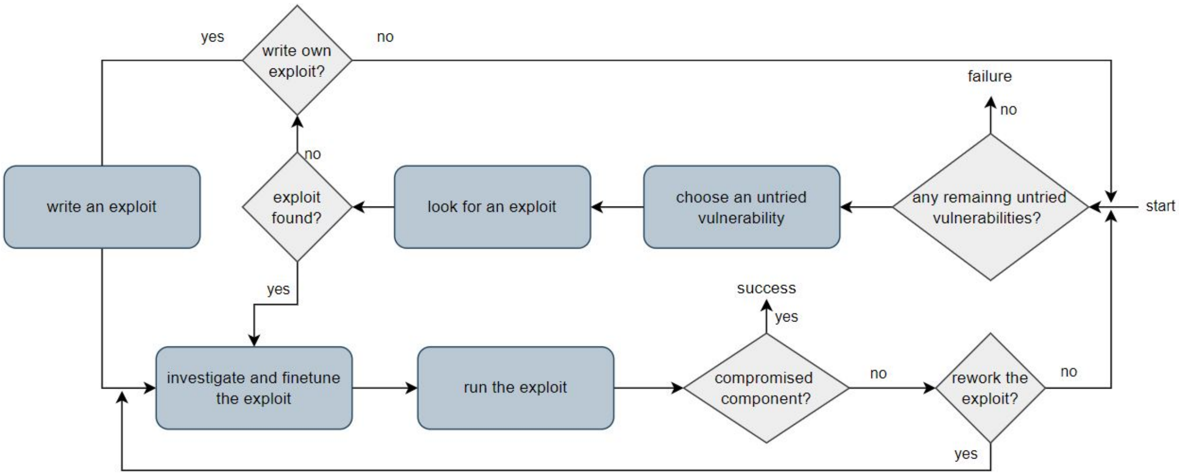


Figure 6.4: Flowchart outlining Time to Compromise subprocess 2 (adaptation of [20])

Probability estimation of subprocess 2 As mentioned before, processes 1 and 2 are mutually exclusive. This means that when *V* > 0, then the probability that the malicious actor is in the second process is equal to the complement of the first process. This leads to:

$$P_2 = exp\left(-\frac{VM}{V_{tot}}\right) = 1 - P_1 \tag{6.2}$$

where P_2 is equal to the probability that the malicious actor is not in possession of a readily available exploit that would lead to the compromise of the component under study.

Time estimation of subprocess 2 As previously discussed, in process 2 the malicious actor must develop an exploit on their own account to capitalize on the known vulnerability under study. According to results obtained by [20] the average time that is required per try is presumed to be 5.8 *days*. Subsequently, the mean time of the second process is computed by the following equation

$$t_2 = 5.8 * ET \tag{6.3}$$

where t_2 is defined as the mean time for completing the second process and *ET* is equal to the amount of expected tries that are needed for the malicious actor to realize an exploit. The amount of expected tries (*ET*) is calculated by

$$ET = \left(\frac{AM}{V}\right) * \left(1 + \sum_{tries=2}^{V-AM+1}\left[tries * \prod_{i=2}^{tries}\left(\frac{NM - i + 2}{V - i + 1}\right)\right]\right) \tag{6.4}$$

where *AM* is defined as the average amount of vulnerabilities for which an exploit can be developed or discovered by the malicious actor taking into account their respective skill level. *NM* is described as the amount of vulnerabilities that the skill level of the malicious actor is not able to deploy, and *V* is still the amount of vulnerabilities of the component under study. The ratio of known vulnerabilities that can be targeted by the malicious actor is

$$AM = int(f_c * V) \tag{6.5}$$

where f_c is proportional to the malicious actor's skill level. The given values for f_c are based on the values calculated by [20] and are as follows: expert (1.0), intermediate (.55), beginner (.30), and novice (.15).

6.2.4. Subprocess 3

The model of subprocess 3 is depicted in figure 6.5 as a flow chart.

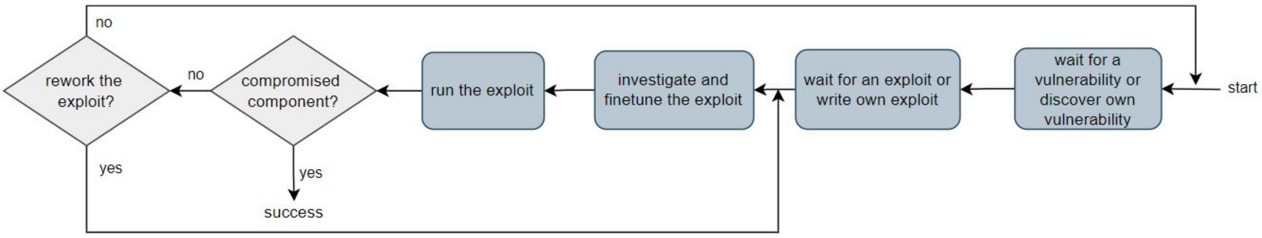

no
rework the exploit?
no
compromised component?
run the exploit
investigate and finetune the exploit
wait for an exploit or write own exploit
wait for a vulnerability or discover own vulnerability
start
yes
yes
success

Figure 6.5: Flowchart outlining Time to Compromise subprocess 3 (adaptation of [20])

Time estimation of process 3 Lastly, the third process considers that the tempo of new vulnerability development becomes a constant over time, while the implementation of the skill level of the malicious actor is applied conform the study by [100]. In order to compute this, the probability variable *u* has to be known, which specifies whether or not the second process is unsuccessful:

$$u = (1 - k)^V \tag{6.6}$$

where *k* is still equal to the skill level of the malicious actor. Following [20] this skill level is used in the discrete domain, so for each skill class, a constant value is given (i.e. a value of 1 for experts and .30 for beginners. From this, we conclude that the mean time to go through the third process is calculated by

$$t_3 = \left(\left(\frac{1}{V_{tot}}\right) - \left(\frac{1}{2}\right)\right) * 30.42 + 5.8 \tag{6.7}$$

where the number 30.42 is presumed to be the mean time between the discovery of novel vulnerabilities in *days*. The exact value of this number can be adjusted according to expert validation. In the end, the overall TTC for a given component can be calculated by:

$$TTC = t_1 * P_1 + t_2 * P_2 * (1 - u) + t_3 * u * P_2 \tag{6.8}$$

6.2.5. Continuous TTC calculation

This subsection discusses how the discrete TTC can be transformed into a continuous variable. The work of [20] differentiates four separate discrete classes of skill level for the malicious actors that correspond to a certain numerical value. A major shortcoming in this approach is the fact that it contemplates these actors' skill as being located on an integer number line [18]. In other words, a malicious actor is for example either an intermediate or an expert, but could not be quantified as a highly experienced intermediate. Therefore, the malicious actor's skill level should be defined as a probability distribution instead of a discrete variable. This drawback was also mentioned by [100].

The primary underlying assumption for this adjustment comes down to the fact that a substation may be attacked by a malicious actor whose attack-efficiency is dependent on their respective skill level. According to [101] there are certain syndicates that are extraordinarily skilled in cyber attacks. The MITRE ATTCK for industrial control systems database describes such syndicates. Nevertheless, such groups are more the exception than the rule. Organizations may consider such advanced adversaries as a worst-case scenario instead of the expert-level standard [18].

This work follows the numerical adjustments made by [18] which consist predominantly of a revision of the skill level mapping of the malicious actors, and the subsequent parameters that will be affected by this change. It is assumed that the skill level is defined by a normal distribution which is based on their skill class. Instead of four skill classes, only three are considered. The distributions of said parameters are shown in Table 6.2.

attacker skill level	value	normal distribution (mean - variance)
beginner	100	(0,8 - 0,04)
intermediate	250	(0,55 - 0,07)
beginner	360	(0,2 - 0,05)

Table 6.2: Skill level

The procedural adjustment starts with giving skill level *k* a value which is derived from the probability distribution of table 6.2. Furthermore, the amount of readily available exploits that the malicious actors could use taking into account their respective skill level is calculated as follows:

$$m = M * V_{tot} \tag{6.9}$$

where *M* is defined as the total number of readily available exploits which could be used by the malicious actor. This means that a malicious actor with a 0.2 skill level and 100 available exploits could only use 20 of these exploits.

An additional adjustment has been made to the amount of vulnerabilities that a malicious actor could target, based on the skill level of the malicious actor. This gives the following equation:

$$AM = K * V_{tot} \tag{6.10}$$

This adjustment is linked to the description of the second subprocess of the TTC model.

6.3. Methodology for risk assessment

As mentioned before, the goal of this study is to quantify the change in risk of cyber attacks on CPPS substations by applying countermeasures based on the quantified model. In order to quantify this risk, an overall risk equation has been proposed by [28] which is adopted and elaborated in this work. This methodology incorporates both the attack graph's probabilistic analysis and the dynamic CPS model's quantitative impact analysis. The simplified risk equation can be calculated by:

$$Risk(j) = Likelihood(j) * I_{ph}(j) * I_{fin}(j) \tag{6.11}$$

In this equation *Risk(j)* is defined as the risk which is computed for a given scenario *j*, *Likelihood(j)* is the success likelihood of the scenario, denominated in $(1/year)$, while I_{ph} expresses the quantified impact assessment of the scenario on the physical power system in a dimensionless unit. The term I_{fin} expresses the financial impact of a successful cyber attack in $.

However, Equation 6.11 does not contain a temporal dimension. To calculate the yearly risk over time, in this case, infinity, Equation 6.11 has to be adjusted.

$$Risk_i(j) = \frac{Likelihood(j) * I_{ph}(j) * I_{fin}(j)}{(1+r)^i} \tag{6.12}$$

Where, *r* is defined as the discount rate that is equal to 2%, and *i* is the amount of years. Since the likelihood is expressed in $(1/year)$, I_{ph} in a dimensionless unit, and I_{fin} is denominated in $, it makes that the annual risk is given in $\$/year$. From the previous equation, we can also compile the Total Risk (TR), which is given below:

$$TR(j) = \sum_{i=1}^{\infty} R_i(j) = \frac{Likelihood(j) * I_{ph}(j) * I_{fin}(j)}{r} \tag{6.13}$$

The TR over an infinite period of time is equal to $(Likelihood * Impact)/r$, assuming that there is no time dependency in either the Likelihood, Impact, or discount rate. This results in a TR in $, which will ultimately be compared against the cost of potential investments, which are also expressed in $.

In Paragraph 6.6.1. and Chapter 7, the calculation for I_{fin} is described. Therein, it can be seen that I_{fin} is based on two elements. Namely, the VoLL (expressed in \$/MWh) and the load that has been lost due to a successful cyber attack (expressed in MWh). The latter element is calculated based on an adjusted Sigmoid function and I_{load} (and thus also on I_{ph}). By multiplying these two elements, I_{fin} functions as total impact measure I_{total}, which implicitly takes into account the physical impact I_{ph}, but is expressed in \$. This makes it possible to simplify the TR equation to the following form:

$$TR(j) = \sum_{i=1}^{\infty} R_i(j) = \frac{Likelihood(j) * I_{total}(j)}{r} \tag{6.14}$$

This work is mainly focused on the likelihood (reduction) segment of the risk equations, however, the other parts of this equation are also relevant to eventually calculate the TR. In the original equation by [18] there are additional variables that describe among others, the impact on the cyber layer and the difficulty of restoring certain generators. This has been omitted in this version of the equation, as the focus will be mainly on the physical consequences of a successful cyber attack. The underlying models of the other segments are taken as given, so as to keep the scope of the study manageable. To see the entire equation with an explanation of all the variables, see: [18]. The remaining individual components are further described in the subsequent paragraphs.

6.3.1. Attack scenario likelihood

In this study there is a primary underlying assumption regarding attack scenarios. This assumption is that the risk has a direct relation to the time which a malicious actor requires to compromise its target [20]. Furthermore, the Mean-Time-To-Detect (MTTD) metric is considered. This is a Key Performance Indicator (KPI) that represents the time span between problem emergence and detection by the Blue Team (a team that performs system analysis to ensure cyber security) or the designated process. This KPI is commonly used by experts in the cyber security field [102]. In this study, it is assumed that the base value of the MTTD metric is defined and set by the CPPS cyber security experts, and is a constant which can be employed for the assessment of the likelihood. These assumptions and definitions follow the work of [28], but in this work, the MTTD is expanded upon by adjusting the base value of the MTTD with a multiplier that takes into account the maturity of the organization's cyber security policies. The multiplier is called the maturity factor *m* and can take a value between 0.25 and 2.5. These values are based on the timeliness of incident detection categorization by [103] which is performed by the MITRE Corporation and the U.S. Space Force. The adjusted MTTD is called the Relative MTTD (RMTTD) and is defined as follows:

$$RMTTD = m * MTTD, \quad let \;\; m = \{0.25, 0.5, 0.75, 1, 2.5\} \tag{6.15}$$

Ultimately, likelihood is formulated as:

$$Likelihood(j) = \frac{RMTTD}{TTC_{avg}(j) + RMTTD} \tag{6.16}$$

In this equation $TTC_{avg}(j)$ is defined as the average TTC as calculated by the attack graph security assessment, and *RMTTD* is equal to the previously defined variable which represents the (relative) detection capabilities of the defenders. This equation produces a result in the [0,1] range. When $TTC_{avg}(j)$ « *RMTTD*, then *Likelihood(j)* approaches 1, and if the component is strongly guarded and is impenetrable by the malicious actor, then:

$$\lim_{TTC_{avg} \to \infty} \frac{RMTTD}{TTC_{avg}(j) + RMTTD} = 0 \tag{6.17}$$

The exact calculation of the MTTD is beyond the scope of this study and follows [28] in the assumption that it is equal to 14 *days* and is defined by an industry expert.

6.3.2. Assessment of power system impact

The power system has a wide variety of protection and control schemes to ensure static (whether or not a system settles at a newer post-disturbance operating level that satisfies the physical constraints) and dynamic security (if the system survives pre-fault to post-fault transition) [104] [105]. The impact of cyber attacks on this system can be researched. In order to do this, two states are compared. The pre-fault (or pre-attack) state and the post-attack (equilibrium is restored) state. This analysis has a prime assumption, which is that there are no other remedial actions taken, besides those executed by the considered automation and protection systems that are implemented. In this study the physical impact is calculated by a shortened version of the impact calculation by [18] and only considers the impact on the load. It is defined as follows:

$$I_{ph}(j) = w_L * I_{load} \tag{6.18}$$

The variable w_L is a weight factor whereas I_{Load} is a function that considers the impact related to the change in frequency, and load. Because in this adapted version of the equation we only consider the impact on the load, the weight factor w_L is equal to 1. The deviation of the loss of load indicator is calculated as follows:

$$I_{Load}(j) = \sum_{i=1}^{N_{Loads}} \frac{\Delta P_{Load,i}(j)}{P_{initial,i}(j)} \tag{6.19}$$

in which $P_{Load,i}$ is defined as the discrepancy between the set initial $P_{initial,i}$ and the definitive active power of every load. The range for $I_{Load}(j)$ is [0, N_Loads]. These equations are based upon the equations by [35].

Another element that is taken into consideration in other works is the possible formation of fragments within the overall system. This fragmentation causes the grid as a whole to be separated into one or multiple smaller remote areas (also called islands), due to a cascading failure. In this work that phenomenon is neglected to decrease the complexity of the study.

6.4. Simulated countermeasures

During the second literature review, which surveyed possible cyber attack countermeasures, many possible defensive policies have been identified. From 22 different papers 23 distinct countermeasure categories are presented, which belong to either the technical or organizational countermeasures group. An overview of these countermeasures is shown in Table 5.2 which can be found in chapter 5. Modelling all these 23 countermeasures would be ideal, but given the time limitation, only four countermeasures are considered and modelled in this study. These are: Intrusion Detection Systems, Zoning, Honeypots, and Remote attestation (vendor security requirements). These options have been chosen due to their frequent occurrence in the analyzed papers and the required time to model. In the coming paragraphs, the method of modelling for each of the countermeasures is described. The Python code of both the first and second model can be found in Appendix H, and Appendix I, respectively.

6.4.1. IDS

An intrusion detection system (IDS) uses network sensors to detect suspicious behaviour in a system that could be caused by a malicious actor [106]. IDSs can come in three categories; anomaly, signature, and specification based. These different types detect possible malicious actors based on a defined attack pattern, abnormal component behaviour, and predetermined rule deviation, respectively [107] [108] [109]. According to [107] specification based IDSs are the most common in digital substations. The specification based IDS (or rule-based IDS) uses a subset of specifications (or rules) that describe the permitted behaviour of certain systems, which correspond to normal operating behaviour. The rules are used to flag suspicious behaviour which triggers an appropriate incident response by an automatic system or by a human cyber security specialist. Not all flagged suspicious behaviour is, in fact, caused by a malicious actor. An IDS can produce false positives and false negatives. The probability for these false positives and negatives, given a certain condition (whether or not there is, in fact, an intrusion), can be calculated by using Bayes' theorem [109].

Model implementation In the Python model, IDS is coded by selecting certain nodes that function as the network sensors. Whenever a calculated shortest path contains both network sensor nodes, then the TTC for that attack path is doubled. This should change the dominant attack path of the malicious actor, as the penalty makes that the previously shortest path is now longer, and becomes much less attractive. Another option to code the IDS was to continuously change the TTC based on the probability to be caught in either of the two sensor nodes and decrease the TTC by a set amount based on the probability. However, this approach was not feasible given the limitations of the used algorithm, combined with the Monte Carlo simulation and the envisioned outcome.

6.4.2. Zoning

Another countermeasure against cyber attacks is the use of communication zoning. This countermeasure divides a system, in this case, the digital substation, into separate compartments (communication zones) with distinct access control levels. Critical components within the substation can be protected through zoning by limiting access to these components. The different communication zones can be protected by data diodes or firewalls. Data diodes are far more secure than firewalls but have limited flexibility. This is because data diodes allow communication traffic to flow only one way. When using IEDs and MUs this is not desirable as these components have to both receive data and perform actions (e.g. send data or trigger circuit breaker) [83]. For this reason, firewalls are used in the digital substation under study. An example of communication zoning (also known as network segmentation) is given by [86] which illustrates how the substation can be divided into three zones with three firewalls. Zone 1 contains the station and process bus, IEDs, and MU. Zone 2 encompasses the HMI and gateway, and Zone 3 is a Demilitarized Zone (DMZ) which contains only the gateway to the enterprise network. The firewalls are placed between zone 1 and 2, between zone 2 and 3, and a firewall after zone 3. An example zoning configuration is shown in Figure 6.6. Zoning makes it harder for malicious actors to reach components that are further in the substation and are in a different zone than the component from where the attack step is launched.

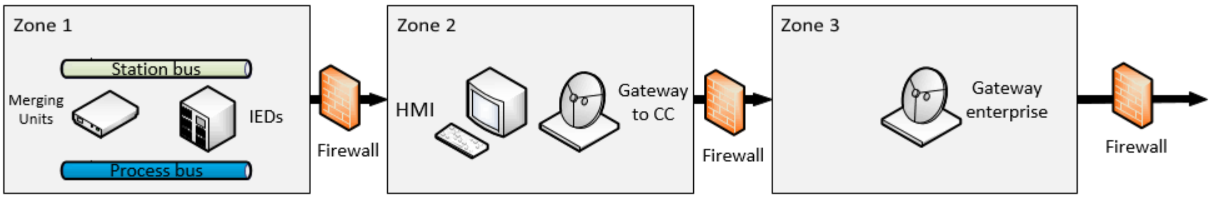


Figure 6.6: Example zoning configuration

Model implementation In the Python model, zoning is coded by dividing three subsets of nodes into groups. Whenever the shortest path from one node to another end node contains an attack step that crosses zone boundaries, then the TTC is increased by a certain value.

6.4.3. Remote attestation

Protecting the digital substation can also be done through remote attestation. This is a cyber attack countermeasure that enables devices to verify whether the software (or firmware) that is installed, is in fact legitimate or a compromised version installed by a malicious actor. This measure counters remote malware injections or physical attacks on the substation network [83]. Remote attestation can be applied in two ways: hardware based attestation and software based attestation. The former employs a trusted platform module (a co-processor that protects crypto keys and records the computing platform's software state [110]) which executes a challenge-response authentication protocol to verify the software integrity of a component. This protocol makes use of a public key encryption scheme. The general remote attestation procedure is shown in Figure 6.7. Hardware based attestation is most suited for more advanced computing systems (e.g. servers and computers), while software based attestation is more commonly used for lower-end embedded systems [111] [83][112]. A cyber attack on hardware based attestation devices are deemed infeasible because the core components used in this type of attestation are considered as secured hardware [113]. Hardware based remote attestation also has some other limitations. Generating the attestation key, which is a testament to the validity of the system's soft- and firmware, requires a large amount of computations, and thus time. Whenever this has to be done for an extensive computing system such as a personal computer, then the key should include hash values for all the software, such as: the operating system, firmware, run-able applications, and the bootloader [110]. Therefore, the application of hardware based remote attestation should only be done for system components (nodes) that can suffer such a delay. Besides attestation per component, there is also swam attestation, which is the attestation of an entire network of computing devices [113]. In this study, swarm attestation is not considered as this would be too extensive to model within the given time.

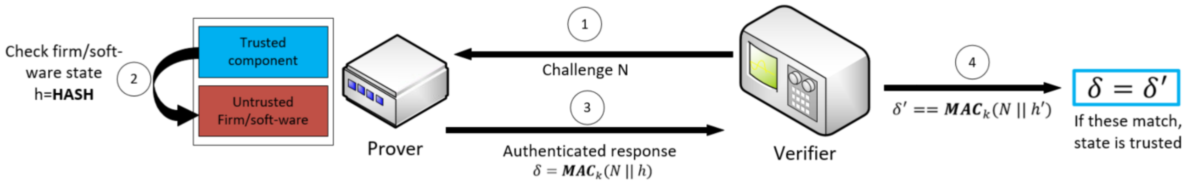

Check firm/soft-ware state
h=HASH
2
Trusted component
Untrusted Firm/soft-ware
Prover
1
Challenge N
3
Authenticated response
$\delta = MAC_k(N \,||\, h)$
Verifier
4
$\delta' == MAC_k(N \,||\, h')$
$\delta = \delta'$
If these match, state is trusted

Figure 6.7: General attestation procedure adapted from [112]

Model implementation In the Python model, remote attestation is coded in the first part of the model, which calculates the TTC for individual components. This is because remote attestation is a nearly impenetrable countermeasure, but only against attacks that compromise the hardware and/or soft/firmware of a component. So, for components (nodes) that have been determined suitable for hardware based remote attestation, the number of vulnerabilities in their TTC calculation has been decreased by the number of vulnerabilities that depend on hardware and/or soft/firm-ware being compromised.

6.4.4. Deception: Honeypots

The last countermeasure to be covered in this study is the use of honeypots. Honeypots are decoy components in a system whose sole purpose is to be attacked by a malicious actor. Honeypots come in a myriad of shapes and sizes. [114] Differentiates between two different types of honeypots based on their level of interaction. On one hand, there are low-interaction honeypots, which simulate just one or few services and functions that the malicious actor can attack. On the other hand, there are high-interaction honeypots, which emulate a large array of functions and services and can hardly be distinguished from a genuine system component. Both types have advantages and disadvantages [115]. Low interaction honeypots get fewer hits compared to high interaction honeypots [116], but are easier to set up and maintain. Conversely, high-interaction honeypots have a higher risk of being broken out of by the malicious actor, due to the realism and allowed functionalities of the decoy system component [114]. It has been reported that honeypots were able to predict an eventual exploit up to three days before the first connection to the honeypot [114]. Honeypots are not only standalone countermeasures but work together well with IDS to act as network sensors, which increase the accuracy of IDSs [117].

Model implementation The use of honeypots has been implemented within the Graph-oriented Python model. To emulate the workings of a honeypot, an additional node and edge can be connected to another node in the testbed system. Whenever a malicious actor's attack path contains that node in its shortest path, it incurs a TTC penalty of 270. The exact value of the penalty cannot be completely underwritten by scientific literature, as there is not enough exact data regarding the time spent in high-interaction honeypots that delay a malicious actor. Therefore, a number is chosen that leads to an extreme overall increase in average TTC, because the triggering of a honeypot trap alerts the defenders of an ongoing cyber attack. This causes the defenders to take defensive actions to oust the malicious actor, severely hindering that attack.

6.5. Physical and cyber system

In order to study the effects of a cyber attack on the physical and cyber part of the CPPS environment, the results of a study by [18] is used. Those results are obtained during his study with DIgSILENT PowerFactory and Mininet, which can assess the impact on the physical and cyber sections of the CPPS, respectively. Such results have been used in other similar studies before [118][119].

6.6. Financial calculation

In order to show the added value of using a quantified model for cyber security investment decisions, the business case for such a model has to be made. Within the setting under study, the case comes down to the following: companies invest money (and time) in applying countermeasures to cyber attacks of malicious actors. This constitutes one part of the cost-benefit balance. The other part is the damage (measured in \$) incurred when a successful attack causes a measurable disruption for society in the power system. When the made cyber security investments lead to a decrease in the incurred costs after a successful attack, then this is seen as a benefit. So, the business case for such a quantified model is to maximize the decrease in incurred damages, while minimizing the costs needed for realizing the decrease in incurred damages. The incurred damage per scenario is equal to the TR as calculated in Equation 6.14. The costs of the countermeasures are estimated based on the available papers and documentation. The financial calculations are further elaborated upon both in this sub-chapter as well as in the following chapters.

6.6.1. Cost-benefit analysis

The countermeasures suggested in this study to obtain the benefit of risk reduction in digital substations, obviously cannot be implemented for free. Fundamentally, the utility company has to pay for these investments. In the Netherlands, due to the vital nature of electrical critical infrastructure, this is done by the state-owned 'private' company TenneT [120][121]. Investing large quantities of money into public infrastructure projects has to be based on the expected return - i.e. added benefit - of the project. Notice that the return here is 'expected', this implies that there is an inherent uncertainty in the investment, as is reasonable. In order to better deal with these uncertainties, governments, and government-owned enterprises should perform a SCBA [122].

Such an SCBA is part of a much larger decision-making process for public infrastructure projects and usually consists of 10 main elements such as: project, and non-project alternatives, scenarios, market and competition analysis, external effects quantifiable in money, and those not quantifiable in money, a business analysis, partial and general cost-benefit analysis, indirect and national economic effect analysis, and finally the decision [123].

Given the relatively limited scope of the suggested investments, in comparison to the large infrastructural projects that SCBAs are usually associated with, only a select part of the SCBA will be applied in this case. Figure 6.8 gives an overview of how the different SCBA elements work together, and what parts are included (green) and excluded (red) from this work.

Project alternatives and zero-alternative The project alternatives in this case could be different types of countermeasures, as discussed in Table 5.2. However, given the limited time of the study, the focus will be kept on the suggested countermeasures and combinations thereof. Also, the zero-alternative is taken into consideration. A zero-alternative is not equal to doing nothing, but means using a different investment to deal with the 'challenge' faced [123]. Since this study already compares multiple investment options the zero-alternative has been taken into account. Eventually, of all the different possible options, the most feasible according to the SCBA should be chosen by the decision maker.

Scenario formulation As said before, the expected benefits of (ICT infrastructural) projects are often uncertain. Especially when these benefits are expected to occur in the (far away) future. In the case of substation countermeasures, this uncertainty is even greater as the expected benefits are based on the occurrence (or better: avoidance) of a successful cyber attack. This risk should be acknowledged with scenario- and sensitivity analyses [123].

To deal with the risk and uncertainty in this case, different scenarios are made for the attacker types and suggested countermeasures. Just as in regular investment analyses, cyber security investment analyses also make use of a sensitivity analysis to determine the effects of changing parameters on the business case [124].

Transport effects/competition analysis Since this concerns an investment in a public good which has no national competitors (regulated monopoly [125]), a competition analysis would be of limited use.

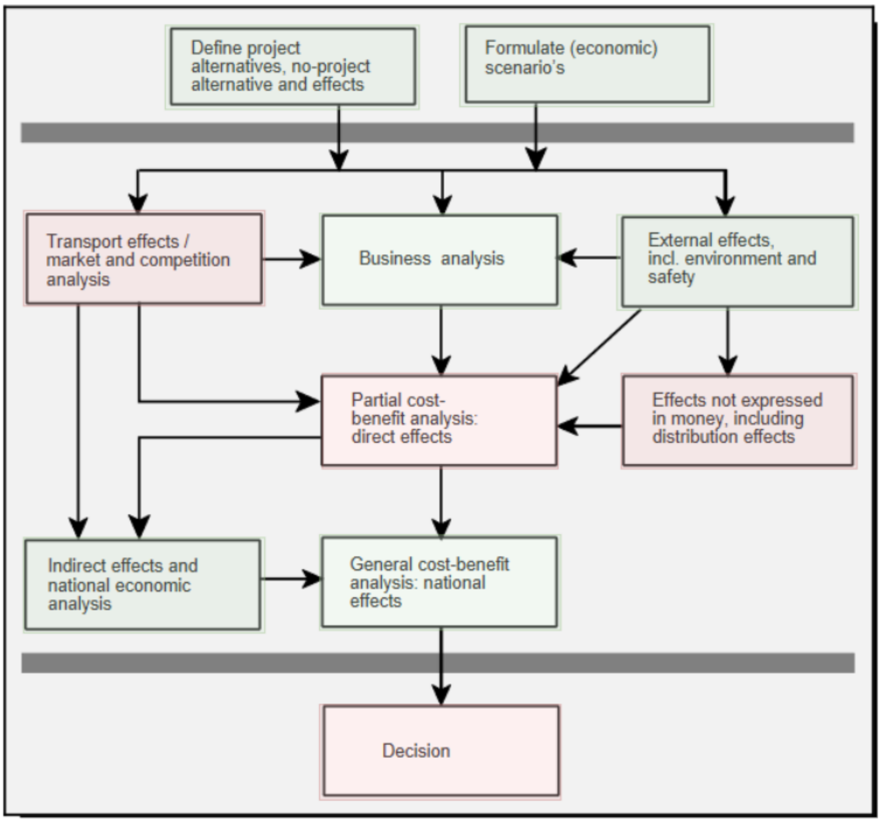

Define project alternatives, no-project alternative and effects
Formulate (economic) scenario's
Transport effects / market and competition analysis
Business analysis
External effects, incl. environment and safety
Partial cost-benefit analysis: direct effects
Effects not expressed in money, including distribution effects
Indirect effects and national economic analysis
General cost-benefit analysis: national effects
Decision

Figure 6.8: Included main elements of an infrastructural SCBA project (adjusted from: [123])

Indirect effects (national-economic analysis) It might seem self-evident that investing in the cyber security of the national power grid brings positive indirect effects. Nevertheless, this should still be substantiated as best as possible. According to [126] there are three additional added benefits of investing in cyber security that are generally neglected in traditional CBAs. These benefits are:

1. *Synergetic effects* which are achieved by new and increased public-private partnerships. Public examples are: leveraging relationships with private actors to gain new insights in the cyber security field, and increasing the momentum of cyber security progress as private actors are able to advance and perfect cyber security practices in their processes, which inspire others. A private example is: gains in credibility and resources by partnering with government agencies.
2. *Indirect impact* which can be obtained through the long-term operation of cyber security. Examples of public benefits are: raising the (likelihood) of an increase or maintenance of the cyber security budget. Examples of private benefits are: marketing effects (reputation gains and engagement of community), Sustainability/durability through stabilization of operating environment and processes, and meeting the moral obligation as a company.
3. *Shared values* which come in the form of the ability to influence participating actors and communities. Public examples are: knowledge sharing and the improvement of supplier quality and the quality of life of the people in the community. Private examples are: job creation, and providing appropriate training and infrastructure.

A quantitative study by [127] has shown that there is a positive relation between cyber commitment (among others: telecommunication infrastructure quality and access to ICT) by government agencies and Macro-economic growth (albeit mediated by the increase in the use of ICT in business practices). It should be mentioned here that this concerns correlation and not causation. Therefore, one should take into account that other factors contribute to macroeconomic growth besides cyber commitment. Nevertheless, this is a quantified effect of the investment in cyber security that should be weighed into the SCBA.

Another indirect (national-economic) beneficial effect is the mitigation of economic damage due to power outages. The consequences of a cyber attack can vary greatly, from a minor disruption all the way to a complete blackout. Quantifying the monetary impact of such consequences is done through an economic indicator called the Value of Lost Load (VoLL). The value of lost load gives a monetary evaluation of the disruption (damage) of electrical supply for society [128]. The VoLL can be measured in a direct and indirect way. The direct approach looks at among others: willingness to pay and/or costs directly incurred by the unavailability of electricity. The indirect technique looks at factors such as macroeconomic impact and revealed preference (mitigation costs that users are willing to take). Besides the indirect and direct measurement techniques, the VoLL also has a structure that takes into account a differentiation between affected parties (private or commercial) and damage and mitigation costs. The VoLL is dependent on the location of impact. In this study, the Dutch VoLL will be considered which, according to the ACM, is equal to 68.887 €/MWh [129] or 73.802,22 $/MWh (as per the exchange rates of June 3rd). The results of the scenarios in the first and second Python models will be combined with the results of [18] to extract the amount of lost load of a test bus system to eventually calculate the monetary damage of the cyber attack. By calculating the change in TR through countermeasures, the potential (national-economic) indirect effect can be computed.

The lost load values are extracted from a data set generated during a study of [18]. In the data set, I_{Load} is given for different types of substations and different combinations of sequences in which the circuit breakers are opened. I_{Load} Has a range of [0, 19], as there are 19 loads connected to the substation in total. The value of I_{Load} indicates how many of the 19 loads are disconnected. In equation 6.18 the value of I_{Load} is a float. Therefore, the rounded-up integer of that value should be considered, as a load is either connected or disconnected. From the rounded-up I_{Load} value, the amount of lost load in MW can be estimated. In this estimation, the fraction of disconnected loads is multiplied by the total value of all the loads. Which is 6.097,1 MW. However, because the VoLL is given in €/MWh, the MW values have to be multiplied by the number of hours that the load is lost (blackout duration). In order to realistically mimic how the load in a power system is restored, an inverse logistic function (or Sigmoid function) is used. This function emulates a fast restoration of a large amount of load in the first phase of a blackout, and a slower restoration near the end of the blackout. This function has been fitted based on percentage recovery quantities over time of the 2003 Italian blackout [130]. This fit is taken as the generalized recovery function for all calculated outages. The equation used goes as follows:

$$y = \frac{L}{1 + e^{-5.99(Dx-0.43)}} \tag{6.20}$$

Where, *y* is equal to the lost load, *x* is equal to the elapsed time, L is defined as the share of lost load relative to nominal operation (in MW) of the substation, and D is the duration of the blackout. The integral of this function from 0 to *x* hours is then calculated in order to get the lost load in MWh. Because L is dependant on the value of I_{Load}, and the total amount of load during normal operations *T*, L can be rewritten as follows:

$$L = \frac{\lceil I_{Load} \rceil}{I_{LoadTotal}} * T \tag{6.21}$$

Where, $I_{LoadTotal}$ is equal to the sum of the loads. When these equations are combined, the following equation is created:

$$y = \frac{\left(\frac{\lceil I_{Load} \rceil}{I_{LoadTotal}} * T\right)}{1 + e^{-5.99(Dx-0.43)}} \tag{6.22}$$

For the sake of simplicity, three blackout duration categories (based on historical blackout data) are considered to calculate the VoLL in MWh:

1. small blackout, where a maximum of 15% of the physical grid is down, and has a duration of 2 hours [131]

2. medium blackout, where a maximum of 75% of the physical grid is down, and has a duration of 19 hours [130]
3. large blackout, where a minimum of 75% of the physical grid is down, and has a duration 72 hours [132]

Based on the aforementioned categorizations a suited subset of data from the data set is chosen. This subset is substation 2, and the circuit breaker opening sequences are 2-1-3, 3-2-1, and 3-1-2. The first sequence causes an I_{Load} of 0.52 which corresponds with a physical system failure of 8.14%. This classifies it as a small blackout. The second sequence causes an I_{Load} of 7.46 which corresponds to a physical system failure of 30.33%. This would be classified as a medium blackout. The last sequence causes an I_{Load} of 18.16 which corresponds with a physical system failure of 91.63%. This is considered a large blackout according to the categorization. The large blackout duration category happens to be based on the US-Canada power outage, consisting of a large part of the New England power system, which is the same power system that is used by [18] whom provides the physical grid properties of a successful cyber attack. In Chapter 7: Simulation scenarios, the calculations for the lost load in MWh and in monetary value will be provided. Together the lost load in MWh, which is based on I_{load}, and the VoLL in ($/MWh) can be multiplied to obtain the total impact in $, calculated in an implicit manner. The provided figures and assumptions in this paragraph are heavily reliant on assumptions and generalizations of blackout types. It is taken into consideration that in reality there is far more variety than the three described categories.

External effects The external effects, according to [123], should include effects on environment and safety. In the case of substation cyber security, the effects on the environment are negligible. However, the effects on safety are rather important. Increased cyber security of substations should lead to a decrease in risk of cyber attacks on the national power grid. This does not only increase the national safety of the country against foreign malicious (state) actors, but also prevents accidents from happening due to power outages, and protects people who are dependent on power for vital functions, such as hospitals and care centres (if an uninterruptible power source is absent or depleted). Besides these benefits, there are also possible negative side effects of increased cyber security. According to [133] there are also unintended harms of cyber security countermeasures, and identified seven different types. Of these seven only three are taken into consideration in this study. These external effects according to [133] are:

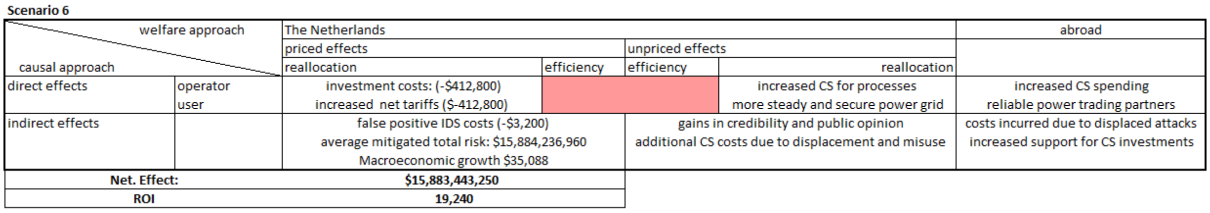
1. *Displacement:* The displacement of cyber crime refers to a possible change in location, method or period, as a result of cyber countermeasures [134]. This displacement results in the need for (additional) countermeasure investments in other, possibly related, systems of critical infrastructure.
2. *Additional costs:* Besides the monetary cost of countermeasures there are also other (possibly overseen) costs that can occur in the form of resource depletion. For this reason, a cost-benefit analysis for cyber security countermeasures is advised [135]. In the current case, the possible known additional costs could be a disruption of operations due to false negatives in detection systems, or a slowdown in response time to accidents due to the implementation of additional countermeasures.
3. *Misuse:* The implementation of novel countermeasures with the aim of stopping or disrupting malicious actors, could backfire and cause additional attack vectors through misuse. A well-known example of this is the danger that high-interaction honeypots could be broken out of and used as a springboard for attacks on the rest of the system [114].

Partial CBA: direct effects Given the relatively limited size of the investment and the low amount of direct benefits, a partial CBA would have limited added value in this case.

Business analysis The business analysis maps out the total financial effects of the prospective project for the operator. This is a limited analysis that focuses solely on the financial cost-effectiveness of the project. The investment costs for the operator are calculated in Appendix F.

Non-monetary expressed effects Close to all non-monetary expressable effects are covered in the indirect effects and external effects elements, therefore adding the non-monetary expressed effects is superfluous.

General cost-benefit analysis The table below shows the composition of the general societal cost-benefit analysis:



Scenario 6

welfare approach		The Netherlands				abroad
		priced effects		unpriced effects		
causal approach		reallocation	efficiency	efficiency	reallocation	
direct effects	operator user	investment costs: (-$412,800) increased net tariffs ($-412,800)	🟥	🟥	increased CS for processes more steady and secure power grid	increased CS spending reliable power trading partners
indirect effects		false positive IDS costs (-$3,200) average mitigated total risk: $15,884,236,960 Macroeconomic growth $35,088		gains in credibility and public opinion additional CS costs due to displacement and misuse		costs incurred due to displaced attacks increased support for CS investments
Net. Effect:		$15,883,443,250				
ROI		19,240				

Figure 6.9: General Societal Cost Benefit Analysis

The general SCBA shown above is that of scenario 6. In this analysis, it is assumed that the average mitigated TR is equal to the benefit of the countermeasures implemented in that scenario. This general SCBA template is based on the template by [123], and distinguishes between priced effects and unpriced effects. These effects are then divided into direct and indirect effects in The Netherlands and abroad. In Appendix J the general SCBA for each non-base case scenario can be found. The values for the false negative costs have been calculated with Bayes' theorem

Decision The eventual decision on whether or not to perform the suggested investments is up to the respective decision-makers. This is deemed out of the scope of this research. Therefore this element is excluded from this work.

6.7. Model validation and verification

This section is dedicated to the validation and verification of the built and adjusted model(s).

6.7.1. Validation

The act of validation is defined as the procedure of auditing to what extent the built model serves as a genuine portrayal of its real-world counterpart seen from the perspective of the model's intended use [136]. Checking the validity of a model is not always easy to do, this is even more true in this case, as reliable figures for TTC of successful cyber attacks are all but abundant. However, the model can attain its validity from, among other things, the fact that the attack steps have been based on real vulnerabilities from the National Vulnerability Data by NIST. But, also several papers have been consulted that describe possible attack paths within digital substations.

A sensitivity analysis of both models has also been performed to assess the validity of the created models. In the performed sensitivity analysis one parameter in the model is changed at a time to see what the result is on the model outcome. In tables 6.3 and 6.4 below, the tested parameters and their respective values are given. For model part 1, the output of the variables total time to compromise for the expert level (Tot_TTC_e) and the TTC distribution for the expert level was evaluated. For the second model, the output (including the paths) of the variables shortest total TTC path, average shortest total TTC path, and the highest total TTC was evaluated. The results of the sensitivity analysis consist of seven different Excel sheets filled with model outcomes. This is because the sensitivity analysis had to be done "by hand", since the current version of the models doesn't support automated sensitivity analysis. In order to keep the size of the report at a minimum, these pages are not taken up in the report. Only outcomes which are worth mentioning are taken up in the appendix.

The sensitivity analysis has not shown irregularities in the outcome, which are not explainable by the intention behind the code. Besides the aforementioned, validation of the generated TTC values has also been performed by comparing the results of this study with the study performed by [18] to see whether or not these values were similar. After comparison, this seemed to be the case. Consequently, as far as it is possible, the results of the model have been verified. However, as will be discussed in

Table 6.3: Sensitivity analysis: parameter values for model part 1

parameter	'normal' value	test range
number of vulnerabilities (V)	3	[8, 10, 12, 14]
total vulnerabilities (K)	6100	[400, 5000, 7000, 8000]
skill level expert (M_e)	360	[180, 270, 450, 540]
time to exploit process 1 (tte1)	1	[0, 0.5, 1.5, 2]
time to exploit process 2 (tte2)	5.8	[2.9, 4.35, 7.25, 8,7]
time to exploit process 3 (tte3)	30.42	[15.21, 22.82, 38.03, 45.63]
number of runs	10000	[2500, 5000, 20000, 40000]

Table 6.4: Sensitivity analysis: parameter values for model part 2

parameter	'normal' value	test range
Penalty_IDS	80	[40, 60, 100, 120]
Penalty_HoneyPot	270	[135, 203, 338, 406]
zoning_penalty	15	[7.5, 11.25, 18.75, 22.5]
number of runs	15000	[5000, 10000, 40000, 80000]

Chapter 9, there are limitations and conditions to these results of the model and the metrics that have been created therefrom.

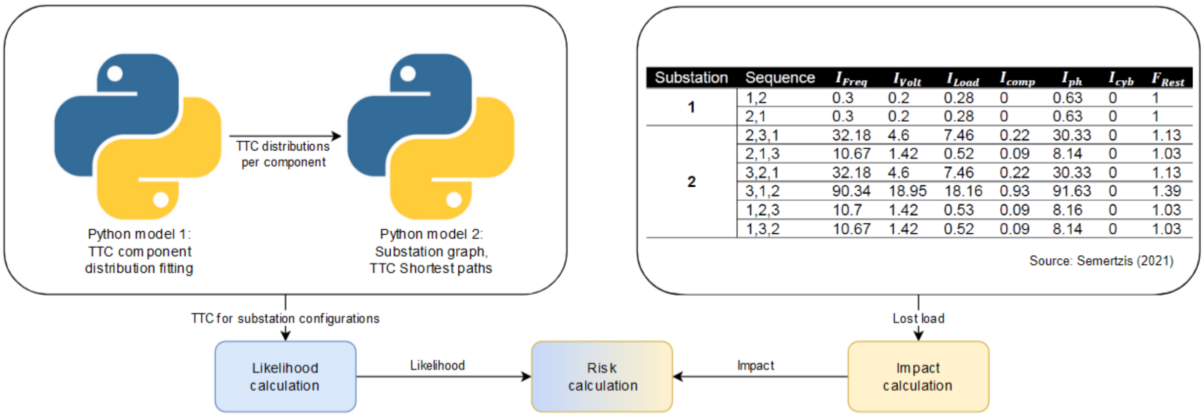
6.7.2. Verification

Verification of a model is defined as the process of examining if the implementation (coding) of a model is an accurate representation of the conceptual depiction of the developer's model and its solution [136]. The concept of the model was that it would mimic the possible paths that a malicious actor could take to disrupt the power supply. These attack paths, and therewith the TTC of the corresponding paths, would then be altered by means of the application of countermeasures. In chapters 7, 8, and 9 the results, limitations, and capabilities of the model are fully described. In brief, the model is able to present the desired metrics. So, it can be concluded that the conceptual model has been successfully implemented in the model code.

6.8. Model flow

In order to ultimately calculate the change in risk, two variables should be known. These are the likelihood that an event (in this case successful cyber attack) happens, and the impact that such an event has. In this study, the likelihood is calculated via the two Python models that supply the TTC per component and ultimately the overall attack path TTC for a substation with (or without) certain countermeasures. This TTC is then converted into likelihood by means of equation 6.16. The impact is calculated by using the study results of [18], which supply the lost load in the power system due to a successful cyber attack. Together with other variables the Impact is calculated according to equation 6.18. Once both the impact and likelihood are known, the risk can be calculated according to equation 6.11. In figure 6.10 an overview of the model flow is given.

Since there are a lot of assumptions made in this chapter that have an effect on the model outcome, all the assumptions have been placed in an overview below:



Substation	Sequence	I_{Freq}	I_{Volt}	I_{Load}	I_{comp}	I_{ph}	I_{cyb}	F_{Rest}
1	1,2	0.3	0.2	0.28	0	0.63	0	1
	2,1	0.3	0.2	0.28	0	0.63	0	1
2	2,3,1	32.18	4.6	7.46	0.22	30.33	0	1.13
	2,1,3	10.67	1.42	0.52	0.09	8.14	0	1.03
	3,2,1	32.18	4.6	7.46	0.22	30.33	0	1.13
	3,1,2	90.34	18.95	18.16	0.93	91.63	0	1.39
	1,2,3	10.7	1.42	0.53	0.09	8.16	0	1.03
	1,3,2	10.67	1.42	0.52	0.09	8.14	0	1.03

Source: Semertzis (2021)

Figure 6.10: Flow of used models and data set

Chapter 6 assumptions:

- the malicious actor always takes the path with the shortest TTC (also called the dominant attack path);
- only the modelled attack steps and respective components are considered;
- all known CVEs are in the database and none have been patched yet;
- the malicious actor is always in one of the three TTC processes;
- the total number of vulnerabilities V_{tot} is set at 6100;
- the time needed to successfully execute a readily available exploit for a given vulnerability t_1 is equal to 1 day;
- the average time needed per try for a malicious actor to develop an exploit for a given vulnerability is equal to 5.8 days;
- tempo of new vulnerability development becomes a constant over time;
- the mean time between the discovery of novel vulnerabilities is equal to 30.42 days;
- skill level of malicious actors is defined by a normal distribution with set base values, mean, and variance;
- attack efficiency is dependent on the respective skill level;
- only physical (and financial) impact are explicitly studied in this work;
- risk has a direct relation to the time which a malicious actor requires to compromise its intended target;
- base value of the MTTD is defined and set by CPPS cyber security experts, but in this work is equal to 14 days;
- IDS functions as an attack path deterrent, forcing malicious actors to choose a path without network sensors;
- zone trespassers always receive a TTC penalty of 15 days;
- Remote Attestation renders all firmware based attacks useless;
- modelled honeypot cannot be broken out of.

7 Simulation scenarios

This chapter is dedicated to outlining the different scenarios that have been run in the models. A total of six scenarios are chosen and executed. Each subchapter deals with one specific scenario. Per subchapter, the proposed countermeasures, the topology, the shortest attack path, change in TTC and risk, and lastly, the SCBA is discussed. The chapter ends with an overview of the different scenarios.

7.1. Malicious actor scenario setting

In order to study the impact of different countermeasures on the testbed, the fixed settings of the scenario must be defined. These fixed settings are the malicious actor's goal and their entry point.

7.1.1. Attacker goal

In this work, only one attack goal scenario is considered. This scenario refers to the malicious actor's ultimate goal (or final attack step). The considered scenario is the opening of the circuit breakers which are connected to the physical grid. The reason for choosing only this scenario is two-fold. Firstly, this is one of the only scenarios in which the final attack step immediately and measurably impacts the physical part of the CPPS. Secondly, calculating the benefits of a protective measure aimed at preventing this scenario is easier, due to the more direct relationship between the opening of circuit breakers and the loss of load in the physical part of the system [137][18]. This, compared against DoS scenarios which indirectly (and over a longer period) impact the physical system through the cyber layer [138].

7.1.2. Entry point

For all the scenarios that are run, the same entry point is chosen. This keeps the number of combinations between countermeasures and scenario settings at a manageable level. That is, due to the limited amount of time to complete this study. The entry point for the executed scenarios is the Gateway/RTU via the WAN. This is in accordance with the preceding study by [18], and this entry point is also mentioned by [139][140][34] and [141].

7.2. Defender detection time

In paragraph 6.3.1, the MTTD and RMTTD have been discussed. The multiplication factor, representing the capabilities of the defender to detect the malicious actor within a certain time, is assumed to be 1 for all scenarios.

Chapter 7 assumptions:

- the malicious actor has only one end-goal, which is opening the circuit breaker(s);
- for all the scenarios the same entry point is considered, namely the gateway, so the malicious actor already has a foothold in the system;
- the previous steps in the kill-chain are neglected;
- the multiplication factor of the RMTTD is equal to 1,

7.3. Scenario 1: Base case

The first scenario that will be run is the base case scenario. In this scenario, only the bare minimum of cyber security is applied. This bare minimum consists of a firewall that (partly) protects the Gateway/RTU from receiving unwanted and possibly malicious traffic. However, as mentioned in Paragraph 7.1.2. the Gateway/RTU is assumed to be a compromised entry point. Besides the firewall, there is also compartmentalization on CPPS system-wide level. This divides the entire CPPS system into different separate sectors, as can be seen in Figure 5.1, but this compartmentalization does not affect the TTC if the malicious actor is already in the substation system, as is the case in the constructed scenarios. This scenario will be used as the baseline measurement for comparison to the other countermeasure scenarios.

7.3.1. Dominant attack path

The dominant attack path in the first scenario can be seen in Figure 7.1. This attack path makes use of the man-in-the-middle vulnerability of the server controller, allowing the malicious actor to transfer the command of the server controller, and abuse this command to open the circuit breaker by modifying the parameters of the IED. This shortest path has an average minimum shortest path of 8.24 days.

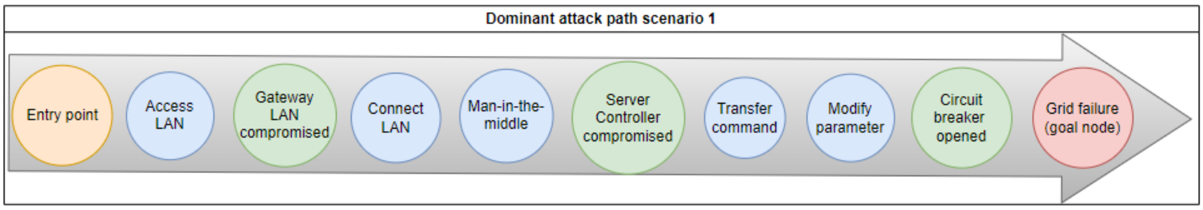


Figure 7.1: Dominant attack path scenario 1

7.3.2. TTC

As mentioned in paragraph 7.2.1, the TTC_{avg} of the shortest path is approximately 8.24 days. The average TTC of all the unique shortest attack paths is circa 20.92 days and the longest attack path takes around 33.06 days.

7.3.3. Likelihood & Risk

Calculating the risk for this scenario, in accordance with equation 6.11 requires the likelihood and the impact. Firstly, the likelihood for this scenario is calculated with equation 6.16. For the calculation of the likelihood the TTC_{avg} of the shortest (dominant) attack path is used, as this is the path of least resistance for the malicious actor and is the path that has the highest chance to be chosen as the definite attack path. Choosing the attack path with the shortest TTC_{avg} also ensures that the worst-case scenario is assumed.

$$Likelihood(scen1) = \frac{14}{8.24 + 14} = 0.629 \tag{7.1}$$

The impact on the load is dependent on the sequence in which the compromised circuit breakers are opened. In keeping with the categorization, as discussed in Chapter 6, the impact is given for the three levels of severity of the subsequent blackout.

By multiplying the likelihood with the total impact, and then dividing that by the discount rate, the TR per scenario type can be calculated:

$$TR(scen\ 1_{small}) = \frac{0.629 * \$21,910,711}{0.02} = \$689,091,861$$
$$TR(scen\ 1_{medium}) = \frac{0.629 * \$3,352,740,038}{0.02} = \$105,443,674,200$$
$$TR(scen\ 1_{large}) = \frac{0.629 * \$30,966,169,273}{0.02} = \$973,886,023,600$$

blackout severity	I_{Load}	total lost load	I_{total}	Total Risk
small	0.52	296.89 MWh	$21,910,711	$689,091,861
medium	7.46	45428 MWh	$3,352,740,038	$105,443,674,200
large	18.16	419538 MWh	$30,966,169,273	$973,886,023,600

Table 7.1: Blackout outcomes scenario 1

7.3.4. SCBA

Scenario 1 is the base case scenario in which no investments in countermeasures are made. Due to this fact, it is not of added value to perform and describe the SCBA of this scenario. However, because this is the base case scenario against which the other scenarios are compared, the total risk expressed in monetary value and the I_{total} should be considered. The I_{total} is calculated by multiplying the total lost load of that sub-scenario (expressed in MWh) with the VoLL (in $/MWh) The results are shown in Table 7.1.

The TR for the first scenario varies between $689,091,861 for a small blackout up to $973,886,023,600 for a large blackout. The average TR of the three severity sub-scenarios is $360,006,263,220.

The total value of lost load is equal for all scenarios, as the sub-scenario categorizations (levels of blackout severity) have identical I_{Load} values and duration per category. So, the differences per scenario (and thus sub-scenarios) lie in the change in likelihood.

7.4. Scenario 2: IDS

In the second scenario the base case model is expanded by the implementation of an Intrusion Detection System (IDS). This IDS should be able to detect a certain portion of malicious actors, through the use of network sensors, and increase their TTC for that attack. A more elaborate description of the workings of an IDS is given in section 6.4.1.

7.4.1. Dominant attack path

The dominant attack path in the second scenario is shown in Figure 7.2. It has a TTC_{avg} of 14.48 days. This attack path makes use of a vulnerability in the firmware of the station switch to eventually use the same type of firmware vulnerability on the HMI. Once the HMI is compromised the IED parameters are changed and the circuit breakers are opened as a consequence.

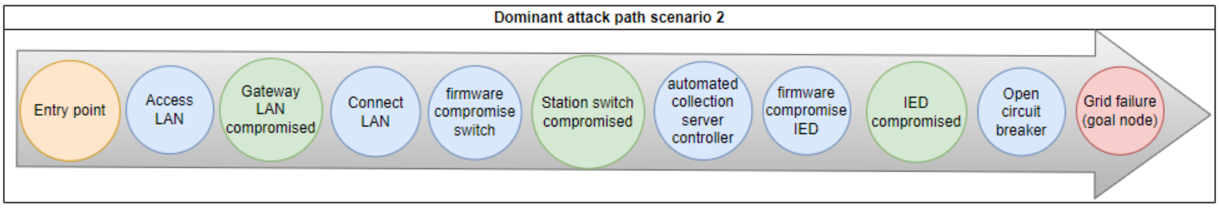


Figure 7.2: Dominant attack path scenario 2

7.4.2. TTC

As mentioned in paragraph 7.3.1, the TTC_{avg} of the shortest path is approximately 14.48 days. The average TTC of all the unique shortest attack paths is circa 27.03 days and the longest attack path takes around 88.48 days.

7.4.3. Likelihood & Risk

The likelihood for scenario 2, while once again presuming the shortest path will be executed by the malicious actor, is calculated below:

$$Likelihood(scen2) = \frac{14}{14.48 + 14} = 0.492 \tag{7.2}$$

7.4.4. SCBA

The cost part of the Societal Cost Benefit Analysis consists of direct costs which in turn, are made up of investment, maintenance, and exploitation costs [123]. The initial investment costs for deploying an IDS are estimated to be around $107.000. According to [142] the yearly maintenance costs of an IDS are equal to 15% of the initial investment costs. Which in this case is equal to $ 16.050. The exploitation costs are 4.8 hours per policy. The hourly rate for tuning such systems was estimated to be 75 $/hr in 2014 [143]. Adjusting this figure for an average increase in rates of 3% brings the hourly rate to $98 in 2023. Since there is only one IDS policy in this scenario the exploitation costs per year is approximately $5645. This brings the total direct costs to nearly $130.000. The entire calculation can be found in appendix F.

The TR for the second scenario varies between $539,003,491 for a small blackout up to $761,767,764,100 for a large blackout. The average TR of the three severity sub-scenarios is $281,594,724,174. The TR for each subscenario of scenario 2 can be found in Table 7.2

blackout severity	Total Risk
small	$539,003,491
medium	$82,477,404,930
large	$761,767,764,100

Table 7.2: Total Risk per blackout subscenario of scenario 2

7.5. Scenario 3: Zoning

As a third scenario the application of zoning in within the digital substation has been chosen. In essence, zoning comes down to dividing the digital substation into different ’zones’ that are separated from each other and force the network traffic through certain entry points that are monitored and protected. A more elaborate description of the workings of zoning is given in section 6.4.2.

7.5.1. Dominant attack path

When zoning is applied as a countermeasure against malicious actors a dominant attack path becomes apparent. This attack path is shown in Figure 7.3 and has a TTC_{avg} of 23.04 days. In this figure, it is shown that the attack path utilizes the man-in-the-middle approach to compromise the server controller and send commands to the IED to modify parameters which leads to the opening of the circuit breaker and thus, the eventual goal of the malicious actor.

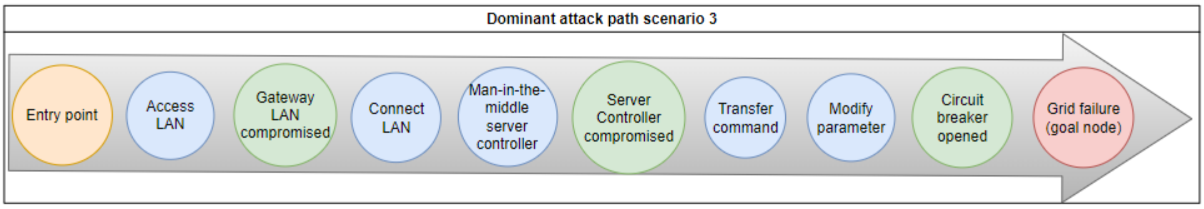


Figure 7.3: Dominant attack path scenario 3

7.5.2. TTC

As mentioned in paragraph 7.4.1, the TTC_{avg} of the dominant attack path of scenario 3 is approximately 23.04 days. The average TTC of all the unique shortest attack paths is circa 34.21 days and the longest attack path takes around 49.91 days.

7.5.3. Likelihood & Risk

The likelihood for scenario 3, while once again presuming the shortest path will be executed by the malicious actor, is calculated below:

$$Likelihood(scen3) = \frac{14}{23.04 + 14} = 0.378 \quad (7.3)$$

7.5.4. SCBA

The initial costs for implementing zoning (or network segmentation) vary greatly, but an average of $50.000 is used in a business case by [144]. This same business case approximates the maintenance/exploitation costs to be between $40.000 and $100.000. In this study, the average of this given estimate is taken, which is $70.000. This brings the total direct costs to $120.000

The TR for the third scenario varies between $414,112,438 for a small blackout up to $585,260,599,300 for a large blackout. The average TR of the three severity sub-scenarios is $216,347,166,152. The TR for each subscenario of scenario 3 can be found in Table 7.3

blackout severity	Total Risk
small	$414,112,438
medium	$63,366,786,720
large	$585,260,599,300

Table 7.3: Total Risk per blackout subscenario of scenario 3

7.6. Scenario 4: Remote attestation

The fourth scenario is the implementation of remote attestation to a certain set of components. Remote attestation protects the firm/soft-ware from certain types of attack steps. Thus, making it harder for a malicious actor to perform the entire kill chain and ultimately launch an attack at its intended target. A more elaborate description of the workings of remote attestation is given in section 6.4.3.

7.6.1. Dominant attack path

Application of remote attestation on the eligible components has brought forth the dominant attack path can be seen in Figure 7.4. This dominant path has a TTC_{avg} of 8.01 days. In this path, the malicious actor would exploit a vulnerability in the server controller through a man-in-the-middle attack which allows the malicious actor to gain command of the server controller and modify the IED parameters. Ultimately this would lead to the opening of the circuit breakers and thus the end goal of the malicious actor.

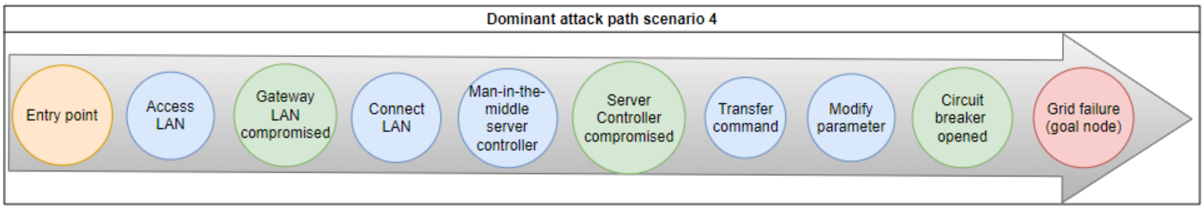


Figure 7.4: Dominant attack path scenario 4

7.6.2. TTC

As previously discussed in paragraph 7.5.1, the TTC_{avg} of the dominant attack path of scenario 4 is approximately 8.01 days. The average TTC of all the unique shortest attack paths is circa 40.11 days and the longest attack path takes around 63.21 days.

7.6.3. Likelihood & Risk

The likelihood for scenario 4, which still assumes that the shortest path will be executed by the malicious actor, is calculated below:

$$Likelihood(scen4) = \frac{14}{8.01 + 14} = 0.636 \tag{7.4}$$

7.6.4. SCBA

As (advanced) remote attestation only gained popularity in the last decade and commercial usages in digital substations are slim, not much financial information can be found on this subject [113]. According to [145] a Trusted Platform Module (TPM), the AtmelAT97SC3203S to be precise, has a price of around $5. This is relatively cheap compared to the other countermeasures. However, this is only the price of the materials. To calculate the entire direct costs one has to take into account not only hardware costs, but also software costs for managing all the remote attestation hardware, integration costs for implementing these modules in the existing infrastructure, but also training and auditing costs. In total the direct costs for implementing remote attestation are approximately $46.000. The entire calculation can be found in appendix F.

The TR for the fourth scenario varies between $696,760,610 for a small blackout up to $984,724,182,900 for a large blackout. The average TR of the three severity sub-scenarios is $364,012,692,237. The TR for each subscenario of scenario 4 can be found in Table 7.4

blackout severity	Total Risk
small	$696,760,610
medium	$106,617,133,200
large	$984,724,182,900

Table 7.4: Total Risk per blackout subscenario of scenario 4

7.7. Scenario 5: Deception

The last distinct countermeasure is the use of deception technology in digital substations to lure and delay a malicious actor. This is done through the use of honeypots in the system. Honeypots - as the name implies - attract malicious actors by acting as an alluring target, which in reality is a custom-made trap for the malicious actor to spend time in and be detected. A more elaborate description of the workings of an IDS is given in section 6.4.4.

7.7.1. Dominant attack path

The use of a honeypot countermeasure generates not one, but two dominant attack paths. The first attack path has a TTC_{avg} of 276.14 days and is shown in Figure 7.5. In this path the malicious actor compromises the gateway and goes straight for the honeypot which is (from the malicious actor's perspective) a relatively easy target (TTC = 1 day), and which leads directly to IED access. Because this honeypot is a trap that is only accessed by actors with bad intent, the defenders know with certainty that an attack is occurring and can take protective measures to fend off the malicious actor. This causes the TTC_{avg} to increase substantially and the attack path stops at the honeypot. This causes the formation of another attack path that does not include the honeypot steps. The alternative attack path without the honeypot is shown in Figure 7.6 and has a TTC_{avg} 278.78 days.

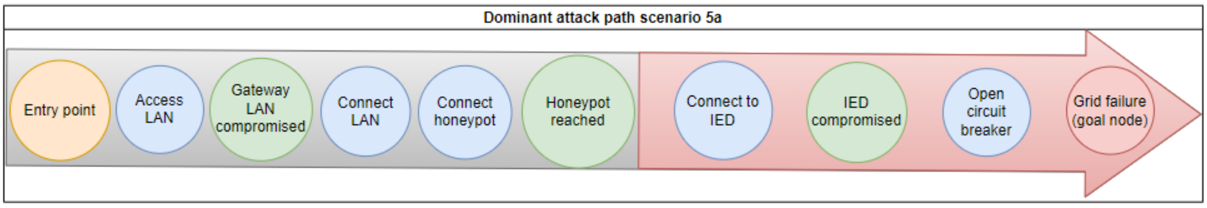


Figure 7.5: Dominant attack path scenario 5a

7.7.2. TTC

The primary dominant attack path has a TTC_{avg} of approximately 276.14 days. The average TTC of all the unique shortest attack paths is circa 293.73 days and the longest attack path takes around 314.01 days.

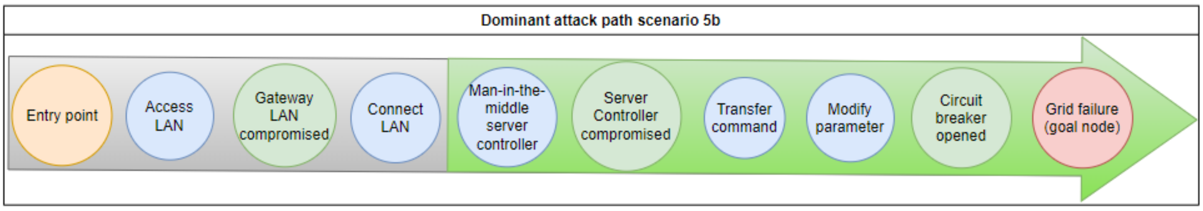


Figure 7.6: Dominant attack path scenario 5b

7.7.3. Likelihood & Risk

The likelihood for scenario 5, assuming that the shortest path will be executed by the malicious actor, is calculated below:

$$Likelihood(scen5) = \frac{14}{278.78 + 14} = 0.048 \tag{7.5}$$

7.7.4. SCBA

The direct costs for the deployment of a honeypot as deception technology consists of licensing costs for the software, setup costs for configuring the software, monthly policy maintenance, and integration costs. The total costs are approximately $31.000, and the detailed calculation for this can be found in Appendix F.

The TR for the fifth scenario varies between $52,585,706 for a small blackout up to $74,318,806,260 for a large blackout. The average TR of the three severity sub-scenarios is $27,472,656,019. The TR for each subscenario of scenario 5 can be found in Table 7.5

blackout severity	Total Risk
small	$52,585,706
medium	$8,046,576,091
large	$74,318,806,260

Table 7.5: Total Risk per blackout subscenario of scenario 5

7.8. Scenario 6: All-in

The sixth scenario is the last scenario and consists of a combination of all the aforementioned countermeasures. Every countermeasure is active at the same time.

7.8.1. Dominant attack path

When all the countermeasures of the previously described scenarios are combined into a single scenario, a dominant attack path (which excludes the honeypot) becomes clear which can be seen in 7.7. This path has a TTC_{avg} of around 304.47 days. The malicious actor takes advantage of the authentication breach vulnerability in the station switch to gain transfer the command of the HMI via a man-in-the-middle attack. This ultimately enables the malicious actor to modify the parameters of the IED which triggers the opening of the circuit breaker connected to said IED. Once the circuit breaker is opened, a failure in the grid is imminent and thus the goal of the malicious actor is reached.

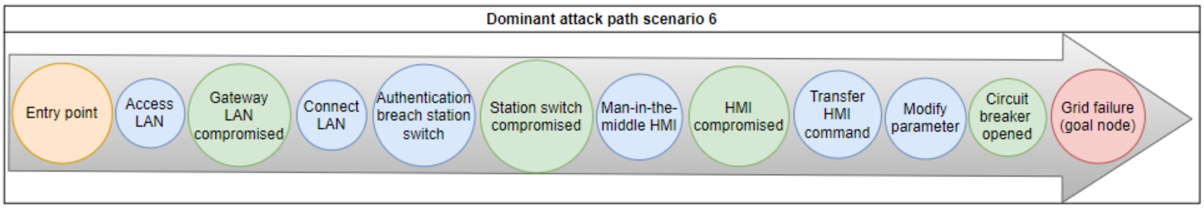


Figure 7.7: Dominant attack path scenario 6

7.8.2. TTC

The dominant feasible attack path has a TTC_{avg} of circa 304.47 days. The average TTC of all the unique shortest attack paths is circa 332.15 days and the longest attack path takes around 377.57 days.

7.8.3. Likelihood & Risk

The likelihood for scenario 6, presuming that the shortest path will be executed by the malicious actor, is calculated below:

$$Likelihood(scen6) = \frac{14}{304.47 + 14} = 0.044 \tag{7.6}$$

7.8.4. SCBA

The direct cost of deploying all of the countermeasures at once is equal to the sum of the individual direct costs, plus additional costs for implementation and integration of all the individual countermeasures into a coherent cyber security system. The summed cost of the individual countermeasures is $322.800. If the financial worst-case scenario is taken, a supplementary amount of around $90.000 could be needed to stitch the patchwork of individual countermeasures into a single unified cyber security blanket. This includes change management, implementation, integration costs, operations, and IT service management [144]. This would bring the total direct costs to $412.800. The entire calculation can be found in Appendix F.

The TR for the last scenario varies between $48,203,564 for a small blackout up to $68,125,572,400 for a large blackout. The average TR of the three severity sub-scenarios is $25,183,268,016. The TR for each subscenario of scenario 6 can be found in Table 7.6

blackout severity	Total Risk
small	$48,203,564
medium	$7,376,028,084
large	$68,125,572,400

Table 7.6: Total Risk per blackout subscenario of scenario 6

7.9. Overview

This section of chapter 7 will provide an overview of the results of all the scenarios that have been carried out. This overview will be provided in Table 7.7 below:

scenario	dominant path TTC_{avg}	average TTC_{avg} of unique shortest paths	TTC_{avg} longest unique path	likelihood	Average Total Risk	costs
1: base case	8.24	20.92	30.06	0.629	$360,006,263,220	$0
2:IDS	14.48	27.03	88.48	0.492	$281,594,724,174	$129,000
3: zoning	23.04	34.21	49.91	0.378	$216,347,166,152	$120,000
4: RA	8.01	40.11	63.21	0.636	$364,012,692,237	$46,000
5: honeypot	276.14	293.73	314.01	0.048	$27,472,656,019	$27,800
6: all-in	304.47	332.15	377.57	0.044	$25,183,268,016	$412,800

Table 7.7: Overview scenario results

8 Discussion

In this chapter the results of the scenarios, as described in Chapter 7, are discussed. The scenarios are addressed one by one and the most interesting findings will be highlighted. The analyses of these scenarios are used as input for the next chapter, Chapter 9: Conclusions, where the sub-questions, and ultimately the main research question, will be answered.

8.1. Results 1: Base case

The first scenario, concerns a digital substation that consists of solely the components without any substation-level cyber attack countermeasures in place. This scenario will be used as a zero-measurement against which the other scenarios are compared to see the added value of their respective countermeasure(s).

8.1.1. (dominant) attack path TTC

The dominant attack path of the base case scenario has a TTC_{avg} of 8.24 days and the average TTC_{avg} of all the unique shortest paths is 20.92. The dominant attack path first goes through the Gateway asset, as in this study setup, it is the only point through which to proceed to subsequent attack steps. The Gateway is in all cases compromised by the access_LAN attack step, as this step has a significantly lower TTC distribution. Afterwards, the server controller is compromised through a man-in-the-middle attack. The choice for this specific attack step becomes clear when the entire attack steps graph is analyzed. Once the Gateway is compromised the malicious actor can proceed to connect to the LAN for zero TTC. From that point, there are three options of which the man-in-the-middle option has the lowest TTC distribution mean. This should not guarantee that this step is always the quickest option, as the distributions of the other two steps (firmware compromise and authentication breach of the station switch) have TTCs that are close to that of the man-in-the-middle step. Due to the combination of the random distribution of these steps and the Monte Carlo simulation, it becomes possible for the other two steps to become the shortest step to take within a simulation run. However, because all the attack steps that succeed the man-in-the-middle server controller step have a TTC of 0, taking that step will pay off in the end as this yields the lowest overall TTC for that attack path.

When the average TTC_{avg} 20.92 of all the unique shortest paths is also taken into consideration, it is clear that this dominant attack path with an average TTC_{avg} of 8.24 is quite favourable when compared to the other paths.

8.1.2. Likelihood, risk, and financial aspects

Ultimately, the likelihood and average TR that has been calculated in this scenario are 0.629 and \$360,006,263,220 respectively. This likelihood, and therefore automatically the risk, are higher than empirical observations. According to one observation of a study by [146] the likelihood of a certain cyber attack can lie in the range of [0.029, 0.271]. An explanation for this discrepancy between practice and theory can be found in the malicious actor scenario setting, as discussed in Paragraph 7.1. In the constructed scenarios and the underlying assumptions of the malicious actor scenario setting, only the end part of the kill chain is taken into account and modelled. All the previous steps which had to be executed are taken as given. A result of this assumption is that the likelihood of the studied scenarios is relatively high, compared to empirical observations. In order to incorporate the difficulty for a malicious actor to reach the starting point of the scenarios in the current study, the likelihoods of all scenarios are multiplied by a scaling factor. This scaling factor will take into account the select boundary demarcation of the kill chain in the studied scenarios. This multiplication factor is called the Kill chain Scoping Factor (KSF). The height of the KSF is based on the number of presumably executed steps needed by the malicious actors in the 2015 Ukraine cyber attack on the power grid. According to [137], there were two stages in the 2015 Ukraine cyber attack. The first stage was the intrusion stage, in which the malicious actor gained entry into the industrial control system of the CPPS. This first stage is assumed to be equal to all the steps that have been performed before the malicious actor is at the starting point of the studied scenarios. The second stage of the kill chain is the attacking stage, which is assumed to be equal to the attack steps in the scenarios of this work. According to [137], the first stage consists of eight steps, and the second stage consists of five steps. So, of the presumed 13 total steps, only 5 are considered in this study. So if we want the likelihood of the studied scenarios to be 'normalized' the KSF should be equal to (5/13), which is 0.38. This would change the likelihood of the first scenario to 0.242, which is within the range of the empirical observations by [146].

The adjusted TR calculation would then yield the following results:

$$TR(scen\ 1_{small}) = \frac{0.629 * \$21,910,711}{0.02} * 0.38 = \$261,854,490$$
$$TR(scen\ 1_{medium}) = \frac{0.629 * \$3,352,740,038}{0.02} * 0.38 = \$40,068,596,196$$
$$TR(scen\ 1_{large}) = \frac{0.629 * \$30,966,169,273}{0.02} * 0.38 = \$370,076,688,968$$

The KSF has been placed next to the division so it is more clearly visible in the calculation. Since this limitation is present in all scenarios, the KSF factor will also be applied in all scenarios to see the normalized results.

Furthermore, the average TR of the base case scenario is $136,802,379,900. This amount might seem high but is close to the expected sum range. For example, similar sums ($4.8 billion - $10 billion) are estimated in the U.S. - Canada power outage [132], which is a medium-sized outage, according to the classification of this study. However, the TR in this study is calculated with the VoLL of 2023, which is equal to €68,887/MWh, while the U.S. - Canada power outage occurred in 2003, when the VoLL in Europe was just equal to €8600/MWh [147]. Thus, the VoLL of 2023 is 8.01 times higher than the VoLL of 2003. Note that the European VoLLs (expressed in €) are used as these are the only VoLL values that can be found for both years. So, in order to better compare the TR amount of the current study with the amount of damage in the study by [132], the TR has to be divided by a factor of 8.01. This makes the average TR of the base scenario equal to $17,078,948,882, which deviates only 70% of the ceiling value of the U.S. - Canada range. For the small, medium, and large subscenarios the VoLL adjusted TR values are: $32,690,947, $5,002,321,622, and $46,201,833,830 respectively. Nevertheless, this monetary value that is allocated is not an absolute sum that is received if the countermeasure is implemented, but is merely an indicator of the relative potential added value that this countermeasure might produce in a certain scenario under the given assumptions and generalizations.
Now that the highlights of the base case have been addressed it is time to discuss how an Intrusion Detection System would alter the outcome as has been generated by the base case scenario.

8.2. Results 2: IDS

In the second scenario an Intrusion Detection System is added as a countermeasure against malicious actors. The network sensors of this intrusion detection system are placed in two places in the attack path graph. The choice has been made to only use two network sensors since applying many sensors has diminishing returns and increases the costs and the manual effort that is required for monitoring the potential intrusions which are flagged by the sensors [148]. The two sensors have been placed at the Gateway component and in the server controller component. The choice for these two specific locations has been made on the knowledge that in the base case the server controller was the weakest link in the attack chain.

8.2.1. (dominant) attack path TTC

The placement of the IDS sensors in these locations has had the desired effect. The dominant attack path for this simulation has been altered. This attack path has deviated from the server controller man-in-the-middle vulnerability and has instead moved to compromise the station switch and then used automated collection on the station controller. This does not require the server controller to be compromised but only collects log files to see which devices are connected to the station controller. Once the malicious actor has seen all the connected devices the focus is put on the IED which is compromised by taking advantage of a vulnerability in the IED firmware. Once the IED is compromised the circuit breaker is opened and the goal of the malicious actor is reached. This dominant attack path has a TTC_{avg} of 14.48 days. Compared to a TTC_{avg} of 8.24 days for the dominant attack path from the base case scenario, the application of an Intrusion Detection System has led to an increase in TTC of 75.7%. The average TTC_{avg} of the unique shortest paths in the second scenario has increased by 29% and the TTC_{avg} of the longest unique path has increased by 194.3%. This large disparity between the increase in TTC_{avg} of the longest unique path and the average TTC_{avg} of the unique shortest paths can be explained by the fact that the IDS heavily penalizes the unique shortest path that goes through both sensor nodes ”gateway_LAN_compromised” and ”server_controller_compromised” by adding 80 TTC to that path.

8.2.2. Likelihood, risk, and financial aspects

The adjusted TR calculation for the second scenario would yield the following results:

$$TR(scen\ 2_{small}) = \frac{0.492 * \$21,910,711}{0.02} * 0.38 = \$204,821,327$$

$$TR(scen\ 2_{medium}) = \frac{0.492 * \$3,352,740,038}{0.02} * 0.38 = \$31,341,413,870$$

$$TR(scen\ 2_{large}) = \frac{0.492 * \$30,966,169,273}{0.02} * 0.38 = \$289,471,750,400$$

The likelihood and average TR that has been calculated in scenario 2 are 0.492 (times 0.38 KSF, makes 0.187) and $107,005,995,200 respectively. If the VoLL comparison factor (of 8.01) is applied, then the average TR is equal to $13,359,050,590. Both the likelihood and TR of scenario 2 have decreased by 21.8% compared to the base case scenario.

This change in dominant attack path and increase in TTC_{avg} of 75.7% of that same attack path, combined with a likelihood and total risk reduction of 21.8% comes at a price of approximately $129.000. However, the average TR would decrease from $17,078,948,800 to $13,359,050,590, meaning that this countermeasure could achieve a $3,719,898,215 reduction.

These are the KPIs that are leading in this study, as these are required for answering the subquestions of the main research question.

8.3. Results 3: Zoning

The third scenario to be discussed is the deployment of zoning in digital substations as countermeasures against malicious actors. Deploying the zoning countermeasure has led to an interesting result; the dominant attack path of scenario 3 is the same as the dominant attack path of the base case scenario (scenario 1). There seems to be a logical explanation for this outcome which, once again, can be found in the application of the penalty in the Python code. The zoning countermeasure has effectively cut the digital substation into two separate parts. For the sake of convenience, these zones are called Zone 1 and Zone 2. The first zone consists of all the components (and related attack steps) up and including the station switch, the server controller, and the HMI. The remainder of the components and their corresponding attack steps are part of the second zone. Whenever a malicious actor takes an attack step that crosses the boundary of Zone 1 into Zone 2, then a penalty of 15 TTC is applied to that corresponding attack path.

8.3.1. (dominant) attack path TTC

Because all of the attack paths need to cross this boundary, all of these attack paths will receive the same penalty. Therefore the original dominant attack path of the first scenario is once again the dominant attack path in this scenario, but this time with a higher TTC_{avg}. The TTC_{avg} of the dominant attack path of scenario 3 is 23.04 days, which is 179.6% higher than the TTC_{avg} of the dominant attack path from scenario 1. The average TTC_{avg} of the unique shortest paths in the third scenario has increased by 63.5% and the TTC_{avg} of the longest unique path has increased by 66%. In this scenario, the TTC_{avg} of the dominant attack path is much higher than its base case counterpart, while the other TTC_{avg} metrics are in a similar range as those in scenario 1. The explanation of the large discrepancy between the dominant attack path metrics of these two scenarios can be attributed to how the shortest path algorithm responds to the application of the zoning penalty. As previously described, the zoning penalty of plus 15 TTC is applied when an attack path crosses over into another zone, which always happens given the possibility of attack steps. As every possible path gains this additional 15 TTC the dominant attack path will also gain this penalty. This is completely different from how the Intrusion Detection System operated, which penalized a specific part of the solution space. That way the algorithm looks for the shortest path from the source node ”entry_point” to the ”goal_node” forces it to look for the shortest path including the zoning penalty. That is why even the shortest path (dominant attack path) has a relatively higher TTC_{avg} and the other metrics are similar to that of the base case scenario.

From the results, it can be concluded that applying a form of zoning in a digital substation is an even (or spread out) method that increases the TTC across the board. This is in stark contrast to the Intrusion Detection System countermeasure of scenario 1, which penalized just a certain subset of possible paths.

8.3.2. Likelihood, risk, and financial aspects

The adjusted TR calculation for the third scenario would yield the following results:

$$TR(scen\ 3_{small}) = \frac{0.378 * \$21,910,711}{0.02} * 0.38 = \$157,362,726$$
$$TR(scen\ 3_{medium}) = \frac{0.378 * \$3,352,740,038}{0.02} * 0.38 = \$24,079,378,950$$
$$TR(scen\ 3_{large}) = \frac{0.378 * \$30,966,169,273}{0.02} * 0.38 = \$222,399,027,700$$

The likelihood and average TR that has been calculated in the third scenario are 0.378 (times 0.38 KSF, makes 0.144) and \$82,211,923,140 respectively. If the VoLL comparison factor (of 8.01) is applied, then the average TR is equal to \$10,263,660,820. This amount of total average risk nearly falls within in the range that has been reported in the U.S. - Canada power outage. Both the likelihood and TR of scenario 3 have been decreased by 39.9% compared to the base case scenario.

This preservation of the dominant attack path, but an increase in TTC_{avg} of 179.6% of that same attack path, combined with a likelihood and total risk reduction of 39.9%, comes at a cost of approximately \$120.000. Though, the average TR would decrease from \$17,078,948,800 to \$10,263,660,820, meaning that this countermeasure could achieve a \$6,815,287,984 reduction.

8.4. Results 4: Remote attestation

As a fourth scenario, the remote attestation of certain components within the digital substation has been chosen as a countermeasure against cyber attacks from malicious actors. As described in paragraph 6.4.3. the use of (hardware based) remote attestation makes it nigh impossible to compromise the firmware of those components.

8.4.1. (dominant) attack path TTC

The application of remote attestation has had a similar result as what happened in the third scenario. Namely, the dominant attack path has stayed the same as it has been in the base case scenario. Once again the server controller based attack path has come out as the shortest route to the goal node of the malicious actor. The explanation for this result can be found in the specifics of the remote attestation countermeasure. Due to the fact that remote attestation only 'patches' vulnerabilities that are related to the compromise of the firmware of components, it leaves other attack steps that don't depend on those vulnerabilities, unaffected. On the grounds that the server controller can be compromised without abusing a firmware compromise vulnerability, the dominant attack path stays unchanged compared to the base case scenario. As a matter of fact, the TTC_{avg} of the dominant attack path of scenario 4, is actually lower than that of the base case scenario. Scenario 4 has a TTC_{avg} of 8.01 days compared to a TTC_{avg} of 8.01 in the first scenario. This discrepancy can be explained by the random distributions of all relevant attack steps in the dominant attack path. During every run, the values of the TTC of the relevant attack steps are randomly chosen according to their given distribution. This causes deviations in the total TTC of attack paths. In this case, the randomness has caused the overall TTC of the dominant attack path to have been 2.8% lower than that of the dominant attack path run in the first scenario. However, the average TTC_{avg} of the unique shortest paths in scenario 4 has increased by 92.7% and the TTC_{avg} of the longest unique path has increased by 110.3%. This difference between this scenario and the base case in average TTC_{avg} of the unique shortest paths and the TTC_{avg} of the longest unique path, finds its origin in the same reasoning as the difference in the TTC_{avg} of the dominant attack path between the aforementioned scenarios.

It is clearly visible that remote attestation, as could be expected, is a specific countermeasure that makes firmware compromise based attack paths very unattractive for malicious actors.

8.4.2. Likelihood, risk, and financial aspects

The adjusted TR calculation for the fourth scenario would yield the following results:

$$\begin{aligned}
TR(scen\ 4_{small}) &= \frac{0.636 * \$21,910,711}{0.02} * 0.38 = \$264,769,032 \\
TR(scen\ 4_{medium}) &= \frac{0.636 * \$3,352,740,038}{0.02} * 0.38 = \$40,514,510,620 \\
TR(scen\ 4_{large}) &= \frac{0.636 * \$30,966,169,273}{0.02} * 0.38 = \$374,195,189,500
\end{aligned}$$

The likelihood and average TR that has been calculated in the third scenario are 0.636 (times 0.38 KSF, makes 0.242) and \$138,324,823,100 respectively. If the VoLL comparison factor (of 8.01) is applied, then the average TR is equal to \$17,269,016,610. The likelihood, as well as the risk calculated in this fourth scenario, have been increased by 1.1% compared to the base case scenario.

The conservation of the dominant attack path and a decrease in TTC_{avg} of 2.8% of that same attack path, combined with a likelihood and risk increase of 1.1%, comes at a price of approximately \$46.000 and increases the average TR by \$190,067,811. These "negative" values, as previously mentioned, are caused by the randomness of the TTC distributions combined with the Monte Carlo simulation, and should not be taken at face value. Instead, these numbers that show a very low deviation from the base case, should be regarded as a non-significant change in outcome. That is if such a countermeasure is implemented in isolation, under similar assumptions as the given scenario.

8.5. Results 5: Deception

The fifth scenario is the last scenario that introduces a singular cyber security countermeasure against attacks by malicious actors. This last countermeasure uses deception to trick the malicious actor into a trap by use of a honeypot.

8.5.1. (dominant) attack path TTC

In this fifth scenario the most anomalies occur. Initially, the dominant attack path goes through the added honeypot attack steps, as these steps give access to the IED (and thus the circuit breaker) while taking the shortest amount of TTC. This attack path has a TTC_{avg} of 276.14 days. However, this attack path is not feasible for the attacker as the honeypot is merely a trap, and (as assumed in this study) cannot be broken out of in order to continue to the IED and successive attack steps. Be that as it may, it is still possible for the attacker to return a couple of steps and try again through another attack path. This attack path is the unique shortest attack path with the second lowest overall TTC, which in this case is equivalent to a TTC_{avg} of 278.78 days. As the attacker has been inside the honeypot, the cyber security specialists of the TSO are alerted to the presence of the attacker and know that a cyber attack is imminent. Because of this precognition, the defenders can take adequate measures to make it harder for the malicious actor to reach its target node. This has been modelled in the model as a substantial penalty of an additional 270 TTC. Therefore, making it extraordinary for the attacker to continue the attack after being caught red-handed. This high penalty for being caught in the honeypot has led to an exorbitant increase in TTC_{avg} of 3226.9% for the dominant attack path, compared to the base case scenario. This large surge in TTC_{avg} for the dominant attack path is caused by the penalty which is triggered when the malicious actor falls for the honeypot trap. This exact same cause also explains the large increase in both the average TTC_{avg} of the unique shortest path which has increased by 1304.1% and the TTC_{avg} of the longest unique path which has grown by 944.6%.

A big assumption in this scenario and the way the dominant path is chosen and calculated, is that the malicious actor is presumed to always fall into the honeypot trap first and then looking for an alternative route while being heavily penalized. This excludes the possibility that this expert-level (as assumed in this study) malicious actor has prior knowledge of the possibility of the presence of such a honeypot. If a malicious actor has fallen into such traps before, then that actor might recognize its patterns and tactics, and decide to skip the attractive easy route that includes the honeypot. Instead, it will go for the second shortest path which will then be executed without receiving a penalty. This scenario is therefore subject to the condition that the attacker indeed goes for the honeypot first.

8.5.2. Likelihood, risk, and financial aspects

The adjusted TR calculation for the fifth scenario would yield the following results:

$$TR(scen\ 5_{small}) = \frac{0.048 * \$21,910,711}{0.02} * 0.38 = \$19,982,568$$
$$TR(scen\ 5_{medium}) = \frac{0.048 * \$3,352,740,038}{0.02} * 0.38 = \$3,057,698,915$$
$$TR(scen\ 5_{large}) = \frac{0.048 * \$30,966,169,273}{0.02} * 0.38 = \$28,241,146,380$$

The likelihood and average TR that has been calculated in the third scenario are 0.048 (times 0.38 KSF, makes 0.02) and $10,439,609,290 respectively. If the VoLL comparison factor (of 8.01) is applied, then the average TR is equal to $1,303,322,009. Both the likelihood and risk as calculated in the fifth scenario have been decreased by 92.4% compared to the base case scenario.

The eventual conservation of the dominant attack path and an increase in TTC_{avg} of 4336.9% of that same attack path, combined with a likelihood and average TR reduction of 92.4%, comes at a price of approximately $27.800. So, the average TR would decrease from $17,078,948,800 to $1,303,322,009, meaning that this countermeasure could achieve a $13,359,050,590 reduction.

8.6. Results 6: All-in

The sixth and final scenario to be discussed is the all-in scenario. In this scenario, all the aforementioned countermeasures are deployed simultaneously.

8.6.1. (dominant) attack path TTC

The concurrent application of all the countermeasures has led to the emergence of a novel dominant attack path which has not been seen as a dominant path before in the preceding scenarios. This attack path makes consists of (among others) the compromising of the station switch by means of an authentication breach, followed by taking over the control of the HMI via a man-in-the-middle attack. This gives the malicious actor the possibility to manually adjust the parameters of the IED which leads to the opening of the circuit breaker and thus the goal of the malicious actor. This dominant attack path has a TTC_{avg} of 304.47 days and, similar to scenario 5, is the shortest attack path without the honey pot nodes. Which is an increase of 3595% compared to the base case scenario. Compared to the base case scenario the average TTC_{avg} of the unique shortest paths in this last scenario has increased by 1487.7% and the TTC_{avg} of the longest unique path has increased by 1156.1%.

In this scenario the same big assumption applies here, as the one that was made in scenario 5. Making this scenario also conditional.

8.6.2. Likelihood, risk, and financial aspects

The adjusted TR calculation for the final scenario would yield the following results:

$$TR(scen\ 6_{small}) = \frac{0.044 * \$21,910,711}{0.02} * 0.38 = \$18,317,354$$
$$TR(scen\ 6_{medium}) = \frac{0.044 * \$3,352,740,038}{0.02} * 0.38 = \$2,802,890,672$$
$$TR(scen\ 6_{large}) = \frac{0.044 * \$30,966,169,273}{0.02} * 0.38 = \$25,887,717,510$$

The likelihood and average TR that has been calculated in the third scenario are 0.044 (times 0.38 KSF, makes 0.02) and $9,569,641,846 respectively. If the VoLL comparison factor (of 8.01) is applied, then the average TR is equal to $1,194,711,841. The likelihood and average TR as calculated in scenario 6 have both been decreased by 93% compared to the base case scenario.

This change in dominant attack path and the increase in TTC_{avg} of 3595% of that same attack path, combined with a likelihood and risk decrease of 93%, comes at a price of approximately $412.800, and could in theory, decrease the average TR from $17,078,948,800 to $1,194,711,841. This would constitute a reduction in the average TR of $15,884,236,960.

8.7. Adjusted overview

In this section of the discussion chapter an overview of the adjusted scenarios, as discussed in the previous sections, is presented in the table below:

scenario	likelihood	adjusted average TR	Δ adjusted average TR	costs
1: base case	0.242	$17,078,948,800,	N.A.	$0
2:IDS	0.187	$13,359,050,590.	-$3,719,898,215	$129,000
3: zoning	0.144	$10,263,660,820	-$6,815,287,984	$120,000
4: RA	0.242	$17,269,016,610	$190,067,811	$46,000
5: honeypot	0.02	$1,303,322,009	-$13,359,050,590	$27,800
6: all-in	0.02	$1,194,711,841	-$15,884,236,960	$412,800

Table 8.1: Overview of the adjusted results per scenario

The adjusted values as shown in Table 8.1. are used for calculating the general SCBA for each non-base case scenario. The results of this can be found in Appendix J. According to these general SCBAs, the best Return-on-Investment (ROI)/cost-effectiveness of investment is the investment in a honeypot (scenario 5) which has an ROI of 247,390, and the worst is the investment in RA (scenario 4), which has an ROI of -2,066.

9 Conclusions

This final chapter contains the conclusions of the performed study. It begins with providing answers to the individual sub-questions. The collective answers to these sub-questions are used to answer the main research question in the subsequent paragraph. Once the main research question is answered, the research difficulties and recommendations will also be discussed in separate paragraphs. This chapter, and the report as a whole, will come to an end after the final paragraph which outlines possible future research.

9.1. Sub-question answers

To keep this section as orderly as possible the sub-questions will be repeated once again and the answers to these sub-questions are given directly after each respective sub-question.

9.1.1. Sub-question 1

The first sub-question of the research went as follows:

What is the current state of substation cyber security and what are the weaknesses?

This sub-question has been answered by extensive document analysis. In this document analysis over 40 different sources have been consulted. These sources have unveiled a large number of weaknesses in the cyber security of substations. The weaknesses are listed as follows in no particular order:

1. lack of communication/coordination in the cyber security supply chain;
2. legacy infrastructure;
3. communication networks (and corresponding communication policies);
4. remote access points;
5. vulnerable soft- and firmware;
6. physical security;
7. lack of incident reporting

This list is a mere collection of the weaknesses that have been found during this study and were deemed important enough to report. The actual entirety of weaknesses of substation cyber security is near impossible to encapsulate in a study that takes less than 21 weeks and is performed by one researcher.

The lack of communication/coordination in the cyber security supply chain is the first weakness. If the digital substation receives components that are not cyber secure then these components make it easier for an attacker to infiltrate the system and carry out attacks.

Due to the fact that some older substations can still contain legacy infrastructure after being retrofitted, causes the coexistence of legacy infrastructure with more modern technology. This blend of different technologies increases the possible attack surface for malicious actors.

The centralized command and control of digital substations makes them (the digital substations) reliant on TCP/IP-based protocols (such as IEC 61850). These protocols provide efficiency in the monitoring and controlling of the energy system but also instigate possible cyber security risks. Vulnerabilities in these communication protocols can be exploited by malicious actors.

Considering the fact that it is expensive to physically send an engineer to the digital substation site to fix a certain malfunction or perform regular maintenance, the provision of remote access to engineers has been introduced to perform these tasks at a distance. However, these remote access points through which components such as the HMI can be accessed, can also be abused by malicious actors to gain entry into the digital substation and perform attacks from inside.

Components within digital substations, such as IEDs, are heavily reliant upon soft- and firmware. The soft- and firmware often contains vulnerabilities that can be exploited by a malicious actor if access to the IEDs is obtained.

Besides cyber attacks that come through connections that go to and from the substation, it is also possible that a malicious actor (or a proxy) can gain physical access to the digital substation and tamper with the system to cause damage or perform an attack step in a larger kill chain. If security measures are not sufficiently taken then this is a weakness that can be misused by actors with bad intent.

When cyber attacks have occurred, often these attacks are not extensively reported and shared. This is done for a variety of reasons, but the result stays the same, a lack of insight, knowledge, and awareness about such events. Therefore, it is harder for utility companies and other relevant organizations to prepare for cyber attacks. There are many known unknowns and unknown unknowns which are unusable at this moment.

9.1.2. Sub-question 2

The second sub-question of the research went as follows:

What are possible cyber security policies/interventions that can be implemented to decrease the likelihood of a successful attack on substations?

This sub-question has been analyzed by a combination of document analysis and literature review which contained over 16 sources. This resulted in the identification of 23 possible countermeasures that could be implemented at digital substations to decrease the likelihood of a successful attack on substations. These countermeasures are shown in Table 2.2, which can be found in Chapter 5. Once again, this list is not exhaustive and is merely the result of the performed research during this study. There are most likely cyber security countermeasures which are not mentioned in this table, but might still be relevant.

9.1.3. Sub-question 3

The third sub-question of the research goes as follows:

How do the identified policies/interventions influence the likelihood of a successful attack on a substation?

This sub-question has been answered by modeling attack graphs in Python and using shortest path algorithms to find the shortest paths to the node which would correspond to a successful cyber attack. The shortest paths, expressed in TTC (days), are converted into likelihood by equation 6.16.

The resulting change in likelihood per scenario, relative to the base case scenario, is repeated below for the sake of readability:

scenario	Δ likelihood
scenario 2: Intrusion Detection System	-21.8%
scenario 3: zoning	-39.9%
scenario 4: Remote Attestation	+1.1%
scenario 5: honeypot	-92.4%
scenario 6: all-in	-93%

Table 9.1: Change in likelihood per scenario

9.1.4. Sub-question 4

The last sub-question of the research went as follows:

What are the (financial) consequences of a successful substation attack on the grid?

To find the answer to this sub-question, the results of the first executed scenario will be used, as this scenario does not include any of the suggested countermeasures. According to this scenario and its underlying sub-scenarios (levels of blackout following the opening of certain circuit breakers), the consequences affect the amount of load in the grid and cause economic damage to the corresponding region of that substation. The consequence for the grid load, for simplicity's sake, has been categorized into three possible classes, and using the DIgSILENT and Mininet based physical grid figures provided by [18], the following has been estimated:

The consequences can range between a small blackout (maximum of 15% grid failure and duration of up to 2 hours [131]), a medium blackout (maximum of 75% grid failure and duration of up to 19 hours [130]) and a large blackout (minimum of 75% grid failure and duration of approximately 72 hours [132]). The total risk of these subscenarios varies between \$32,690,947 and \$46,201,833,830, caused by a respective loss of load between 296.89 MWh (8.14% grid failure) and 41,9538 MWh (91.63% grid failure). According to the general SCBAs (located in Appendix J), the best ROI/cost-effectiveness of investment, is the investment in a honeypot (scenario 5) which has an ROI of 247,390, and the worst is the investment in RA (scenario 4), which has an ROI of -2,066.

9.2. Main research questions answer

The main research question of this study was formulated as follows:

"To what extent can cyber security measures decrease the risk of cyber-attacks on CPPS substations?"

Answering this research question required the completion of all the underlying sub-question. Now that these sub-questions have been answered, the main research question can be answered.

The quantified model for cyber security in digital substations has proven to provide some relevant cyber security metrics. Such metrics as the TTC_{avg} of the dominant attack path, the average TTC_{avg} of all unique paths, Δ *likelihood* and Δ *risk*. While the generated values that result from the model are most likely not the exact values that would be observed in practice, these values are still usable as indicators and guidelines for creating and evaluating cyber security countermeasures. The model should not be seen as the ultimate truth, but as a supporting tool to aid in running thought experiments and quantify ideas and proposals for investments in the cyber security of digital substations, as this is badly needed. The addition of financial figures to the model further enables and supports the investment decision-making process.

The suggested measures in the current iteration of the quantified model have limited application. Due to the conditionality of the attack scenario setting and the finite combinations of countermeasures, the results of the scenarios are only partly applicable in situations that are similar to the set constraints and limitations of the constructed scenarios and setting.

Nevertheless, the quantified model has shown to produce results which indicate a reduction (between

21.8% and 93%) in the total risk of certain scenarios of attacks against digital substations by malicious actors.

9.3. Research difficulties

This section of the chapter is dedicated to the description of difficulties that have been while conducting this study.

9.3.1. Lack of digital substation incident reports

As already touched upon in section 9.1.1., there is a limited amount of information available about successful (or even thwarted) cyber attacks on digital substations. The reluctance of organizations to disclose this valuable information has made it rather difficult to find many different attack steps. This information was needed to formulate various attack steps that a malicious actor could take to eventually reach its intended target, which in this case was the opening of the circuit breakers.

9.3.2. Unavailability of SecuriCAD software

At the beginning of the study, it became apparent that the intended quantitative threat modelling tool, SecuriCAD, had been bought by Google's parent company Alphabet Inc. Once acquired by Alphabet, the software was only sold as an enterprise business to business product, and is no longer free to use for researchers or academics. This complicated this study a lot since this software was planned to be an integral part of the modelling research. In order to get similar results as would have been produced via SecuriCAD, two separate Python models had to be built. However, the author had very limited Python experience. This made the modelling process substantially more difficult than it was first envisioned to be. Fortunately, the author of the work upon which this study continues, Ioannis Semertzis, had a Python code available which eventually served as the foundation of the first part of the Python model. Nevertheless, this change in the modelling approach has caused some delays in the progression of the research. However, besides the negative consequences, there was a positive externality. Namely, the added flexibility and transparency that a Python model provides, compared to the black-box approach of the SecuriCAD software.

9.3.3. Algorithm limitations

During the finalisation of the second Python model, a constraint of the previously used Dijkstra's Single Source Shortest Path Algorithm was stumbled upon. This constraint was that it is very difficult or impossible to force the algorithm to recalculate the shortest path, while within the Monte Carlo simulation if a certain condition of the shortest path was met. This functionality was needed to apply the penalty of a modelled countermeasure and extract the shortest unique paths between the source node and the target node. This required that the entire core of the second Python model had to change from a Dijkstra's Algorithm to a Depth-First-Search algorithm, which provided the needed flexibility in penalty application modelling. Due to the late realization, this constraint had to be fixed quickly and caused some minor delays in the progress of the research.

9.3.4. Cyber security SCBA documentation

In order to analyze the business case for the use of the proposed quantified model to aid in investment decision-making, a limited societal cost-benefit analysis of the suggested countermeasures has been performed. A format for this SCBA was sought in order to lend credence to the execution of the SCBA. However, there were no examples or formats to be found for critical infrastructure cyber security SCBAs. This was peculiar, as such cyber security investments into critical infrastructure should be more common in 2023, especially given the current and outlined cyber security trends. However, not being able to find a proper format, an older SCBA for critical infrastructure format has been adjusted accordingly, and used to serve as a guideline.

9.3.5. 'Manual' sensitivity analysis

Due to the fact that both models have been built in Python and don't have built-in sensitivity analysis options, the sensitivity analyses of these models had to be performed 'by hand'. This means that for all the parameters that have been changed within a certain range, the adjustment had to be inserted and run one by one. As a result, the sensitivity analysis has taken a lot of time to perform, in order to execute them for every scenario and parameter adjustment. In order to keep the time spent on the sensitivity analysis at a manageable level, only one parameter had been changed at a time, while leaving the remaining parameters at their 'normal' level. This resulted in a sensitivity analysis that consists of single parameter adjustments, and not combinations of different parameters. Running combinations of adjusted parameters might have resulted in interesting behaviour which is now unknown.

9.4. Limitations

In this part of the conclusion the limitations of the performed study are outlined.

9.4.1. Attack scenarios

Within the constructed scenarios only a single attack setup is considered. Namely, the combination of entry at the set Gateway entry point, and a single goal of the malicious actor, which was the opening of the circuit breaker. In reality, there are other entry points from which the malicious actor could start its attack path. However, given the limited time to complete the study, only this single entry point was considered. The same goes for the goal of the malicious actor. There are other goals that a malicious actor could have to (eventually) affect the power system, such as a DoS attack against the Gateway/RTU in order to prevent the SCADA system to communicate with the targeted substation.

Because this study only focused on the assumed attack scenario setting, the results of this research can only be applied to cases where one is only interested in that specific attack setting.

9.4.2. Attack path possibilities

Given the limited knowledge available of possible attack steps within the digital substation, the constructed attack paths consist of a small and finite subset of the potential attack paths that could be taken by a malicious actor in practice. In order to increase the usability of the quantified model(s) a pen-testing team, specialized in digital substations could provide a more extensive list of possible attack steps.

9.4.3. Separate sensitivity analyses

As mentioned in paragraph 9.3.5, the long duration of the sensitivity analysis has led to the choice to limit some parts of the sensitivity analysis. One important limitation of the sensitivity analysis is that the evaluated output values of the first model, have not been used as input for the second model. Since these models are coupled, the potential behaviour of the model-pair has not been examined. The coupling of the individual sensitivity analyses might have resulted in interesting behaviour which remains unobserved at this point.

9.4.4. TTC calculation constants

In equations 6.2 and 6.7, two constants have been assumed, that were in line with previous authors. The same goes for the time estimation of subprocess 1, where t_1 = *1 day*. However, in hindsight, these constants should have been converted to, for example, normal distributions, to emulate the uncertainty in these assumptions and better reflect the broader range of possible values that could be encountered in reality. This has not been noted before the study was completed. Therefore, it was not possible to implement this improvement post hoc, to see the results. It might have been possible that the change from constants to distributions could have led to significantly different outcomes, but this could not be studied in time. Therefore, this is a limitation of this work.

9.5. Future work

In the final part of this study, possible directions for future work are given. These directions can be pursued by another researcher who wants to continue the research in this field.

9.5.1. Additional attack and defense scenarios

In this work, as previously described in this chapter and the previous chapter, either only one countermeasure is simulated at a time, or all the countermeasures at once. This leaves a lot of other combinations in the solution space which are not researched. This is also true for the entry point and goal of the malicious actor. Performing further research into this unexplored solution space might yield interesting results in terms of defensive efforts and change in likelihood and risk.

This future work suggestion is part of a larger recommendation that is done after performing this study, namely, the creation of a more professionalized combined quantitative model which consists of more extended attack paths based on input from cyber experts and field operators. This model should be user-friendly and have a built-in sensitivity analysis to ensure practicability and ease of use. This way the quantitative model would be more accurate and of greater added value to decision-makers.

9.5.2. Experimentation with different cyber security maturity

In this work an addition to the likelihood calculation has been proposed. This addition adjusts the Mean-Time-To-Detect, based on the maturity factor of the organization's cyber security policies and capabilities. In this study the maturity factor m was assumed to be equal to 1, to keep the complexity of the study to a manageable degree. However, the calculation and effects of m on the change in likelihood and risk might be interesting to research further.

9.5.3. Increased cyber security (supply chain) cooperation

Another common theme that was encountered throughout this research was the lack of, and potential gains in, increased cyber security collaboration. Not only do some of the biggest vulnerabilities lie in the lack of cooperation in the cyber security supply chain, but also possible solutions to these (and larger) ”challenges” can be found in forms of increased cooperation, such as self-governance of critical infrastructure. Tighter collaboration makes it easier to appoint the least-cost avoider(s), and might even decrease the necessary cyber security investment costs or the potentially incurred monetary damage. This topic has only been slightly addressed in this study by means of a very limited use of the IAD framework to apply theories such as the least-cost avoider. However, the superficiality of this study with regard to that topic can be elaborated upon. Further research could be done to see what the exact benefits of increased cyber security (supply chain) cooperation are, and how this could influence the (financial) consequences of a successful cyber attack on a digital substation.

References

[1] Council of European Energy Regulators. “Distribution Systems Working Group (DS WG) New Services and DSO Involvement A CEER Conclusions Paper”. In: (2019).

[2] G Murray, M N Johnstone, and C Valli. “The convergence of IT and OT in critical infrastructure”. In: (2017), pp. 149–155. DOI: 10.4225/75/5a84f7b595b4e. URL: https://ro.ecu.edu.au/ism.

[3] R Khan et al. “Real-time Control and Monitoring in Smart Grid”. In: (2016), pp. 53–63. DOI: 10.14236/ewic/ICS2016.7.

[4] Gartner Inc. *Predicts 2020: Security and Risk Management Programs*. 2019. URL: https://www.gartner.com/en/documents/3976275.

[5] Åsa Fritzon et al. “Protecting Europe’s Critical Infrastructures: Problems and Prospects”. In: *Journal of Contingencies and Crisis Management* 15 (1 Mar. 2007), pp. 30–41. ISSN: 1468-5973. DOI: 10.1111/J.1468-5973.2007.00502.X. URL: https://onlinelibrary.wiley.com/doi/full/10.1111/j.1468-5973.2007.00502.x%20https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1468-5973.2007.00502.x%20https://onlinelibrary.wiley.com/doi/10.1111/j.1468-5973.2007.00502.x.

[6] Alessandro Liberati et al. “The PRISMA statement for reporting systematic reviews and meta-analyses of studies that evaluate health care interventions: explanation and elaboration”. In: *PLoS medicine* 6 (7 July 2009). ISSN: 1549-1676. DOI: 10.1371/JOURNAL.PMED.1000100. URL: https://pubmed.ncbi.nlm.nih.gov/19621070/.

[7] Rajaa Vikhram Yohanandhan et al. “Cyber-Physical Power System (CPPS): A Review on Modeling, Simulation, and Analysis with Cyber Security Applications”. In: *IEEE Access* 8 (2020), pp. 151019–151064. ISSN: 21693536. DOI: 10.1109/ACCESS.2020.3016826.

[8] D Kundur et al. “Towards modelling the impact of cyber attacks on a smart grid”. In: *Int. J. Security and Networks* (2011).

[9] Qin Wang, Guangyuan Zhang, and Fushuan Wen. “A survey on policies, modelling and security of cyber-physical systems in smart grids”. In: *Energy Conversion and Economics* 2 (4 Dec. 2021), pp. 197–211. ISSN: 2634-1581. DOI: 10.1049/ENC2.12051. URL: https://onlinelibrary.wiley.com/doi/full/10.1049/enc2.12051%20https://onlinelibrary.wiley.com/doi/abs/10.1049/enc2.12051%20https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/enc2.12051.

[10] Vetrivel Subramaniam Rajkumar et al. “Cyber Attacks on Power System Automation and Protection and Impact Analysis”. In: IEEE, Oct. 2020, pp. 247–254. ISBN: 978-1-7281-7100-5. DOI: 10.1109/ISGT-Europe47291.2020.9248840.

[11] Ricardo G Sanfelice. “Analysis and Design of Cyber-Physical Systems: A Hybrid Control Systems Approach”. In: (2016).

[12] Jia Guo et al. “Modeling and vulnerability analysis of cyber-physical power systems considering network topology and power flow properties”. In: *Energies* 10 (1 2017). ISSN: 19961073. DOI: 10.3390/EN10010087. URL: https://www.researchgate.net/publication/312343796_Modeling_and_Vulnerability_Analysis_of_Cyber-Physical_Power_Systems_Considering_Network_Topology_and_Power_Flow_Properties.

[13] Shujun Xin et al. “Cyber-physical modeling and cyber-contingency assessment of hierarchical control systems”. In: *IEEE Transactions on Smart Grid* 6 (5 Sept. 2015), pp. 2375–2385. ISSN: 19493061. DOI: 10.1109/TSG.2014.2387381.

[14] Mahdi Bahrami, Mahmud Fotuhi-Firuzabad, and Hossein Farzin. “Reliability Evaluation of Power Grids Considering Integrity Attacks against Substation Protective IEDs”. In: *IEEE Transactions on Industrial Informatics* 16 (2 Feb. 2020), pp. 1035–1044. ISSN: 19410050. DOI: `10.1109/TII.2019.2926557`.

[15] Ying Chen, Junho Hong, and Chen Ching Liu. “Modeling of Intrusion and Defense for Assessment of Cyber Security at Power Substations”. In: *IEEE Transactions on Smart Grid* 9 (4 July 2018), pp. 2541–2552. ISSN: 19493053. DOI: `10.1109/TSG.2016.2614603`.

[16] Xiaofei Zhang, Luolin Zheng, and Rui Zhou. “The Petri Net Based Probability Transfer Analysis of Cyber-Physical Power System Under False Data Injection Attacks”. In: 2020, pp. 859–870. DOI: `10.1007/978-981-13-9779-0_70`.

[17] Gabriel A. Weaver et al. “Cyber-Physical models for power grid security analysis: 8-substation case”. In: *2016 IEEE International Conference on Smart Grid Communications (SmartGridComm)* (Dec. 2016), pp. 140–146. DOI: `10.1109/SMARTGRIDCOMM.2016.7778752`.

[18] Ioannis Semertzis. “Risk Assessment of Cyber Attacks on Cyber-Physical Power Systems: A Quantitative Analysis using Attack Graphs”. In: (2021). URL: `https://repository.tudelft.nl/islandora/object/uuid%5C%3A77b5490c-92b6-47a6-90e2-c0274fa6f771`.

[19] Koji Yamashita et al. “Establishment of Cyber-Physical Correlation and Verification Based on Attack Scenarios in Power Substations”. MICHIGAN TECHNOLOGICAL UNIVERSITY, 2020.

[20] Miles A. McQueen et al. “Time-to-compromise model for cyber risk reduction estimation”. In: *Advances in Information Security* 23 (2006), pp. 49–64. ISSN: 15682633. DOI: `10.1007/978-0-387-36584-8_5/COVER`. URL: `https://link.springer.com/chapter/10.1007/978-0-387-36584-8_5`.

[21] Stewart Robinson. *Simulation: the practice of model development and use*. John Wiley and Sons, Ltd, 2014.

[22] Joshua M Epstein. “Why Model?” In: *Journal of Artificial Societies and Social Simulation* 11 (4 2008), p. 12.

[23] Maurice Landry, Jean Louis Malouin, and Muhittin Oral. “Model validation in operations research”. In: *European Journal of Operational Research* 14 (3 Nov. 1983), pp. 207–220. ISSN: 0377-2217. DOI: `10.1016/0377-2217(83)90257-6`.

[24] Stewart Robinson and Roger J. Brooks. “Independent Verification and Validation of an Industrial Simulation Model”. In: *http://dx.doi.org/10.1177/0037549709341582* 86 (7 July 2009), pp. 405–416. ISSN: 00375497. DOI: `10.1177/0037549709341582`. URL: `https://journals.sagepub.com/doi/abs/10.1177/0037549709341582`.

[25] Bart W. Tuinema et al. “Cyber-Physical System Modeling for Assessment and Enhancement of Power Grid Cyber Security, Resilience, and Reliability”. In: *Probabilistic Reliability Analysis of Power Systems* (2020), pp. 237–270. DOI: `10.1007/978-3-030-43498-4_8`. URL: `https://link.springer.com/chapter/10.1007/978-3-030-43498-4_8`.

[26] Juliet Corbin and Anselm Strauss. “Basics of Qualitative Research (3rd ed.): Techniques and Procedures for Developing Grounded Theory”. In: *Basics of Qualitative Research (3rd ed.): Techniques and Procedures for Developing Grounded Theory* (Apr. 2012). DOI: `10.4135/9781452230153`.

[27] Elias G. Carayannis et al. “Ambidextrous Cybersecurity: The Seven Pillars (7Ps) of Cyber Resilience”. In: *IEEE Transactions on Engineering Management* 68 (1 Feb. 2021), pp. 223–234. ISSN: 15580040. DOI: `10.1109/TEM.2019.2909909`.

[28] Ioannis Semertzis et al. “Quantitative Risk Assessment of Cyber Attacks on Cyber-Physical Systems using Attack Graphs”. In: *2022 10th Workshop on Modelling and Simulation of Cyber-Physical Energy Systems, MSCPES 2022* (2022), pp. 1–6. DOI: `10.1109/MSCPES55116.2022.9770140`. URL: `https://research.tudelft.nl/en/publications/quantitative-risk-assessment-of-cyber-attacks-on-cyber-physical-s`.

[29] Yichi Zhang et al. “Power System Reliability Evaluation With SCADA Cybersecurity Considerations”. In: *IEEE Transactions on Smart Grid* 6 (4 July 2015), pp. 1707–1721. ISSN: 19493053. DOI: `10.1109/TSG.2015.2396994`.

[30] C.R. Bayliss and B.J. Hardy. “Substation Layouts”. In: *Transmission and Distribution Electrical Engineering* (2012), pp. 93–119. DOI: 10.1016/B978-0-08-096912-1.00003-4.

[31] D Ishchenko and D Nuqui. “Secure communication of intelligent electronic devices in digital substations”. In: *IEEE/PES Transmission and Distribution Conference and Exposition (TD)* (2018). URL: https://ieeexplore.ieee.org/abstract/document/8440438/.

[32] Rinaldo Pagani. “ABB Digital Substation The state of art substation Introduction-ABB Ability™”. In: (2017).

[33] ABB Kunsman Steven. “IEC 61850 in Digital Substation and Cyber security”. In: (May 2018).

[34] Athar Khodabakhsh et al. “Cyber-Security Gaps in a Digital Substation: From Sensors to SCADA”. In: *2020 9th Mediterranean Conference on Embedded Computing, MECO 2020* (June 2020). DOI: 10.1109/MECO49872.2020.9134350.

[35] Alexandru Stefanov et al. “SCADA modeling for performance and vulnerability assessment of integrated cyber-physical systems”. In: *International Transactions on Electrical Energy Systems* 25 (3 Mar. 2015), pp. 498–519. ISSN: 20507038. DOI: 10.1002/ETEP.1862.

[36] Jingyu Wang and Dongyuan Shi. “Cyber-Attacks Related to Intelligent Electronic Devices and Their Countermeasures: A Review”. In: Institute of Electrical and Electronics Engineers Inc., Nov. 2018. ISBN: 9781538629109. DOI: 10.1109/UPEC.2018.8542059.

[37] Junho Hong et al. “Cyber-physical security testbed for substations in a power grid”. In: *Power Systems* 79 (2015), pp. 261–301. ISSN: 18604676. DOI: 10.1007/978-3-662-45928-7_10.

[38] ENISA. “Interoperable EU Risk Management Toolbox”. In: (2023). DOI: 10.2824/713364. URL: www.enisa.europa.eu..

[39] TenneT Holding B.V. *Integrated Annual Report 2021*. 2021.

[40] Tania Wallis, Greig Paul, and James Irvine. “Organisational Contexts of Energy Cybersecurity”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 13106 LNCS (2022), pp. 384–402. ISSN: 16113349. DOI: 10.1007/978-3-030-95484-0_22/TABLES/2. URL: https://link.springer.com/chapter/10.1007/978-3-030-95484-0_22.

[41] U.S. Homeland Security and U.S. Department of Energy. “ELECTRICITY SUBSECTOR CYBERSECURITY CAPABILITY MATURITY MODEL (ES-C2M2)”. In: (2014).

[42] Chris Roberts. “Biometric attack vectors and defences”. In: *Computers and Security* 26 (1 Feb. 2007), pp. 14–25. ISSN: 01674048. DOI: 10.1016/J.COSE.2006.12.008.

[43] David Formby, Srikar Durbha, and Raheem Beyah. *Out of Control: Ransomware for Industrial Control Systems*. Georgia Institute of Technology, 2017.

[44] Pubudu Eroshan Weerathunga and Anca Cioraca. “The importance of testing Smart Grid IEDs against security vulnerabilities”. In: *69th Annual Conference for Protective Relay Engineers, CPRE 2016* (Apr. 2017). DOI: 10.1109/CPRE.2016.7914920.

[45] Luis A Garcia et al. “Hey, My Malware Knows Physics! Attacking PLCs with Physical Model Aware Rootkit”. In: (2017). DOI: 10.14722/ndss.2017.23313. URL: http://dx.doi.org/10.14722/ndss.2017.23313.

[46] Huan Yang, Liang Cheng, and Mooi Choo Chuah. “Detecting Payload Attacks on Programmable Logic Controllers (PLCs)”. In: (2018).

[47] Naman Govil, Anand Agrawal, and Nils Ole Tippenhauer. “On ladder logic bombs in industrial control systems”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 10683 LNCS (2018), pp. 110–126. ISSN: 16113349. DOI: 10.1007/978-3-319-72817-9_8/FIGURES/9. URL: https://link.springer.com/chapter/10.1007/978-3-319-72817-9_8.

[48] Ralf Spenneberg, Maik Brüggemann, and Hendrik Schwartke. “PLC-Blaster: A Worm Living Solely in the PLC”. In: (2016).

[49] Center for Internet Security. *Election Security Spotlight – Hardware, Software, and Firmware*. Accessed: 14-03-2023. URL: https://www.cisecurity.org/insights/spotlight/cybersecurity-spotlight-hardware-software-and-firmware.

[50] Onion.io. *Pin Multiplexing | Onion Omega2 Documentation*. Accessed on: 14-03-2023. URL: https://docs.onion.io/omega2-docs/pin-multiplexing.html.

[51] Ali Abbasi and Majid Hashemi. "Ghost in the PLC Designing an Undetectable Programmable Logic Controller Rootkit via Pin Control Attack". In: (2016).

[52] Ang Cui, Michael Costello, and Salvatore J Stolfo. "When Firmware Modifications Attack: A Case Study of Embedded Exploitation". In: (2013). DOI: 10.7916/D8P55NKB. URL: https://academiccommons.columbia.edu/doi/10.7916/D8P55NKB%20https://doi.org/10.7916/D8P55NKB.

[53] C Johnson and M Evangelopoulou. "Defending against firmware cyber attacks on safety-critical systems". In: *Journal of System Safety* (2018). URL: https://jsystemsafety.com/index.php/jss/article/view/83.

[54] Jacob Wurm et al. "Introduction to Cyber-Physical System Security: A Cross-Layer Perspective". In: *IEEE Transactions on Multi-Scale Computing Systems* 3 (3 July 2017), pp. 215–227. ISSN: 23327766. DOI: 10.1109/TMSCS.2016.2569446.

[55] Orlando Arias et al. "Privacy and Security in Internet of Things and Wearable Devices". In: *IEEE Transactions on Multi-Scale Computing Systems* 1 (2 June 2015), pp. 99–109. ISSN: 23327766. DOI: 10.1109/TMSCS.2015.2498605.

[56] A. I. Tarmizi, Mihai D. Rotaru, and Jan K. Sykulski. "Electromagnetic compatibility studies within smart grid automated substations". In: *Proceedings of the Universities Power Engineering Conference* (Oct. 2014). DOI: 10.1109/UPEC.2014.6934675.

[57] William A. Radasky and Richard Hoad. "An overview of the impacts of three high power electromagnetic (HPEM) threats on Smart Grids". In: *IEEE International Symposium on Electromagnetic Compatibility* (2012). ISSN: 10774076. DOI: 10.1109/EMCEUROPE.2012.6396770.

[58] Anupam Chattopadhyay et al. "Toward Threat of Implementation Attacks on Substation Security: Case Study on Fault Detection and Isolation". In: *IEEE Transactions on Industrial Informatics* 14 (6 June 2018), pp. 2442–2451. ISSN: 15513203. DOI: 10.1109/TII.2017.2770096.

[59] Atef Abdrabou and A. M. Gaouda. "Uninterrupted Wireless Data Transfer for Smart Grids in the Presence of High Power Transients". In: *IEEE Systems Journal* 9 (2 June 2015), pp. 567–577. ISSN: 19379234. DOI: 10.1109/JSYST.2013.2265179.

[60] ICF International. *Electric Grid Security and Resilience: Establishing a Baseline for Adversarial Threats*. ICF International, 2016.

[61] Michael Assante and Scott Swartz D. *Industrial Control Systems (ICS) Cybersecurity Response to Physical Breaches of Unmanned Critical Infrastructure Sites Whitepaper*. Jan. 2014. URL: https://www.sans.org/white-papers/industrial-control-systems-ics-cybersecurity-response-to-physical-breaches-of-unmanned-critical-infrastructure-sites-whitepaper/.

[62] David Formby et al. "A physical overlay framework for insider threat mitigation of power system devices". In: *2014 IEEE International Conference on Smart Grid Communications, SmartGridComm 2014* (Jan. 2015), pp. 970–975. DOI: 10.1109/SMARTGRIDCOMM.2014.7007774.

[63] Nils Ole Tippenhauer et al. "On the requirements for successful GPS spoofing attacks". In: *Proceedings of the ACM Conference on Computer and Communications Security* (2011), pp. 75–85. ISSN: 15437221. DOI: 10.1145/2046707.2046719.

[64] Zhenghao Zhang et al. "Time synchronization attack in smart grid: Impact and analysis". In: *IEEE Transactions on Smart Grid* 4 (1 2013), pp. 87–98. ISSN: 19493053. DOI: 10.1109/TSG.2012.2227342.

[65] Sergei Skorobogatov. *Design and Security of Cryptographic Algorithms and Devices (ECRYPT II) Fault attacks on secure chips: from glitch to flash*. May 2011. URL: http://www.cl.cam.ac.uk/~sps32.

[66] EEEGuide. *Differential Pilot Wire Protection | Pilot Wire Differential Protection Scheme*. Accessed on: 18-03-2023. URL: https://www.eeeguide.com/differential-pilot-wire-protection/.

[67] IEEE Power et al. “IEEE standard for electric power systems communications–distributed network protocol (DNP3)”. In: *IEEE standards* (Oct. 2012), p. 779.

[68] Dong Hoon Shin et al. “Low-complexity secure protocols to defend cyber-physical systems against network isolation attacks”. In: *2013 IEEE Conference on Communications and Network Security, CNS 2013* (2013), pp. 91–99. DOI: 10.1109/CNS.2013.6682696.

[69] Yikai Xu et al. “Review on cyber vulnerabilities of communication protocols in industrial control systems”. In: *2017 IEEE Conference on Energy Internet and Energy System Integration, EI2 2017 - Proceedings* 2018-January (June 2017), pp. 1–6. DOI: 10.1109/EI2.2017.8245509.

[70] Bradley Reaves and Thomas Morris. “Discovery, infiltration, and denial of service in a process control system wireless network”. In: *2009 eCrime Researchers Summit, eCRIME ’09* (2009). DOI: 10.1109/ECRIME.2009.5342612.

[71] Saleh Soltan, Dorian Mazauric, and Gil Zussman. “Cascading failures in power grids - Analysis and algorithms”. In: *e-Energy 2014 - Proceedings of the 5th ACM International Conference on Future Energy Systems* 14 (2014), pp. 195–206. DOI: 10.1145/2602044.2602066. URL: https://dl.acm.org/doi/10.1145/2602044.2602066.

[72] Maria Vrakopoulou et al. “Cyber-attacks in the automatic generation control”. In: *Power Systems* 79 (2015), pp. 303–328. ISSN: 18604676. DOI: 10.1007/978-3-662-45928-7_11.

[73] Qingyu Yang et al. “Toward Data Integrity Attacks Against Optimal Power Flow in Smart Grid”. In: *IEEE Internet of Things Journal* 4 (5 Oct. 2017), pp. 1726–1738. ISSN: 23274662. DOI: 10.1109/JIOT.2017.2709252.

[74] MTA Rashid and S Yussof. “A review of security attacks on IEC61850 substation automation system network”. In: *Proceedings of the 6th International Conference on Information Technology and Multimedia (ICIM)* (2014). URL: https://ieeexplore.ieee.org/abstract/document/7066594/?casa_token=VAbS6IofYToAAAAA:K0xgSMRd_au6Yg_FRkCFQ-xaTkxqCqNqXRpEcRHOTn3THMFNJpLcqkFrhse4o-jMpSYCojtu4g.

[75] Yousu Chen. “Weighted-Least-Square(WLS) State Estimation”. In: *PNNL* (2015).

[76] Rajesh Kalluri et al. “Simulation and impact analysis of denial-of-service attacks on power SCADA”. In: *2016 National Power Systems Conference, NPSC 2016* (Feb. 2017). DOI: 10.1109/NPSC.2016.7858908.

[77] Stéphane Mocanu and Jean Marc Thiriet. “Real-Time Performance and Security of IEC 61850 Process Bus Communications”. In: *Journal of Cyber Security and Mobility* 10 (2 2021), pp. 305–346. ISSN: 22454578. DOI: 10.13052/JCSM2245-1439.1021. URL: https://www.researchgate.net/publication/350749212_Real-Time_Performance_and_Security_of_IEC_61850_Process_Bus_Communications.

[78] Carol Taylor, Axel Krings, and Jim Alves-Foss. “Risk Analysis and Probabilistic Survivability Assessment (RAPSA): An Assessment Approach for Power Substation Hardening 1”. In: (2002).

[79] NCCIC. “INCIDENT RESPONSE/VULNERABILITY COORDINATION IN 2014”. In: *ICS-CERT Monitor* (2014).

[80] Verizon. “DBIR Data Breach Investigations Report”. In: (2022).

[81] MZ Gunduz, R Das - Computer networks, and undefined 2020. “Cyber-security on smart grid: Threats and potential solutions”. In: *Elsevier* (2020). mag blijven, zie paragaaf 3.3. URL: https://www.sciencedirect.com/science/article/pii/S1389128619311235?casa_token=3J7bHswd0-UAAAAA:BMHl6QZMPx31Vnd3r-5ZHRe2rdfl9sjYvcTJdWmpMad5RkzAHGsbWi_hTjOg3D1AN-3VMvJ8ZMA.

[82] Samonas S and Coss D. “The CIA strikes back: Redefining confidentiality, integrity and availability in security.” In: *Journal of Information System Security* (2014). URL: https://www.proso.com/dl/Samonas.pdf.

[83] HC Tan et al. “Tabulating cybersecurity solutions for substations: Towards pragmatic design and planning”. In: *IEEE Innovative Smart Grid Technologies - Asia (ISGT Asia)* (2019). URL: https://ieeexplore.ieee.org/abstract/document/8881706/?casa_token=jxH7g1YiVDUAAAAA:iZNTKfH7yN-iJXdOTQqh2ieEwu8laCV0ilrlVai1llhpQ1ZASRM2RFP7INVNQDPk5Wg5tiOXmw.

[84] Yi Deng and Sandeep Shukla. “Vulnerabilities and Countermeasures – A Survey on the Cyber Security Issues in the Transmission Subsystem of a Smart Grid”. In: *Journal of Cyber Security and Mobility* 1 (2-3 Apr. 2012), pp. 250–276-250–276. ISSN: 2245-4578. DOI: 10.13052/JCSM2245-1439.1236. URL: https://journals.riverpublishers.com/index.php/JCSANDM/article/view/6115%20https://journals.riverpublishers.com/index.php/JCSANDM.

[85] Chee Wooi Ten, Chen Ching Liu, and Manimaran Govindarasu. “Vulnerability assessment of cybersecurity for SCADA systems using attack trees”. In: *2007 IEEE Power Engineering Society General Meeting, PES* (2007). DOI: 10.1109/PES.2007.385876.

[86] M N Dazahra et al. “A Defense-in-depth Cybersecurity for Smart Substations”. In: *International Journal of Electrical and Computer Engineering (IJECE)* 8 (6 2018), pp. 4423–4431. ISSN: 2088-8708. DOI: 10.11591/ijece.v8i6.pp4423-4431.

[87] Shahrin Sadik et al. “Toward a Sustainable Cybersecurity Ecosystem”. In: *Computers 2020, Vol. 9, Page 74* 9 (3 Sept. 2020), p. 74. ISSN: 2073-431X. DOI: 10.3390/COMPUTERS9030074. URL: https://www.mdpi.com/2073-431X/9/3/74/htm%20https://www.mdpi.com/2073-431X/9/3/74.

[88] Saqib Ali et al. “Cyber security for cyber physical systems”. In: *Computational Intelligence 768* (2018). wellicht uitsluiten. URL: https://link.springer.com/content/pdf/10.1007/978-3-319-75880-0.pdf.

[89] A Georgiadou et al. “Assessing mitre attck risk using a cyber-security culture framework”. In: *Sensors* (2021). DOI: 10.3390/s21093267. URL: https://www.mdpi.com/1101874.

[90] Leandros Maglaras et al. “Threats, countermeasures and attribution of cyber attacks on critical infrastructures”. In: *EAI Endorsed Transactions on Security and Safety* (2018). DOI: 10.4108/15-10-2018.155856. URL: https://eudl.eu/doi/10.4108/eai.15-10-2018.155856.

[91] Diane Gan et al. “Cyber security countermeasures to combat cyber terrorism”. In: *Elsevier* (2013). DOI: 10.1016/B978-0-12-407191-9.00020-X. URL: https://www.sciencedirect.com/science/article/pii/B978012407191900020X.

[92] Tala Talaei Khoei, Hadjar Ould Slimane, and Naima Kaabouch. “A Comprehensive Survey on the Cyber-Security of Smart Grids: Cyber-Attacks, Detection, Countermeasure Techniques, and Future Directions”. In: *Astrophysics Data System* (2022). URL: https://ui.adsabs.harvard.edu/abs/2022arXiv220707738T/abstract.

[93] David Whyte. “Using a systems-theoretic approach to analyze cyber attacks on cyber-physical systems”. In: *MIT Library* (2017). URL: https://dspace.mit.edu/handle/1721.1/110143.

[94] M Boeding et al. “Survey of Cybersecurity Governance, Threats, and Countermeasures for the Power Grid”. In: *Energies* (2020). DOI: 10.3390/computers9030074. URL: https://www.mdpi.com/1996-1073/15/22/8692.

[95] Y Yang et al. “Impact of cyber-security issues on smart grid”. In: *IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT Europe)* (2011). URL: https://ieeexplore.ieee.org/abstract/document/6162722/?casa_token=oqYfd5tNfC4AAAAA:wW5ydkZqiHcHenMlRHn4KVORLZq2Ab5pss1F0yiL8JLEtbtEl8jW5K604FSGZOA_B9i683BRCA.

[96] Young Ho Chae, Chanyoung Lee, and Hyun Seong. “Cyber-attack path suggestion system by using markov model and Dijkstra’s algorithm”. In: *Transactions of the Korean Nuclear Society Virtual spring Meeting* (2021).

[97] Robert Tarjan. “DEPTH-FIRST SEARCH AND LINEAR GRAPH ALGORITHMS”. In: *SIAM JOURNAL OF COMPUTING* 1 (2 1972). URL: https://epubs.siam.org/terms-privacy.

[98] S Vidalis et al. “Using Vulnerability Trees for Decision Making in Threat Assessment”. In: *School of Computing Technical Report CS-03-2* (2003). URL: www.comp.glam.ac.uk.

[99] JOHN A. MAJOR. “Advanced Techniques for Modeling Terrorism Risk”. In: *The Journal of Risk Finance* 4 (1 Apr. 2002), pp. 15–24. ISSN: 1526-5943. DOI: 10.1108/EB022950. URL: https://www.researchgate.net/publication/239597825_Advanced_Techniques_for_Modeling_Terrorism_Risk.

[100] David John Leversage and Eric James Byres. "Estimating a system's mean time-to-compromise". In: *IEEE Security and Privacy* 6 (1 Jan. 2008), pp. 52–60. ISSN: 15407993. DOI: 10.1109/MSP.2008.9.

[101] Heena Goyal. "Technology Crime and Organized Syndicates in Cybercrime: Critical Analysis". In: *International Journal of Law Management & Humanities* 5 (5 2022). URL: https://heinonline.org/HOL/Page?handle=hein.journals/ijlmhs19%5C&id=266%5C&div=%5C&collection=.

[102] Sergio Iglesias Pérez, Santiago Moral-Rubio, and Regino Criado. "A new approach to combine multiplex networks and time series attributes: Building intrusion detection systems (IDS) in cybersecurity". In: *Chaos, Solitons and Fractals* 150 (Sept. 2021), p. 111143. ISSN: 0960-0779. DOI: 10.1016/J.CHAOS.2021.111143.

[103] Sean Kinser et al. "SSC20-P1-03 Scoring Trust Across Hybrid-Space: A Quantitative Framework Designed to Calculate Cybersecurity Ratings, Measures, and Metrics to Inform a Trust Score". In: *34th Annual Small Satellite Conference* (2020).

[104] Federica Bellizio, Jochen L. Cremer, and Goran Strbac. "Machine-learned security assessment for changing system topologies". In: *International Journal of Electrical Power and Energy Systems* 134 (Jan. 2022). ISSN: 01420615. DOI: 10.1016/J.IJEPES.2021.107380.

[105] Emanuele Ciapessoni et al. "Probabilistic Risk-Based Security Assessment of Power Systems Considering Incumbent Threats and Uncertainties". In: *IEEE Transactions on Smart Grid* 7 (6 Nov. 2016), pp. 2890–2903. ISSN: 19493053. DOI: 10.1109/TSG.2016.2519239.

[106] Longe Olumide Babatope, Lawal Babatunde, and Ibitola Ayobami. "Strategic Sensor Placement for Intrusion Detection in Network-Based IDS". In: *International Journal of Intelligent Systems and Applications* 6 (2 Jan. 2014), pp. 61–68. ISSN: 2074904X. DOI: 10.5815/IJISA.2014.02.08.

[107] SE Quincozes et al. "A survey on intrusion detection and prevention systems in digital substations". In: *Elsevier* (2020). URL: https://www.sciencedirect.com/science/article/pii/S1389128620312895?casa_token=4wlTXXAc2hUAAAAA:AXRzqBatXqib3p-M4jNeIjXS4YDKg3Nt_6DexxBuOO9qXjOX6h2esGd4DXLQnviL8lafRQhjakc.

[108] Hamid Bostani and Mansour Sheikhan. "Hybrid of anomaly-based and specification-based IDS for Internet of Things using unsupervised OPF based on MapReduce approach". In: *Computer Communications* 98 (Jan. 2017), pp. 52–71. ISSN: 0140-3664. DOI: 10.1016/J.COMCOM.2016.12.001.

[109] Huseyin Cavusoglu, Birendra Mishra, and Srinivasan Raghunathan. "A model for evaluating IT security investments". In: *Communications of the ACM* 47 (7 July 2004), pp. 87–92. ISSN: 00010782. DOI: 10.1145/1005817.1005828. URL: https://dl.acm.org/doi/10.1145/1005817.1005828.

[110] Tigist Abera et al. "Invited - Things, trouble, trust: On building trust in IoT systems". In: *Proceedings - Design Automation Conference* 05-09-June-2016 (June 2016). ISSN: 0738100X. DOI: 10.1145/2897937.2905020.

[111] N. Asokan et al. "SEDA: Scalable embedded device attestation". In: *Proceedings of the ACM Conference on Computer and Communications Security* 2015-October (Oct. 2015), pp. 964–975. ISSN: 15437221. DOI: 10.1145/2810103.2813670. URL: https://dl.acm.org/doi/10.1145/2810103.2813670.

[112] E Dushku et al. "Disseratation: Remote attestation to ensure the security of future Internet of Things services". In: *Thesis Sapienza University of Rome, Dipartmento di Informatica, Italy* (Feb. 2020). URL: https://sites.google.com/di.uniroma1.it/dushku/.

[113] Alexander Sprogø Banks, Marek Kisiel, and Philip Korsholm. "Remote Attestation: A Literature Review". In: (May 2021). URL: https://arxiv.org/abs/2105.02466v2.

[114] Lance Spitzner. "Honeypots: tracking hackers". In: *Addison Wesley* (2003). URL: http://www.it-docs.net/ddata/792.pdf.

[115] Christos Dalamagkas et al. “A Survey on honeypots, honeynets and their applications on smart grid”. In: *Proceedings of the 2019 IEEE Conference on Network Softwarization: Unleashing the Power of Network Softwarization, NetSoft 2019* (June 2019), pp. 93–100. DOI: 10.1109/NETSOFT.2019.8806693.

[116] E. Alata et al. “Lessons learned from the deployment of a high-interaction honeypot”. In: *Proceedings - Sixth European Dependable Computing Conference, EDCC 2006* (2006), pp. 39–44. DOI: 10.1109/EDCC.2006.17.

[117] W A Alzoubi, Wael Ahmad Alzoubi, and Maen T Alrashdan. “The effect of using honeypot network on system security”. In: *Canada. International Journal of Data and Network Science* 6 (2022), pp. 1413–1418. DOI: 10.5267/j.ijdns.2022.5.010. URL: www.GrowingScience.com/ijds.

[118] FM Gonzalez-Longatt and JL Rueda. “PowerFactory applications for power system analysis”. In: (2014). URL: https://books.google.com/books?hl=nl&lr=&id=v7TzBQAAQBAJ&oi=fnd&pg=PR5&dq=DIgSILENT+PowerFactory&ots=qTCIYu-ovq&sig=GYUrqrjc6LMruM3SOFFyZ5TbRqc.

[119] Zhiyuan Yang et al. “Implementation of Risk-Aggregated Substation Testbed Using Generative Adversarial Networks”. In: *IEEE Transactions on Smart Grid* 14 (1 Jan. 2023), pp. 677–689. ISSN: 19493061. DOI: 10.1109/TSG.2022.3192522.

[120] TenneT Holding BV. “TenneT Group legal overview”. In: (2023).

[121] PricewaterhouseCoopers. “Onderzoeksrapport: TenneT; Markt, Organisatie en Eigendom”. In: (2018). URL: www.pwc.nl.

[122] Carl Koopmans, Menno Van Benthem, and Walter Hulsker. “ICT-projecten van de overheid: Omgaan met onzekerheid”. In: *TPEdigitaal* 13 (2 2019), pp. 50–59.

[123] Carel Eijgenraam et al. “EVALUATIE VAN INFRASTRUCTUURPROJECTEN LEIDRAAD VOOR KOSTEN-BATENANALYSE”. In: *Centraal Planbureau Nederlands Economisch Instituut* (2000).

[124] Anna Nagurney and Shivani Shukla. “Multifirm models of cybersecurity investment competition vs. cooperation and network vulnerability”. In: *European Journal of Operational Research* 260 (2 July 2017), pp. 588–600. ISSN: 03772217. DOI: 10.1016/J.EJOR.2016.12.034.

[125] European Network of Transmission System Operators for Electricity. “European Network of Transmission System Operators for Electricity European Electricity Transmission Grids and the Energy Transition Why remuneration frameworks need to evolve”. In: (2021).

[126] Yiseul Cho and Joel Clark. “Strategic Philanthropy for Cyber Security: An extended cost-benefit analysis framework to study cybersecurity”. In: *Massachusetts Institute of Technology Library* (June 2012). URL: https://dspace.mit.edu/handle/1721.1/72880.

[127] Ben Krishna and Sebastian M.P. “Examining the relationship between e-government development, nation’s cyber-security commitment, business usage and economic prosperity: a cross-country analysis”. In: *Information and Computer Security* 29 (5 Nov. 2021), pp. 737–760. ISSN: 2056497X. DOI: 10.1108/ICS-12-2020-0205/FULL/HTML.

[128] Thomas Schröder and Wilhelm Kuckshinrichs. “Value of lost load: An efficient economic indicator for power supply security? A literature review”. In: *Frontiers in Energy Research* 3 (DEC Dec. 2015), p. 55. ISSN: 2296598X. DOI: 10.3389/FENRG.2015.00055/BIBTEX.

[129] Autoriteit Consument en Markt (ACM). “The value of lost load for electricity in the Netherlands Final report Client: The Netherlands Authority for Consumers and Markets”. In: (2022).

[130] UCTE. “FINAL REPORT of the Investigation Committee on the 28 September 2003 Blackout in Italy”. In: (2003).

[131] UCTE. “Final Report System Disturbance on 4 November 2006 union for the co-ordination of transmission of electricity”. In: (2006).

[132] US Canada Power System Outage Task Force. “Canada Power System Outage Task Force Final Report on the Implementation of Task Force Recommendations”. In: (2006).

[133] Yi Ting Chua et al. “Identifying Unintended Harms of Cybersecurity Countermeasures”. In: *eCrime Researchers Summit, eCrime* 2019-November (Nov. 2019). ISSN: 21591245. DOI: 10.1109/ECRIME47957.2019.9037589.

[134] Russell G. Smith, Glenn. Worthington, and Nicholas. Wolanin. “E-crime solutions and crime displacement”. In: *Crime and criminal justice* (Jan. 2003), p. 6.

[135] Rok Bojanc and Borka Jerman-Blažič. “An economic modelling approach to information security risk management”. In: *International Journal of Information Management* 28 (5 Oct. 2008), pp. 413–422. ISSN: 0268-4012. DOI: 10.1016/J.IJINFOMGT.2008.02.002.

[136] Ben H Thacker et al. “Concepts of model verification and validation”. In: (2004).

[137] R. M Lee, M. J Assante, and T Conway. “Analysis of the cyber attack on the Ukrainian power grid”. In: *Information Sharing and Analysis Center (E-ISAC)* (2016). URL: https://africautc.org/wp-content/uploads/2018/05/E-ISAC_SANS_Ukraine_DUC_5.pdf.

[138] S Ashraf et al. “Denial-of-service attack on iec 61850-based substation automation system: A crucial cyber threat towards smart substation pathways”. In: *Sensors* (2021). DOI: 10.3390/s21196415. URL: https://www.mdpi.com/1286370.

[139] Mansi Girdhar et al. “Cybersecurity of Process Bus Network in Digital Substations Machine Learning based ECU Detection for Automotive Security View project Cybersecurity of Process Bus Network in Digital Substations”. In: *2021 International Conference on Electronics, Information, and Communication (ICEIC)* (2021). DOI: 10.1109/ICEIC51217.2021.9369743. URL: https://www.researchgate.net/publication/349978451.

[140] Yiming Wu, Florin Stelea, and Anders Johnsson. “AN INVESTIGATION ON DATA GATEWAY FUNCTIONALITIES FOR ENTERPRISE ANCILLARY SERVICES IN DIGITAL SUBSTATIONS”. In: *25th International Conference on Electricity Distribution* (2019), pp. 3–6.

[141] Athar Khodabakhsh et al. “Cyber-Risk Identification for a Digital Substation”. In: *ARES ’20: Proceedings of the 15th International Conference on Availability, Reliability and Security* (2020). DOI: 10.1145/3407023.3409227. URL: https://doi.org/10.1145/3407023.3409227.

[142] M Garuba, C Liu, and D Fraites. “Intrusion techniques: Comparative study of network intrusion detection systems”. In: *IEEE Xplore: Fifth International Conference on Information Technology: New Generations* (2007). URL: https://ieeexplore.ieee.org/abstract/document/4492545/.

[143] Eugene E Schultz. “Calculating Total Cost of Ownership on Intrusion Prevention Technology A SANS Analyst Product Review TCO Exercises Favor Automated Management PAgE 5 Sponsored by Sourcefire”. In: *SANS Institute InfoSec* (2014).

[144] CEO AirGap.io Ritesh Argawal. *business-case-for-microsegmentation*. Accessed: 25-05-2023. URL: https://airgap.io/business-case-for-microsegmentation/.

[145] Hailun Tan, Wen Hu, and Sanjay Jha. “A TPM-enabled Remote Attestation Protocol (TRAP) in wireless sensor networks”. In: *PM2HW2N’11 - Proceedings of the 6th ACM International Workshop on Performance Monitoring, Measurement, and Evaluation of Heterogeneous Wireless and Wired Networks* (2011), pp. 9–16. DOI: 10.1145/2069087.2069090.

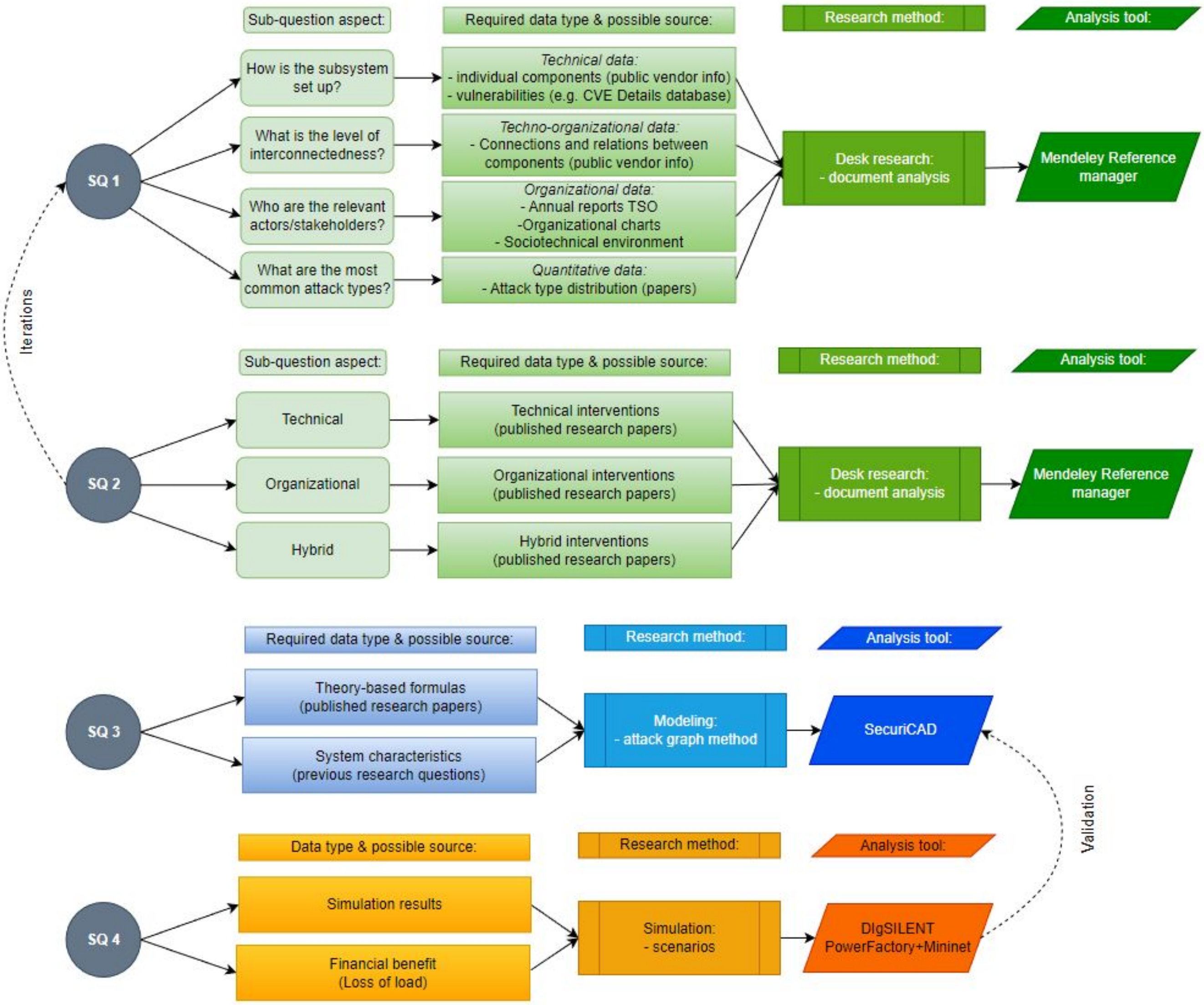
[146] Y Hao et al. “Likelihood analysis of cyber data attacks to power systems with Markov decision processes”. In: *IEEE TRANSACTIONS ON SMART GRID* 9 (4 2018), pp. 3191–3203. URL: https://ieeexplore.ieee.org/abstract/document/7742922/.

[147] Carlijn (C.C.) Bijvoet et al. “”Gansch het raderwerk staat stil” : de kosten van stroomstoringen”. In: (2003). URL: https://www.researchgate.net/publication/254768729_Gansch_het_raderwerk_staat_stil_de_kosten_van_stroonstoringen.

[148] H Chen, JA Clark, and SA Shaikh. “Optimising IDS sensor placement”. In: *ARES ’10 International Conference on Availability, Reliability, and Security* (2010), pp. 315–320. URL: https://ieeexplore.ieee.org/abstract/document/5438075/.

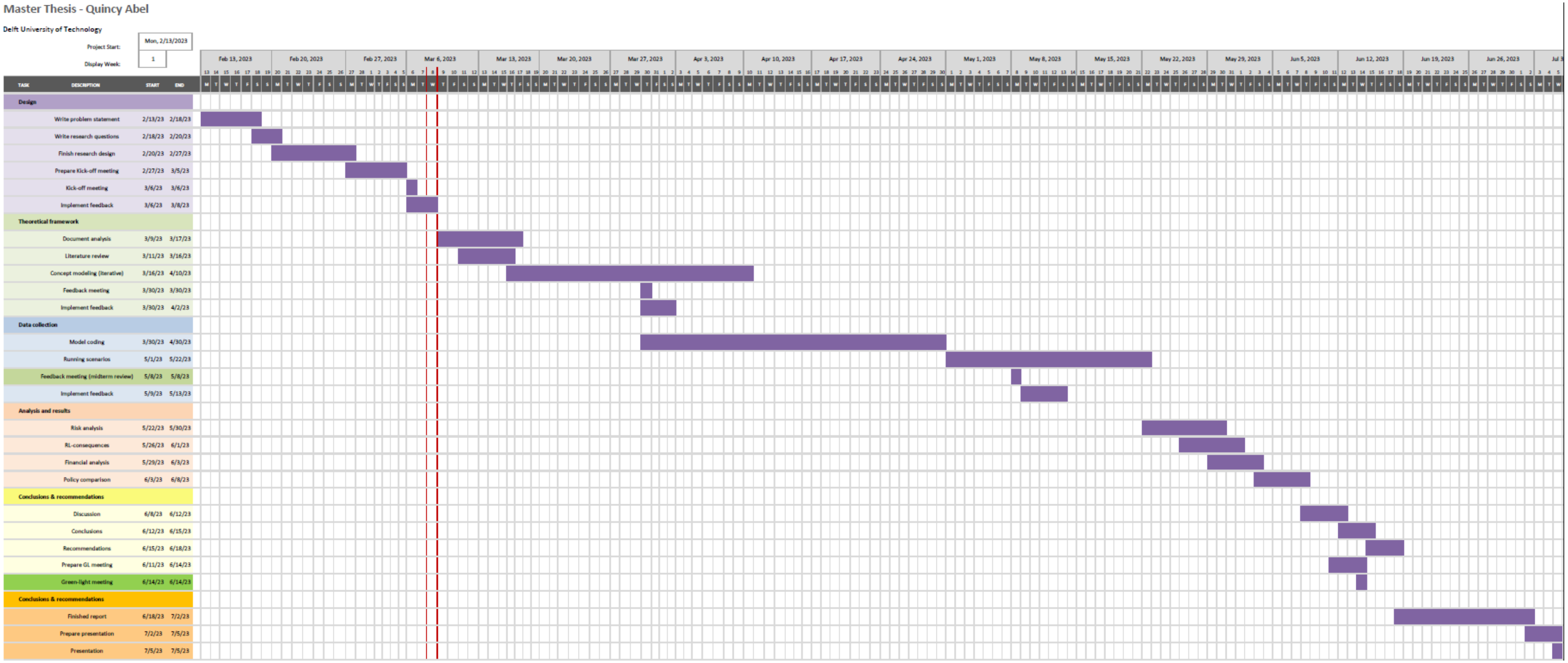
A

Research methods, data types and analysis tools


Sub-question aspect:
Required data type & possible source:
Research method:
Analysis tool:
SQ 1
How is the subsystem set up?
What is the level of interconnectedness?
Who are the relevant actors/stakeholders?
What are the most common attack types?
Technical data:
- individual components (public vendor info)
- vulnerabilities (e.g. CVE Details database)
Techno-organizational data:
- Connections and relations between components (public vendor info)
Organizational data:
- Annual reports TSO
-Organizational charts
- Sociotechnical environment
Quantitative data:
- Attack type distribution (papers)
Desk research:
- document analysis
Mendeley Reference manager
Iterations
Sub-question aspect:
Required data type & possible source:
Research method:
Analysis tool:
SQ 2
Technical
Organizational
Hybrid
Technical interventions
(published research papers)
Organizational interventions
(published research papers)
Hybrid interventions
(published research papers)
Desk research:
- document analysis
Mendeley Reference manager
Required data type & possible source:
Research method:
Analysis tool:
SQ 3
Theory-based formulas
(published research papers)
System characteristics
(previous research questions)
Modeling:
- attack graph method
SecuriCAD
Validation
Data type & possible source:
Research method:
Analysis tool:
SQ 4
Simulation results
Financial benefit
(Loss of load)
Simulation:
- scenarios
DIgSILENT PowerFactory+Mininet

B

Gantt Chart



Master Thesis - Quincy Abel

Delft University of Technology

Project Start: Mon, 2/13/2023

Display Week: 1

TASK	DESCRIPTION	START	END
Design			
	Write problem statement	2/13/23	2/18/23
	Write research questions	2/18/23	2/20/23
	Finish research design	2/20/23	2/27/23
	Prepare Kick-off meeting	2/27/23	3/5/23
	Kick-off meeting	3/6/23	3/6/23
	Implement feedback	3/6/23	3/8/23
Theoretical framework			
	Document analysis	3/9/23	3/17/23
	Literature review	3/11/23	3/16/23
	Concept modeling (iterative)	3/16/23	4/10/23
	Feedback meeting	3/30/23	3/30/23
	Implement feedback	3/30/23	4/2/23
Data collection			
	Model coding	3/30/23	4/30/23
	Running scenarios	5/1/23	5/22/23
	Feedback meeting (midterm review)	5/8/23	5/8/23
	Implement feedback	5/9/23	5/13/23
Analysis and results			
	Risk analysis	5/22/23	5/30/23
	RL-consequences	5/26/23	6/1/23
	Financial analysis	5/29/23	6/3/23
	Policy comparison	6/3/23	6/8/23
Conclusions & recommendations			
	Discussion	6/8/23	6/12/23
	Conclusions	6/12/23	6/15/23
	Recommendations	6/15/23	6/18/23
	Prepare GL meeting	6/11/23	6/14/23
	Green-light meeting	6/14/23	6/14/23
Conclusions & recommendations			
	Finished report	6/18/23	7/2/23
	Prepare presentation	7/2/23	7/5/23
	Presentation	7/5/23	7/5/23

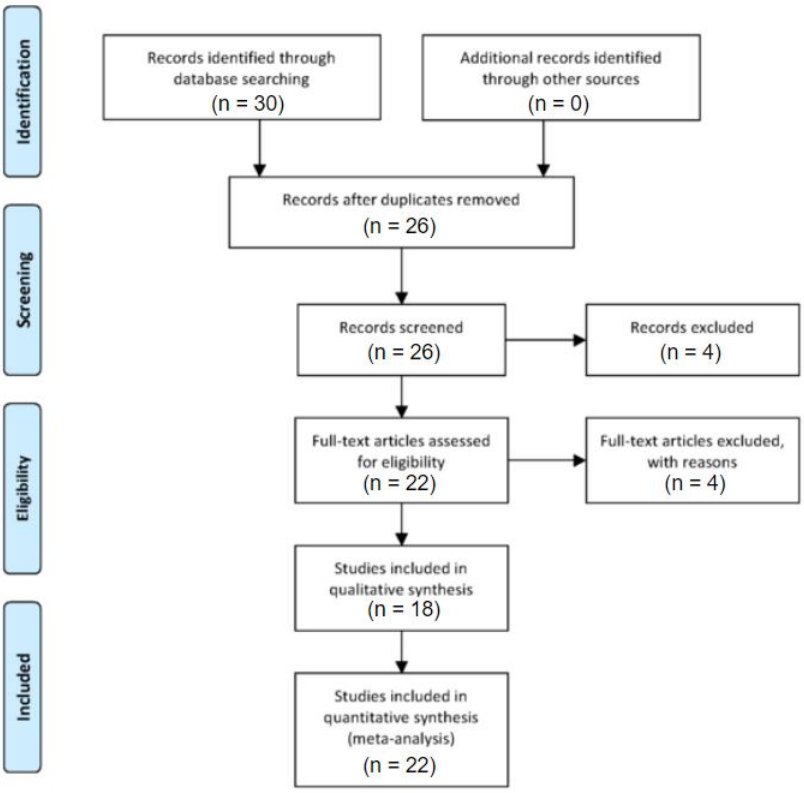
C
Stakeholders

Table C.1: Description of the stakeholders-actors

Name:	Role:	Description:
Employees	Actor	Employees directly influence the cybersecurity and functioning of the substation. Every day the employees interact with the system
NGOs	Stakeholder	NGOs try to influence the (cyber security) direction of the TSO in an indirect way through actions such as e.g. reports and protests
Governments and policy-makers	Actor	Governments (national and intergovernmental) directly influence the operations and direction of TSOs through binding regulation and oversight, including cyber security practices.
Customers	Stakeholder	Customers, despite being essential to deliver the service, are not able to actively influence the substations of the TSO as it is a natural monopoly. They don't influence the cybersecurity of substations
Regulators	Stakeholder	Regulators (such as ENTSO-E) indirectly influence the cybersecurity of substations. This association of TSOs advises the national and intergovernmental policy makers on proposed electricity related policies
Suppliers	Actors	Suppliers are essential actors in the (cybersecurity) supply chain. Without the components and services supplied by the suppliers the process cannot function. Substations consist of many components of suppliers. If these components are not cyber secure the whole system is not cyber secure. Security related actions of the suppliers thus directly influence the security of substations
Shareholders	Stakeholder	Shareholders can only influence big business decisions such as take-overs and fusions. They don't directly influence the cyber security of substations
Energy market participants	Stakeholder	Energy market participants are required for supplying energy to the consumers, but do not directly influence the cybersecurity of substations
Hackers	Actor	Hackers are malicious actors that actively try to disrupt the functioning of the substation or try to temper with it's data(structure)

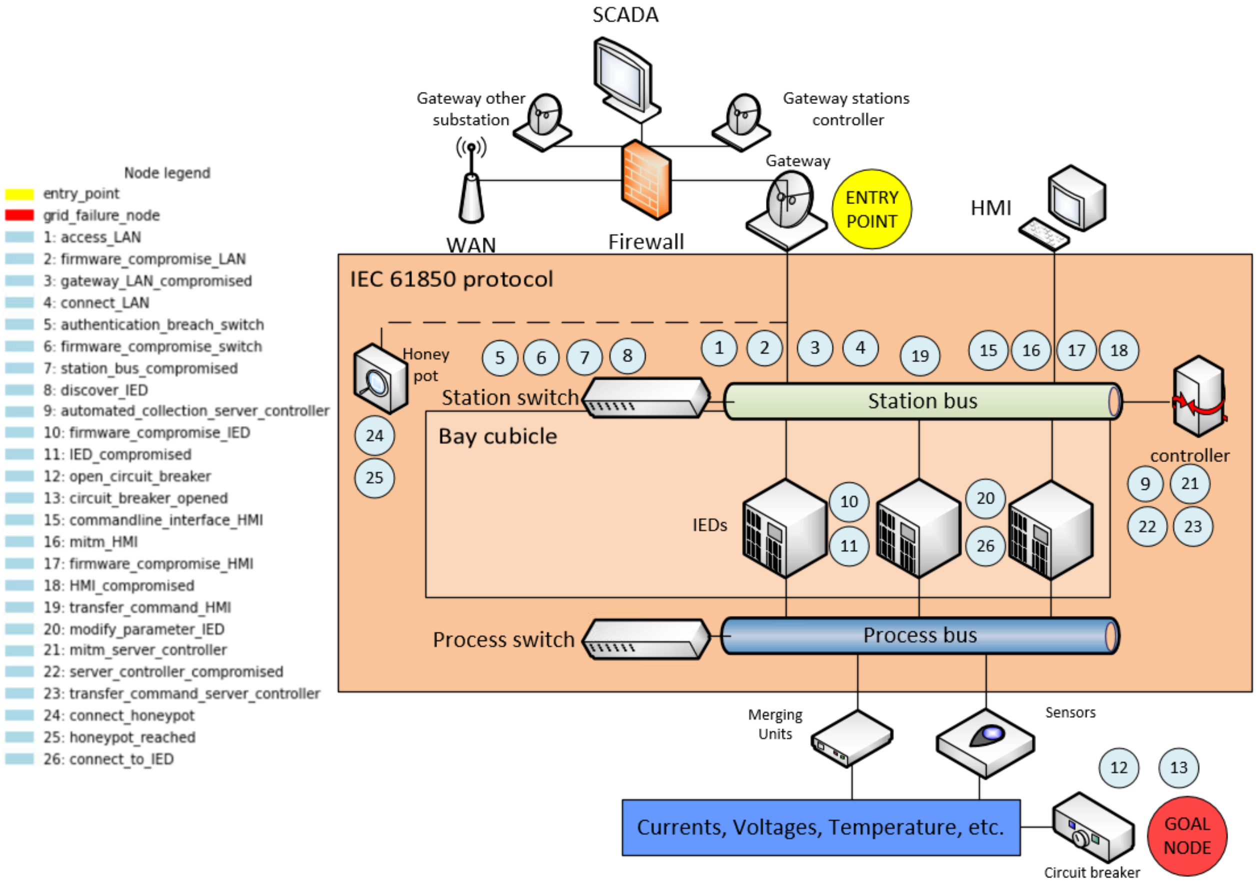
D

LRA overview defensive policies


Identification
Screening
Eligibility
Included
Records identified through database searching (n = 30)
Additional records identified through other sources (n = 0)
Records after duplicates removed (n = 26)
Records screened (n = 26)
Records excluded (n = 4)
Full-text articles assessed for eligibility (n = 22)
Full-text articles excluded, with reasons (n = 4)
Studies included in qualitative synthesis (n = 18)
Studies included in quantitative synthesis (meta-analysis) (n = 22)

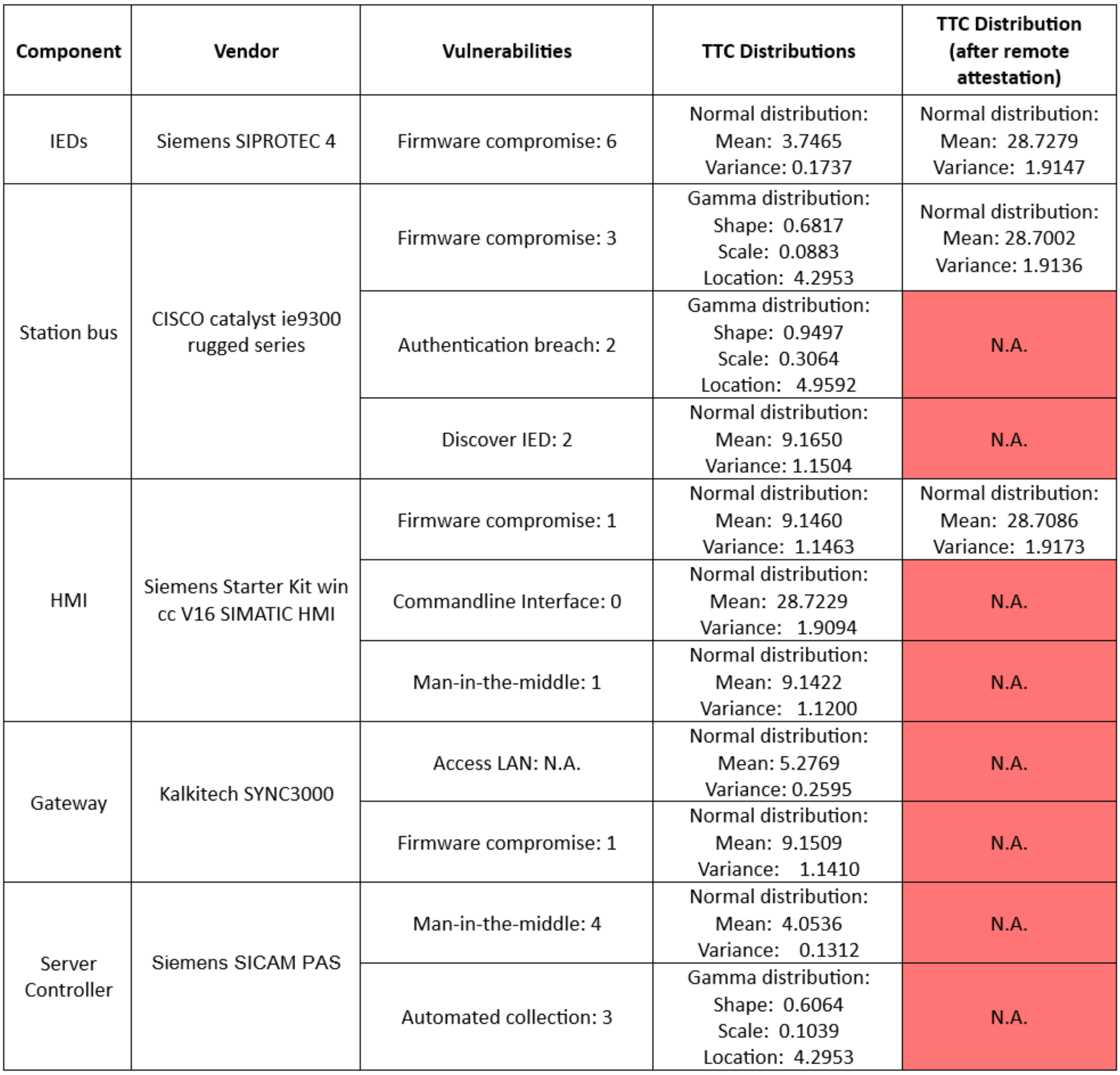
E

Substation topology and attack steps



F

TTC distributions per component

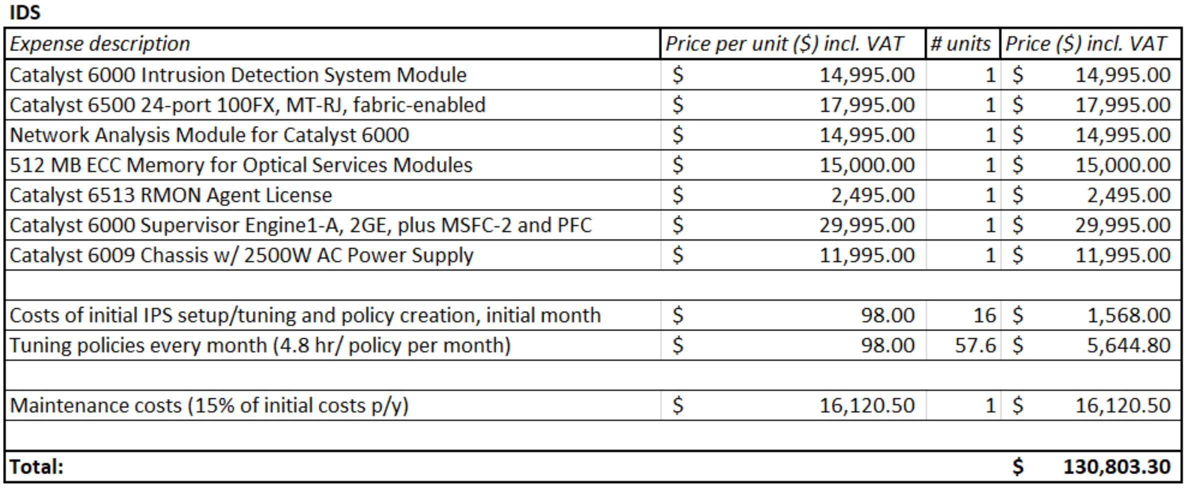


Component	Vendor	Vulnerabilities	TTC Distributions	TTC Distribution (after remote attestation)
IEDs	Siemens SIPROTEC 4	Firmware compromise: 6	Normal distribution: Mean: 3.7465 Variance: 0.1737	Normal distribution: Mean: 28.7279 Variance: 1.9147
Station bus	CISCO catalyst ie9300 rugged series	Firmware compromise: 3	Gamma distribution: Shape: 0.6817 Scale: 0.0883 Location: 4.2953	Normal distribution: Mean: 28.7002 Variance: 1.9136
		Authentication breach: 2	Gamma distribution: Shape: 0.9497 Scale: 0.3064 Location: 4.9592	N.A.
		Discover IED: 2	Normal distribution: Mean: 9.1650 Variance: 1.1504	N.A.
HMI	Siemens Starter Kit win cc V16 SIMATIC HMI	Firmware compromise: 1	Normal distribution: Mean: 9.1460 Variance: 1.1463	Normal distribution: Mean: 28.7086 Variance: 1.9173
		Commandline Interface: 0	Normal distribution: Mean: 28.7229 Variance: 1.9094	N.A.
		Man-in-the-middle: 1	Normal distribution: Mean: 9.1422 Variance: 1.1200	N.A.
Gateway	Kalkitech SYNC3000	Access LAN: N.A.	Normal distribution: Mean: 5.2769 Variance: 0.2595	N.A.
		Firmware compromise: 1	Normal distribution: Mean: 9.1509 Variance: 1.1410	N.A.
Server Controller	Siemens SICAM PAS	Man-in-the-middle: 4	Normal distribution: Mean: 4.0536 Variance: 0.1312	N.A.
		Automated collection: 3	Gamma distribution: Shape: 0.6064 Scale: 0.1039 Location: 4.2953	N.A.

Figure F.1: Component vulnerabilities and TTC distributions

G

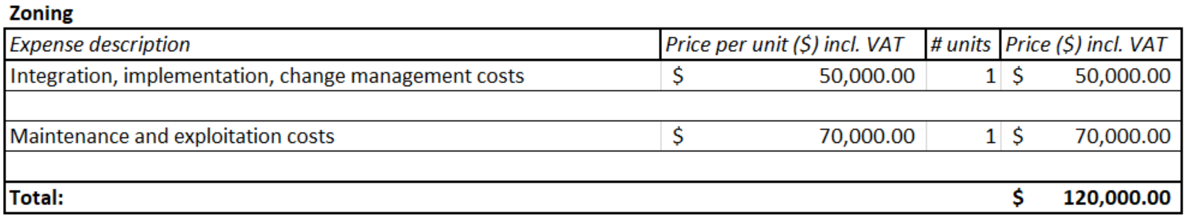
Financial calculations countermeasures



IDS

Expense description	*Price per unit ($) incl. VAT*	*# units*	*Price ($) incl. VAT*
Catalyst 6000 Intrusion Detection System Module	$ 14,995.00	1	$ 14,995.00
Catalyst 6500 24-port 100FX, MT-RJ, fabric-enabled	$ 17,995.00	1	$ 17,995.00
Network Analysis Module for Catalyst 6000	$ 14,995.00	1	$ 14,995.00
512 MB ECC Memory for Optical Services Modules	$ 15,000.00	1	$ 15,000.00
Catalyst 6513 RMON Agent License	$ 2,495.00	1	$ 2,495.00
Catalyst 6000 Supervisor Engine1-A, 2GE, plus MSFC-2 and PFC	$ 29,995.00	1	$ 29,995.00
Catalyst 6009 Chassis w/ 2500W AC Power Supply	$ 11,995.00	1	$ 11,995.00
Costs of initial IPS setup/tuning and policy creation, initial month	$ 98.00	16	$ 1,568.00
Tuning policies every month (4.8 hr/ policy per month)	$ 98.00	57.6	$ 5,644.80
Maintenance costs (15% of initial costs p/y)	$ 16,120.50	1	$ 16,120.50
Total:			**$ 130,803.30**

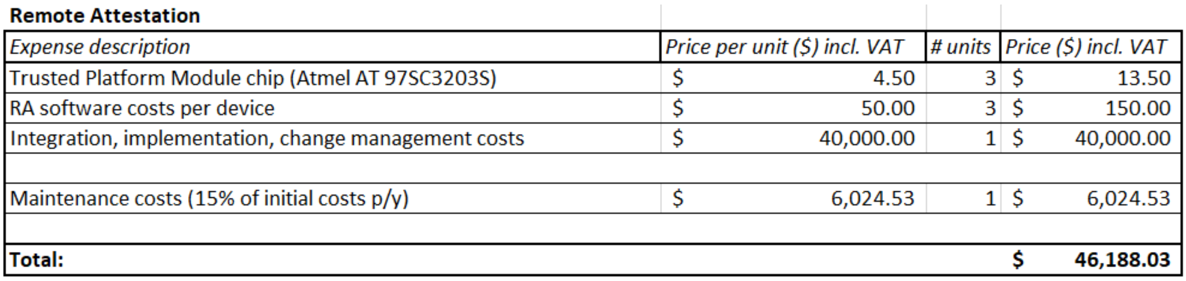
Figure G.1: Estimated cost composition IDS countermeasure



Zoning

Expense description	*Price per unit ($) incl. VAT*	*# units*	*Price ($) incl. VAT*
Integration, implementation, change management costs	$ 50,000.00	1	$ 50,000.00
Maintenance and exploitation costs	$ 70,000.00	1	$ 70,000.00
Total:			**$ 120,000.00**

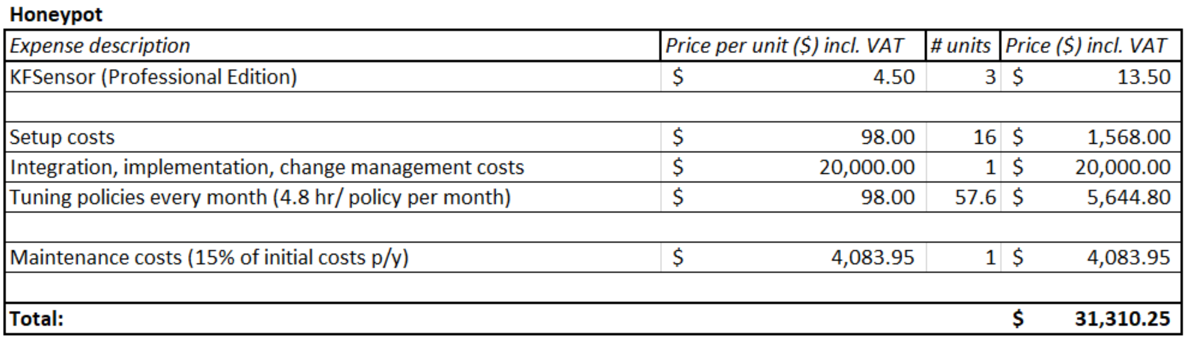
Figure G.2: Estimated cost composition Zoning countermeasure



Remote Attestation

Expense description	*Price per unit ($) incl. VAT*	*# units*	*Price ($) incl. VAT*
Trusted Platform Module chip (Atmel AT 97SC3203S)	$ 4.50	3	$ 13.50
RA software costs per device	$ 50.00	3	$ 150.00
Integration, implementation, change management costs	$ 40,000.00	1	$ 40,000.00
Maintenance costs (15% of initial costs p/y)	$ 6,024.53	1	$ 6,024.53
Total:			**$ 46,188.03**

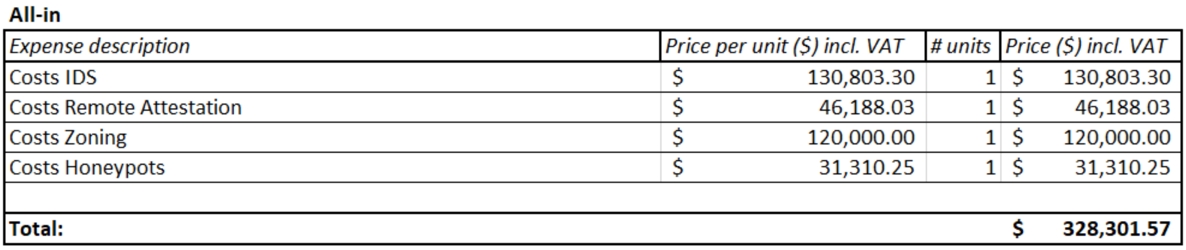
Figure G.3: Estimated cost composition Remote Attestation countermeasure



Honeypot

Expense description	*Price per unit ($) incl. VAT*	*# units*	*Price ($) incl. VAT*
KFSensor (Professional Edition)	$ 4.50	3	$ 13.50
Setup costs	$ 98.00	16	$ 1,568.00
Integration, implementation, change management costs	$ 20,000.00	1	$ 20,000.00
Tuning policies every month (4.8 hr/ policy per month)	$ 98.00	57.6	$ 5,644.80
Maintenance costs (15% of initial costs p/y)	$ 4,083.95	1	$ 4,083.95
Total:			**$ 31,310.25**

Figure G.4: Estimated cost composition Honeypot countermeasure



All-in

Expense description	*Price per unit ($) incl. VAT*	*# units*	*Price ($) incl. VAT*
Costs IDS	$ 130,803.30	1	$ 130,803.30
Costs Remote Attestation	$ 46,188.03	1	$ 46,188.03
Costs Zoning	$ 120,000.00	1	$ 120,000.00
Costs Honeypots	$ 31,310.25	1	$ 31,310.25
Total:			**$ 328,301.57**

Figure G.5: Estimated cost composition All-in countermeasures

H Python code Model 1

H.1. Python code

```python
import scipy.stats
import pandas as pd
import numpy as np
import numpy.random as random
import matplotlib.pyplot as plt
import math

from scipy.stats import norm
from scipy.stats import truncnorm
from scipy.stats import gamma
from scipy.stats import expon
from scipy.stats import lognorm
from scipy.stats import pareto


#time that malicious actor needs to exploit the target asset if the actor can access a certain vulnerability
tte1 = 1

#time that malicious actor needs to develop novel exploit for a known vulnerability
tte2 = 5.8

#time that malicious actor needs to exploit zero-day vulnerability
tte3 = 30.42

#amount of unique vulnerabilities of an asset which are know to malicious actor
#decrease this number by the amount of hardware and/or soft/firm-ware based vulnerabilities when remote attestation is on
RA = 0
V = 0 - RA

#amount of total vulnerabilities according to Zhang(2015) but adjusted from 7k to 6.1k due to Monte Carlo fitting limitations
K = 6100

#number of Monte-Carlo samples
samples = 10000

#initial matrice results
rows = samples
TTC_results_e = [0 for i in range(rows)]
```

```python
TTC_results_i = [0 for i in range(rows)]
TTC_results_b = [0 for i in range(rows)]
p1_results_e = [0 for i in range(rows)]
p1_results_i = [0 for i in range(rows)]
p1_results_b = [0 for i in range(rows)]

Attacker1='expert'
Attacker2='intermediate'
Attacker3='beginner'

for s in range(0,samples,1):

    if Attacker1 == 'expert':
        k_expert=np.random.normal(0.8, 0.04,size=None)

        fc_e = k_expert

        M_e = 360*k_expert

    if Attacker2 == 'intermediate':
        k_intermediate=np.random.normal(0.55, 0.07,size=None)

        fc_i = k_intermediate

        M_i = 250*k_intermediate

    if Attacker3 == 'beginner':
        k_beginner = np.random.normal(0.2, 0.05,size=None)

        fc_b = k_beginner

        M_b = 100*k_beginner

    #process 1 - Probability of exploiting known vulnerability with a known exploit. success probability
    #per malicious actor level (K = V_tot!!)
    p1_e= 1-math.exp(-(V*M_e)/K)
    p1_i= 1-math.exp(-(V*M_i)/K)
    p1_b= 1-math.exp(-(V*M_b)/K)

    #Time to compromise for process 1
    TTCp1_e = tte1*p1_e
    TTCp1_i = tte1*p1_i
    TTCp1_b = tte1*p1_b

    #process 2 Probability time needed to create an exploit for a known vulnerability. success probability
    #per malicious actor level
    p2_e = 1-p1_e
    p2_i = 1-p1_i
    p2_b = 1-p1_b

    #vulnerabilities that can be targeted by malicious actor (AM) and those which the malicious actor cannot target(NM)
    AM_e = int(fc_e*V)
    AM_i = int(fc_i*V)
    AM_b = int(fc_b*V)

    NM_e = V - AM_e
    NM_i = V - AM_i
    NM_b = V - AM_b

    #initialization
    SumProbability_e = 1
    SumProbability_i = 1
    SumProbability_b = 1
    Probability_e = 1
    Probability_i = 1
    Probability_b = 1

    #Number of tries to create exploit for known vulnerability
    for i in range(2, V-AM_e+1,1):
        for j in range(2,i,1):
            Probability_e = Probability_e*((NM_e-j+2)/(V-j+1))
            SumProbability_e = SumProbability_e + i*Probability_e

    for i in range(2, V-AM_i+1,1):
        for j in range(2,i,1):
            Probability_i = Probability_i*((NM_i-j+2)/(V-j+1))
            SumProbability_i = SumProbability_i + i*Probability_i

    for i in range(2, V-AM_b+1,1):
        for j in range(2,i,1):
            Probability_b = Probability_b*((NM_i-j+2)/(V-j+1))
            SumProbability_b = SumProbability_b + i*Probability_b

    #estimation of number of tried needed by malicious actor
    ET_e = k_expert*(SumProbability_e)
    u3_e = (1-k_expert)**V

    ET_i = k_intermediate*(SumProbability_i)
    u3_i = (1-k_intermediate)**V

    ET_b = k_beginner*(SumProbability_b)
    u3_b = (1-k_beginner)**V

    u2_e = 1-u3_e
    u2_i = 1-u3_i
    u2_b = 1-u3_b

    #Time to compromise for process 2
    TTCp2_e = tte2*p2_e*ET_e*u2_e
    TTCp2_i = tte2*p2_i*ET_i*u2_i
    TTCp2_b = tte2*p2_b*ET_b*u2_b

    #process 3 needed time for novel zero day exploit
    TTCp3_e = (((1/k_expert)-0.5)*tte3+tte2)*p2_e*u3_e
    TTCp3_i = (((1/k_intermediate)-0.5)*tte3+tte2)*p2_i*u3_i
    TTCp3_b = (((1/k_beginner)-0.5)*tte3+tte2)*p2_b*u3_b


    #total TTC calculation
    Tot_TTC_e = TTCp1_e + TTCp2_e + TTCp3_e
    Tot_TTC_i = TTCp1_i + TTCp2_i + TTCp3_i
    Tot_TTC_b = TTCp1_b + TTCp2_b + TTCp3_b


    TTC_results_e[s]=Tot_TTC_e
    p1_results_e[s]=TTCp1_e

    TTC_results_i[s]=Tot_TTC_i
    p1_results_i[s]=TTCp1_i

    TTC_results_b[s]=Tot_TTC_b
    p1_results_b[s]=TTCp1_b

print(Tot_TTC_e)

#distributions

shape1, loc1, scale1 = scipy.stats.distributions.gamma.fit(TTC_results_e)
shape2, loc2, scale2 = scipy.stats.distributions.gamma.fit(TTC_results_i)
shape3, loc3, scale3 = scipy.stats.distributions.gamma.fit(TTC_results_b)
#shape1, loc1, scale1 = scipy.stats.distributions.gamma.fit(TTC_results_e, floc=3)

mean1, var1 = scipy.stats.distributions.norm.fit(TTC_results_e)
print("\n Normal distribution for expert level attacker: mean =",mean1, "and
    variance=",var1)

mean2, var2, loc2, s2 = scipy.stats.distributions.truncnorm.fit(TTC_results_e)

print("\n Truncated Normal for expert level attackers -> mean:", mean2, ",variance
    :",var2, ",location:", loc2)

a, b = scipy.stats.distributions.expon.fit(TTC_results_e)

print("\n Exponential for expert level attackers ->",a ,"and ",b)

print ("\n Gamma Distribution Fitting for all level of attackers")
print("\n shape =",shape1, "loc =",loc1,"and scale=",scale1)
print("\n shape =",shape2, "loc =",loc2,"and scale=",scale2)
print("\n shape =",shape3, "loc =",loc3,"and scale=",scale3)

"Manually set the plotting space"
if Attacker1 == 'expert':
    x1 = np.linspace(3,5,100)
    range_A = [3,5]

if Attacker2 == 'intermediate':
    x2 = np.linspace(2.5,20,100)
    range_B = [2.5,20]

if Attacker3 == 'beginner':
    x3 = np.linspace(0,300,100)
    range_C = [0,300]

y_e = gamma.pdf(x1, shape1, loc1, scale1)
y_i = gamma.pdf(x2, shape2, loc2, scale2)
y_b = gamma.pdf(x3, shape3, loc3, scale3)

n_e = norm.pdf(x1, mean1, var1)
tn_e = truncnorm.pdf(x1, mean2, var2, loc2, s2)
p_e = expon.pdf(x1, a, b)

"Plotting the histograms of TTC per attacker category"
fig1 = plt.figure(1)
fig1, axs = plt.subplots(3, 1, figsize=(16,9))
fig1.tight_layout(pad=1.0)

axs[0].hist(TTC_results_e, range=range_A, bins=200, density=True, label = "Expert"
    , color = 'red')
axs[0].legend(loc ='upper left', prop={"size":14})
axs[0].set_xlabel('TTC in days \n\n a)', fontsize = 14)
```

```python
axs[0].set_ylabel('Percentage of Samples', fontsize = 14)

axs[1].hist(TTC_results_i, range=range_B, bins=200, density=True, label = "Intermediate", color="green")
axs[1].legend(loc ='upper left', prop={"size":14})
axs[1].set_xlabel('TTC in days \n\n b)', fontsize = 14)
axs[1].set_ylabel('Percentage of Samples', fontsize = 14)

axs[2].hist(TTC_results_b, range=range_C, bins=200, density=True, label = "Beginner", color = "orange")
axs[2].legend(loc ='upper right', prop={"size":14})
axs[2].set_xlabel('TTC in days \n\n c)', fontsize = 14)
axs[2].set_ylabel('Percentage of Samples', fontsize = 14)

"Plot the legends"
axs[0].legend()
axs[1].legend()
axs[2].legend()
fig1.tight_layout(pad=1.0)
#plt.savefig("Histograms.png")


"Plotting the distribution fitting"
fig2 = plt.figure(2)
fig2, axs = plt.subplots(1, 1, figsize=(16,9))

axs.hist(TTC_results_e, range=range_A, bins=100, density=True,  label = "Histogram", linewidth = 2.0)
axs.plot(x1, p_e, label = "Exponential Fit", linewidth = 3.0, linestyle = 'dashdot', color ='k')
axs.plot(x1, n_e, label = "Normal Fit", linewidth = 3.0)


axs.plot(x1, y_e, label = "Gamma Fit", linewidth=3.0, linestyle = 'dashed', color = 'r')

axs.legend(loc ='upper left', prop={"size":20})
axs.set_xlabel('TTC in days', fontsize = 20)
axs.set_ylabel('Percentage of Samples', fontsize = 20)
axs.tick_params(axis='x', which='both', labelsize=20)
axs.tick_params(axis='y', which='both', labelsize=20)
axs.set_ylim(0.01, 25)
#axs.set_xlim(4.2, 4.7)

fig2.tight_layout(pad=1.0)
#plt.savefig("Histograms.png")
plt.show()

print('P1')
print(p1_e)
print('P2')
print(p2_e)
```

I

Python code Model 2

I.1. Python code

```python
import plotly.graph_objects as go
import networkx as nx
import random
from textwrap import wrap
import matplotlib.pyplot as plt
import matplotlib.patheffects as pe
import numpy as np
from adjustText import adjust_text
from collections import defaultdict
from matplotlib.path import Path
from matplotlib.ticker import MaxNLocator
import matplotlib.patches as patches
import matplotlib.lines as mlines
from networkx.drawing.nx_agraph import graphviz_layout

# Define the graph randomgamma(shape, scale, 1)[0] + loc
G = nx.MultiDiGraph()

honeypot_enabled = True # if this is true then the honeypot is added to the graph
RA_enabled = True #if this is true then the TTC of the corresponding edges is increased and if false then the edges keep their lower TTC

G.add_node("entry_point")
G.add_edge("entry_point", "access_LAN", TTC=np.random.normal(5.2769, 0.2595, 1)[0], name="EP-AL", color="black")
G.add_node("access_LAN")
G.add_node("firmware_compromise_LAN")
G.add_edge("entry_point", "firmware_compromise_LAN", TTC=np.random.normal(9.150897, 1.140997, 1)[0], name="EP-FCLAN", color="black")
G.add_edge("firmware_compromise_LAN", "gateway_LAN_compromised", TTC=0, name="FCLAN-GLC", color="green")
G.add_edge("access_LAN", "gateway_LAN_compromised", TTC=0, name="AL-GLC", color="green")
G.add_node("gateway_LAN_compromised")
G.add_edge("gateway_LAN_compromised", "connect_LAN", TTC=0, name="GLC-CL", color="green")
G.add_node("connect_LAN")
G.add_node("authentication_breach_switch")
G.add_node("firmware_compromise_switch")
G.add_edge("connect_LAN", "authentication_breach_switch", TTC=np.random.gamma(0.949702, 0.30644, 1)[0] + 4.959169, name="CL-ABS", color="black")
```

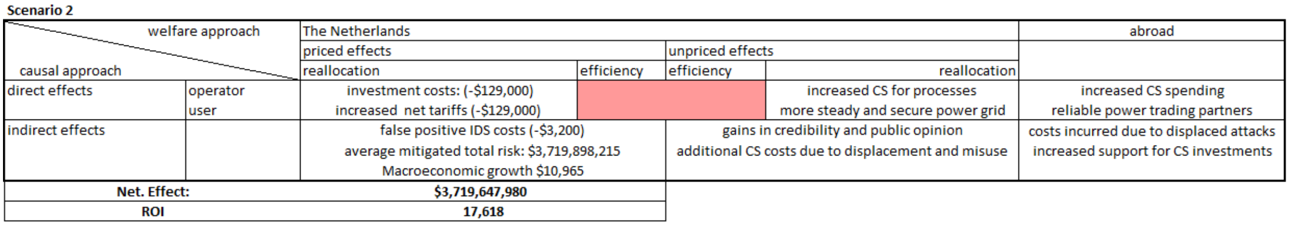
```python
G.add_edge("connect_LAN", "firmware_compromise_switch", TTC=np.random.gamma(0.681717, 0.088328, 1)[0] + 4.29532, name="CL-FCS", color="black")
G.add_edge("authentication_breach_switch", "station_bus_compromised", TTC=0, name="ABS-SBC", color="green")
G.add_edge("firmware_compromise_switch", "station_bus_compromised", TTC=0, name="FCS-SBC", color="green")
G.add_node("station_bus_compromised")
G.add_edge("station_bus_compromised", "discover_IED", TTC=np.random.normal(9.16497, 1.150367, 1)[0], name="SBC-DIED", color="black")
G.add_edge("station_bus_compromised", "automated_collection_server_controller", TTC=np.random.gamma(0.606349, 0.10392, 1)[0] + 4.29532, name="SBC-ACSC", color="black")
G.add_node("automated_collection_server_controller")
G.add_node("discover_IED")
G.add_node("firmware_compromise_IED")
G.add_edge("firmware_compromise_IED", "IED_compromised", TTC=0, name="FCIED-IEDC", color="green")
G.add_node("IED_compromised")
G.add_edge("IED_compromised", "open_circuit_breaker", TTC=0, name="IEDC-OCB", color="green")
G.add_node("open_circuit_breaker")
G.add_edge("open_circuit_breaker", "circuit_breaker_opened", TTC=0, name="OCB-CBO", color="green")
G.add_node("circuit_breaker_opened")
G.add_edge("circuit_breaker_opened", "grid_failure_node", TTC=0, name="CBO-GFN", color="green")
G.add_node("grid_failure_node")
G.add_node("commandline_interface_HMI")
G.add_node("mitm_HMI")
G.add_node("firmware_compromise_HMI")
G.add_edge("station_bus_compromised", "commandline_interface_HMI", TTC=np.random.normal(28.72292, 1.90938, 1)[0], name="SBC-CIHMI", color="black")
G.add_edge("commandline_interface_HMI", "HMI_compromised", TTC=0, name="CIHMI-HMIC", color="green")
G.add_edge("station_bus_compromised", "mitm_HMI", TTC=np.random.normal(9.14217, 1.119987, 1)[0], name="SBC-MITMHMI", color="black")
G.add_node("HMI_compromised")
G.add_edge("mitm_HMI", "HMI_compromised", TTC=0, name="MITMHMI-HMIC", color="green")
G.add_edge("firmware_compromise_HMI", "HMI_compromised", TTC=0, name="FCHMI-HMIC", color="green")
G.add_node("transfer_command_HMI")
G.add_edge("HMI_compromised", "transfer_command_HMI", TTC=0, name="HMIC-TCHMI", color="green")
G.add_node("modify_parameter_IED")
G.add_edge("transfer_command_HMI", "modify_parameter_IED", TTC=0, name="TCHMI-MP", color="green")
G.add_edge("modify_parameter_IED", "circuit_breaker_opened", TTC=0, name="MP-CBO", color="green")
G.add_edge("connect_LAN", "mitm_server_controller", TTC=np.random.normal(4.05357, 0.131204, 1)[0], name="CL-MITMSC", color="black")
G.add_node("mitm_server_controller")
G.add_node("server_controller_compromised")
G.add_edge("mitm_server_controller", "server_controller_compromised", TTC=0, name="MITMSC-SCC", color="green")
G.add_node("transfer_command_server_controller")
G.add_edge("server_controller_compromised", "transfer_command_server_controller", TTC=0, name="SCC-TCSC", color="green")
G.add_edge("transfer_command_server_controller", "modify_parameter_IED", TTC=0, name="TCSC-MP", color="green")

```

```python
if honeypot_enabled:
    G.add_node("connect_honeypot")
    G.add_edge("connect_LAN", "connect_honeypot", TTC= 1, name="SBC-CHP", color="black")
    G.add_node("honeypot_reached")
    G.add_edge("connect_honeypot", "honeypot_reached", TTC= 0, name="CHP-HPR", color="green")
    G.add_node("connect_to_IED")
    G.add_edge("honeypot_reached", "connect_to_IED", TTC= 0, name="HPR-CTIED", color="green")
    G.add_edge("connect_to_IED", "IED_compromised", TTC= 0, name="CTIED-IEDC", color="green")

if RA_enabled:
    G.add_edge("discover_IED", "firmware_compromise_IED", TTC=np.random.normal(28.7279, 1.91468, 1)[0], name="DIED-FCIED", color="black")
    G.add_edge("automated_collection_server_controller", "firmware_compromise_IED", TTC=np.random.normal(28.7279, 1.91468, 1)[0], name="ACSC-FCIED", color="black")
    G.add_edge("connect_LAN", "firmware_compromise_switch", TTC=np.random.normal(28.7002, 1.91364, 1)[0], name="CL-FCS", color="black")
    G.add_edge("station_bus_compromised", "firmware_compromise_HMI", TTC=np.random.normal(28.70862, 1.917329, 1)[0] + 4.841127, name="SBC-FCHMI", color="black")
else:
    G.add_edge("discover_IED", "firmware_compromise_IED", TTC=np.random.normal(3.746522, 0.173731, 1)[0], name="DIED-FCIED", color="black")
    G.add_edge("automated_collection_server_controller", "firmware_compromise_IED", TTC=np.random.normal(3.74252, 0.15001, 1)[0], name="ACSC-FCIED", color="black")
    G.add_edge("connect_LAN", "firmware_compromise_switch", TTC=np.random.gamma(0.681717, 0.088328, 1)[0] + 4.29532, name="CL-FCS", color="black")
    G.add_edge("station_bus_compromised", "firmware_compromise_HMI", TTC=np.random.normal(9.145946, 1.146286, 1)[0] + 4.841127, name="SBC-FCHMI", color="black")

# Define the Monte Carlo simulation parameters
num_trials = 20000
penalty_IDS = 80
penalty_HP = 270

simulation_paths = {}
shortest_paths = defaultdict(dict)
all_shortest_paths = defaultdict(dict)
penalized_edges = defaultdict(int)  # Stores whether a penalty was applied to an edge
shortest_paths_list = []

# Function to calculate TTC for a path
def calculate_ttc_for_path(G, path):
    total_ttc = 0
    for i in range(len(path) - 1):
        total_ttc += G[path[i]][path[i + 1]][0]['TTC']
    return total_ttc

# Function to find all paths from source to target #penalty zoning = 15 for analysis
def find_paths(graph, start, end, path=[], ttc=0, IDS_enabled=True, zoning_enabled=True, shortest_path=None, honeypot_penalty=True):
    path = path + [start]
    if start == end:
        if IDS_enabled and "gateway_LAN_compromised" in path and "server_controller_compromised" in path:
            ttc += penalty_IDS
        if zoning_enabled and any(node in path for node in ["discover_IED", "automated_collection_server_controller", "transfer_command_HMI", "transfer_command_server_controller"]):
            ttc += 15
        return [(path, ttc)]
    if start not in graph:
        return []
    paths = []
    for node in graph[start]:
        edge_ttc = min(graph[start][node][k]['TTC'] for k in graph[start][node])
        if node not in path:
            newpaths = find_paths(graph, node, end, path, ttc + edge_ttc, IDS_enabled, zoning_enabled, shortest_path, honeypot_penalty)
            for newpath in newpaths:
                paths.append(newpath)
    return paths

# setup a dictionary to store names and colors for each edge
edge_colors = {(u, v): d['color'] for u, v, d in G.edges(data=True)}

# Create edge_names dictionary
edge_names = {}
for u, v, k, data in G.edges(keys=True, data=True):
    edge_names[(u, v, k)] = data['name']

# Initialize an empty dictionary to store TTC values for each edge
edge_ttc_values = {}
for u, v, k in G.edges(keys=True):
    edge_ttc_values[(u, v, k)] = []
    G[u][v][k]['TTC_original'] = G[u][v][k]['TTC']

# update shortest path lengths for multiple trials

shortest_path_lengths = []
source_node = "entry_point"
target_node = "grid_failure_node"
# Initialize the shortest_paths and all_shortest_paths dictionaries for the source node
shortest_paths[source_node] = {target_node: {'Path': None, 'TTC': float('inf')}}
all_shortest_paths[source_node] = {target_node: None}

# Initialize the shortest path and its total TTC
shortest_path = None
shortest_ttc = float('inf')
shortest_ttc_dict = {}

# Initialize the overall shortest path and its total TTC
overall_shortest_path = []
overall_shortest_ttc = float('inf')

for i in range(num_trials):
    G.edges["entry_point", "access_LAN", 0]["TTC"] = np.random.normal(5.2769, 0.2595, 1)[0]
    G.edges["entry_point", "firmware_compromise_LAN", 0]["TTC"] = np.random.normal(9.150897, 1.140997, 1)[0]
    G.edges["access_LAN", "gateway_LAN_compromised", 0]["TTC"] = 0
    G.edges["gateway_LAN_compromised", "connect_LAN", 0]["TTC"] = 0
```

```python
    G.edges["connect_LAN", "authentication_breach_switch", 0]["TTC"] = np.random.gamma(0.949702, 0.30644, 1)[0] + 4.959169
    G.edges["connect_LAN", "firmware_compromise_switch", 0]["TTC"] = np.random.gamma(0.681717, 0.088328, 1)[0] + 4.29532
    G.edges["authentication_breach_switch", "station_bus_compromised", 0]["TTC"] = 0
    G.edges["firmware_compromise_switch", "station_bus_compromised", 0]["TTC"] = 0
    G.edges["station_bus_compromised", "discover_IED", 0]["TTC"] = np.random.normal(9.16497, 1.150367, 1)[0]
    G.edges["station_bus_compromised", "automated_collection_server_controller", 0]["TTC"] = np.random.gamma(0.606349, 0.10392, 1)[0] + 4.29532
    G.edges["discover_IED", "firmware_compromise_IED", 0]["TTC"] = np.random.normal(3.74252, 0.15001, 1)[0]
    G.edges["automated_collection_server_controller", "firmware_compromise_IED", 0]["TTC"] = np.random.normal(3.74252, 0.15001, 1)[0]
    G.edges["firmware_compromise_IED", "IED_compromised", 0]["TTC"] = 0
    G.edges["IED_compromised", "open_circuit_breaker", 0]["TTC"] = 0
    G.edges["open_circuit_breaker", "circuit_breaker_opened", 0]["TTC"] = 0
    G.edges["circuit_breaker_opened", "grid_failure_node", 0]["TTC"] = 0
    G.edges["station_bus_compromised", "commandline_interface_HMI", 0]["TTC"] = np.random.normal(28.72292, 1.90938, 1)[0]
    G.edges["commandline_interface_HMI", "HMI_compromised", 0]["TTC"] = 0
    G.edges["HMI_compromised", "transfer_command_HMI", 0]["TTC"] = 0
    G.edges["transfer_command_HMI", "modify_parameter_IED", 0]["TTC"] = 0
    G.edges["modify_parameter_IED", "circuit_breaker_opened", 0]["TTC"] = 0
    G.edges["connect_LAN", "mitm_server_controller", 0]["TTC"] = np.random.normal(4.05357, 0.131204, 1)[0]
    G.edges["mitm_server_controller", "server_controller_compromised", 0]["TTC"] = 0
    G.edges["server_controller_compromised", "transfer_command_server_controller", 0]["TTC"] = 0
    G.edges["transfer_command_server_controller", "modify_parameter_IED", 0]["TTC"] = 0

    if honeypot_enabled:
        G.edges["connect_LAN", "connect_honeypot", 0]["TTC"] = 1
        G.edges["connect_honeypot", "honeypot_reached", 0]["TTC"] = 0
        G.edges["honeypot_reached", "connect_to_IED", 0]["TTC"] = 0
        G.edges["connect_to_IED", "IED_compromised", 0]["TTC"] = 0

    if RA_enabled:
        G.edges["discover_IED", "firmware_compromise_IED", 0]["TTC"] = np.random.normal(28.7279, 1.91468, 1)[0]
        G.edges["automated_collection_server_controller", "firmware_compromise_IED", 0]["TTC"] = np.random.normal(28.7279, 1.91468, 1)[0]
        G.edges["connect_LAN", "firmware_compromise_switch", 0]["TTC"] = np.random.normal(28.7002, 1.91364, 1)[0]
        G.edges["station_bus_compromised", "firmware_compromise_HMI", 0]["TTC"] = np.random.normal(28.70862, 1.917329, 1)[0]
    else:
        G.edges["discover_IED", "firmware_compromise_IED", 0]["TTC"] = np.random.normal(3.746522, 0.173731, 1)[0]
        G.edges["automated_collection_server_controller", "firmware_compromise_IED", 0]["TTC"] = np.random.normal(3.74252, 0.15001, 1)[0]
        G.edges["connect_LAN", "firmware_compromise_switch", 0]["TTC"] = np.random.gamma(0.681717, 0.088328, 1)[0] + 4.29532
        G.edges["station_bus_compromised", "firmware_compromise_HMI", 0]["TTC"] = np.random.normal(9.145946, 1.146286, 1)[0]

    # Find all paths
    all_paths = find_paths(G, "entry_point", "grid_failure_node", shortest_path=shortest_path)

    # For each path, if it's a new path or if the total TTC is shorter than the previous shortest, update the variables
    for path, total_ttc in all_paths:
        path_tuple = tuple(path)
        if total_ttc < shortest_ttc:
            shortest_ttc = total_ttc
            shortest_path = path[:]
        if path_tuple not in shortest_ttc_dict or total_ttc < shortest_ttc_dict[path_tuple]:
            shortest_ttc_dict[path_tuple] = total_ttc

    # Update the overall shortest path and its TTC if necessary
    if shortest_ttc < overall_shortest_ttc:
        overall_shortest_ttc = shortest_ttc
        overall_shortest_path = shortest_path[:]

    # Check if the overall shortest path contains the node "honeypot_reached"
    if "honeypot_reached" in overall_shortest_path:
        # Apply the penalty to all paths
        for path_tuple, total_ttc in shortest_ttc_dict.items():
            shortest_ttc_dict[path_tuple] += penalty_HP

        # Update the overall shortest path and its TTC after penalty has been applied
        overall_shortest_path, overall_shortest_ttc = min(shortest_ttc_dict.items(), key=lambda x: x[1])

    # Store TTC values for each edge
    for path, total_ttc in all_paths:
        for u, v in zip(path[:-1], path[1:]):
            edge_ttc_values[(u, v, 0)].append(G[u][v][0]['TTC'])

    # Update the overall shortest path and its TTC if necessary
    if shortest_ttc < overall_shortest_ttc:
        overall_shortest_ttc = shortest_ttc
        overall_shortest_path = shortest_path[:]

# Sort the dictionary items by their TTC in ascending order
sorted_ttc_items = sorted(shortest_ttc_dict.items(), key=lambda item: item[1])

# If there are at least 2 unique paths, print the shortest and the second shortest
if len(sorted_ttc_items) >= 2:
    # Print the overall shortest path
    path_tuple, shortest_ttc = sorted_ttc_items[0]
    path = list(path_tuple)
    print("Overall Shortest Path:")
    for node in path:
        print(node)
    print(f"Shortest total TTC: {shortest_ttc}\n")

    # Find and print the second shortest unique path that does not contain "honeypot_reached"
    second_shortest_path = None
    for i in range(1, len(sorted_ttc_items)):
        path_tuple, shortest_ttc = sorted_ttc_items[i]
        if path_tuple != sorted_ttc_items[0][0] and "honeypot_reached" not in path_tuple:
            second_shortest_path = path_tuple
            break

    if second_shortest_path:
        path = list(second_shortest_path)
        print("Second Overall Shortest Path (without honeypot):")
        for node in path:
            print(node)
        print(f"Shortest total TTC: {shortest_ttc}\n")
    else:
        print("No second shortest unique path found without 'honeypot_reached'.")

# If there's only one path, just print that one
elif sorted_ttc_items:
    path_tuple, shortest_ttc = sorted_ttc_items[0]
    path = list(path_tuple)
    print("Overall Shortest Path:")
    for node in path:
        print(node)
    print(f"Shortest total TTC: {shortest_ttc}\n")
else:
    print("No paths found.")

# After all simulations, update the overall shortest path one last time
overall_shortest_path, overall_shortest_ttc = min(shortest_ttc_dict.items(), key=lambda x: x[1])

# Print the overall shortest path and its total TTC
print("The overall shortest path is:", overall_shortest_path)
print("Its total TTC is:", overall_shortest_ttc)

# Print the shortest total TTC for each unique path
for path_tuple, shortest_ttc in shortest_ttc_dict.items():
    path = list(path_tuple)
    print("Path:")
    for node in path:
        print(node)
    print(f"Shortest total TTC: {shortest_ttc}\n")
```

J

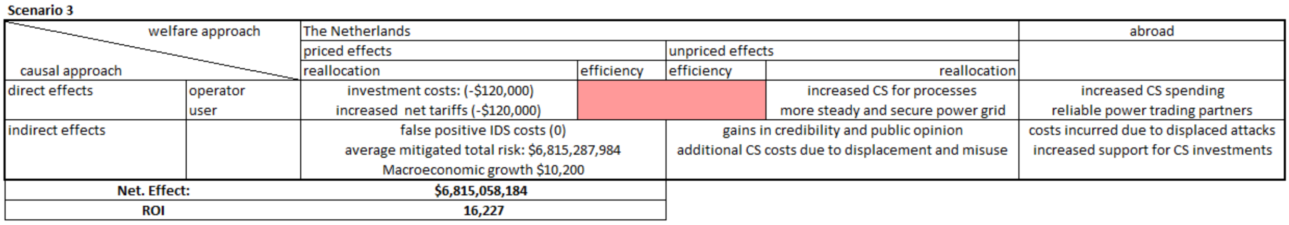
General SCBAs for each scenario



Scenario 2

welfare approach		The Netherlands				abroad
		priced effects		unpriced effects		
causal approach		reallocation	efficiency	efficiency	reallocation	
direct effects	operator user	investment costs: (-$129,000) increased net tariffs (-$129,000)	🟥	🟥	increased CS for processes more steady and secure power grid	increased CS spending reliable power trading partners
indirect effects		false positive IDS costs (-$3,200) average mitigated total risk: $3,719,898,215 Macroeconomic growth $10,965		gains in credibility and public opinion additional CS costs due to displacement and misuse		costs incurred due to displaced attacks increased support for CS investments
Net. Effect:		$3,719,647,980				
ROI		17,618				

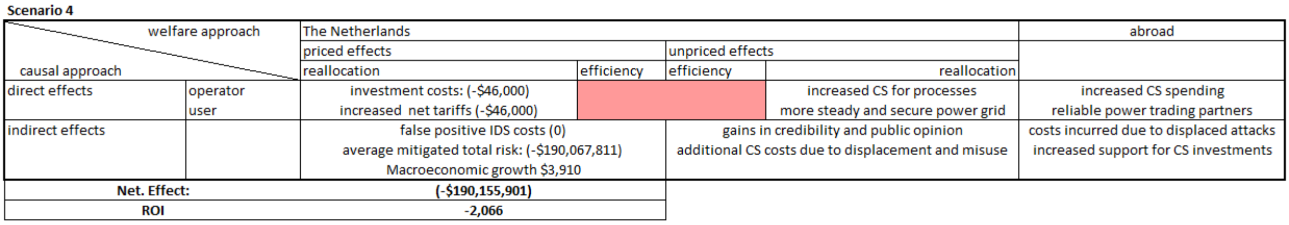
Figure J.1: GSCBA scenario 2



Scenario 3

welfare approach		The Netherlands				abroad
		priced effects		unpriced effects		
causal approach		reallocation	efficiency	efficiency	reallocation	
direct effects	operator user	investment costs: (-$120,000) increased net tariffs (-$120,000)	🟥	🟥	increased CS for processes more steady and secure power grid	increased CS spending reliable power trading partners
indirect effects		false positive IDS costs (0) average mitigated total risk: $6,815,287,984 Macroeconomic growth $10,200		gains in credibility and public opinion additional CS costs due to displacement and misuse		costs incurred due to displaced attacks increased support for CS investments
Net. Effect:		$6,815,058,184				
ROI		16,227				

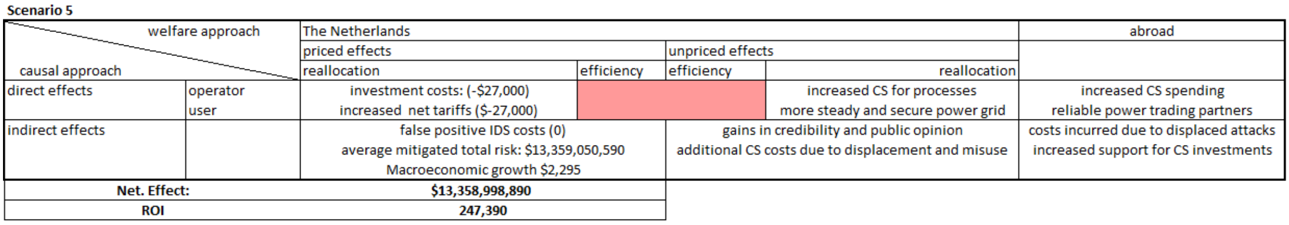
Figure J.2: GSCBA scenario 3



Scenario 4

welfare approach		The Netherlands				abroad
		priced effects		unpriced effects		
causal approach		reallocation	efficiency	efficiency	reallocation	
direct effects	operator user	investment costs: (-$46,000) increased net tariffs (-$46,000)	🟥	🟥	increased CS for processes more steady and secure power grid	increased CS spending reliable power trading partners
indirect effects		false positive IDS costs (0) average mitigated total risk: (-$190,067,811) Macroeconomic growth $3,910		gains in credibility and public opinion additional CS costs due to displacement and misuse		costs incurred due to displaced attacks increased support for CS investments
Net. Effect:		(-$190,155,901)				
ROI		-2,066				

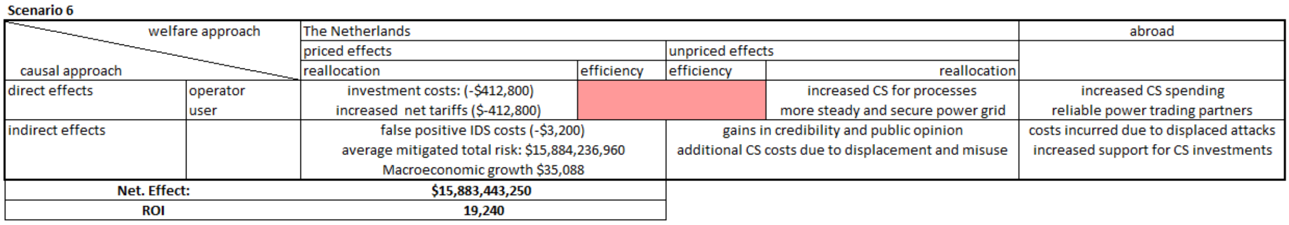
Figure J.3: GSCBA scenario 4



Scenario 5

welfare approach		The Netherlands				abroad
		priced effects		unpriced effects		
causal approach		reallocation	efficiency	efficiency	reallocation	
direct effects	operator user	investment costs: (-$27,000) increased net tariffs ($-27,000)	🟥	🟥	increased CS for processes more steady and secure power grid	increased CS spending reliable power trading partners
indirect effects		false positive IDS costs (0) average mitigated total risk: $13,359,050,590 Macroeconomic growth $2,295		gains in credibility and public opinion additional CS costs due to displacement and misuse		costs incurred due to displaced attacks increased support for CS investments
Net. Effect:		$13,358,998,890				
ROI		247,390				

Figure J.4: GSCBA scenario 5



Scenario 6

welfare approach		The Netherlands				abroad
		priced effects		unpriced effects		
causal approach		reallocation	efficiency	efficiency	reallocation	
direct effects	operator user	investment costs: (-$412,800) increased net tariffs ($-412,800)	🟥	🟥	increased CS for processes more steady and secure power grid	increased CS spending reliable power trading partners
indirect effects		false positive IDS costs (-$3,200) average mitigated total risk: $15,884,236,960 Macroeconomic growth $35,088		gains in credibility and public opinion additional CS costs due to displacement and misuse		costs incurred due to displaced attacks increased support for CS investments
Net. Effect:		$15,883,443,250				
ROI		19,240				

Figure J.5: GSCBA scenario 6