NTNU
Norwegian University of
Science and Technology

Master's thesis

2021

Lars Halvdan Flå

NTNU
Norwegian University of
Science and Technology
Faculty of Information Technology and Electrical
Engineering
Department of Engineering Cybernetics

Master's thesis

Lars Halvdan Flå

Threat Modeling Framework for Smart Grids

February 2021

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of
Science and Technology

Threat Modeling Framework for Smart Grids

Lars Halvdan Flå

Cybernetics and Robotics
Submission date: February 2021
Supervisor: Mary Ann Lundteigen
Co-supervisor: Ravishankar Borgaonkar, SINTEF

Norwegian University of Science and Technology
Department of Engineering Cybernetics

Preface

This master thesis is part of the master’s degree in Cybernetics and Robotics from the Department of Engineering Cybernetics at the Norwegian University of Science and Technology (NTNU). The work was carried out during the fall of 2020 in cooperation with the research institute SINTEF. SINTEF suggested the topic for the thesis and has provided regular support throughout the project. This includes biweekly meetings with a supervisor, extra meetings with additional cyber security experts, a meeting with power grid experts, and meetings with relevant industry actors. The thesis assumes the reader to be a master's student in electrical engineering, with basic knowledge of communication technology.

Trondheim, 01.02.2021

Lars Halvdan Flå

Lars Halvdan Flå

Acknowledgment

I would like to thank several persons for their help during the work with this thesis. Supervisors professor Mary Ann Lundteigen at the Department of Engineering Cybernetics at NTNU and Dr. Ravishankar Borgaonkar at the Department of Software Engineering, Safety and Security in SINTEF Digital for invaluable guidance, advice, and feedback throughout the project. Martin Gilje Jaatun and Inger Anne Marie Tøndel at the Department of Software Engineering, Safety and Security in SINTEF Digital for advice and feedback on the Microsoft Threat Modeling Tool and the threat modeling framework. Merkebu Zenebe Degefa and Santiago Sanchez-Acevedo at the department of Energy Systems in SINTEF Energy Research for help with the smart grid scenario used for the threat modeling. Lastly, I would like to thank SINTEF for proposing and guiding the work with the thesis and for arranging meetings with relevant industry actors.

This work has been supported by CINELDI WP2 - Centre for intelligent electricity distribution, an 8-year Research Centre under the FME-scheme (Centre for Environment-friendly Energy Research, 257626/E20). The authors gratefully acknowledge the financial support from the Research Council of Norway and the CINELDI WP2 partners.

L.H.F

Executive summary

The future power grid, called the smart grid, is expected to include distributed generation, bidirectional flow of power, large scale data gathering and processing, and numerous information and communication technology devices. According to an EU report, the smart grid provides an increased attack surface for adversaries to cause noteworthy disturbance to critical infrastructures. The primary aim of this thesis is to assist in identifying cyber threats in the future power grid infrastructure using threat modeling concepts. In particular, we develop a threat modeling framework[1] for cybersecurity in the smart grid. While developing our framework, we also present a survey of smart grid relevant cyber threats.

Threat modeling is about identifying and mitigating unwanted incidents caused by an attacker. It originates from software security and has seen limited use in complex cyber-physical systems such as the smart grid, involving both Information Technology (IT) and Operation Technology (OT) systems. In this thesis, we investigate the applicability of threat modeling to the complex cyber-physical system that is the smart grid. In general, threat modeling consists of creating a model of the system of interest, analyzing for threats, addressing these threats, and verifying the result. The threat model developed in this thesis uses a model of the data flow of the smart grid. The next step is to analyze each interaction in the model for threats that could lead to Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege types of attacks. These threat categories form the mnemonic STRIDE, which is the name of the threat modeling method.

In this thesis we develop a new generic framework for the smart grid, allowing for automatic detection of relevant cyber threats. The framework is easy to use. It contains modules relevant to the smart grid and provides a drag and drop interface for creating data flow models of arbitrary smart grid use cases. Once the model is created the framework automatically analyzes it and produces a systematically categorized list of relevant cyber threats. Each of the threats in the list applies to an instance of communication between two components in the use case. The developed generic framework and its overall results from the thesis were validated with relevant security experts (from SINTEF and the CINELDI project[2]).

The framework is developed using The Microsoft Threat Modeling Tool (Microsoft TMT). Microsoft TMT may be used to both create new frameworks and to analyze use cases using existing frameworks. The Microsoft TMT is freely available and allows for easy use of the framework developed in this thesis by others. The developed framework can be downloaded from Github.[3] It is our intention that the framework developed in this thesis may be used as a starting point for other smart grid actors demanding a slightly different set of threats or functionality. The framework is designed in such a way that this can be done with ease.

After having created the smart grid framework it is then used to analyze a use case from the smart grid realized in a lab environment. This is done to demonstrate feasibility and usability aspects. The smart grid scenario is provided by SINTEF via the CINELDI project. The scenario relates to the control of a portion of the power grid. A total of 355 threats are identified, and a subset is selected for closer review.

The thesis provides a survey of previously known cyber threats affecting the smart grid. The results of this survey are used as input for the threats included in the framework. The survey identifies and groups threats according to the component they affect and threat category. The result of the survey is presented

[1] Framework and template are used synonymously. Template is the term used by the Microsoft Threat Modeling Tool.

[2] https://www.sintef.no/projectweb/cineldi/

[3] https://github.com/larshfl/MS-TMT-Smart-Grid-Template

in a way that allows it to be used as input for other cybersecurity projects in the smart grid, not necessarily related to threat modeling.

The results obtained from analyzing the smart grid scenario indicate that the developed framework can be used for threat modeling in the smart grid. We believe that it has several advantages. Firstly, it can be used for an initial analysis performed by people not possessing expert knowledge in cyber-security. The idea is that power grid professionals can look at the potential consequences of an attack and classify threats into different levels. This classification can be made to depend on the grid operator's security standards and tolerance for risk. All or a subset of the threats may then be forwarded to smart grid cyber-security professionals for closer review. This may, for example, be challenging threats that have not yet been addressed in the design phase.

Secondly, regardless of the security knowledge of the user, the framework helps ensure that threats are not forgotten and encourages reflection on how threats may arise in the smart grid after deployment. The framework generates a structured report that contains all identified threats, where they arise, how they have been evaluated, how they potentially are mitigated, and justification for the choices made.

Several disadvantages are identified and discussed as well. The number of generated threats quickly grow large even for a scenario of moderate complexity. Reviewing the threat modeling results provided by the framework may be a laborious process due to the number of threats.

Sammendrag

Det fremtidige strømnettet, ofte kalt det smarte strømnettet, er forventet å inkludere distribuerte strømproduksjon, toveis flyt av strøm, omfattende innsamling og prosessering av data og et stort antall enheter relatert til informasjons og kommunikasjonsteknologi (IKT). Ifølge en rapport fra EU medfører det smarte strømnettet en økt angrepsflate som kan utnyttes til å angripe kritisk infrastruktur. Hovedmålet til denne oppgaven er å bidra til å identifisere cybertrusler i fremtidens smarte strømnett ved hjelp av trusselmodellering. For å oppnå dette utvikler vi et rammeverk[4] for cybersikkerhet i smart grid. Vi presenterer også en oversikt over relevante cybertrusler mot det smarte strømnettet.

Trusselmodellering handler om å indentifisere og forhindre uønskede hendelser forårsaket av en angriper. Teknikken har bakgrunn i sikkerhet for programvare og har blitt lite brukt i komplekse cyber-fysiske systemer slik som det smarte strømnettet som involverer både informasjons og kommunikasjonsteknologi (IKT) og operasjonell teknologi (OT). I denne oppgaven undersøker vi hvorvidt trusselmodellering er egnet for det komplekse cyber-fysiske systemet som strømnettet er. Trusselmodellering handler generelt om å lage en modell av systemet, analyserer modellen for trusler, adressere disse truslene og verifisere resultatet. Trusselmodellen som utvikles i denne oppgaven bruker en modell av dataflyten i strømnettet. Neste steg er å analysere interaksjonene i modellen for trussler som kan føre til Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service og Elevation of Privilege i strømnettet. Disse trusselkategoriene former mnemonicen STRIDE, som også er navnet på metoden.

I denne oppgaven utvikler vi et generisk rammeverk for det smarte strømnettet som muliggjør automatisk deteksjon av relevante cybertrusler. Rammeverket er enkelt å bruke. Det inneholder moduler som er relevant for det smarte strømnettet og tilbyr et klikk og dra basert brukergrensesnitt for å modellere vilkårlige scenario i det smarte strømnettet. Deretter analyserer rammeverket modellen produserer en systematisk liste over cybertrusler. Hver av truslene relaterer til kommunikasjon mellom to enheter i scenarioet. Det utviklede rammeverket er validert ved hjelp av sikkerhetseksperter fra SINTEF og CINELDI prosjektet[5].

Rammeverket er utviklet ved bruk av Microsoft Threat Modeling Tool (Microsoft TMT). Dette verktøyet kan brukes til å lage nye rammeverk o analysere scenario ved bruk av eksisterende rammeverk. Microsoft TMT er fritt tilgjengelig og gjør det enkelt for andre å benytte rammeverket utviklet i denne oppgaven. Rammeverket kan lastes ned fra Github[6]. Det er vår intensjon at rammeverket som har blitt utviklet skal kunne tjene som et utgangspunkt for andre aktører relatert til det smarte strømnettet som krever en noe annen funksjonalitet eller et annet sett med trusler. Rammeverket er utviklet på en slik måte at dette lett lar seg gjøre.

Etter at vi har utviklet rammeverket bruker vi det til å analysere et scenario fra det smarte strømnettet som for tiden er realisert i et laboratorium. Dette gjøres for å demonstrere nyttigheten av rammeverket. Scenarioet er lever av SINTEF via CINELDI prosjektet. Scenarioet relaterer til kontroll av en del av det smarte strømnettet, og totalt 355 trusler identifiseres. En liten del av disse velges ut for nærmere analyse.

Denne oppgaven inneholder en studie på tidligere kjente cybertrusler mot det smarte strømnettet. Resultatet av denne studien er brukt som input til de truslene som er lagt til i rammeverket. Studien identifiserer og grupperer trusler etter hvilke deler av det smarte strømnettet de påvirker, og hvilken kategori de tilhører. Resultatene fra studien er presentert på slik en måte at de kan brukes som input til

[4] Rammeverk og template benyttes synonymt. Microsoft Threat Modeling Tool bruker begrepet template.

[5] https://www.sintef.no/projectweb/cineldi/

[6] https://github.com/larshfl/MS-TMT-Smart-Grid-Template

andre prosjekter som omhandler cybersikkerhet i det smarte strømnettet. Dette inkluderer prosjekter som ikke er direkte relatert til trusselmodellering.

Resultatene fra analysen av strømnett-scenarioet indikerer at det utviklede rammeverket kan brukes til å trusselmodulering strømnettet. Vi mener rammeverket har en rekke fordeler. For det første kan det bli brukt av mennesker som ikke besitter spisskompetanse på cybersikkerhet til å gjennomføre en første analyse av et system. Tanken er at aktører i strømnettet kan evaluere konsekvenser av et angrep og klassifisere trusler inn i ulike nivåer. Denne klassifiseringen kan avhenge av aktørens toleranse for risiko og sikkerhetsstandarder. Alle eller en del av truslene kan videresendes for nærmere ettersyn av personer med spisskompetanse på cybersikkerhet i det smarte strømnettet. Dette kan for eksempel være utfordrende trusler i designfasen som enda ikke har blitt håndtert.

For det andre kan rammeverket bidra til at trusler ikke blir utelatt fra analysen. Rammeverket bidrar videre til å fremme refleksjon over hvordan trusler kan oppstå i det smarte strømnettet. Rammeverket genererer videre en strukturert rapport over alle identifiserte trusler, hvor i systemet de oppstår, hvordan de har blitt evaluert, hvordan de håndtert og begrunnelse for valgene som er tatt.

Flere ulemper er i tillegg identifisert og diskutert. Antallet identifiserte trusler vokser raskt selv for scenario av moderat kompleksitet. Evaluering av truslene som rammeverket identifiserer kan være en omfattende prosess på grunn av det høye antallet trusler.

Contents

1 Chapter 1 Introduction

1.1 Background

The smart grid may become the most complex cyber-physical system created, merging the discipline of communication and information technology with that of electrical power engineering. Cybersecurity in this domain is of great interest for several reasons. The power grid is a critical infrastructure and a necessity for modern life. Consequently, the grid has strict requirements on power availability. Due to its immense size and the increasing usage of Information and Communication Technology (ICT) in the grid, the attack surface is equally large. An EU report from ENISA by Moulinos and Mattioli (2016, p. 6) argue that the increasing use of bidirectional communication in the smart grid leads to an increase in the power grid attack surface. The ICT components in the smart grid transmit customer data, consumption data, and operator control commands, amongst others. Attacks on this communication are claimed to have the potential to cause a blackout, device malfunction, and violation of privacy.

The 2015 and 2016 cyber-attacks on the Ukraine power grid demonstrated the vulnerability of the power grid. According to an alert by ICS-CERT U.S Department of Homeland Security (2016), the 2015 attack compromised three distribution companies and caused a blackout[7] affecting 225 000 customers. According to Slowik (2019), the 2016 attack was less severe with regards to impact but indicated an increase in the attacker's ambitions. Slowik argues that a more widespread blackout than seen in the 2015 attack, along with potential physical destruction of equipment, may have been the original objective of the attack. These attacks on the Ukrainian power grid can be viewed in a larger context of increases in cyber-attacks on industrial control systems. Examples of such attacks include the 2010 Stuxnet attack, discussed by Falliere et al. (2011), and the 2017 Triton[8] attack, discussed by Johnson et al. (2017). The Stuxnet attack is believed to have targeted Iranian uranium enrichment centrifuges. The Triton attack is believed to have targeted the safety instrumented system in a Saudi Arabia petrochemical plant.

To protect the future energy grid from attackers, potential threats to the energy grid must be identified before the deployment and addressed. Due to the size and complexity of the grid, these threats are expected to be numerous if left unmitigated. Potential threats may target the availability of power, the generation of power, the trade of power, the operation of grid ICT infrastructure, or the large amounts of sensitive information generated about the consumers in the grid. To further increase the difficulty, the threats emerge in between the disciplines of power engineering and information technology. Few of the actors in the grid can be expected to manage both these disciplines.

This thesis develops a smart grid framework[9] for the Microsoft Threat Modelling Tool. The framework allows asset owners to model use cases in the smart grid. The modeling process enumerates potential threats to the smart grid, provides an environment to systematically treat and classify discovered threats and provides a framework for creating more extensive and specialized templates in the future.

[7] A blackout is the loss of power in an area.

[8] Triton is also known under the names Trisis and HatMan

[9] In the following chapters the word “template” is used instead of framework, as this is more consistent with the Microsoft TMT.

Threat modeling is a technique that originates from software, and it has not seen wide application in cyber-physical systems, and particularly not for industrial control systems. We are, at the time of writing, not aware of any attempt to automatically identify threats in the smart grid.

There exist some attempts to apply the Microsoft TMT and STRIDE in other cyber-physical domains. The cybersecurity firm nccgroup develops a template[10] for automotive threat modeling. Microsoft creates three templates[11] for the domains of Azure Cloud services, Medical Devices, and a default general IT template. Khan et al. (2017) develop a five-stage methodology for applying STRIDE to cyber-physical systems.

The security of the smart grid is investigated by other techniques. Olayemi et al. (2017) investigate the threats to Smart Home solutions using STRIDE combined with a method for assessing the risk of the threats. Tøndel et al. (2013) investigate the threats to an AMI configuration using STRIDE. Jiang et al. (2014) investigate the threat of energy theft using attack trees. Liu et al. (2015) use Petri Nets to analyze threats to communication and information in a smart meter. Cardenas et al. (2009) propose and apply a new method for investigating threats to Supervisory Control and Data Acquisition (SCADA). Ding et al. (2017) propose and apply a new method for analyzing threats to critical infrastructure. Suleiman et al. (2015) analyze the threats to the Smart Grid using Security Quality Requirements Engineering.

Various techniques are employed to investigate the security of the smart grid. Common to all is that they are manual procedures. A tool for modeling and automatic identification of threats does not exist. To our knowledge, at the time of writing, a framework for prioritizing threats in the smart grid is also absent. Such a tool can improve the process by ensuring that threats are not forgotten and improve efficiency by automatically generating relevant threats.

1.2 Objective

The objective of this thesis is to assist in improving cybersecurity management by developing a threat modeling framework for the smart grid. This objective is reached through the six tasks outlined below.

1. Identify and describe the ICT components of the smart grid.
2. Identify and describe cyber threats against ICT components in the smart grid.
3. Identify and describe threat modeling methods and describe some examples of widely used tools.
4. Study of the Microsoft Threat Modelling Tool and techniques.
5. Develop a generic framework for the smart grid.
6. Develop and apply a threat model for a smart grid use case.

1.3 Approach

The approach for solving the objectives outlined above is shown in Figure 1. The color coding corresponds to the different tasks. The figure details what input went into the different tasks and how the different tasks contribute to achieving the main objective.

[10] https://github.com/nccgroup/The_Automotive_Threat_Modeling_Template

[11] https://github.com/microsoft/threat-modeling-templates

The first task was achieved by studying articles mainly from IEEEXplore and Elsevier and by studying the NIST framework. The second task was achieved by consulting books, articles, and standards.

The third task was achieved mostly through working with objective two, four, and five. Relevant books on threat modeling studied for objective two provided a theoretical introduction to the relevant threat modeling technique. The work on tasks four and five provided the necessary practical experience with the Microsoft TMT.

The fourth task was to create a generic framework for the smart grid. This was completed mainly by including relevant stencils and threats into the framework. The stencils were added based on the components identified in task one, input from relevant experts in SINTEF Energy, and from the use case provided by SINTEF. The threats were derived from literature, existing threat modeling framework, and previous cyber-attacks. Necessary information on stencils and threat categories needed to structure the findings into a framework was provided by the second task.

The final task was achieved by applying the framework to a use case provided by the Department of Software Engineering, Safety, and Security in SINTEF Digital. Necessary information on threat modeling was provided by task two.

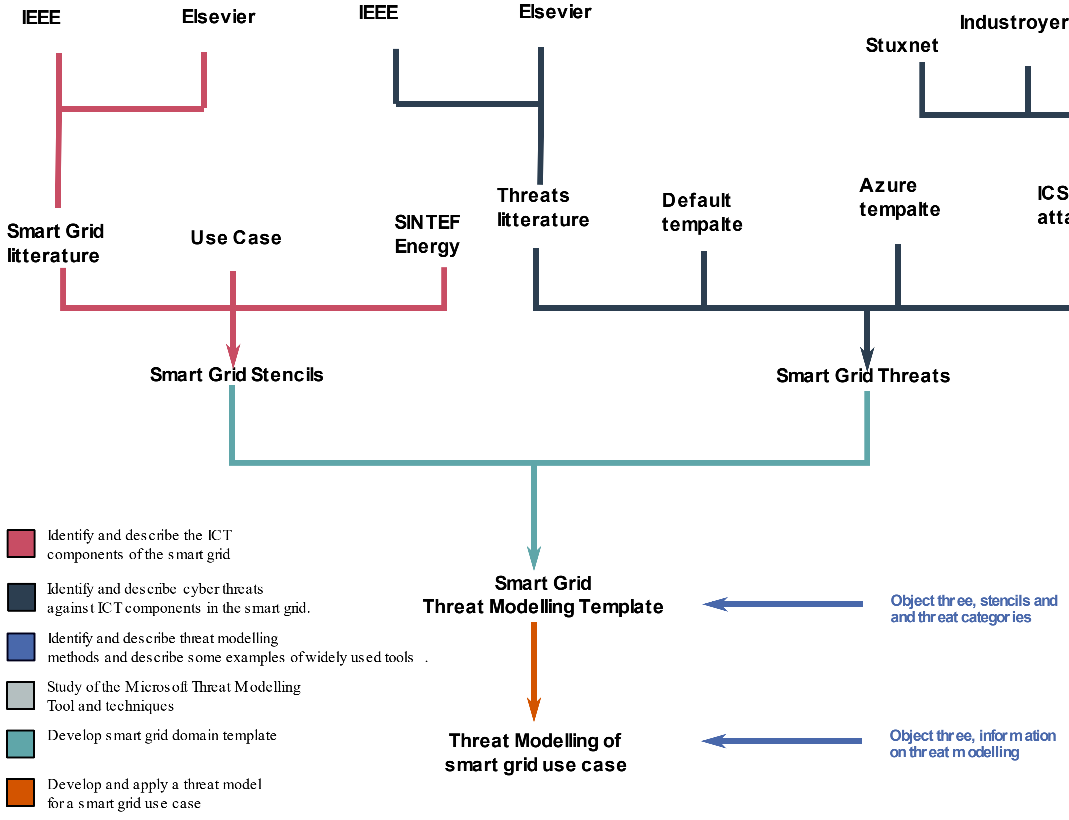

IEEE
Elsevier
IEEE
Elsevier
Industroyer
Stuxnet
Triton
Threats litterature
Default tempalte
Azure tempalte
ICS attacks
Smart Grid litterature
Use Case
SINTEF Energy
Smart Grid Stencils
Smart Grid Threats
Identify and describe the ICT components of the smart grid
Identify and describe cyber threats against ICT components in the smart grid.
Identify and describe threat modelling methods and describe some examples of widely used tools .
Study of the Microsoft Threat Modelling Tool and techniques
Develop smart grid domain template
Develop and apply a threat model for a smart grid use case
Smart Grid Threat Modelling Template
Object three, stencils and and threat categories
Threat Modelling of smart grid use case
Object three, information on threat modelling

Figure 1: Thesis approach to solving the objective

1.4 Contributions

This thesis develops a framework for threat modeling in the smart grid domain. The framework consists of stencils representing various ICT and power grid systems in the smart grid, communication between these systems, and relevant smart grid threats. The framework and the accompanying tool, Microsoft TMT, offers an easy way of both modeling smart grid use cases and identifying cyber threats to the smart grid before the actual deployment.

The framework enables asset owners (for example, the energy company Statnett) and other actors involved in the smart grid ecosystem to easily and quickly generate relevant cyber threats for the smart grid infrastructure. Use of the framework only requires knowledge of the design of the use case with regards to ICT components and how they communicate. With this knowledge, a model can be created, and threats automatically generated using the framework developed in this thesis. Initial threat modeling of a use case can be conducted by actors not specialized in cybersecurity. In the initial phase, power grid specialists may filter out less critical threats based on the description of the threat combined with knowledge of grid operations and the asset owners' objectives. Less obvious threats may be passed on to security professionals.

Additionally, the framework provides a starting point that can be extended for asset owners and use cases in need of different capabilities. The framework is made publicly available and can easily be changed and extended.

Lastly, the thesis provides a literature study of cyber threats against ICT components in the smart grid. The cyber threats are presented in section 4.3 and summarized in tables Table 4 to Table 14. The threats have been used in this thesis as input for the threats created for our smart grid framework. The literature study can be used as input or inspiration for other projects related to cybersecurity in the smart grid, not necessarily related to threat modeling.

1.5 Limitations

The created framework focuses on the transmission, distribution, and operation domains of the smart grid. Additional domains are the market domain, the generation domain, the customer domain, and the service provider domain. Threats originating inside these domains or in the interactions with these domains are not thoroughly covered.

The generated threats are based on identified threats in literature, existing frameworks covering other domains, such as cloud infrastructure, and previous cyber-attacks on industrial control systems. Included threats are in most cases generalized and subject to smart grid assumptions. This was done to limit the possibility of missing threats when performing the threat modeling.

Identified threats are a subset of the total number of existing threats. Inevitably there are threats existing in the smart grid that are not covered. This holds particularly true due to the complex size and structure of the smart grid.

Mitigation of identified threats and validation of the threat model is considered part of the threat modeling process, as discussed in Chapter 3. Threat mitigation is considered out of scope for this thesis. The threat modeling process and the threat model have not been validated in a formal way. The benefits of the framework have been evaluated with smart grid industry experts.

1.6 Structure of report

Chapter 2 introduces the smart grid and its ICT components. Chapter 3 introduces and discusses threat different threat modeling techniques. Chapter 4 outlines the security requirements of the smart grid and presents cyber threats identified in the literature. With input from Chapters 2, 3, and 4 on components, threats, requirements, and threat modeling, Chapter 5 describes the framework we have developed. Chapter 6 shows the created framework applied to a specific use case. Chapter 7 discusses the applicability for threat modeling on the smart grid, the STRIDE technique, the created framework, and the results from Chapter 6. Chapter 8 contains conclusion, discussion, and recommendations for further work.

Please note that this thesis uses an explicit reference style as opposed to an implicit style. For instance, with regards to smart meters, “ Yan et al. (2012) describe a smart meter as …“ is used instead of “A smart meter is described as … Yan et al. (2012) ”.

The terms “template” and “framework” are used synonymously. Framework is mainly used in Chapter 1, and template is used in the remaining chapters. Framework is used as we believe it is more relatable, and that the developed template essentially constitutes a framework. Template is used to remain consistent with Microsoft TMT terminology.

This thesis makes extensive use of the terms stencil and element. A stencil refers to either a process, data store, data flow, or trust boundary inside the Microsoft TMT. Examples of stencils can be seen in Appendix B. An element refers to the same elements in the context of general Data Flow Diagrams, as shown in Table 15, not necessarily related to Microsoft TMT.

2 Chapter 2

The Smart Grid

This chapter identifies and discusses the ICT components of the smart grid. The components are related to the smart grid with the help of the conceptual model described below.

2.1 Smart Grid Domain overview

This section describes the conceptual domain first introduced by NIST in 2010. This section is based on a draft by Gopstein et al. (2020, pp. 97 - 115), which will be used for version 4.0 of the document. The model gives a high-level description of the smart grid, as shown in Figure 2. The model is made up of seven domains. Some of the domains have subdomains.

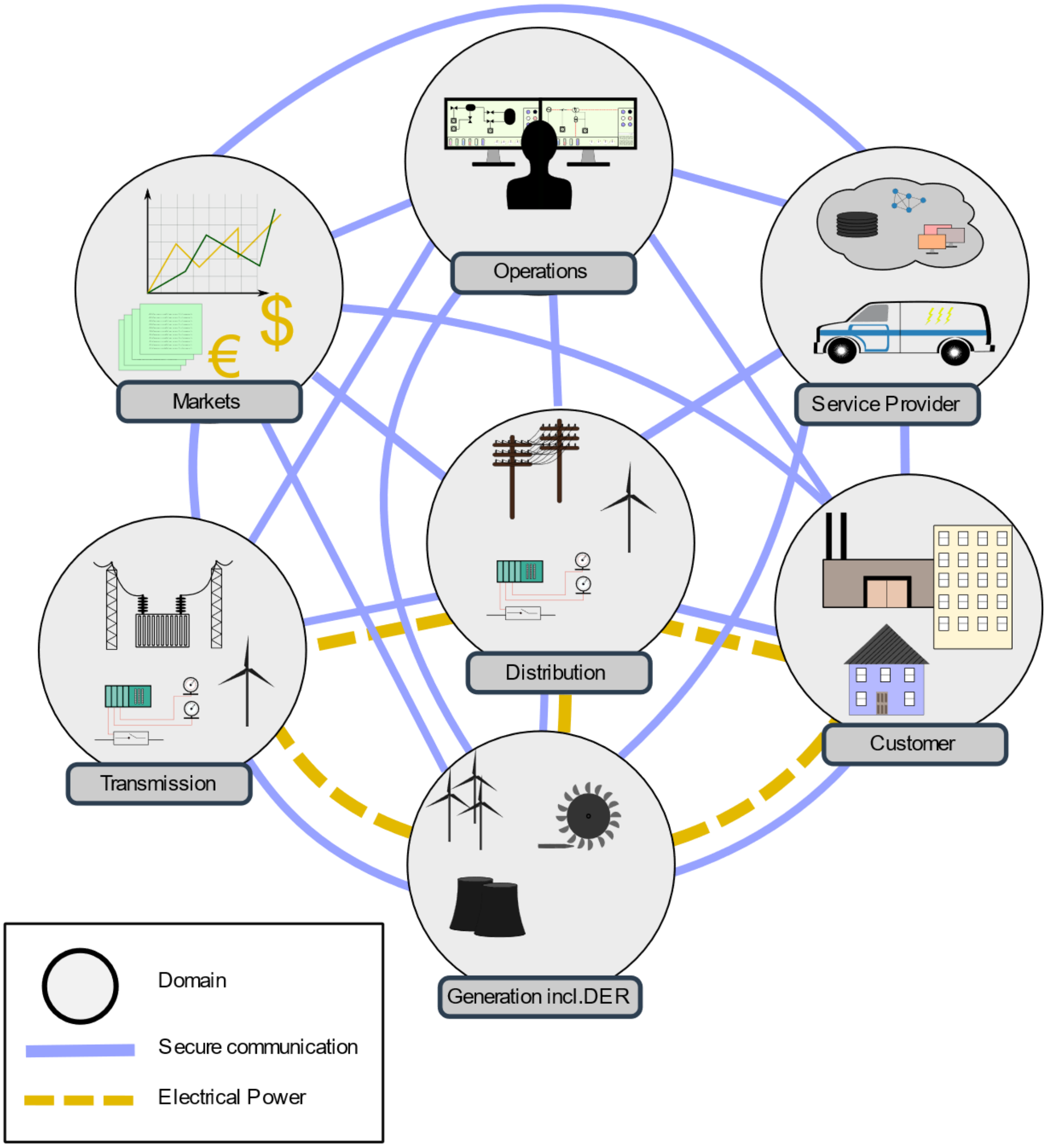


Figure 2 Smart Grid Conceptual Model. Gopstein et al. (2020).

2.1.1 Distribution

The distribution domain is what connects the transmission domain with the customer domain. The domain also contains substations and components to control, measure, protect, record, stabilize and optimize the power flow. With the introduction of the smart grid, DERs can also be placed in this domain.

The distribution domain communicates with the market domain, transmission domain, operation domain, and customer domain. Communication with the operation domain happens in real-time.

The introduction of the market domain affects consumption and generation through communication with the distribution domain. Compared to the traditional grid, the distribution domain in the smart grid will have greater sensing and control capabilities.

2.1.2 Transmission

The transmission domain connects the generation domain with the distribution domain. This domain typically includes several substations. As with the distribution domain, the transmission domain includes components to control, measure, protect, record, stabilize and optimize the power flow. DERs can also be found in this domain. The domain is typically controlled by the operator through a SCADA system.

The transmission domain communicates with the operations domain, markets domain, distribution domain, and generation domain. Communication with the market domains can be used by the latter to procure energy. Their energy transfer is scheduled and operated from the operations domain.

2.1.3 Generation including DER

This domain includes a wide variety of sources producing electrical energy in the form of bulk generation or Distributed Energy Resources (DERs). Bulk generation is used for generation of more than 300 MW. DERs are smaller generation, for instance smaller solar and wind installations. DERs are included in this domain but can also be found in other domains.

The generation including DER domain communicates with all other domains in the model. The distribution and transmission domain receives information on key performance and quality of service. A lack of generation capacity can be addressed either through the Operations or Markets domains.

2.1.4 Operations domain

The operations domain is responsible for the functioning of the power grid. The domain communicates with all other domains. A substantial amount of communication can also be found within the domain itself. The operations domain has a sub domain, the network operations domain. It is worth noting that the smart grid enables some of the traditional tasks performed by the operations domain to be transferred to the market domain and service provider domain.

2.1.5 Service Provider Domain

The service domain provides services to the actors in the smart grid. The service provider role can be occupied either by existing or new parties.

The service provider domain will communicate with the operations, distribution, generation including DER, customer and markets domains. Communication with the operations domain is important to the control and situational awareness of the grid. Communication with the markets and customer domains is important to enable the emergence of new and innovative services. The conceptual model claims that the interfaces in the service provider domain will have to support a wide range of network technologies.

2.1.6 Market Domain

The markets domain provides a market for grid assets and services. The domain communicates with all other domains in the model. Communication with the market domain can facilitate different price models and allow the consumers to participate in the market in a more active way.

The market domain will enable efficient matching of production and consumption of energy. This may partially happen through DER aggregation, as many DERs are believed to be too small to take part in the market individually.

2.1.7 Customer Domain

The customer domain is where the electricity is consumed. With the introduction of the smart grid, the customer domain will also be a domain where electricity is managed and generated. The customer domain communicates with the market, service provider, operation, distribution, and generation including DER domains. The customer domain has three sub domains: industrial, commercial, and residential.

2.2 ICT components

2.2.1 Smart meters and advanced metering infrastructure

It is widely agreed upon that smart meters and advanced metering infrastructures will play an important role in the smart grid. The advanced metering infrastructure (AMI) is one of the areas being prioritized by Greer et al. (2014) in the third version of the NIST Framework and Roadmap for Smart grid interoperability. AMI is the combination of smart meters, the communication link, and a management system for the metering data. A great number of smart meters have already been deployed in various countries.

According to Greer et al. (2014), AMI consist of the hardware and software for communication and data management needed to enable near real-time, two-way communication between smart meters and business utilities. AMI can perform a range of tasks. According to Mak and Farah (2012), the tasks of the AMI can be grouped into four categories. The categories are customer service and demand management, optimization and reliability service, detection of unbalance and asset management, and monitoring of power quality.

Smart Meter

Yan et al. (2012) describe a smart meter as a device capable of measuring the power consumed with greater accuracy than before and with the capability to send and receive data from other actors in the smart grid. Barai et al. (2015) describe smart meters as a combination of hardware, software, and calibration systems. Among the components are real-time clocks, metering system-on-chip, a data communication module, memory, and tamper detection

Meter Data Management System

The Meter Data management system collects the data from the distributed smart meters in the grid. According to Barai et al. (2015), various services may use the data stored in the meter data management system. Examples of such services are billing, demand management, and demand response.

AMI Architecture

Barai et al. (2015) envision an architecture where the smart meters communicate with a data concentrator via the neighborhood area network. Petruševski et al. (2014) propose an architecture where smart meter data is transported via two-layered data concentrators. A local meter concentrator (LMC) collects data from several smart meters and forwards it to a transformer meter concentrator (TMC). The TMC is placed on the substation level and aggregates data from several LMCs. Karimi and Namboodiri (2013) propose an architecture where Wi-Fi routers are placed along the electricity poles, forming a communication chain between the smart meters and the data management unit. One weakness of the architecture is that it includes a substantial number of nodes. This will negatively affect the reliability. The latency may also be unacceptable for real-time applications. An experimental setup of 10 km with 100 nodes showed a latency of 4-8 seconds. Nielsen et al. (2017) base their work on an architecture where smart meters and PMUs send data to a joint data management platform. Relevant applications can then subscribe to the data they need.

Mak and Farah (2012) propose an architecture where substation SCADA and AMI are joined with the intent of optimizing the power distribution. The smart meters function as sensors for the SCADA. In this architecture, AMI and Smart meters are separated. The AMI facilitates communication from the substation SCADA and the smart meters to the control station, customer service, and database management system. The database management system provides data to both the control and the customer service units. Smart meters and sensors are connected through an intelligent device, offering an interface to the communications infrastructure.

Protocols

A variety of protocols and standards can be utilized to realize the communication in the AMI. Karimi and Namboodiri (2013) compare WiMAX, Wi-Fi, ZigBee, and GSM/UMTS as backhaul technologies. Wi-Fi was evaluated as the best suited based on criteria such as cost, range, flexibility, and robustness. A weakness of the article is that more modern cellular technologies such as 4G and 5G are not included. Petruševski et al. (2014) present various possible technologies for the LMC and TMC based architecture. Communication between the smart meter and LMC can take place via UART, EURIDIS, or RS485. Communication between the LMCs and TMC can utilize 3G, B-PLC, or GPRS. xDSL, 3G, B-PLC, and FO can be used between the TMC and the data management system. Barai et al. (2015) list technologies such as RF communication, RS485, and PLC for smart meter to data concentrator communication. Proposed technologies for data concentrator to Meter Data Management System (MDMS) communication are Ethernet, Wi-Fi, cellular, cable, and fiber.

On higher levels of the communication stack, other protocols are used. According to Barai et al. (2015), the ANSI C12.22 standard defines the communication between the smart meters and the data management unit in North America.

2.2.2 Communication medium

Power Line Communication

Power line communication (PLC) uses the existing utility cables as the medium for data transfer. The economic aspects of using already existing power lines are the main advantage of power line communication. Ancillotti et al. (2013) describe two classes of power line communication, Narrowband PLC (NB-PLC) and Broadband PLC (BB-PLC). NB-PLC is reported to have a bandwidth up to 500 Kbps, while BB-PLC is reported to have a bandwidth of up to 200 Mbps. The higher frequencies used in BB-PLC reduces coverage and reliability and are mostly considered for in-home usage. According to Sayed and Gabbar (2017), PLC is mostly used for protective purposes in transmission lines.

A drawback is the limited bandwidth compared to other mediums. Karimi and Namboodiri (2013) report a bandwidth of 11 Kbps. Other potential problems illustrated by the authors include noise, attenuation of the signal, and interference with components in the grid, such as voltage regulators and reclosures. Another significant disadvantage is that a loss of power will cause a loss of communication.

Wired connection

Karimi and Namboodiri (2013) claim wired alternatives and fiber to be less advantageous. Copper wires are believed to have problems with interference and attenuation. Another disadvantage pointed to by Ancillotti et al. (2013) is the cost of paying a fee to the telecommunication company maintaining the network.

According to Ancillotti et al. (2013), fiber is used for connecting substations due to their high bandwidth and resilience against electrical interference. Fiber to the customers is claimed to be expensive by Karimi and Namboodiri (2013). Karimi and Namboodiri claim that both PLC and wired connections will suffer from lost communication in case electrical poles go down. This is certainly true for PLC but only applies to wired alternatives if they use the electrical poles. The simultaneous loss of communication and power is undesirable, as communication often is used to restore the grid to normal operation.

Wireless

Ancillotti et al. (2013) have reviewed Wi-Fi, WiMAX, and 3G/4G. Advantages of Wi-Fi include high data rate and unlicensed frequency spectrum. Disadvantages include a limited range of 300 meters. WiMAX has a speed of 100 Mbps and a range of 7-10 kilometers. Among the advantages of WiMAX is that it can support thousands of individual users. Among the disadvantages is that it operates in the licensed spectrum. 3G/4G is a cellular technology that offers a wide range and high speed. Utility companies have already utilized previous generations of cellular technology for SCADA advanced metering reading.

Karimi and Namboodiri (2013) advocate for the use of wireless communication. Among the advantages highlighted in the article is the possibility of introducing redundancy without extra cost. Flerchinger et al. (2018) claims cost, right of way, location, and deploy time to be among the main advantages of wireless over wired alternatives. Disadvantages of wireless communication include interference. This is a problem that can be reduced but not eliminated by using licensed spectrums. Nielsen et al. (2017) simulate delays of PMUs and smart meters in a 4G/LTE-based architecture. In the architecture, smart meters and PMUs in the distribution domain sends data to a data-sharing platform. Applications can then receive relevant data from this platform. The article presents two bottlenecks for smart grid sensors in the existing cellular network. When many devices would like to connect to the 4G network, connection attempts may collide, resulting in a failed attempt. Many devices attempting to send data at the same time causes competition for the limited uplink capacity, resulting in a delay. These bottlenecks can be remedied by acquiring more LTE capacity, but this might not be economically viable. Cosovic et al. (2017) argue that 4G does not provide sufficient services regarding reliability and latency for PMU communication.

Flerchinger et al. (2018) conduct an experiment with five PMUs sending data to a Phasor Data Concentrator in the distribution domain. Serial radio and 3G were compared. The test showed that serial radio had lower latency and higher reliability but that 3G could provide higher data rates.

In their discussion of 5G and state estimation in the grid, Cosovic et al. (2017) propose an architecture based on 5G base stations. In this architecture, both RTUs, PMUs, and smart meters communicate through base stations. RTUs and PMUs communicate using ultra-reliable low latency communication (URLLC) services due to strict requirements on reliability and latency. Smart meters communicate using massive Machine Type Communication services, for instance, using Narrowband-IoT. The article assumes that smart metering data has been sent through data aggregation units when delivered. The architecture includes Mobile Cloud Computing (MCC) reachable via the internet, and Mobile Edge Computing (MEC) located in the vicinity of the base stations.

2.2.3 Smart grid monitoring and controlling

This section discusses the monitoring and control of the smart grid. This process is performed by the utility company or power grid operators. Consequently, the involved domains in the conceptual model are Generation including DER, Transmission, Operations, and Distribution domain.

SCADA systems are used for the central control and monitoring of industrial systems, typically using distributed sensors, actuators, and programable logic. An example SCADA system is illustrated in Figure 3. Sallam and Malik (2011) divide the SCADA system for the electrical grid into four components. The components are instrumentation, remote station, communication networks, and master terminal unit (MTU). The plant or equipment in the system is typically controlled from a central control room, receiving data and sending commands across a communication network. At the other end of the network are remote stations. These stations are connected to sensors and actuators, connected to the equipment under control.

Instrumentation: instrumented components are either sensors or actuators. Sensors transform a physical quantity into an analog or digital signal readable to a computer. Actuators transform an analog or digital signal into a desired action on the controlled equipment. Sensors and actuators are typically connected to Programmable Logical Controllers (PLC) or Remote Terminal Units (RTU).

Remote stations: Remote stations are typically PLCs or RTUs. Both components typically communicate with the MTU. According to Sallam and Malik (2011), the functions of the remote stations is twofold. First, to gather data from its sensors and transmit this to the MTU, and secondly, to receive and apply commands from the MTU. Particularly PLCs are in addition capable of issuing commands based on inputs and their programmed logic without interference from the MTU. Remote stations can be configured to communicate in between themselves.

According to Bentarzi et al. (2018), traditional SCADA systems in the smart grid receive the majority of their measurements from RTUs. With the transition to the smart grid, Phasor Measurement Units (PMUs) might take over this role.

Communication networks: enable the communication between remote stations and the MTU. A variety of communication mediums might be utilized, including wireless, wired, and fiber. Various industrial protocols are typically used for communication. Examples include the IEC 61580 standard for substation automation.

Control center: the control center is the top entity of the SCADA System. The MTU receives information from the distributed devices in the SCADA System and makes the information accessible to components and operators in the control center. The HMI provides an interface between the system and operators. The data historian is a database for historical data, storing data for later analysis and use.

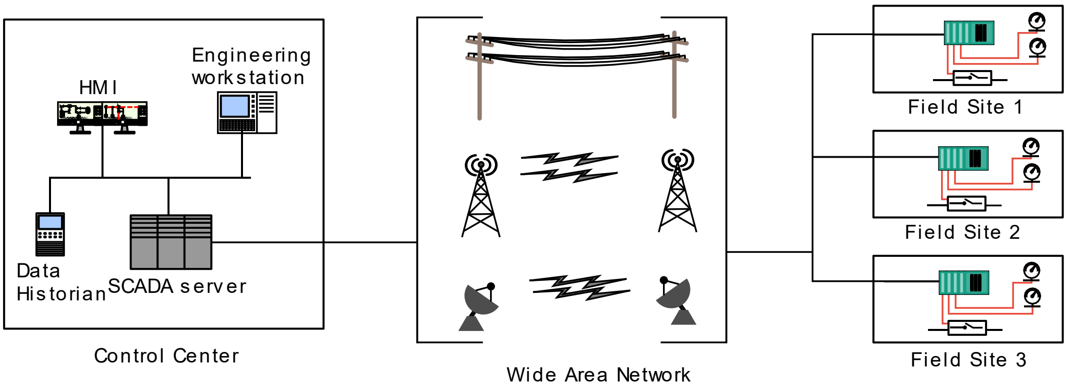


Figure 3: Example of SCADA system in the power grid. Bentarzi et al. (2018).

According to Dhend and Chile (2015), SCADA is typically not used in substations in the distribution domain, but the authors advocate for the necessity of extending SCADA capabilities into this domain. Such an expansion of the SCADA system is believed to improve efficiency and reliability. Extending the SCADA capabilities can be done by utilizing the sensing capabilities and two-way communication of the smart meters deployed in the customer domain. In this way, outage detection and management can be performed. On the other side, Sallam and Malik (2011) claim that SCADA rarely is cost-efficient at the substation and feeder level in the distribution domain. With the technological advancement, the rise of the Internet of Things (IoT), and the vision of a more integrated power grid, this may very well change in the future.

Tom and Sankaranarayanan (2017) discuss the use of the Internet of Things in the distribution part of the power grid. The article divides sensing capabilities in the distribution domain into three categories. Smart meters located in the customer domain, line sensors for voltage and current, and intelligent electronic devices (IEDs). The IEDs can measure parameters such as temperature, loading, and power. One area of application for the IEDs are transformers located in the grid. 6LoWAN is proposed as a technology for smart meters and IEDs. Field mounted routers function as 6LoWAn gateways to the cloud through networks like 3G/4G. In addition, the routers have support for distributed data analytics.

According to Sayed and Gabbar (2017), the integration of SCADA and smart grid enables information from, and control of, the whole electrical grid. This includes power generation, where SCADA is claimed to have many applications. The applications include monitoring of speed, frequency, switches, and protective relays, control of active and reactive power, voltage and frequency load scheduling, and possibly monitoring of weather for solar and wind plants. SCADA systems are also utilized in the transmission domain, amongst others, for real-time monitoring and control, load shedding and load restoration, power control algorithms, and improving power quality.

2.2.4 Wind Turbines

This section is based on a book by Sayed and Gabbar (2017). SCADA is useful in wind parks because it can increase efficiency and reduce maintenance. Individual wind turbines have controllers for roll, pitch, and yaw. The SCADA system typically collects various data in order to monitor the state of the equipment and for maintenance purposes. This SCADA server can be located on-site, or in the control room. The SCADA system can be divided into three main functions. Real/time monitoring and control, handling of events and alarms, and collection and processing of data. The system should control and monitor both the system as a whole and the individual turbines. Among the collected data are wind speed, output power, blade angle, stall level, and yaw.

2.3 Summary

The smart grid can be expected to consist of many heterogenous systems. To describe the smart grid, seven domains have been created. As can be seen in Figure 2, extensive communication between the domains in the grid can be expected. We observe that a substantial amount of the smart grid is not directly related to the supply of electrical power but rather offers services that help to realize the vision of the smart grid.

The AMI system facilitates two-way communication between smart meters and a centralized meter data management unit. The system may allow services such as demand management, customer support, and quality and reliability control. The smart meter is the hardware and software placed in the customer perimeter, measuring power consumption along with other tasks. The communication with the centralized meter data management unit may take place via meter data concentrators. The AMI system may be interconnected at some level with the SCADA system to allow for better control of the grid.

A SCADA system is responsible for monitoring and controlling the power grid. This typically happens from a SCADA control room, communicating with remote controllers, sensors, and actuators in the field. A windmill may be an example of SCADA controlled equipment.

The communication network required to realize the smart grid can be based on PLC, wire, optical fiber, or wireless technology. PLC offers lower data rates and may suffer from interference but has advantages when it comes to cost. Wired alternatives are another technology that may suffer from interference. Fiber optic is by some deemed too expensive for wide deployment in the grid. Wireless alternatives may suffer from interference. Among the advantages of different wireless technologies are cost, deployment time, coverage, and capacity.

The identified ICT components are summarized in the table below. A more elaborate table with references is included in Annex 9.1.

Table 1: Overview of identified ICT components in the smart grid.

Generation including DER	**Transmission**	**Distribution**	**Customer**	**Operations**	**Service**	**Market**
PLC/RTU	PLC/RTU	PLC/RTU	Smart Meter	SCADA server	MDMS	Market platforms
Edge Computing	Edge Computing	Edge Computing	Data Concentrator	MDMS / AMI headend	Cloud	Cloud
PMU	PMU	PMU	Electric Vehicle	Cloud	-	-
Base station	Protection relays	Base station	Appliances	-	-	-
Cloud	Power quality monitors	Cloud	Automation systems	-	-	-
Circuit Switches	Line Sag monitors	Circuit Switches	Base station	-	-	-
Sensors	Fault recorders	Sensors	Cloud	-	-	-
-	Base station	DER	DER	-	-	-
-	Cloud	-	-	-	-	-
-	Substation meter	-	-	-	-	-
-	DER	-	-	-	-	-
-	Sensors	-	-	-	-	-

3 Chapter 3

Threat Modeling

3.1 Definition

Xiong and Lagerström (2019) identify several threat modeling definitions in their literature review. The identified definitions vary but typically include the identification or analysis of threats, security, or vulnerabilities. According to Shevchenko et al. (2018), threat modeling methods are used to create a model of a system and identify attackers, their goals and methods, and potential threats. Swiderski and Snyder (2004, p. 138) claim that "*achieving quantifiable security against a baseline of possible attacks is the driving force behind threat-modeling*."

3.2 Threat Modeling for software

This section is mainly based on two books by Shostack (2014) and Swiderski and Snyder (2004). The combined findings are shown in Figure 4.

Shostack (2014, pp. xxviii - xxix) describes a method for threat modeling consisting of four steps: building a diagram, finding threats, addressing threats, and checking the work. By using a threat model, details are abstracted away in order to provide the full picture. Shostack (2014, pp. xxiii - xxiv) identifies four reasons for threat modeling. The first is to find security bugs early. Identifying issues before the system is built saves expensive and less ideal fixes later in the development. The second is to understand your security requirements. The third is to engineer and deliver better products. Considering requirements and design in the early stages of the process results in a better product. The fourth reason is that threat modeling can help address issues missed by other techniques.

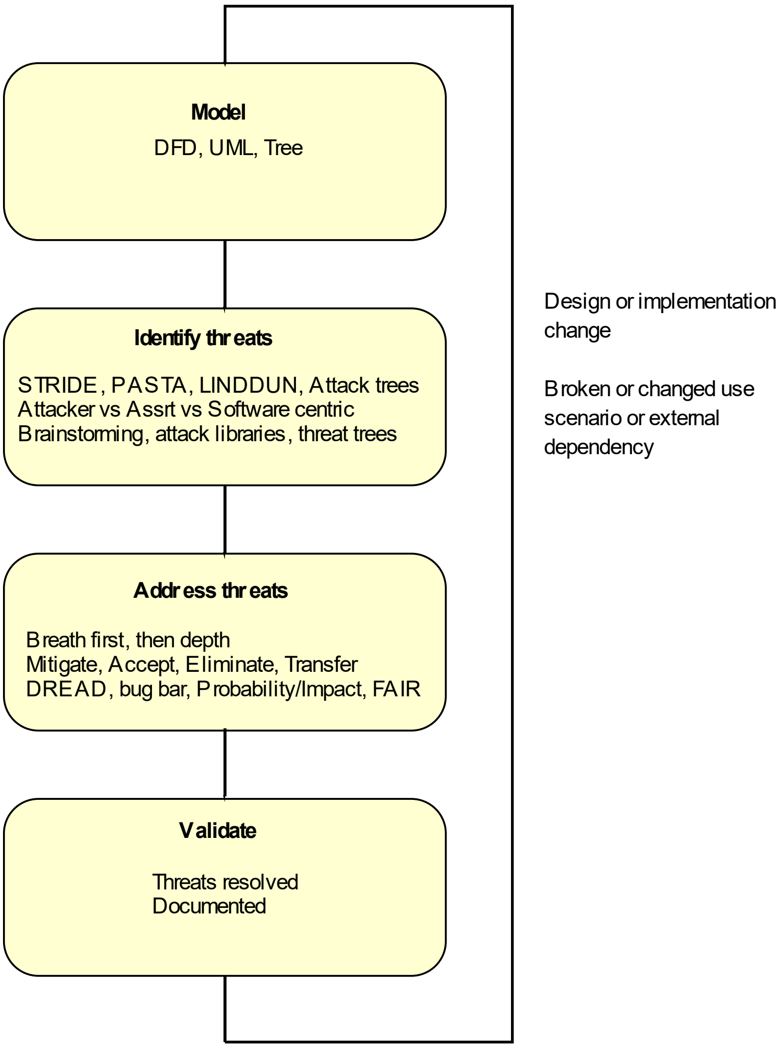

Model
DFD, UML, Tree
Identify threats
STRIDE, PASTA, LINDDUN, Attack trees
Attacker vs Assrt vs Software centric
Brainstorming, attack libraries, threat trees
Address threats
Breath first, then depth
Mitigate, Accept, Eliminate, Transfer
DREAD, bug bar, Probability/Impact, FAIR
Validate
Threats resolved
Documented
Design or implementation change
Broken or changed use scenario or external dependency

Figure 4: The threat modeling process based on Shostack (2014) and Swiderski and Snyder (2004).

3.2.1 Creating a model of the system

Swiderski and Snyder (2004, p. 65) argue that two steps should be undertaken at the start of the threat modeling process. The first is to gather relevant information and determine the scope of the analysis. The second is to create a model in order to understand the processing and data flows. According to Shostack (2014, p. 43), diagrams are a natural way to model software. Shostack (pp. 43 – 48) discusses two variants, Data Flow Diagrams (DFD) and Unified Modelling Language (UML). DFD is a common variant and less complex than UML. Complexity adds expressiveness to the diagram but also makes it more difficult to use. Both diagram types may be used for threat modeling by including trust boundaries.

The choices of how to model the system are often dependent on to threat-modeling method being chosen. LINDDUN and STRIDE often use DFDs. Attack trees use a tree structure to model attack paths. According to Shevchenko et al. (2018), PASTA uses both attack trees and DFDs.

3.2.2 Identifying Threats

Brainstorming is by Shostack (2014, p. 31) claimed to be the most traditional method for identifying threats. Despite this, it is believed to be problematic. Identified threats are believed to be hard to address and dependent on the participants. Another problem raised by Shostack (pp. 30 -34) is the lack of formal exit criteria. Swiderski and Snyder (2004, p. 42) share the critical review of brainstorming and argue that a more systematic approach is beneficial.

Shostack (pp.36 – 42) divides more structured approaches into asset-centric, attacker centric, or software-centric threat modeling. Assets normally refer to something of value. Things the asset owner would like to protect, things an attacker wants, as well as steppingstones to other assets, can all be regarded as assets. Shostack is critical of the asset approach. Identifying assets does not identify the threats against those assets. A method is still needed to go from a list of assets to a list of threats. The critical view of asset identification as an approach to threat modeling appears to be debatable. In their empirical review of STRIDE, Scandariato et al. (2015) claim that threat modeling is based on identifying assets. Swiderski and Snyder (2004, pp. 100 - 102) claim that threats and assets are closely connected. A proposed method by the authors for identifying threats begins with identifying the assets before a list of high-level attack goals is reviewed for each asset.

The attack centric approach uses attackers as a starting point. The description of attackers can have a varying level of detail. An advantage of the method is that it can help make the threats more real. A disadvantage is that a list of attackers often is not enough to deduce the threats the attackers might pose. Attackers have personal skills, backgrounds, and perspectives, which might further complicate the identification of threats. Because of these reasons, Shostack does not recommend focusing threat modeling on attackers. Swiderski and Snyder (2004, pp. 30-31), on the other hand, view taking the attacker's perspective as the first step in threat modeling.

The last approach focuses on software. This approach is claimed to be the best by Shostack. A reason for that is how the developers can be expected to understand software better than a business's assets or potential attackers. Most software is developed according to a model, which is a good starting point for software-centric threat modeling. While Shostack argues for the use of a software-centric approach, Swiderski and Snyder (2004) argue that threat trees are the preferred way of investigating threats. Threat trees are similar to attack trees discussed below. Code review and penetration testing are other ways of investigating threats.

As a more detailed way of identifying threats, Shostack (2014, pp. 101-109) discusses attack libraries. Attack libraries list various attacks and can be used as a more detailed way of identifying concrete threats. Attack libraries should not claim to contain all possible attacks and should try to encourage critical thinking. The discussed libraries are OWASP Top Ten, CAPEC, and checklists. OWASP is a list of the top ten risks for web applications and is updated yearly. CAPEC is a collection of many threat patterns, with a description of, amongst others, execution flow and prerequisites. Checklists are believed to be the most detailed of the three. Shostack argues that checklists are unlikely to help in finding threats not on the list.

3.2.3 Address Threats

Shostack (2014, pp. 128 - 130) advocates doing threat modeling top-down. The process should start from the highest possible view of the entire system. Shostack claim that the bottom-up approach does not work well because merging together lower-level models is challenging. After having chosen a top-down approach, it is recommended to continue investigating threats breadth-first, as opposed to depth-first. This means to first investigate threats in many locations superficially, instead of thoroughly at just one location. This can be done by iterating over either trust boundaries, diagram elements, or a list of threats.

Different techniques for ranking threats exist. Techniques discussed by Shostack (2014, pp. 180 - 184) are DREAD, bug bar, Probability/Impact, and FAIR. DREAD stands for Discoverability, Reproducibility, Affected users, and Damage. Shostack describes it as a subjective method that can give odd results. Swiderski and Snyder (2004, pp. 111 - 112) appear to have a more positive view of the method. Assessing the probability and impact of threats is described as an obvious approach. Despite this, effective implementations are rare. The reason is that probability assessments related to security are hard. FAIR is a way of quantifying risk based on ten steps. According to Shostack, it has many good elements if risk must be quantified. If quantifying risk is not required, a simpler method might be better. Bug bar is described as a method of classifying bugs based on severity. The is done using a set of common criteria defined in a table. Using this classification of severity, the organization can define a level for which bugs with a higher severity must be fixed. This level may change with time.

After a threat is identified, Shostack (2014, p. 168) argue that the level of risk and how the risk should be addressed must be considered. Shostack (pp. 12-13) further claims that identified threats can be addressed in four ways. Threats can be mitigated, eliminated, transferred, or accepted. Mitigating a threat is done by making it harder to take advantage of that threat. Eliminating a threat is often done by removing the feature or component producing the threat. Transferring a threat is done by placing the risk on something or someone else. The last approach is to accept the risk associated with the threat.

3.2.4 Validation and Feedback

The last step is concerned with validating that the identified threats have been addressed. Shostack (2014) relates the validation to testing. This may be more relevant for software than for a critical infrastructure system such as the smart grid. According to Shostack (p.195), important aspects of validation is to ensure that the model matches reality and that all threats have been addressed. Swiderski and Snyder (2004, p. 84) highlight the importance of validating that assumptions made regarding the implementation of the system hold. If this is not the case, they may be a source of vulnerabilities.

Swiderski and Snyder (2004) (p.138-139) argue that for the threat modeling to be complete, entry points must be documented, threats resolved, and documentation reviewed. Swiderski and Snyder (2004, p. 67) give several reasons for the purpose of documentation. Amongst others that it documents how extensive the analysis was and that the threats were investigated.

Swiderski and Snyder (2004, p. 165) argue that threat modeling is an iterative process. The threat model must be updated if the design or implementation changes. The same applies if a use scenario or external dependency is broken or changed (p.139).

3.3 Risk management by IEC 62443-3-2 and ISO 27005

IEC 62443-3-2 and ISO 27005 are standards for managing cybersecurity. They differ from the method outlined above in their primary focus lies on risk and how to manage risk. The IEC 62443 set of standards deals with cybersecurity in IACS systems. This thesis uses the term OT instead of IACS. ISO 27005 deals with information security risk management. The workflow of both standards is shown in Figure 5.

IEC 62443-3-2 deals with assessing security risks in industrial automation and control systems. The standard defines requirements for how to define the system under consideration, partitioning such a system into zones and conduits, assessing the risks of such zones and conduits, establishing the target security level, and documenting the security requirements. A zone is a grouping of assets that by some criteria is believed to belong together. Conduits are groupings of communication channels that share the same security requirements.

The first step in the IEC 62443-3-2 risk assessment is the establishment of the system under consideration. This includes the identification of all access points to the SUC and the security perimeter of the SUC. The second step is to perform a risk assessment to assess the worst-case scenario of an attack. The third step is to group the SUC into zones and conduits, based on some chosen criteria. In the fourth step, the risk of the system is compared with the tolerable risk of the organization to see if it exceeds the allowed limit. If it does, a more detailed assessment is performed in step five. The detailed risk assessment consists of identifying threats, vulnerabilities, and the likelihood of threats occurring if left unmitigated, amongst others. The sixth step focuses on documenting requirements, assumptions and constraints.

ISO 27005 (2018) deals with information security risk management. This paragraph provides a simplified description of the risk management workflow. In the first step, a context is established. This includes determining scop and boundaries and risk acceptance criteria. The second step is to identify, analyze and evaluate risk. In the identification phase, threats, vulnerabilities, and consequences are identified, amongst others. In the risk analysis phase, analysis of risks can be either quantitative or qualitative. In the risk evaluation phase, the level of risk identified is compared to evaluation and risk acceptance criteria, established in the first step. The third step is to treat the risks. Risks can be treated by modification, retention, avoidance, or through risk-sharing. These terms are similar to the terms transfer, accept, mitigate and eliminate used by Shostack (2014, pp. 12-13). In the fourth step, a list of the risks the organization chooses to accept is produced. A couple of conditional branched may alter the workflow. The standard argues that risks are not static and should be continuously monitored and potentially reviewed.

Some differences can be observed between the standards and the method based on Shostack (2014) and Swiderski and Snyder (2004). Most notably, the focus of the standards is to manage risk. Shostack argues that the risk of identified threats should be considered, but the concept of risk appears to have a less prominent position. In addition the standards are more formal and contain more detailed steps.

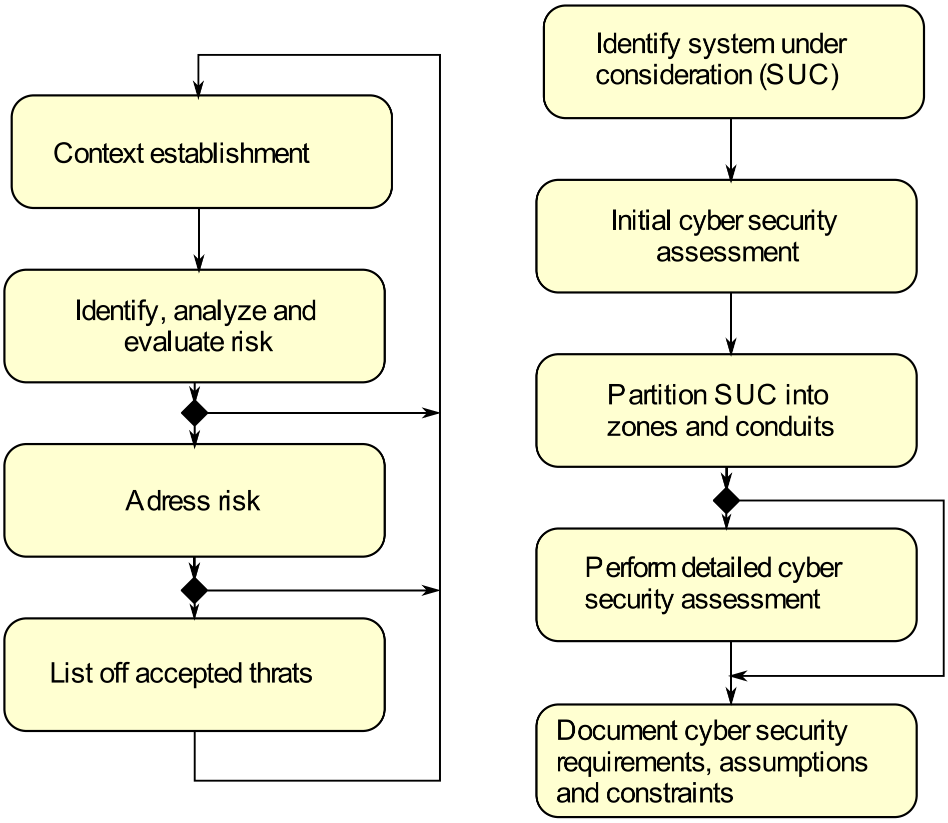

Context establishment
Identify, analyze and evaluate risk
Adress risk
List off accepted thrats
Identify system under consideration (SUC)
Initial cyber security assessment
Partition SUC into zones and conduits
Perform detailed cyber security assessment
Document cyber security requirements, assumptions and constraints

Figure 5: ISO 27005 (2018) workflow (left) and IEC 62443-3-2 (2020) workflow (right)

3.4 Threat modeling in literature

Xiong and Lagerström (2019) perform a systematic literature review on threat modeling. A total of 54 articles were selected for a closer review. The review shows that most articles use a manual method of threat modeling. Fewer articles rely on an automatic method or a combination of the two. Six of the articles are related to the smart grid. The most common methods are depicted in Table 2.

Table 2: Threat modeling methods identified by Xiong and Lagerström (2019).

Method	Number of articles
STRIDE/SDL/MS Threat modeling tool	7
Attack/Threat/Fault tree	6
Petri Nets	4
Dolev–Yao's threat model	3
Others	15

Jiang et al. (2014) create an attack tree for smart meters to illustrate energy theft behavior. The tree has four levels and twelve leaf nodes. Energy theft can occur in three ways. The first method is to prevent the meter from measuring consumed power. The second method is to tamper with the measurement data stored in the smart meter. The third method is to modify the meter data as it is sent across the network.

Liu et al. (2015) use colored Petri nets to describe the communication in a smart meter. The article claims information security to be the main security objective in the smart meter. As a result, threats to communication and information is analyzed based on the architecture modeled with Petri Nets. A strength of Petri Nets is their ability to model concurrent processes. The article does not use a specific technique to identify the threats.

Cardenas et al. (2009) investigate threats to SCADA networks. Known attacks are classified into three different groups. Finally, the attacks are classified based on execution difficulty and impact on security requirements.

Olayemi et al. (2017) investigate threats to Smart Home solutions. First, the assets in the system are identified. The second step is the creation of a DFD. In the next step, entry points to the system are identified. In the fourth step, threats are enumerated according to the STRIDE mnemonic of threats. In the final step, the threats are evaluated using DREAD.

Suleiman et al. (2015) perform an analysis of the Smart Grid using Security Quality Requirements Engineering (SQUARE) and Security Requirements Engineering Process (SREP). The SQUARE method *"Facilitates the exploration of security related issues ..."* Suleiman et al. (2015). SREP is a method for investigating security requirements. A total of 72 requirements, 76 threats and 32 vulnerabilities are identified.

Tøndel et al. (2013) perform a threat modeling on an AMI configuration. Both the STRIDE and attack tree methods are used. STRIDE is used to identify threats after a DFD of the system is created. The system is analyzed for threats in STRIDE, and a total of 30 threats are discovered. Denial of Service contributes the most threats. Attack trees are used in association with the system's assets. First seven assets are determined through brainstorming. Then attack goals are associated with the assets, and attack trees used to analyze how an attack on an asset can be conducted. An attack tree is created for each of the important attack goals. In the next step, each path in the trees should be analyzed to see whether additional measures are needed. These measures should be prioritized according to attack goal importance. Brainstorming is mentioned as a possible weak spot, as it highly depends on the people participating. The STRIDE process is characterized as somewhat laborious, but the maintainability is believed to be good. Another advantage of the method is the coverage of a wide amount of security issues.

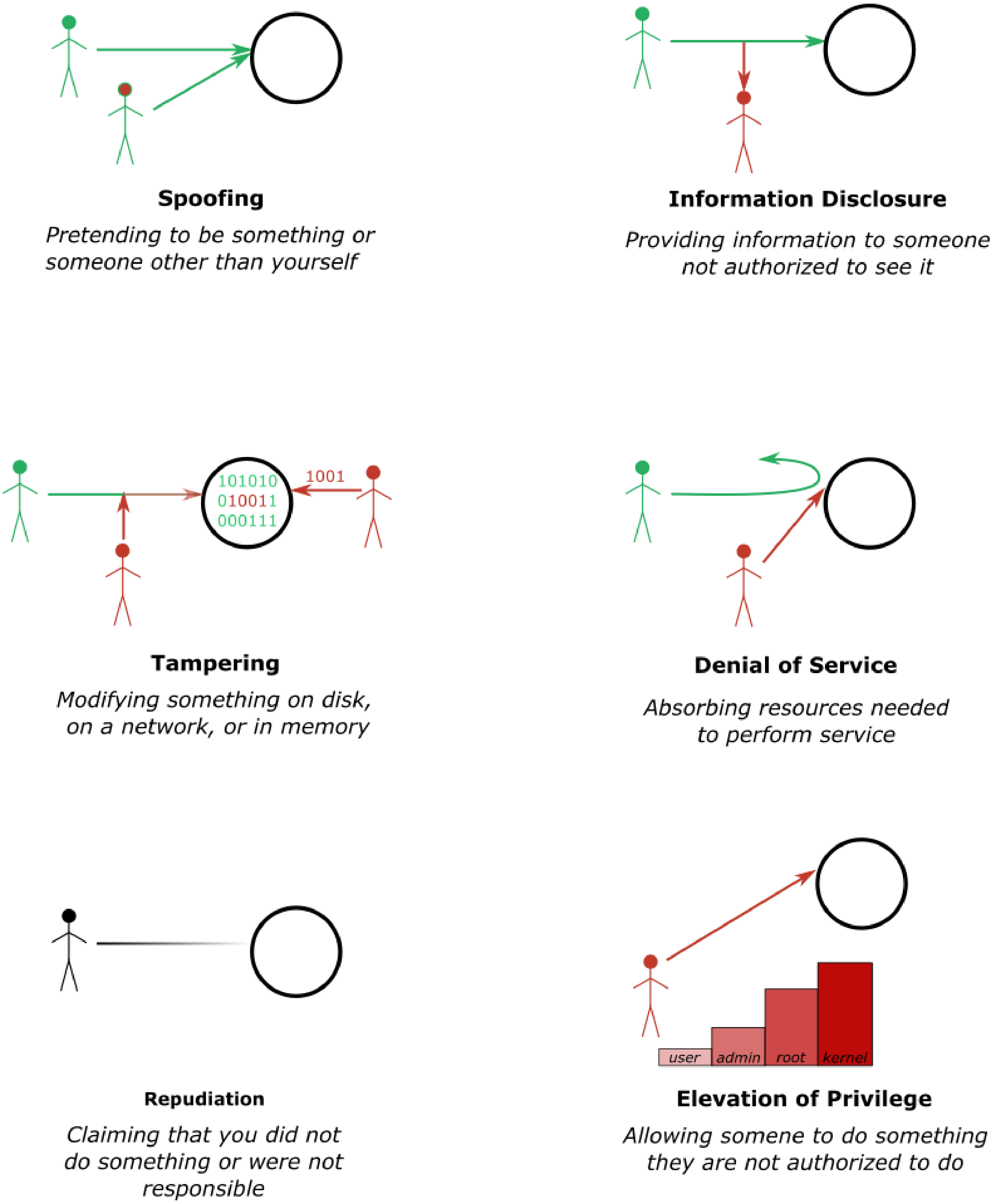
3.5 Different types of threat models

This section gives an overview of different threat modeling techniques. Much of the section is based on Shevchenko et al. (2018).

3.5.1 STRIDE

STRIDE is a mnemonic for Spoofing, Tampering, Repudiation, Integrity, Denial of Service and Elevation of privilege. Definitions along with illustrations of the threat categories are shown in Figure 6. According to Scandariato et al. (2015) threat modeling with STRIDE consists of four steps. In the first step, a DFD of the system is created. Security assumptions are also listed in this step. Step two maps threats to DFD elements. Step three elicits the threats. This is done by further detailing the generic threats mapped to elements in step two. The generic threat can form the root of a tree where the nodes elicit different ways the generic threat can be realized. Spoofing against an external entity may for instance have its own tree, and tampering against a process may have another. Some of the trees are reused. Howard and Lipner (2006) provide 12 such trees in their book on the Security Development

Lifecycle. The relevance of the various nodes depends on the assumptions from step one. The fourth step is to document the threats. STRIDE does not impose requirements on how the threats are documented.



Spoofing

Pretending to be something or someone other than yourself

Information Disclosure

Providing information to someone not authorized to see it

Tampering

Modifying something on disk, on a network, or in memory

Denial of Service

Absorbing resources needed to perform service

Repudiation

Claiming that you did not do something or were not responsible

Elevation of Privilege

Allowing somene to do something they are not authorized to do

Figure 6: Illustration and definition of the STRIDE threats. Adapted from Shostack (2014, pp. 62-63)

Shostack (2014, pp. 78 - 85) distinguishes between STRIDE-per-element and STRIDE-per-interaction. STRIDE-per-element takes advantage of the fact that certain threats are more relevant to certain elements. What these threats are may vary from system to system. How many threats that are chosen for each element is a trade-off. Choosing many threats per element makes the analysis more comprehensive and thorough. STRIDE-per-interaction was invented to address the fact that threats emerge in the interactions of the system. The approach enumerates threats based on a tuple of (origin, destination, and interaction).

According to Shostack, STRIDE-per-element and STRIDE-per-interaction lead to the same number of threats. Khan et al. (2017) argue that STRIDE-per-component is the most complex of the two. This is the method that received the most attention in the cyber-physical adaption of STRIDE, as discussed below. Despite this, the authors argue that STRIDE-per-interaction normally is easier to perform and that its protection strategies normally are sufficient to protect the system.

Both STRIDE-per-Element and STRIDE-per-Interaction have tables for threat applicability. These tables show what threats are relevant for which elements or interactions. The threat applicability table for STRIDE-per-interaction is shown in Table 3. From the table, it becomes clear that the first interaction of a process sending data to a data store is vulnerable to Spoofing and Information disclosure threats. The tool used in this thesis, the Microsoft TMT, uses STRIDE-per-Interaction. Since most of the discussion in this thesis revolves around STRIDE-per-Interaction rather than STRIDE-per-Element, only the threat applicability for STRIDE-per-Interaction is included.

Table 3: Threat Applicability for STRIDE-per-Interaction. Shostack (2014, p. 81).

#	Element	Interaction	S	T	R	I	D	E
1	Process	Process has outbound data flow to data store	X			X		
2	Process	Process sends output to another process	X		X	X	X	X
3	Process	Process sends output to external interactor (code)	X		X	X	X	
4	Process	Process sends output interactor (human)			X			
5	Process	Process has inbound data from flow from data store	X	X			X	X
6	Process	Process has inbound data flow from a process	X		X		X	X
7	Process	Process has inbound data flow from external interactor	X				X	X
8	Data Flow	Crosses machine boundary		X		X	X	
9	Data Store	Process has outbound data flow to data store		X	X		X	X
10	Data Store	Process has inbound data flow from data store		X	X	X		
11	External Interactor	External Interactor passes input to process	X		X	X		
12	External Interactor	External Interactor gets input from process	X					

STRIDE in cyber-physical systems

Khan et al. (2017) propose a methodology for applying STRIDE to cyber-physical systems. According to the authors, such a methodology does not exist in the literature. The methodology has five steps, shown in Figure 7. The authors chose the STRIDE method as they argue it is a comprehensive and systematic approach that provides an understanding of how vulnerabilities in components affect the overall system. With the increasing integration of software in physical systems, Shevchenko et al. (2018) claim that the use of threat modeling can help traditional manufacturers of infrastructure to detect threats they might not consider.

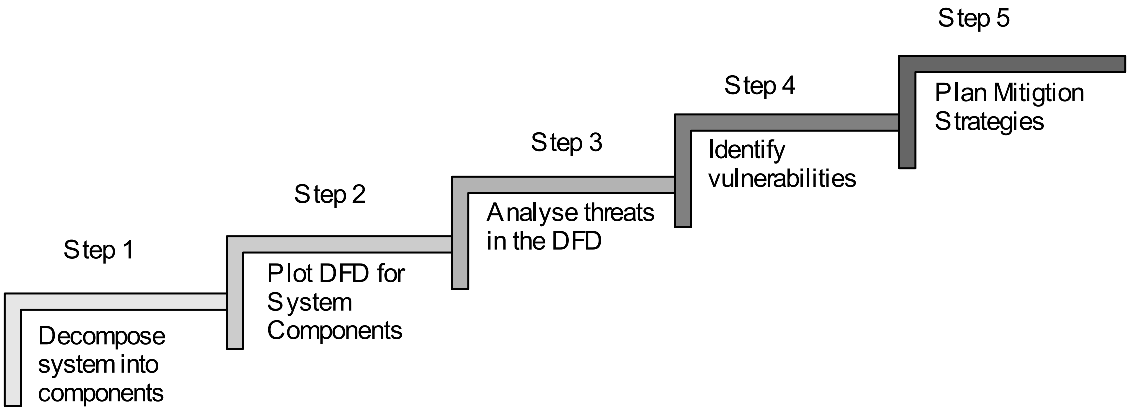

Step 5
Plan Mitigtion Strategies
Step 4
Identify vulnerabilities
Step 3
Analyse threats in the DFD
Step 2
Plot DFD for System Components
Step 1
Decompose system into components

Figure 7: Proposed method for STRIDE in cyber physical systems. Khan et al. (2017)

In the first step, the system is decomposed into its logical or structural components. Components can be internal to the system or external components that communicate with the system. The second step is to create a DFD for each of the components. A DFD consists of five symbols: external entity, process, data flow, trust boundaries, and data store. The third step is to investigate which STRIDE threats are relevant to which DFD. The fourth step is to investigate the vulnerabilities causing the threats. Their fifth step is to identify strategies to mitigate the vulnerabilities.

3.5.2 Attack tree

An attack tree uses the tree data structure in order to represent the different ways an undesirable event can be achieved. According to Shostack (2014, p. 89), the root node of the tree can be the adversary's goal, an unwanted state, or a component. If the root node is a component, the sub-nodes investigate what can go wrong with the component. After a root node has been decided, the tree is extended with sub-nodes. Schneier (1999), often credited as the first to introduce attack trees, labels the root node as the goal of the attack. The paths from the leaf nodes to the root node represent various ways to exploit the described system.

Shostack (2014, p. 88) refers to the use of either AND trees or OR trees. Schneier (1999) makes the distinction between AND and OR nodes in the same tree. For an AND node, all sub-nodes must be reached for the node itself to be reached. For an OR tree, it is sufficient that one sub-node is reached for the node itself to be reached. According to Nagaraju et al. (2017), the leaf nodes represent the actions the attacker can take against the system. The intermediate nodes represent intermediate adversary goals. Schneier (1999) proposes to add a value or attribute to each node. One alternative is to assign a monetary value to each node. The value represents the cost of the activity in the node.

3.5.3 PASTA

PASTA was invented by UcedaVélez and stands for Process for Attack Simulation & Threat Analysis. According to Shevchenko et al. (2018), it is a seven-step method resulting in the impact of threats to application and business. Step one defines the business and security objectives. Step two defines the technical scope. Step three decomposes the application. Step four analyses the threats. Step five analyses weaknesses and vulnerabilities. Step six enumerates, and models attacks and exploits. Step seven analyses risk and impact and deploys countermeasures.

The method appears to be extensive, consisting of more than 20 sub-steps. The sub-steps include tasks and methods such as DFDs, attack trees, and the Common Vulnerability Scoring System (CVSS). Shevchenko et al. claim that the method seeks to combine business objectives and technical requirements and that the method requires input from operations, governance, architecture, and development.

3.5.4 LINDDUN

Deng et al. (2011) present a framework called LINDDUN. LINDDUN is a mnemonic for the following threats: linkability, identifiability, non – repudiation, detectability, disclosure of information, unawareness, and non-compliance.

Linkability is the ability to decide whether two or more actions or items are related. Identifiability is the ability to decide whether a subject is associated with an item or action. Non – repudiation is the ability to prove that an actor has done something. Interestingly it can be observed that while repudiation is the threat in STRIDE, non – repudiation is considered the threat in LINDDUN. This is an example of a scenario where security and privacy may be in conflict. Detectability is the ability to decide whether an item exists or not. Unawareness is when a user is unaware of what information is being shared with the system. Non-compliance means that there is no guarantee that a system follows its own privacy policies.

The method focuses on privacy in software systems. LINDDUN and STRIDE use a similar approach. Both methods use the DFDs to model the system in the first step. The second step is to map threats to the DFD element. Not all threats are relevant to all elements. Threat trees are used to create specific threat instances relevant to the system, based on the general LINDDUN threats. In this way, many threats may be identified. These threats are prioritized in step four. Based on the specific threats, privacy requirements are elicited in step five. In the sixth and final step, mitigation strategies are chosen.

3.6 Discussion of threat methods

STRIDE

Based on the review by Xiong and Lagerström (2019), STRIDE is one of the more popular methods. Shevchenko et al. (2018) claim it to be the most mature method. With regards to complexity, Shevchenko et al. claim the method to be time-consuming and that complex systems can have a high number of threats. This will likely be relevant to a system as complex as the smart grid. According to Tøndel et al. (2013), the method is time-consuming to complete but creates easily maintainable systems if the changes are minor.

Deng et al. (2011) claim that STRIDE does not cover privacy adequately. While privacy was not the main focus, Tøndel et al. (2013) argue that STRIDE could have been used to model privacy threats in their analysis of an AMI configuration. Information disclosure appears to be the most relevant part of STRIDE with regards to privacy.

Attack tree

According to Nagaraju et al. (2017), the advantages of attack trees lies in the possibility of identifying attacks and countermeasures. An advantage of assigning values to the nodes, as proposed by Schneier (1999), is that the most attractive attack path can be identified. Attack paths where the cost exceeds the perceived value of the attack can similarly be identified. This can in turn help prioritize the implementation of countermeasures. Effects of changes to the system can also be analyzed if the attack tree is updated accordingly.

A disadvantage according to Nagaraju et al. (2017) is the difficulty of identifying all attacks and inadequate support for modeling attacks executed using concurrent actions. According to Shostack (2014, pp. 98 - 100), attack trees are hard to create and can be hard to use. One problem is completeness. If certain root nodes are not included, the group of attacks related to it is not discovered. Another problem is scoping. It may be difficult to know what should be included. These problems can be assumed to become relevant in the smart grid due to the vast attack surface and number of actors involved in the grid. Consequently, the attack trees might grow large.

LINDDUN and PASTA

LINDDUN offers an extensive framework for analyzing privacy threats. According to Shostack (2014, p. 121), it is "*... one of the most serious and thought-provoking approaches to privacy threat modeling ...*". Privacy is a key challenge for gathering and storing customer data in the grid. When it comes to controlling and monitoring of the grid, availability, and integrity are the more important requirements. LINDDUN can provide insight into one aspect of smart grid security but is not sufficient for modeling the security for the whole grid. PASTA appears to be the most comprehensive framework among the four discussed. The method may become laborious for larger systems. Both LINDUN and PASTA are relatively new methods that does not appear to be much used.

Others

Shevchenko et al. (2018) review a number of methods. The Common Vulnerability Scoring System (CVSS) assigns a severity score to each vulnerability and describes the key aspects of it. The method can be combined with other methods but is as a standalone method not extensive enough for the smart grid. The Persona non Grata technique focuses on the goals and motivations of human adversaries. Such a method can be used to protect against disgruntled employees or regular customers. One example is the archetype of a financially motivated and technically competent customer stealing power. The method is less suitable for protecting against state actors targeting the availability of the grid. Security cards is a brainstorming technique using a deck of 42 cards. The technique focuses on unusual and complex attacks. This might prove advantageous as attacks on industrial control systems are both unusual and complex. Despite this, the technique is rarely used in the industry. The quantitative Threat Modeling Method is a combination of STIDE, CVSS, and attack trees. The method builds component attack trees for all the threats in STRIDE. Then CVSS scores are given to the nodes of the trees. The authors aimed to address issues related to cyber-physical systems. The method may suffer from some of the same issues as attack trees.

4 Chapter 4

Security Requirements and Threats to the Smart Grid

4.1 Definition of Security requirements

This section defines five fundamental security requirements of the smart grid. These requirements relate to information. This is particularly important when referring to availability, as one can both refer to the availability of information and the availability of power.

Confidentiality: Defined by ISO 7498-2 (1989) as “property that information is not made available or disclosed to unauthorized individuals, entities or processes.”

Integrity: Defined by ISO 7498-2 (1989) as “property that data has not been altered or destroyed in an unauthorized manner.”

Availability: Defined by ISO 7498-2 (1989) as “property of being accessible and useable upon demand by an authorized entity.”

Privacy: Defined by ISO 7498-2 (1989) as “right of individuals to control or influence what information related to them may be collected and stored and by whom and to whom that information may be disclosed.”

Authenticity: Defined by ISO 27000 (2018) as “property that an entity is what it claims to be”.

4.2 Discussion of the security requirements of the Smart Grid

The smart grid contains both IT systems and OT systems. Pillitteri and Brewer (2014) expands on the conceptual model described in section 2 and describes a number of logical interfaces between the actors in the seven domains. The actors can be both systems, devices, and programs. The interfaces are grouped into 22 categories based on similar security-related characteristics. The authors continue to evaluate the effect of a loss of either confidentiality, integrity, or availability for each of the 22 categories. The consequences are evaluated with regards to their effect on the power grid, especially power grid reliability and stability. For seven of the 22 categories, a violation of confidentiality is believed to have severe consequences. For integrity, the same applies to 18 of 22 categories. A violation of availability is believed to have severe consequences in five out of 22 categories. This is shown in Figure 8.

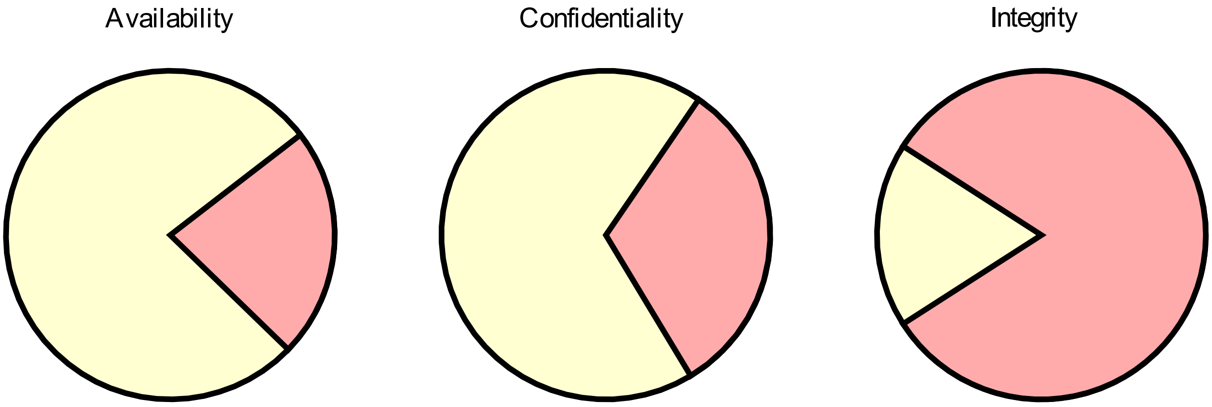


Figure 8: The effect of a loss of Availability, Confidentiality and Integrity on the 22 categories making up the smart grid, as laid out by…, The red area indicates the number of categories expected to be severely affected were a loss of the security property to occur.

Interestingly the loss of availability appears to affect few of the categories in a severe way. The reason for this is unknown. It should be noted that the categories may not be equally large. It may be the case that the categories believed to be severely affected by a loss of availability each contain many individual interfaces.

Despite several affecting fewer categories than the other two properties, Pillitteri and Brewer (2014 p.75) claim availability is the most critical requirement to the reliability of the power grid. Integrity is regarded as second. This largely coincides with IEC 62443-1-1 (2009). IEC claim that traditional information technology systems confidentiality has been regarded as the most important property with integrity and availability ranking as second and third. In automation and control systems this is typically reversed, making availability the most important and confidentiality the least important. The relationship between requirements for IT systems and OT systems according to IEC 62443-1-1 is shown in Figure 9.

Pillitteri and Brewer (2014 p.78) identify confidentiality along with privacy as new issues being introduced by the smart grid. This is introduced by the collection of customer information and more actively participating customers.

European Network and Information Security Agency (2012) take a similar approach to IT and OT security as the more general IEC 62443-1-1 (2009), shown in Figure 9. Confidentiality, integrity and availability is the order of importance in the IT part of the smart grid. In the OT part of the grid, the order of importance is reversed.

European Network and Information Security Agency (2012) introduces two additional security properties. Those are authentication and non-repudiation. Both properties are claimed to be equally important in both the OT and IT parts of the smart grid. In addition, the authors mention privacy, which is believed to be strongly linked to confidentiality.

The availability of power appears to be regarded as the primary security concern in the smart grid, according to Pillitteri and Brewer (2014) and Wang and Lu (2013). Aloul et al. (2012) refer to three main security objectives. These are the availability of power, the integrity of communication, and user data confidentiality.

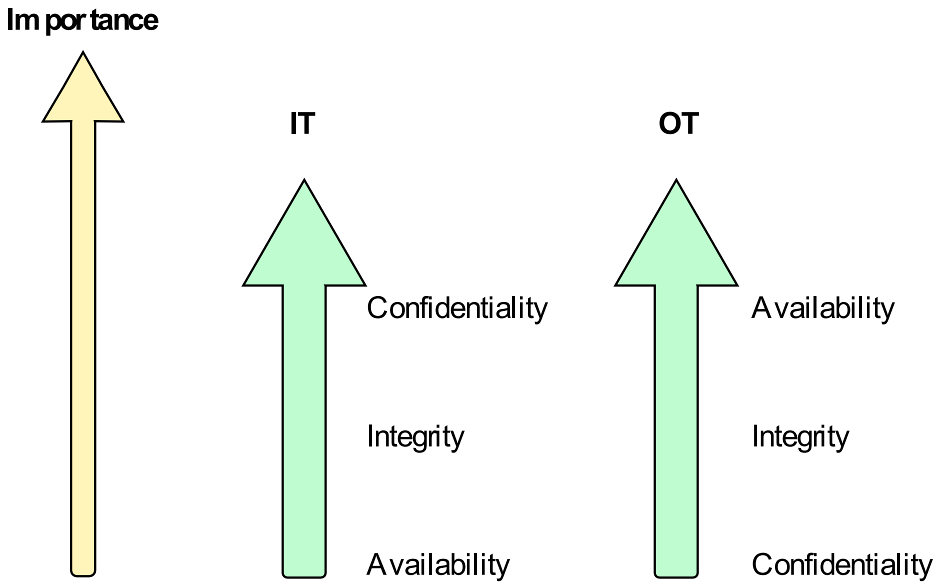


Figure 9: Difference in importance of security properties in IT and OT systems. IEC 62443-1-1 (2009).

4.3 Threats to the Smart Grid in literature

This section presents threats found in the literature grouped by the components they affect.

4.3.1 Smart Meter

1.a Sgouras et al. (2014) demonstrate how the single smart meter will experience a growing queue of incoming packets when it is the victim of a distributed denial of service attack. This can cause increased response times or make the smart meter unresponsive.

1.b ... 1.g These threats are adapted from a threat modeling of AMI by Tøndel et al. (2013). It can be noted that the article considers a broader range of denial of service threats against smart meters. 1.f include both distributed denial of service attacks, as discussed by Sgouras et al. (2014), and other types of denial of service attacks. Such attacks may be specially crafted messages or malware that renders the device unavailable. Eavesdropping of smart meter – AMI communication is a threat highlighted by both Tøndel et al. and Sgouras et al. (2014). Sgouras et al. identify data of interests to an adversary to be name, address, social security number, individual preferences, social activities, consumption habits, health issues, and information indicating presence.

1.h Guo et al. (2015) regard injection of malware into a large number of smart meters to be a possible method for performing a DDoS attack on the AMI server. The malware can either be downloaded from the attacker or from a remote server, or it can propagate from meter to meter like a worm.

Table 4: Threats to Smart Meters

Threat	Category	Source	Threat number
DDoS attack on smart meter	Denial of Service	Sgouras et al. (2014)	1.a
Fake AMI server	Spoofing	Tøndel et al. (2013)	1.b
Tampered communication between meter and AMI server	Tampering	Tøndel et al. (2013)	1.c
Meter denies sending or receiving a message	Repudiation	Tøndel et al. (2013)	1.d
Eavesdropping AMI server – smart meter communication	Information disclosure	Sgouras et al. (2014; Tøndel et al. (2013)	1.e
DoS attack on smart meter	Denial of service	Tøndel et al. (2013)	1.f
Unauthorized remote access to smart meter	Elevation of privilege	Tøndel et al. (2013)	1.g
Injection of malware on smart meter	Elevation of Privilege	Guo et al. (2015)	1.h

4.3.2 Circuit breaker

2.a Unauthorized control of circuit breakers in the grid is a threat to power availability. The opening of circuit breakers can cause customers to lose power. Stefanov and Liu (2012) simulate a scenario where an attacker compromises the SCADA network and proceeds to open a circuit breaker, disconnecting a source. Depending on the architecture of the grid, such an attack can cause the frequency in the grid to fall below its allowed value. According to Slowik (2018), malware used in the 2016 Crashoverride[12] attack on the Ukraine power grid contained ICS specific malware designed to open and close circuit breakers in the grid.

2.b According to Slowik (2019), the attackers in the 2016 Crashoverride attack attempted to exploit a known vulnerability in four safety relays. Safety relays monitor several values and protect the electrical system. This is believed to have failed due to a bug in the malware. A patch for the vulnerability did exist, but it is unknown whether the asset owner had patched the targeted safety relay.

Table 5: Threats to circuit breakers

Threat	**Category**	**Source**	**Threat Number**
Unauthorized open and close	Elevation of privilege	Stefanov and Liu (2012) Slowik (2018)	2.a
Denial of service through a malformed message	Denial of service	Slowik (2019)	2.b

4.3.3 SCADA server

3.a Isozaki et al. (2015) investigate a method of detecting falsified sensor values in a distribution network. Falsifying the sensor values delivered to the voltage controller can cause the voltage at connected loads to deviate outside the allowed range. It is believed that this may cause damage to equipment. How the measurements may be falsified is not discussed.

3.b Rizzetti et al. (2015) argue that flooding types of Denial of Service attacks are a threat to SCADA systems. These attacks are conducted by sending many requests to a resource. In this way, legitimate requests are blocked or not processed. This attack is called a Distributed Denial of Service attack when it is carried out by multiple sources at once.

3.c D'Antonio et al. (2011) highlight how the database storing PMU values is vulnerable to a SQL attack if it does not sanitize the values. As this thesis does not include the phasor concentrator unit; the mentioned database is assumed to reside at the SCADA server. Authenticating the PMU communication makes the database less vulnerable. The database should still sanitize the data in case an adversary is able to compromise the PMU itself, gaining access to encryption keys.

3.d Yang et al. (2012) argue that an attacker can conduct an Address Resolution Protocol (ARP) Man-in-The-middle attack in order to intercept communication between a SCADA host and an IED. This is attempted in a SCADA testbed implemented on Windows computers. The intercepted communication

[12] Crashoverride is also known under the name Industroyer

can be both read and tampered with. As ARP is unable to traverse network interfaces, the attacker must first establish a presence on the LAN before an attack can be conducted.

Table 6: Threats to SCADA Servers

Threat	**Category**	**Source**	**Threat Number**
Falsifying voltage measurements	Tampering	Isozaki et al. (2015)	3.a
Distributed denial of service attack on a server	Denial of service	Rizzetti et al. (2015)	3.b
SQL attack on Phasor Data Concentrators database.	Tampering	D'Antonio et al. (2011)	3.c
ARP Man-In-The-Middle Attack	Spoofing	Yang et al. (2012)	3.d

4.3.4 PMU

4.a Shepard et al. (2012) show that a GPS spoofing attack against a PMU in an experimental setup can cause a deviation in the phasor measurement. The attack is performed by first transmitting signals that are almost identical to the authentic GPS signal. Then the attacker increases its signal strength to be slightly above that of the authentic signal and starts to slowly deviate from the authentic GPS signal. The attack chooses a slow approach to avoid detection. After 680 seconds, the deviation in the PMU measurement is large enough to violate the maximum allowed deviation in the IEEE C37.118 standard for synchrophasors. The attack was made possible due to the use of unencrypted public GPS signals. The article argues that had the attack been launched against PMUs installed in an actual transmission line located in Mexico, it could have caused the generator to disconnect from the grid.

4.b D'Antonio et al. (2011) discuss a scenario where a PMU sends data to a Phasor data concentrator using the C37.118 protocol. As the protocol does not encrypt the message or verify its authenticity, messages sent are vulnerable to information disclosure and spoofing threats.

Table 7: Threats to PMU

Threat	**Category**	**Source**	**Threat Number**
GPS spoofing	Spoofing	Shepard et al. (2012)	4.a
Spoofing and information disclosure of PMU communication	Spoofing Information disclosure	D'Antonio et al. (2011)	4.b

4.3.5 5G network

5.a,…,5.d Borgaonkar and Jaatun (2019) argue that an attacker may be motivated to intercept, modify, or deny 5G IoT traffic. The attacker may also be motivated to attempt to discover the location of IoT devices. These attacks are carried out using hardware and software in the geographical area of the IoT and 5G devices. Other types of attacks can be carried out remotely. Such attacks may be carried out by first compromising systems in the 5G network and then attempt to steal critical smart grid information or launch attacks on smart grid IoT devices.

Table *8: Threats to 5G networks*

Threat	**Category**	**Source**	**Threat Number**
Intercept or modify 5G IoT traffic	Disclosure of Information Tampering	Borgaonkar and Jaatun (2019)	5.a
Deny 5G IoT traffic	Denial of Service	Borgaonkar and Jaatun (2019)	5.b
Steal critical Smart Grid Information	Information Disclosure	Borgaonkar and Jaatun (2019)	5.c
Attack IoT devices	STIDE	Borgaonkar and Jaatun (2019)	5.d

4.3.6 DER

6.a Sundararajan et al. (2018) claim wiretapping to be an attack often seen in the physical layer. Yan et al. (2011) identify wiretapping of optical fiber as a threat. The authors argue that optical fiber is often used as physical medium in the communication architecture of wind farms.

6.b Sundararajan et al. (2018) highlights the risk of TCP and UDP flooding for communication-based on the TCP/IP stack. According to Sundararajan et al., most of the devices in the smart grid rely on this technology. The consequences of this are unknown. Staggs et al. (2017) claim that many wind farms can continue production and delivery of power even in the event of a loss of communication.

6.c By interviewing DER vendors and utilities, Sundararajan et al. (2018) claim to have discovered that no methods exist for validating the integrity of patches and firmware in an isolated environment before applying it to OT infrastructure. Malicious firmware and patches may contain backdoors or harm the system in other ways.

6.d, 6.e In their discussion of cryptography**,** Lai et al. (2019) briefly mention control input spoofing and disclosure of personally identifiable information as possible threats to the operation of DERs.

6.f In their discussion of Wind Farm SCADA security, Yan et al. (2011) identify wiretapping as a threat to fiber optic communication. The authors claim that wiretapping can occur with varying degrees of sophistication and that it is easier than generally believed. More sophisticated wiretapping devices are claimed to be able to inject malicious signals into the fiber.

6.g…6.j In addition to optical wiretapping, Yan et al. (2011) claim there are several threats to wireless communication. The threats are wireless sniffing, wireless spoofing, wireless MITM attacks, and signal jamming attacks. The authors claim that signal jamming can occur even if strong encryption is applied and that it is easy to perform. Two jamming scenarios are further imagined. The first is a scenario where the data flow is slowed down as retransmissions become more likely. The second is that the transmission comes to a complete stop, effectively conducting a denial of service attack.

6.k According to Staggs et al. (2017), vendors often have VPN access to wind farms for maintenance, software updates and monitoring purposes. VPN connections for remote communication in windfarms is also mentioned by Yan et al. (2011). This threat may be particularly serious by the fact that the vendor may have VPN access to many windmills, possibly placed in different geographic locations. If the vendor organization is compromised, a coordinated attack on several windmills may be launched.

6.l, 6.m Staggs et al. (2017) claim technician equipment and the supply chain to be two attack vectors of wind farms. Technician equipment can be infected with malware. In the supply chain either hardware, firmware or software can be compromised before instalment in order to facilitate a cyber-attack. According to Falliere et al. (2011), an infected USB drive and infected vendor software were among the methods used by Stuxnet to compromise its target. Stuxnet is briefly described in section 4.4.

6.n During a security assessment, Staggs et al. (2017) report discovering Programmable Automation Controllers (PAC) which can be accesses through Telnet. The authentication mechanism is reportedly weak or not existent. The authors claim that unauthenticated FTP services are also found.

6.o Staggs et al. (2017) develop a custom tool to demonstrate how a Man-In-The-middle attack can be performed in a wind farm. The tool exploits the ARP protocol to intercept communication between a source and a target. The authors argue that this for instance can be used to intercept OPC messages between an OPC Client and wind turbines. When the attack is completed, an adversary can block, modify and fabricate control messages. Yan et al. (2011) highlight the possibility of performing an MITM attack on fiber optic cables.

6.p After an adversary has compromised a PAC, Staggs et al. (2017) argue that one possible attack is to execute ransomware on the controller. The ransomware can encrypt files, disable services and change credentials and subsequently financially extort the asset owner.

6.q Staggs et al. (2017) develop a custom tool to be able to send malicious commands to a wind turbine in the form of malicious OPC requests. Reverse engineering of the protocol is used to detect suitable commands. The authors highlight that another possibility for controlling wind farms would be to download malware onto a PLC or PAC. The authors argue that these methods could be used to cause physical damage to the windmills.

Table 9: Threats to DER

Threat	**Category**	**Source**	**Threat Number**
Data Wiretapping	Information disclosure	Sundararajan et al. (2018; Yan et al. (2011)	6-1
SYN or UDP flooding	Denial of Service	Sundararajan et al. (2018; Yan et al. (2011)	6.b
Malicious firmware and patches	Elevation of privilege	Sundararajan et al. (2018)	6.c
Control input spoofing	Spoofing	Lai et al. (2019)	6.d
Disclosure of personally identifiable information	Information disclosure	Lai et al. (2019)	6.e
Injection of malicious data in optical fiber	Tampering	Yan et al. (2011)	6.f
Sniffing against wireless communication	Information disclosure	Yan et al. (2011)	6.g
Spoofing against wireless communication	Spoofing	Yan et al. (2011)	6.h
Man-in-the-middle attack against wireless communication	Spoofing	Yan et al. (2011)	6.i
Jamming attack	Denial of service	Yan et al. (2011)	6.j
Accessing wind farm via vendor VPN	Elevation of privilege	Staggs et al. (2017)	6.k
Injection of malware into wind farm using compromised service personnel equipment	Elevation of privilege	Staggs et al. (2017)	6.l
Attack on wind farm via compromised supply hardware, software or firmware.	Elevation of privilege	Staggs et al. (2017)	6.m
Access to programmable automation controller via Telnet or FTP.	Elevation of privilege	Staggs et al. (2017)	6.n
Layer-2 man in the middle attack between control center and windmill.	Spoofing	Staggs et al. (2017)	6.o
Execution of ransomware	Elevation of Privilege	Staggs et al. (2017)	6.p
Issuing malicious commands to the windmill	Spoofing	Staggs et al. (2017)	6.q

4.3.7 IED

7.a Rizzetti et al. (2015) highlights a flooding type of denial of service as a threat to resources in the smart grid. This is done by flooding a resource with many requests such that legitimate traffic cannot be treated by the resource.

7.b Attacks against industrial control systems show that an attacker can inject and execute malicious code on a PLC to carry out an attack. Among the most well-known examples of this are Stuxnet and Triton. Klick et al. (2015) demonstrate how a PLC with internet connectivity can be compromised and turned into a gateway to the network it is connected to. This is achieved by writing a malware in the native PLC language Instruction List. The malware is then uploaded to the PLC and can run alongside the normal PLC program.

7.c..7.e Kalluri et al. (2016) demonstrates how flooding a RTU with network, transport and application layer messages affect the performance of the RTU. In the setup the RTU is connected to a MTU. As the threats apply to the RTU, the threats are by this thesis considered relevant to all RTU communication. Flooding the RTU with IP packets caused the RTU to exhibit abnormal behavior. The exact type of behavior is not specified. The RTU response time rose to 8 seconds when flooded with a stream of 34 000kKbits/s. On the transport layer, the RTU was targeted with a stream of TCP SYN requests. Malicious requests occupied all the connections the RTU can open at one time, leaving no connection possibilities for legitimate traffic. The RTU becomes unresponsive at 850 SYN request per second. On the application layer, denial of service is realized by sending IEE 60870-5-104 messages. The RTU becomes unresponsive when targeted with 580 messages per second.

7.f Niedermaier et al. (2018) investigate the effects of various types of flooding attacks on PLC cycle times against a total of 16 PLCs. Unlike other articles the focus is not on network availability but on the availability of the PLC to perform its assigned tasks. On one incident, an ARP type of flooding attack causes a PLC to stop updating its output variables. As ARP does not propagate across sub-network this attack can only be launched if the attacker gains a foothold on the network.

7.g Niedermaier et al. (2018) report that certain types of PLCs experience increased cycle times when subject to UDP flooding.

Table 10: Threats to IED

Threat	**Category**	**Source**	**Threat Number**
Distributed denial of service on device	Denial of service	Rizzetti et al. (2015)	7.a
Execution of malware on PLC	Elevation of privilege	Falliere et al. (2011) Johnson et al. (2017) Klick et al. (2015)	7.b
IP flooding denial of service	Denial of Service	Kalluri et al. (2016)	7.c
TCP SYN flooding denial of service	Denial of Service	Kalluri et al. (2016)	7.d
IEC 60870-5-104 packet flooding denial of service	Denial of Service	Kalluri et al. (2016)	7.e
ARP flooding against PLC	Denial of service	Niedermaier et al. (2018)	7.f
UDP flooding against PLC	Denial of Service	Niedermaier et al. (2018)	7.g

4.3.8 Measurement unit

8.a Liang et al. (2016) review different types of false data injection attack. The attacks seek to change the estimated state in the system, which could have both a physical and an economic impact. The attacks are based on compromising sensors reporting values to the SCADA system. The article does not discuss how this may be achieved, but we assume that one method could involve injection of malware in the sensors.

Table 11: Threats to Measurement Unit

Threat	Category	Source	Threat Number
Inject malicious code or data in measurement units	Elevation of privilege	Liang et al. (2016)	8.a

4.3.9 AMI Server

9.a, 9.c These threats are adapted from a threat modeling of AMI by Tøndel et al. (2013).

9.b Sgouras et al. (2014) demonstrate in a simulation how a large percentage of smart meters is unable to receive communication from the AMI server when the server is subject to a distributed denial of service attack. Guo et al. (2015) simulate a scenario where many smart meters launch an DDoS attack on the AMI server. The attack can either be an attack on the protocol used, e.g. TCP SYN attack, or seek to generate such amounts of bandwidth that some of the legitimate traffic is dropped. A third approach involving the disruption of routing tables are considered out of scope.

Table 12: Threats to AMI Server

Threat	Category	Source	Threat Number
Smart meter spoofing of AMI server	Spoofing	Tøndel et al. (2013)	9.a
DDoS against AMI server	Denial of Service	Guo et al. (2015; Sgouras et al. (2014)	9.b
Unauthorized remote access to AMI server	Elevation of privilege	Tøndel et al. (2013)	9.c

4.3.10 Communication

10.a Aloul et al. (2012) argue that eavesdropping and traffic analysis is a threat to the smart grid. Examples of information of interest may be power usage, future price information and control structure of the grid. Additionally, SCADA communication may be of interest as it may be used to reverse engineer control commands for later malicious use.

10.b A threat to communication is the tempering of data. One version of this threat is to tamper with voltage measurements in the distribution domain, as discussed by Isozaki et al. (2015).

10.c, 10.d Kalluri et al. (2016) argue that plain text SCADA communication is vulnerable to man-in-the-middle and replay attacks.

Table 13: Threats to communication

Protocol	**Threat**	**Category**	**Source**	
Unspecified protocol				
	Eavesdropping and traffic analysis	Information disclosure	Aloul et al. (2012)	10.a
	Tampering of communication data	Tampering	Isozaki et al. (2015; Rizzetti et al. (2015)	10.b
	Man-in-the-middle attack	STIDE	Kalluri et al. (2016)	10.c
	Replay attack	Spoofing	Kalluri et al. (2016)	10.d

4.3.11 General Smart Grid Process and Data Store

11.a Irmak and Erkek (2018) argue that unused yet open port can constitute a vulnerability. Open ports represent an increased communication interface and hence increase the attack surface.

11.b Irmak and Erkek (2018) argue that SQL injection can pose a threat to the SCADA historian database. If the database fails to validate the query, an attacker may be able to write values to, and read values from a database.

11.c Irmak and Erkek (2018) claim buffer overflow to be a common threat in SCADA systems. The threat occurs when more data than anticipated is given as input. This can cause memory locations not associated with the input functionality to be overwritten. According to the authors, some exploits may use this to execute code with the privileges of the input process.

Table 14: Threats to general processes and data stores.

General Smart Grid Process				
	Unnecessary ports open	Elevation of Privilege	Irmak and Erkek (2018)	11.a
General Data Store				
	SQL injection	Tampering	Irmak and Erkek (2018)	11.b
Process/ Data Store	Buffer overflow	Elevation of Privilege	Irmak and Erkek (2018)	11.c

4.4 Threats from templates and previous attacks

In addition to the literature, two existing templates from Microsoft have been used as sources for threats. These are the templates referred to in the thesis as the default template and the Azure Cloud vices template. The default template includes stencils and threats to model generic software. It was used as input as it was expected to include generic threats likely to be relevant in the smart grid template. The Azure template targets the Azure Cloud Services. This template was used as it was expected to include relevant aspect for modeling cloud services. Both templates included threats that were included or adapted for the smart grid template. Which threats are included from these templates were indicated in the description in section 5.7.

The last source of threats are reports from three attacks on OT systems. These are the Stuxnet attack, discussed by Falliere et al. (2011), the Crashoverride attack discussed by Slowik (2018) and Slowik (2019), and the Triton attack discussed by Johnson et al. (2017). Among the three, only Crashoverride targeted the power grid. Stuxnet and Triton is included as it is assumed that attacks on OT systems in other sectors would be relevant to OT systems in the power grid. Which threats that are inspired by these attacks are indicated in the description of template threats in section 5.7.

Stuxnet is arguably the best-known attack on OT systems. It was discovered in 2010 and is believed to have targeted the Iranian nuclear enrichment facility in Natanz. The malware contained logic to cause specific centrifuge equipment to operate in such a way that they would break. This was implemented through a Man-in-the-middle attack between the operator and the equipment. As a result, the malware would block the operator's commands and feed the HMI with false recordings. The malware had a highly autonomous design and exploited no less than four previously undisclosed vulnerabilities in the Windows operating systems. It is believed to have compromised over 100 000 personal computers worldwide and 24 OT systems in addition to its assumed target.

Crashoverride was an attack on the Ukrainian power grid in 2016. The attackers opened circuit breakers in the grid, but the blackout is reported to have been less severe than in the 2015 attack. Despite the modest impact, analysis of the attack revealed more ambitious goals. The malware contained protocol specific modules for four protocols used in the grid. While not as autonomous as Stuxnet, it contained some functionality for auto discovery of components on the network. Additionally, the malware attempted to deny the service of a safety relay in the grid. This has led some to believe that the attacker had an intention of physically damaging equipment.

Triton was an attack on a Saudi Arabian petro-chemical plant in 2017. The attack lead to a shutdown of the plant caused by an attack on the plants safety instrumented system. The attack is believed to be the first attack on a safety instrumented system and may have had severe consequences had it succeeded.

5 Chapter 5

Threat Modelling Tool and Developed Template

This chapter discusses the Microsoft threat modeling tool and the created Smart Grid template. The template was developed using version 7.3.00929.2 of the Microsoft threat modeling tool. The tool can be downloaded from Microsoft[13]. The template can be downloaded from Github[14].

5.1 The functioning of the threat modeling tool

The Microsoft Threat Modelling Tool provides an environment for analyzing threats to systems. Threat modeling has previously primarily been employed for software systems. The tool provides users with the possibility of defining their own templates. This allows users to include suitable stencils, threats, and configurations. The tool offers great flexibility in defining templates, which facilitates the threat modeling of various types of systems. This thesis has identified templates related to Azure Cloud services, Medical Devices, and general software, all from Microsoft[15], and a template related to Smart Vehicles from nccgroup[16].

The larger threat modeling tool ecosystem can be seen in Figure 10. The contribution of this thesis is marked in green.

[13] https://www.microsoft.com/en-us/securityengineering/sdl/threatmodeling
[14] https://github.com/larshfl/MS-TMT-Smart-Grid-Template
[15] https://github.com/microsoft/threat-modeling-templates
[16] https://github.com/nccgroup/The_Automotive_Threat_Modeling_Template

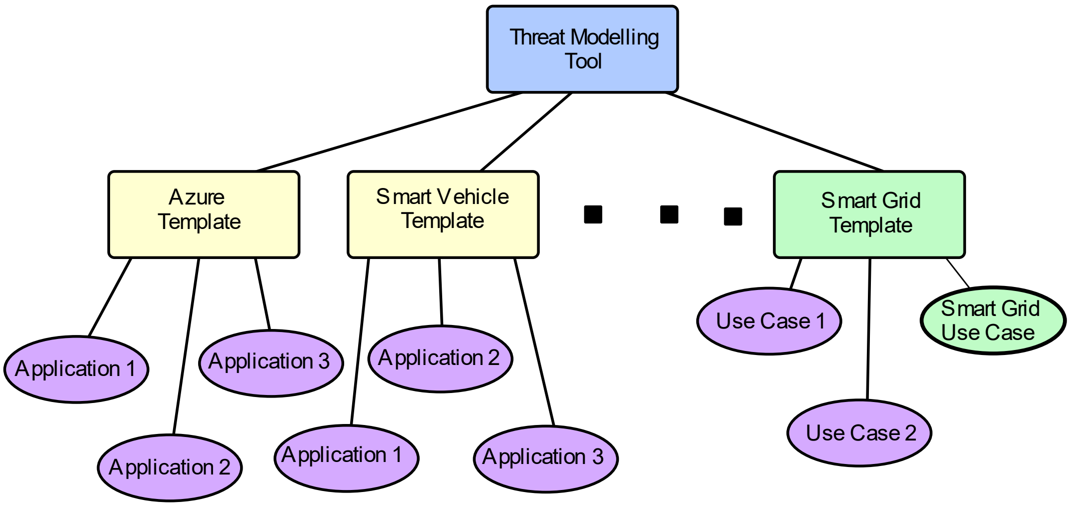


Figure 10: Threat Modelling Tool ecosystem

5.2 Assumptions made for the Smart Grid Template

The template is based on the assumptions listed below.

- The threats do not include natural failure rates due to wear, tear, natural disasters, and similar. Only cyber threats caused by malicious actors are regarded.
- The process stencils represent the functional behavior of equipment and are not directly transferable to the operating system or software processes. Consequently, many of the stencils can be broken down into more detailed units. This is not done in this thesis as it may cause the models to become overly detailed and extensive.
- Only databases are included for storing data. Other forms of storage, for instance, files, memory, cache, or computer permanent storage, are not included.
- Insider threats[17] and physical access related threats are not included.
- Threats originating from external service or maintenance personnel, or their potentially compromised equipment, is not included.
- Threats involving sabotage are not included.
- Threats originating from forgotten data flows are not modeled. As an example, the template does not account for what threats may arise from a VPN connection not included.
- All default properties in the template are set to the values that generate the most threats. This is done to ensure that no threats are overlooked because default properties were not changed.

[17] An insider threat is a threat originating from within the organization, for example, from an IT employee or SCADA operator.

5.3 Data Flow Diagrams

The threat modeling tool is based on creating DFDs of the system or application. DFDs were introduced by Larry Constantine in 1967. According to Shostack, models of data flow are well suited for threat modeling. Swiderski and Snyder (p. 94) regard DFDs as the preferred way to model systems in threat modeling. Advantages, according to Swiderski and Snyder (2004, p. 87), include their hierarchical nature and that various parts can be described in varying degrees of detail. This may rely on the use of a “multiple process” element introduced by the authors. This is a circle with a double line boundary that represents a more detailed DFD. As this element currently is not available in the Microsoft TMT, it is not included here.

An explanation of the elements in a DFD is shown in Table 15. The DFD elements used in this thesis have a slightly different meaning from the classical DFD elements. This is done to make them more suitable for threat modeling in the smart grid.

Table 15: DFD element description. Shostack (2014, p. 45).

Element	Appearance	Traditional DFD Description	Smart Grid DFD Description	Traditional / Smart Grid Examples
Process	○	Any running code	A functional part or system that performs an action	Code written in C, C#, etc. / SCADA Server,
Data flow	↘	Communication between processes, or between processes and data stores.	Flow of data between functional processes, data stores or external entities. Models the flow of application level data. Protocol acknowledgements are not included.	Network connections, HTTP, RPC, LCP / Generic data flow, DNP3
Data store	=	Things that store data	Databases	Files, databases, shared memory segments/ SCADA databases
External entity	▭	People or code outside your control	Actors outside the control of asset owner	Customer, Microsoft.com / Cloud services, Vendor organizations

5.4 Trust boundaries

Shostack (2014, pp. 5 - 6) argues that trusted boundaries should be drawn whenever different actors control different elements. Examples of relevant places include network interfaces, different physical computers, virtual machines, and organizational boundaries. Shostack (p.50) provides a useful question to investigate whether boundaries are needed. If everything in the system has the same privilege and access to everything else in the system, the system can be put into a single trust boundary. Swiderski and Snyder (2004, pp. 89 - 90) refer to boundaries as privilege boundaries, used to separate elements of different privileges.

Our smart grid template uses trust boundaries in a similar way. The trust boundary separates trusted stencils from untrusted stencils. This is similar to how it is used by Khan et al. (2017), who use it to separate trustworthy and untrustworthy systems. Another way to view the smart grid trust boundary is that it should be placed on all communication interfaces where threats are expected. All threats in the template are tied to trust boundaries. During the threat modeling, boundaries should be included on all interfaces the program shall investigate and generate threats for. An example of interfaces that could be of interest is data arriving across untrusted networks. Other interfaces that may be of interest may be data crossing LAN boundaries. A final example could be interfaces sending control commands to processes controlling the power in the smart grid. In that case, these data flows should be modeled to cross trust boundaries.

5.5 Stencils added to the Smart Grid Template

The stencils included in the smart grid template is shown in Table 16. The included stencils are derived from literature, the use case, and discussions with SINTEF energy experts. All stencils are assumed to possess the capability of communicating and executing code. Consequently, most threats relate to the stencil categories rather than the individual stencils. This means that individual stencils, for instance, the Circuit breaker Process and the IED Process in many cases, generates the same threats. The default value of the stencils is the option resulting in the largest attack surface, as explained in section 5.2.

General smart grid processes represent the functional behavior of smart grid components or systems. Of focus are their communication interfaces towards other processes, as threats are generated for (source, flow, target)-tuples. This is shown in Figure 17 and described in section 5.8.

Generic External Interactors represent systems or actors that interact with the smart grid. Threats originating from an external interactor are included. Threats affecting external interactors are not included. For instance, the threat of an attacker exploiting weak credentials on a VPN connection from a vendor organization into a smart grid process is included. The threat of a Denial of Service attack on a vendor organization, originating from the smart grid or elsewhere, is not included.

The generic data store represents storage of data. The smart grid template only included databases. Other forms of data storage, for instance, cache, memory, and disk are assumed to be part of the smart grid processes. Like the case with processes, the data stores do not map directly to ICT components. Data stores include the data being stored and the necessary infrastructure to read, write and protect the data.

Generic data flow represents the flow of all forms of “useful” or application-level data. This means that, for instance, protocol acknowledgments are not included. An introduction to the added protocols is provided in section 5.6.

Table 16: Stencils included in the smart grid template.

General smart grid processes	Generic Trust Border Boundary	Generic External Interactor	Generic Data Store	Generic data flow
AMI Server Process	-	External network	Database	Operator input
Automatic Voltage Regulator Process	-	GPS Clock Process	-	GPS
Circuit breaker Process	-	GPS satellite Process	-	IEC 60840-5-104
IED Process	-	Human Operator	-	-
IoT Cloud Gateway Process	-	Service personnel	-	-
IoT Device Process	-	Vendor organization	-	-
IoT Field Gateway Process	-	-	-	-
Measurement Unit	-	-	-	-
Onload Tap Changers Process	-	-	-	-
PMU Process	-	-	-	-
RTU Process	-	-	-	-
SCADA Server Process	-	-	-	-
Smart Meter Data Concentrator Process	-	-	-	-
Smart Meter Process	-	-	-	-
Substation Process	-	-	-	-
Transformer Process	-	-	-	-
Virtual RTU Process	-	-	-	-
Windmill Process	-	-	-	-

5.6 Application-level protocols included in the template

5.6.1 IEC 60870-5-104

Maynard et al. (2014) claim that the IEC 60870-5-104 protocol is widely used in water, gas, and electricity. According to Slowik (2018), the Crashoverride malware had a payload module targeting this protocol, further illustrating its relevance.

According to Kalluri et al. (2018), the standard does not offer confidentiality, integrity, or authenticity. IEC 60870-5-104 (2006, p. 15) itself claims that security mechanisms are outside of scope. According to Maynard et al. (2014), IEC 104 is compatible with IEC 62351. This can protect against threats such as eavesdropping, replay, and spoofing, among others. The authors claim that IEC 62351 is rarely deployed. Maynard et al. demonstrate how a replay and a man-in-the-middle attack can be conducted on simulated IEC 104 communication. Kalluri et al. conduct tamper and replay attacks on IEC 104 communication between an RTU and MTU. IEC 60870-5-104 (2006, p.

21) indicates that the protocol runs on top of the TCP/IP protocols. Because of this, the protocol can be assumed to be vulnerable to IP and TCP level threats.

Due to its use in the use case and position I OT systems, the IEC 60970-5-104 protocol was added to the template. The properties of confidentiality, integrity, authenticity, replay protection, and non-repudiation were left for the user to determine. This allows for scenarios where IEC 62351 is used, even if this is reported to be rare. The layer 3 and 4 protocols were restricted to IP and TCP.

5.7 Threats added to the Smart Grid Template

This section describes the threats included in the template. The threats are based on threats from literature, existing templates, and cyber-attacks on OT systems. The added threats are grouped into STRIDE-categories. The categories are defined in section 3.5.1. Some of the threats are difficult to place in a particular category. This is especially true for some of the elevation of privilege threats. Originally, this is a category for threats that elevate privileges. Some of the threats included under Elevation of Privilege in this thesis needs elevated privileges as a necessary precondition. Execution of malware is an example. For an attacker to be able to execute malware, one can assume that an elevation of privilege exploit is needed.

The threats in each category are arranged hierarchically in Figure 11 to Figure 16 to help to associate the threats to each other. A threat is a sub-threat of the threat it connects to above. This shares similarity with STRIDE threat trees, used by Shostack (2014, pp. 429 - 476). These trees are generally a bit more specific with regards to threats, and the trees are related to both category and DFD elements. For instance, instead of having one tree for denial of service, there is a separate tree for denial of service against a process, a data store, and a data flow.

5.7.1 Spoofing

Control input spoofing: This is the threat of an attacker sending control input to a process, pretending it originates from a legitimate source. Such sources may be Windmill Process, Substation Process, IED Process, Automatic Voltage Regulator Process, Circuit breaker Process, Onload Tap Changer Process, PMU Process, RTU Process, and virtual RTU Process. In this way, the attacker can cause a process responsible for controlling the grid to behave in a malicious way. The threat is inspired by literature threat 6.d relating to DERs and generalized to the processes mentioned above.

Spoofing the source: This is the threat of an attacker pretending to be a legitimate process, data store, or external interactor. The attacker would attempt to exploit this by making processes or data stores believe the communication originates from a trusted source. This could lead to unauthorized access to a process or to incorrect data being sent to a process. The threat is adapted from the default template.

Spoofing the target: This is the threat of an attacker pretending to be a legitimate process, data store, or external interactor. The attacker would attempt to exploit this by making processes, data stores, or external interactors believe it is sending data to a legitimate target. This may lead to information being sent to the attacker instead of the legitimate process. The threat is adapted from the default template. Additionally, target spoofing is mentioned in literature threat 1.b.

Spoofing of Data Store Source: This is the threat of an attacker sending malicious data to a process by pretending to be a legitimate data store. This could cause the process to behave in a malicious way by tricking it into basing decisions on false data. The threat is included from the default template.

MITM-Attack: This is the threat of an attacker performing a MITM attack on communication between any of the processes, data stores, or external interactors in the grid. This general threat is inspired by the Azure template, where the threat is generated for IoT related traffic. Different forms of MITM attacks have been identified in the literature, as indicated by literature threats 3.d, 6.i, 6.o, and 10.c. The smart grid template generalizes this threat to all types of communication.

ARP MITM-Attack: This is the threat that an attacker performing an ARP MITM-attack against communication between processes or data stores on a local network. The threat is not generated if the communication is authenticated, does not use ARP or crosses cross a network boundary. The last condition is included as ARP is restricted to one network and does not propagate to connected networks. The threat is inspired by literature threat 3.d and 6.o, related to SCADA servers and DERs. The smart grid template generalizes this threat to all ARP communication.

Reuse of authentication tokens: This is the threat of an attacker acquiring cryptographic key material from a legitimate IoT Device Process or an IoT Field Gateway Process and uses it to communicate with an IoT Field Gateway Process or an IoT Cloud Gateway Process. Falsely authenticating as someone else may give the attacker the possibility of sending false data to the process or receiving data meant for someone else. As stolen cryptographic keys can be used to authenticate a malicious process as both a target and a source in a communication, the threat is marked as a sub threat of both target and source spoofing. The threat is included from the Azure template.

GPS spoofing: This is the threat of an attacker sending false GPS signals to a PMU Process. PMUs generally rely on GPS to timestamp their measurements. These measurements may later be used for state estimation, and a successful GPS spoofing attack may cause the grid operators to estimate a wrong state. The threat is inspired by literature threat 4.a.

Replay attack: This is the threat of an attacker capturing a message from the network and resending it at a later time. A replay attack is assumed to be possible for communication between any types of stencils if the data flow does not provide replay protection. A replay attack is essentially a way of providing the sender with bad input. The most serious consequences can be achieved if the attacker has knowledge of the target under attack. The threat is inspired by literature threat 10.d, which claims that plain-text SCADA systems may be vulnerable to replay attacks. The smart grid template generalizes this to all communication.

Auto-generation of valid authentication tokens for IoT Hub: This is the threat of an attacker generating false authentication tokens and using these to authenticate to an IoT Hub. False authentication in this way may give the attacker the option of sending false data to the IoT hub or learning the contents of confidential information. The threat is included from the Azure template. The threat is marked as both a target and source spoofing threat as this thesis assumes that a token can be used to authenticate both as a target and as a source.

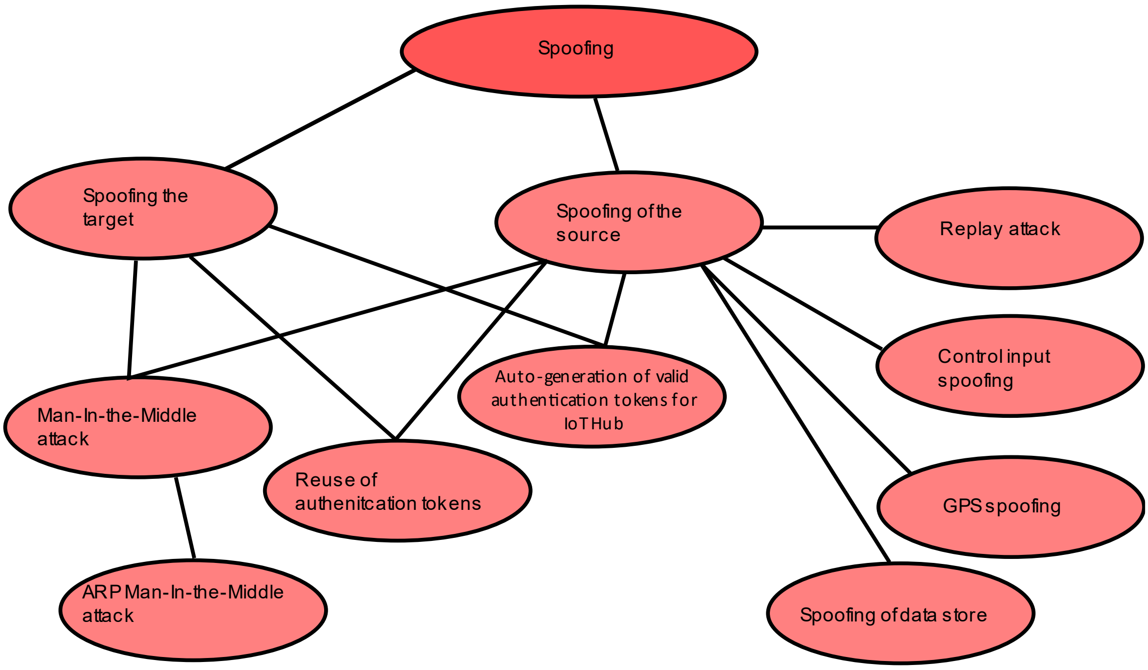


Figure 11: Overview of spoofing threats included in the smart grid template.

5.7.2 Tampering

Tampering of communication: This is the threat of an attacker tampering with a data flow. The threat is not generated if the communication provides integrity or if the communication is human input or output. The threat is included as a general "parent" threat, not tied to any method. The consequence of a successful attack can be that the target stores a false value in a database or otherwise behaves in a malicious manner. An example of tampering can be found in literature threat 1.c.

Injection of data in optical fiber: This is the threat of an attacker injecting data into communication over optical fiber. The consequences are much the same as the for the tampering of communication- threat. The threat is inspired by literature threat 6.f, related to communication with DERs via fiber optical cables. The smart grid template generalizes this to all fiber optic communication.

SQL injection attack: This is the threat of an attacker performing an SQL attack on an SQL relational database that does not sanitize input. An SQL injection attack may corrupt the database content or reveal the content to the attacker. The threat is inspired from literature threats 3.c and 11.b related to SQL attacks on SCADA historian databases and from SQL injection threats in the default template.

Corruption of Data Store by tampering of data flow: This is the threat of an attacker tampering with a data flow going to a data store. The consequence of such an attack is that false data is stored in the data store. The threat is adapted from the default template.

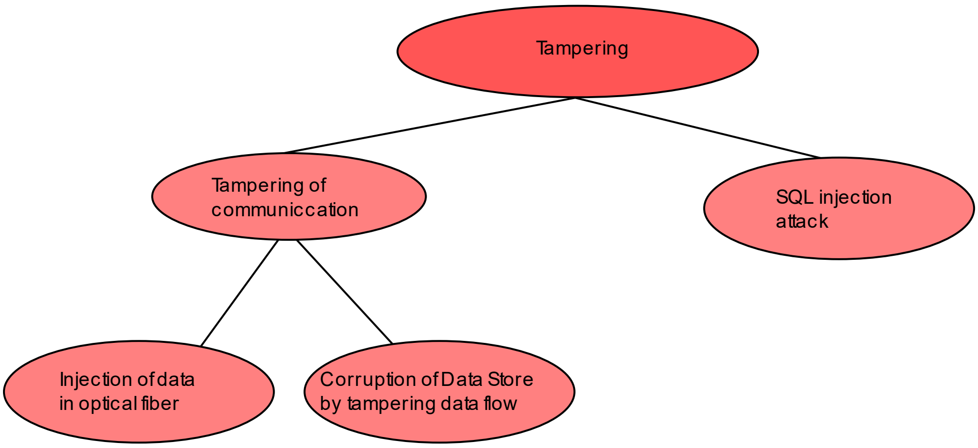

Tampering
Tampering of communiccation
SQL injection attack
Injection of data in optical fiber
Corruption of Data Store by tampering data flow

Figure 12: Overview of tampering threats included in the smart grid template.

5.7.3 Repudiation

Repudiation of received data: This is the threat of not being able to prove whether a process or data store did receive a message. Lack of such proof may make it hard to investigate attacks after they have happened. The threat is not generated if the actions on the database or data store are logged. The threat is adapted from the default template. This threat is highlighted in literature threat 1.d.

Repudiation of sent data: This is the threat of not being able to prove whether a process or external interactor did send data or not. Lack of such proof may make forensics analysis hard. The threat is not generated if the source logs its actions. The threat is inspired by the repudiation of received data threat. This threat is highlighted in literature threat 1.d.

Repudiation of actions on smart grid process: This is the threat of not being able to prove whether an action was committed on a process or not. This can lead to repudiation claims after an attack and make it harder to attribute an attack to an actor. The threat is inspired by similar threats in the Azure template related to databases and cloud gateways.

Repudiation of actions performed on data store: Same as the threat above, but for data stores instead of processes.

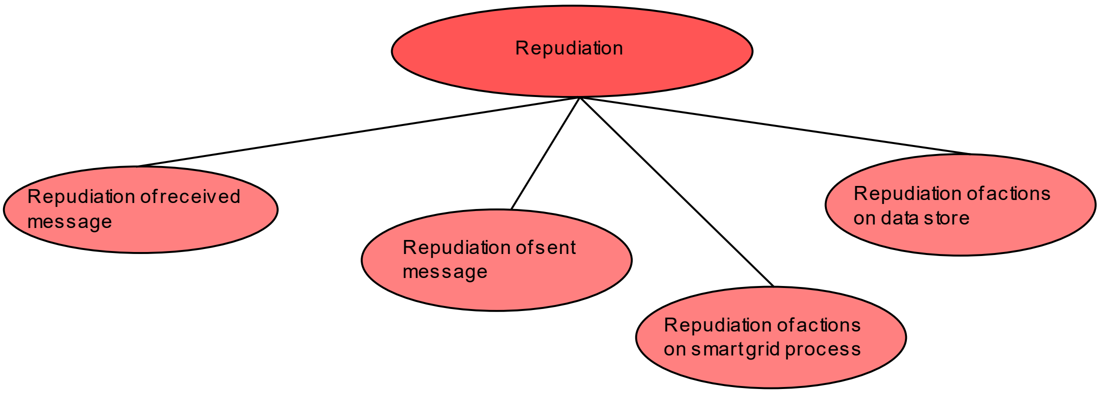

Repudiation
Repudiation of received message
Repudiation of sent message
Repudiation of actions on smart grid process
Repudiation of actions on data store

Figure 13: Overview of repudiation threats included in the smart grid template.

5.7.4 Information Disclosure

Data Flow Sniffing: This is the threat of an attacker learning the contents of a data flow between any components in the grid. If the flow does not offer confidentiality, this could lead to theft of confidential information or be used to reverse engineer commands in preparation for a later attack. The threat is based on literature threats 1.e, 4.b, 5.a, 6.a, 6.g, and 10.a, which relates to the disclosure of transmitted information.

Wiretapping of fiber optic cables: This is the threat of an attacker wiretapping optical fiber cables to learn the content of the communication. If the flow does not offer confidentiality, the consequences are the same as for the data flow sniffing threat. The threat is based on literature threat 6.a, which relates to wiretapping at the physical level and particularly optical fiber.

Weak Credential Transit: This is the threat of an attacker sniffing credentials as they are transmitted between components. If transmitted credentials are not encrypted, they may be sniffed and used to obtain elevated privileges. The threat is adapted from the default template.

Weak Credential Storage: This is the threat of an attacker obtaining credentials from a data store. Such credentials may be used to obtain elevated privileges. The threat is adapted from the default template. The default template argues that stored credentials may be stolen, tampered or disclosed. To prevent this, credentials should ideally be hashed or encrypted.

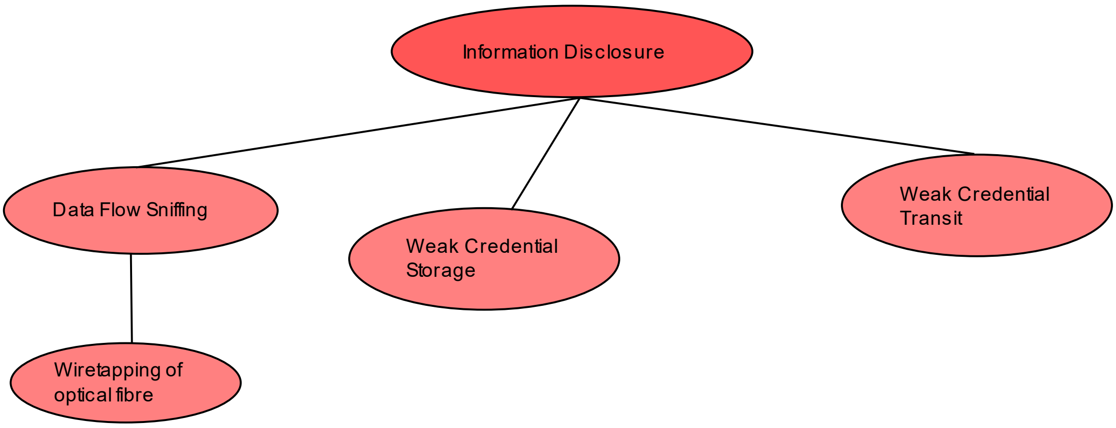


Figure 14: Overview of information disclosure threats included in the smart grid template.

5.7.5 Denial of Service

External distributed denial of service attack: This is the threat of distributed attack on the availability of a process originating from an external network. Such an attack may cause the target to become temporarily unavailable to legitimate communication from other sources. The threat is inspired by literature threats 1.a, 3.b, 1.f, 6.b, 7.a, 7.d, 7.e, 7.f, 7.g, and 9.c, which all deal with generating large amounts of traffic, possibly from distributed hosts. An external network provides the possibility for an attacker to compromise many hosts without the knowledge of the asset owner and use these to launch an attack.

Flooding denial of service attack: This is the threat of an attack on the availability of a process by flooding it with data. A flooding type of attack is created by attempting to exhaust some resource by generating large amounts of traffic. Various forms of this attack exist, as elaborated by literature threats 3.b, 1.a, 1.f, 6.b, 7.a, 7.d, 7.f, 7.g and 9.c. This general threat is included to encourage reflection on what type of flooding attacks may be present in a use case.

Smart meter-based DDoS attack on AMI server: This is the threat of an attacker compromising many smart meters and subsequently using them for a DDoS attack on an AMI Server. Such an attack could leave the AMI server unavailable for a period. The threat is inspired by literature threat 9.c.

Denial of service through specially crafted message: This is the threat of an attack on the availability of a circuit breaker using a specially crafted package. Such an attack can cause the circuit breakers not to react in a proper way to changes in the grid. According to Slowik (2019), an attempt was made in the 2016 Ukraine attack to place a safety breaker in a firmware update mode, leaving it in a state unable to perform its normal function. The attack attempted this by exploiting a vulnerability in the device by sending it a specially crafted UPD packet. Literature threat 1.f claims that specially crafted messages may be a way of denying service.

Signal jamming: This is the threat of an attacker jamming the wireless communication between two components in the grid. The threat is inspired by literature threat 6.j concerning communication with a DER. The smart grid template generalizes this threat to all types of wireless communication.

TCP-SYN or UDP flooding: This is the threat of an attack on the network availability of components in the grid communicating via TCP of UDP. The attack is performed by generating large amounts of network traffic, blocking legitimate traffic. A successful attack causes the target to become unresponsive for a period. The threat is based on literature threat 6.b, 7.d, 7.f, and 7.g. These threats regard DERs and IEDs. The smart grid thesis has generalized the threat to all components using TCP of UDP.

ARP flooding: This is the threat of an attack on the network availability of components in the grid communicating using the ARP protocol. The threat is based on literature threat 7.f, which relates to PLCs. The smart grid thesis generalizes this to all smart grid processes using ARP.

IP flooding: This is the threat of attack on the network availability of components in the grid communicating via IP. The attack is performed by generating large amounts of IP traffic, blocking legitimate IP traffic from accessing the target. The threat is based on literature threat 7.e. This threat concerns IP flooding in an IED scenario. The smart grid template generalizes this threat to all IP based communication.

Interruption of data flow: This is the threat of an attacker disrupting the data flow, attacking the network availability of the target. This is a general threat to encourage reflection of how such an interruption may occur in the use case under consideration. This threat is adapted from the default template.

Data Store Inaccessible: This is the threat of an attacker making the data store inaccessible. This is a general threat to encourage reflection of how such an interruption may occur in the use case under consideration. This threat is adapted from the default template.

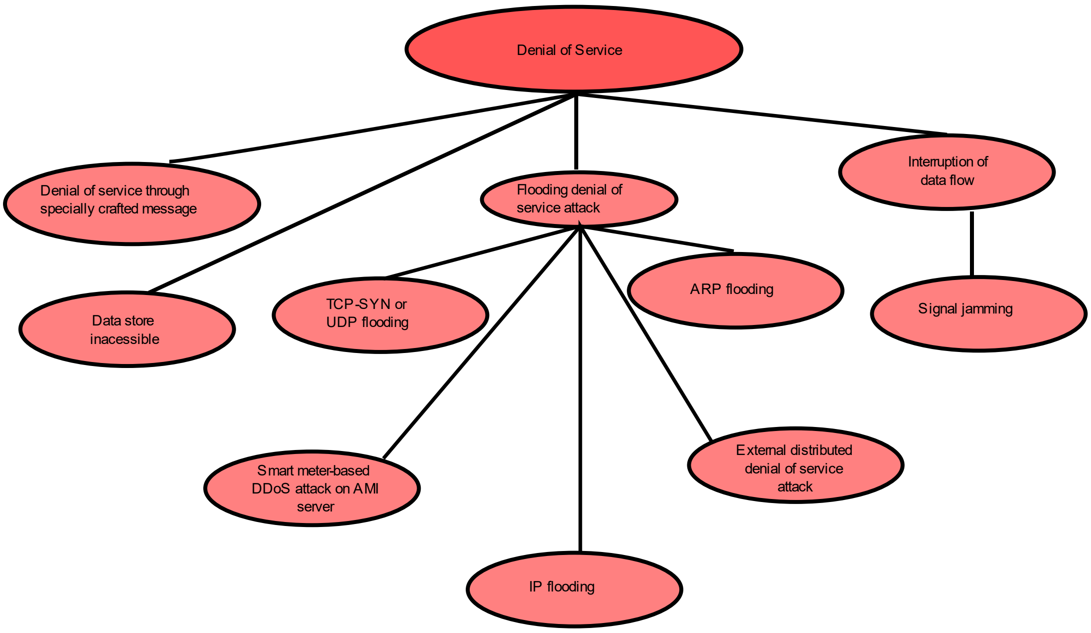

Denial of Service
Denial of service through specially crafted message
Flooding denial of service attack
Interruption of data flow
Data store inacessible
TCP-SYN or UDP flooding
ARP flooding
Signal jamming
Smart meter-based DDoS attack on AMI server
External distributed denial of service attack
IP flooding

Figure 15: Overview of denial of service threats included in the smart grid template.

5.7.6 Elevation of Privilege

Elevation of privilege in database due to poor configurations: This is the threat of an attacker obtaining greater privileges than intended in a database. More specifically, the threat is generated if access to a database is not configured based on least privilege. Least privilege involves that a user does not have more permissions than what is needed. This threat is adapted from the Azure template.

Execution of malware: This is the threat of an attacker executing malware in a process. Execution of malware has been observed in all the three attacks on industrial control systems mentioned earlier, Triton, Crashoverride, and Stuxnet. Injection of malware is also regarded as a threat in the literature, as indicated by threat 1.h, 7.b, 7.d

Exploitation of publicly disclosed vulnerabilities: This is the threat of an attacker exploiting a publicly disclosed vulnerability in a process or a data store in order to obtain elevated privileges and conduct an attack. Vulnerabilities are continuously discovered and disclosed. Failure to update systems after such vulnerabilities have been made publicly known lowers the effort for conducting an attack. According to Slowik (2019), an example of this can be found in the 2016 Ukrainian attack. The attackers attempted to exploit a vulnerability that was already publicly known. The threat is additionally inspired by the Azure Cloud Service template, where the threat is generated for IoT Devices or IoT Gateways. The smart grid template has extended the threat to regard all smart grid processes and data stores.

Exploitation of zero-day vulnerabilities: This is the threat of an attacker exploiting a vulnerability that is not publicly known. This is referred to as a zero-day vulnerability. Attacks leveraging such vulnerabilities have been observed in the Stuxnet and Triton attacks. Defending against such attacks is hard. The threat is included to encourage reflection on such threats. This can, for instance, include reflecting over worst-case consequences if an attacker obtains kernel-level privileges in a process.

Exploitation of remote update functionality: This is the threat of an attacker exploiting remote update functionality in a process in order to execute malware on the system. As elaborated by threat 6.c, failure to verify the integrity of patches and updates can pose a threat to the grid. Execution of malware are key in most cyberattacks and exploiting such update functionality must be regarded as an attractive goal for an attacker.

Exploitation of unused services or features: This is the threat of an attacker exploiting unused services or features in order to access and obtain elevated privileges on a process or data store. The threat is adapted from the Azure Cloud Service template. In that template, the threat is included for IoT devices and IoT gateways. In the smart grid template, this is extended to all smart grid processes and data storage. An example of such services may be open ports, as identified in threat 11.a. In their assessment of the 2015 attack on the Ukraine power grid, ICS-CERT U.S Department of Homeland Security (2016) recommends that ports are closed and unused services turned off. Staggs et al. (2017) argue that system hardening can be used as a mitigation technique. This includes disabling unnecessary remote interfaces, removing unused interfaces and functionality, and adjusting default configurations to fit the operating environment.

Lack of input validation: This is the threat of an attacker giving malicious input to a process or data store in order to obtain elevated privileges. A well-known form of such a threat is a buffer overflow attack, as highlighted in literature threat 11.c. Lack of input is also included in the default template. The template states that improper input is the root cause for many types of threats. The smart grid thesis generalizes the threat to all processes and data stores.

Unauthorized access through vendor VPN: This is the threat of an attacker obtaining access to a process through a vendor VPN. This threat is inspired by literature threat 6.k related to DERs and on the Crashoverride attack. ICS-CERT U.S Department of Homeland Security (2016) reports that a VPN connection may have been used by attackers to open circuit breakers in the 2015 Ukraine attack. The smart grid template generalizes the threat to all processes that are configured to be accessible through VPN.

Unauthorized execution of commands: This is the threat of an attacker executing unauthorized commands on a process or data store. The threat is adapted from the Azure Cloud Service Template, where is included for IoT related communication. The smart grid template generalizes this threat to all processes and data stores.

Weak authentication: This is the threat of an attacker obtaining elevated privileges on a process or data store due to weak authentication mechanisms. This can be the case if the authentication mechanism consists of easily guessable credentials or factory default credentials. The threat is adapted from the default template. In the Azure template, it is included for databases. Based on this, the smart grid template makes the threat relevant to both processes and data stores.

Remote control of circuit breakers: This is the threat of an attacker obtaining control of a remote circuit breaker in the grid. This threat is inspired by the 2015 Ukraine attack where ICS-CERT U.S Department of Homeland Security (2016) reports that the attackers opened circuit breakers in the grid. Malware containing functionality to open circuit breakers was also identified in the 2016 Crashoverride attack, according to Slowik (2018).

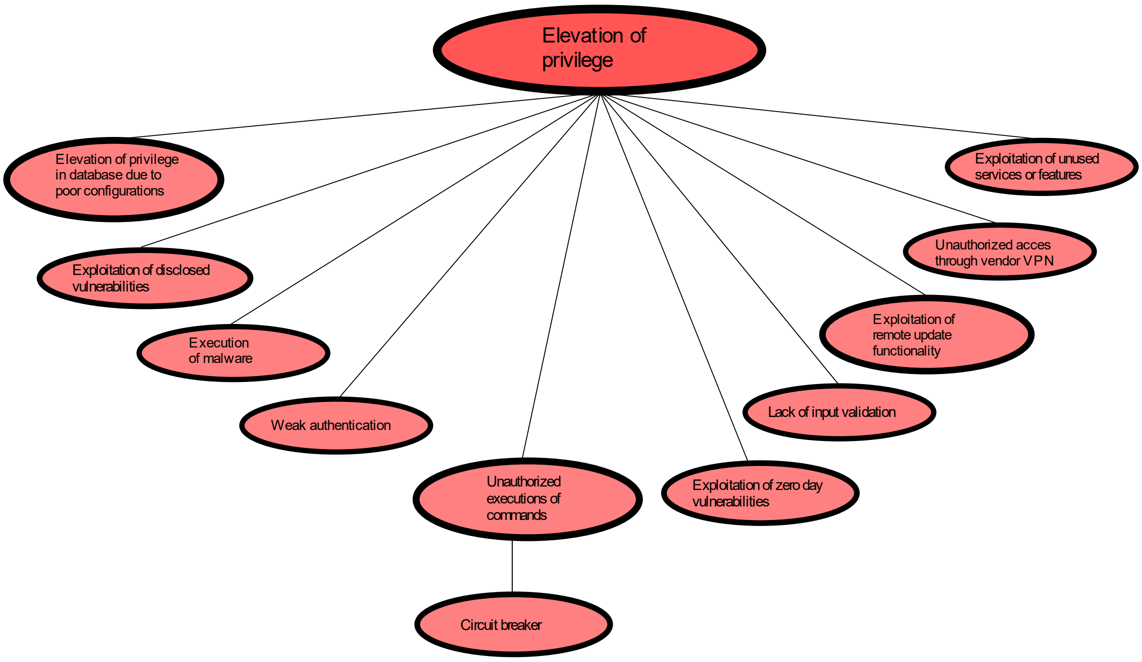

Elevation of privilege
Elevation of privilege in database due to poor configurations
Exploitation of disclosed vulnerabilities
Execution of malware
Weak authentication
Unauthorized executions of commands
Circuit breaker
Exploitation of zero day vulnerabilities
Lack of input validation
Exploitation of remote update functionality
Unauthorized acces through vendor VPN
Exploitation of unused services or features

Figure 16: Overview of elevation of privilege threats included in the smart grid template.

5.8 Generation of threats

Threats in the Threat Modelling Tool are based on a tuple of <source, communication flow, target>. This method of generating threats in STRIDE is referred to as STRIDE-by-interaction, according to Shostack (2014). An illustration is given in Figure 17.

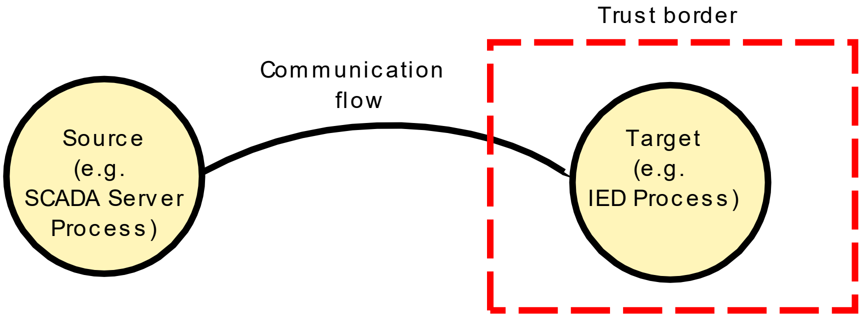
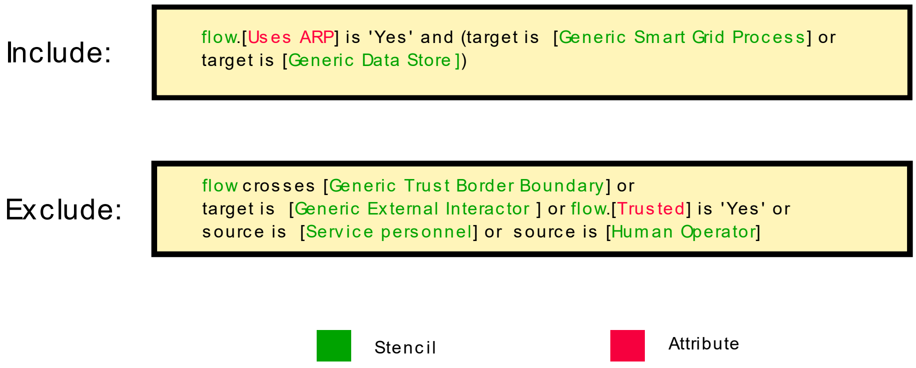


Figure 17: The basis for generating threats in the Threat Modelling Tool

Each threat added to the template is included or excluded in the analysis according to a logic related to the (source, communication flow, target)-tuple. An include logic that evaluates to true determines if the threat is included in the analysis. Threats that otherwise would have been included may be excluded if an exclude logic is true. An example of include and exclude logic is given below in Figure 18. This logic determines if the ARP flooding threat is included in the analysis.



Include:

flow.[Uses ARP] is 'Yes' and (target is [Generic Smart Grid Process] or target is [Generic Data Store])

Exclude:

flow crosses [Generic Trust Border Boundary] or
target is [Generic External Interactor] or flow.[Trusted] is 'Yes' or
source is [Service personnel] or source is [Human Operator]

🟩 Stencil 🟥 Attribute

Figure 18 Examples of include and exclude logic

5.9 Threats modeling process

This section provides a brief explanation of the Microsoft TMT and the threat modeling process. The tool is freely available for download and allows the user to work both with templates and threat models. Existing templates may be edited, or new templates may be created from scratch. These templates may then, in turn, be used to create threat models of use cases. Like the case of templates, threat models can either be created from scratch or edited. Several screenshots from the tool can be seen in Appendix B.

Chapter 5 Threat Modelling Tool and Developed Template

6 Chapter 6

Application of Framework on Use Case

This chapter discusses the threat modeling of a smart grid use case provided by SINTEF via the CINELDI project[18]. The CINELDI projects works with digitalization and modernization of electrical distribution. The result of the threat modeling is discussed in section 7.4.

6.1 Use case description

The use case is based on a setup shown in Figure 19. The setup is a simple example of transmission and distribution line control. A Transmission System Operator (TSO) controls a generation source and the high voltage part of the line. A Distribution System Operator (DSO) controls a windmill and the medium voltage part of the line. The high and medium voltage sections are interconnected by an On-Load-Tap-Changer (OLTC).

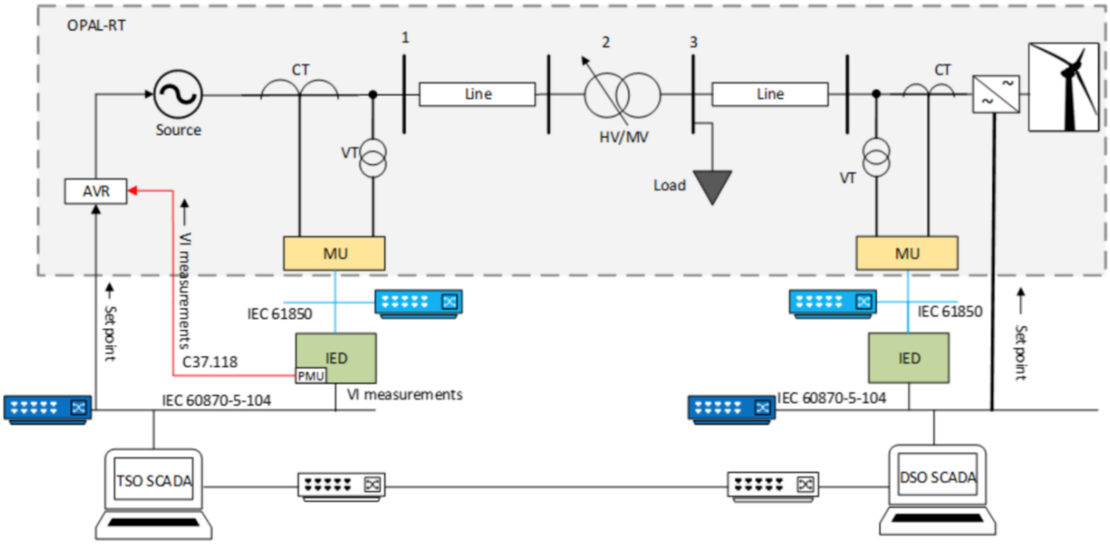


Figure 19: Use case set up. The use case and illustration have been provided by CINELDI WP2.

Modeling of the use case in the Microsoft TMT resulted in the model shown in Figure 20. Network routers, the TSO generation source, the current transformers (CT), and the voltage transformers (VT) were not included. This was done as they were assumed to be of less importance from a cybersecurity point of view. An extra IED was added to control the HV/MV OLTC together with a data flow to the TSO SCADA. Both the TSO and DSO SCADA processes are assumed to interact with a human operator and a database.

[18] https://www.sintef.no/projectweb/cineldi/

Communication between TSO/DSO and IED and Measurement Units (MUs) happen with the protocols IEC 60870-5-104, IEC 61850, and C37.118. The protocol used for TSO and DSO communication is unspecified. All processes are assumed to both send and receive data. The assumed physical mediums used in the communication between stencils are shown in Table 17. As indicated in section 5.8, a few threats are reliant on the underlying medium, but most of the threats are not affected by it.

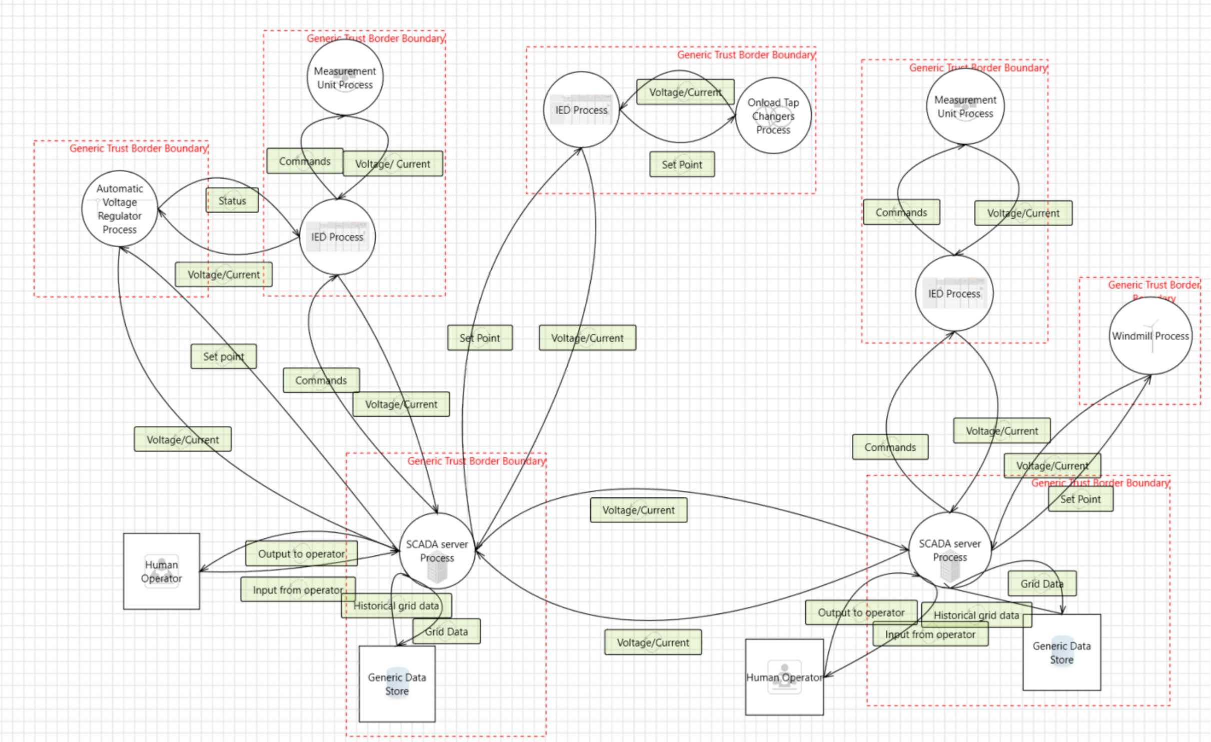


Figure 20: Use case modeled in the Threat Modelling Tool

Table 17: Assumed physical communication mediums in the use case.

Connection	Physical Medium
TSO – DSO	Fiber Optic
TSO – IED	Wireless
TSO – AVR	Wireless
TSO – Data Store	Wired
DSO – IED	Wireless
DSO – Windmill	Wireless
DSO – Data Store	Wired
IED – MU	Wired
IED – OLTC	Wired
IED – AVR	Wireless

6.2 Use Case assumptions and threat modeling results

The use case used for threat modeling in this thesis is from the OT part of the smart grid. OT systems tend to have a much longer lifespan than IT systems. Consequently, many OT systems in the industry can be expected to originate from a time when cybersecurity was less relevant than it is today. Because of this, this thesis assumes the use case consisted of largely insecure units. There is some support for this in the literature. Staggs et al. (2017) report that windmills often transmit command and control messages as clear text. Aloul et al. (2012) claim that current smart grid devices do not have the necessary processing power or storage for advanced cryptographic operations. This resulted in template default configurations being used. The template default settings are set to be insecure, as explained in section 5.2.

A meeting with specialists from SINTEF energy confirmed that with this particular use case, a loss of availability of components is less critical than malicious values or commands. Because of this, the thesis assumes the use case to have adequate backup solutions to deal with denial of service incidents. This may or may not hold true more generally. It is supported by Staggs et al. (2017), who claim that most windmills are able to continue to produce and transmit power even if communication with the control center is lost.

A threat modeling was performed on the system, as shown in Figure 20, with template version 1.0.0.776. The threat model generated 355 threats. Of the 355 threats, 66 threats were related to denial of service, 114 were related to elevation of privilege, 30 related to information disclosure, 47 related to repudiation, 77 related to spoofing, and 21 related to tampering. The threats are discussed in section 7.4

7 Chapter 7

Discussion

7.1 STRIDE-per-interaction and Threat Modelling as a Method for Smart Grid Security

Threat modeling and particularly STRIDE provide a systematic way of analyzing a system for threats. This ensures that threats are less likely to be overlooked. The smart grid is set to become increasingly dependent on ICT components and systems for it to function correctly. The smart grid stands out because of its size, amount of data generated, and the combination of IT and OT systems. Because of this, it may be beneficial to apply or draw inspiration from how the IT industry deals with security both on a component and network level.

A problem of STRIDE is that it does not include a method for evaluating the criticality of a threat. The aspect of evaluating the criticality of a threat in the smart grid is both difficult and potentially useful. The evaluation of criticality is difficult because as it can be expected to depend on the specific use case being studied. The evaluation of criticality is potentially useful as it can help structure an otherwise overwhelming number of threats. As discussed, there are several methods that can be used to evaluate the risk associated with threats. One possible approach that perhaps can be adapted to the smart grid is the use of a bug bar. This is an approach that, according to Shostack (2014, p. 181), is much used at Microsoft. One can imagine that a set of criteria for various levels of criticality can be used to categorize the threats.

It is sometimes difficult to know how to categorize a threat. For instance, one can argue that a man-in-the-middle attack is a spoofing attack, as the core of the attack is to pretend to be someone else. But a MITM attack can also be a starting point for eavesdropping, modification, injection, and DoS. Another example is the injection of data in an optical fiber cable. One can make the argument that it is either tampering of communication, or spoofing, as one would realistically inject data that appear to originate from a legitimate source.

7.2 The Microsoft Threat Modelling Tool

This section discusses the advantages and disadvantages of the Microsoft TMT with regards to threat modeling in the smart grid.

Advantages

The tool offers a large degree of freedom when defining templates. The creator is given the option of freely defining stencils, stencil categories, and stencil properties. This allows for threat modeling that does not need to follow the traditional DFD categories of process, data store, trust boundary, data flow, and external interactor. The same freedom is offered for defining threats, threat categories, and ways of categorizing threats. This provides the possibility of deviating from the STRIDE threats if necessary. The template creator is free to add other ways of classifying threats in addition to the default priority classification. For this purpose, this thesis included the threats shown in Table 18, but different properties could alternatively be included.

The Microsoft TMT provides a structured way to deal with identified threats. For each threat the user can set threat properties, decide whether the threat needs investigation, or provide a justification for the choices being made. The user can then choose to generate an HTML report containing all threats and their status. This report provides good documentation for the state of the security of the system being analyzed.

The Microsoft TMT offers the option of merging templates together. This can allow for convenient expansion of the template in the future. An interesting improvement of this functionality would be to allow merging on a stencil by stencil basis. One can imagine functionality where it is possible to merge just the stencils of interest into another template, along with all threats relating to these stencils.

Disadvantages
The tool allows the template creator to add stencils and derived stencils. The derived stencils inherit the properties of the parent stencil. Only one level of inheritance is allowed. This restriction has not caused inconvenience for the work on the Smart Grid template, but the reason for this limitation is not evident. This limitation may cause inconvenience in future expansions where the number of stencils and level of configuration detail increase.

The tool is not suited for analyzing threats that do not conform to the (source, flow, target)-tuple used in threat generation, shown in Figure 17. An example from this thesis is the desire to model forgotten VPN connections. VPN connections put in place by vendors can pose a threat to the smart grid asset owner. This holds even more true if the asset owner is unaware of any such VPN connections. To model this scenario, configuring all smart grid process stencils to have a VPN connection by default was attempted. The motivation for this was to encourage asset owners to reflect on the possibility of unknown and active VPN connections in their infrastructure. Scenarios of unknown data flows in the infrastructure cannot be conveniently modeled by the tool. By not specifying a flow in the threat generation logic (shown in Figure 18), the same threat is generated multiple times for each flow. Including a threat of unknown VPN connection in the use case threat modeling would have generated four duplicates for the right SCADA Server process in Figure 20. This problem may become even more relevant as cloud connectivity is expected to increase.

The Microsoft TMT did sometimes behave in peculiar ways. Changes made to threat properties in the template, shown in Table 18, did not become visible while performing threat modeling before long after. At times stencil property names in the threat generation expressions were replaced by hex values after the program was started. Fixing this involved manually renaming all affected properties to their original names. At one point, the stencil properties, shown in Appendix B, appeared to be listed several times.

Renaming stencils and properties are very inconvenient. It involves a process of manually updating each logical threat generation expression. A template wide search and replace function is not included.

7.3 Smart grid template

This section discusses the created smart grid template.

The threats in the template have, in many cases, been generalized so that they affect all stencils. Such a generalization may cause the template to generate false threats. By this is meant threats that would not be present in real systems. Choosing a narrower scope when generalizing threats would result in fewer included threats. A decision was made to accept potential false threats to minimize the risk of excluding real threats from the threat modeling.

Chapter 7 Discussion

The individual stencils included in the template do not have many configurable properties. Most properties are added at the generic level. This is because most threats are generalized, as described in the paragraph above. The purpose of properties is to be used in the threat generation logic. An example is shown in Figure 18. As threats generally have been generalized to the generic level, this is where most properties are located.

The process stencils represent the functional behavior of a part or subsystem of the grid. Traditional threat modeling uses the process stencil to represent running code, as indicated in Table 15. A similar approach would be possible for the smart grid but is thought to would have greatly increased the complexity of the threat modeling process. A measurement unit can potentially run a communication process, measurement process, and possibly other processes related to a lightweight operating system. The number of processes can be expected to be greater for more complex template stencils such as a SCADA server or Windmill. This would have caused the number of processes in Figure 20 to more than double. Directly linking template processes to ICT components would increase the number of processes in a similar way. The windmill would have had to be broken down into smaller components. Instead, the functional behavior with a focus on communication interfaces was adopted. This allows for processes to represent systems of various abstraction levels. This is one possible approach. In their article on STRIDE for physical systems, Khan et al. (2017) advocate for creating a DFD for each physical component.

The smart grid template implements a database as the only data store. Other types of data store that could have been included are cache, memory, and files. All these types of storage are assumed to be included in the processes. Consequently, only larger and more dedicated data store was included in the form of the database stencil.

The smart grid template only has a single type of trust boundary. Crossing a trust boundary is a necessary condition for most threats to be included. Other templates, like the default template, includes several different trust boundaries. This thesis did not see a need for more than one type of trust boundary.

Four threat properties were included in the template. These are priority, loss of power, difficulties of implementing mitigation, and affected systems. The properties and values they can hold are shown in Table 18.

Table 18: Smart Grid template threat properties

Priority	**Loss of Power**	**Difficulties of implementing mitigation**	**Affected Systems**
High	Select	Select	Select
Medium	Immediate backout	Low	Both IT and OT
Low	May lead to future blackout	Medium	IT
High	No threat to power availability	High	OT

A weakness of the template is that it inevitably excludes some threats. This is particularly true because of the scope chosen for the various stencils. The threats that may arise inside the chosen boundaries of a SCADA Server process can be expected to be numerous. As an example, Shostack (2014, p. 65) outlines three different ways an attacker can attempt to spoof a file. This threat level is far below what is modelled in this thesis but serves as an illustration of how threats can hide at lower levels in the model.

Different threats have different levels of detail. The more general ones are included as they seek to encourage critical thinking of how such a threat may be realized in the system. The more specific threats are included to serve as more of a checklist.

7.4 Use Case Threat Modelling Results

Threat modeling of the use case shown in Figure 20 resulted in 355 identified threats. This is in line with Shevchenko et al. (2018) claim that STRIDE can generate a high number of threats for complex systems. The high number of threats represents a challenge. Apart from individual review, few good methods of classifying and treating the threats exist. Individual review of the individual threats takes considerable time for a high number of threats. A total of 355 identified threats for a use case of moderate complexity as the one studied in this thesis points towards scalability issues for more complex use cases. It should be noted that model configurations affect the number of generated threats greatly.

Individual review of each threat in the threat modeling is infeasible due to the scope of this thesis. Consequently, a subset of the threats is selected for closer review. The subset consists of ten of the threats for the communication between the left SCADA Server Process and the middle IED Process, connected to the OLTC. The threats are chosen so that every threat category in STRIDE is represented but apart from this the selection is somewhat arbitrary.

The review of the selected threats mostly follows the same procedure as would have been the case in the Microsoft TMT. The threats can be seen below. For each threat, the “Status,” “Priority,” and “Loss of Power” properties are determined. As all the threats in the use case are OT related, the “affected system” is OT for all threats. Consequently, this property is omitted below. The “mitigation difficulty” property has been omitted in accordance with the scope of the thesis, as outlined in section 1.5.

Spoofing

Spoofing the source

Description: This threat consists of an attacker attempting to pose as the SCADA Server process.
Status: Needs investigation
Priority: High
Loss of Power: High
Justification: If this succeeds, an attacker may be able to send false commands to the IED. Such commands could in turn cause the IED to alter the setpoint of the OLTC and potentially cause instabilities in the grid.

Spoofing the target

Description: An attacker may attempt to pose as the IED process, causing SCADA communication to be sent to the attacker instead of to the legitimate process.
Status: Needs investigation
Priority: Medium
Loss of power: High

Justification: Data being sent to the attacker has two consequences. One is the aspect of information disclosure. The other is the aspect of Denial of Service. As confidentiality is the least important requirement in OT, the threat of information disclosure is regarded as less critical. The threat of Denial of Service is evaluated to have a medium criticality, as it is assumed that the use case can handle a loss of network availability without a blackout.

Tampering

Tampering of communication data

Description: An attacker may attempt to tamper with the communication sent between the SCADA server process and the IED process.
Status: Needs investigation
Priority: High
Loss of power: High
Justification: An attacker who successfully tampers the communication may be able to send malicious commands to the IED Process controlling the OLTC Process. This may lead to a blackout scenario.

Repudiation

Repudiation of received data

Description: The IED Process may deny having received a message.
Status: Mitigated
Priority: Low
Loss of power: N/A
Justification: This threat regards the possibility of denying having received a message. Such a threat is quite relevant in, for instance, internet banking. In OT systems, it is mostly relevant for forensics analysis. The priority of this threat depends on the asset owner's tolerance. This thesis assumes that the asset owner accepts the risk, as it poses no threat to power availability. Based on this, it is given the status of mitigated.

Repudiation of sent data

Description: The SCADA Server Process may deny having sent a message
Status: Mitigated
Priority: Low
Loss of power: N/A
Justification: See **Repudiation of received message**

Information Disclosure

Data flow sniffing

Description: An attacker may learn the content of the information transmitted between the SCADA Server Process and the IED Process.
Status: Needs investigation
Priority: Medium
Loss of Power: N/A

Justification: This threat is classified with a medium priority as confidentiality is regarded less important in OT systems. There is a threat that information disclosure can serve as a stepping stone for more critical attacks. An attacker can, for instance, reverse engineer commands which later can be used for malicious intent.

Denial of Service

Signal jamming

Description: An attacker may attempt to jam the communication from the SCADA Server Process to the IED Process. This denies the service of the IED Process.
Status: Needs investigation
Priority: Medium
Loss of Power: N/A
Justification: This threat is classified with a medium priority because of the assumption that a loss of network availability is not enough to cause a blackout.

TCP-SYN or UDP flooding

Description: An attacker may attempt to deny the network availability of the IED Process by flooding it with TCP-SYN requests or UDP packets.
Status: Needs investigation
Priority: Medium
Loss of Power: N/A
Justification: This threat is classified with a medium priority because of the assumption that a loss of availability is not enough to cause a blackout.

Elevation of Privilege

Exploitation of remote update functionality

Description: An attacker may attempt to exploit the remote update functionality of the IED Process to download and subsequently execute malware on the IED.
Status: Needs investigation
Priority: High
Loss of Power: High
Justification: This threat is classified with a high priority, as it may lead to a blackout. An attacker who successfully exploits the remote update functionality may download malware that allows for full control over the IED Process.

Execution of malware

Description: An attacker may attempt to execute malware on the IED Process.
Status: Needs investigation
Priority: High
Loss of Power: High
Justification: This threat is classified with a high priority, as it may, among other things, lead to a blackout. A blackout can be achieved by first gaining control over the IED and then send malicious commands to the OLTC.

Khan et al. (2017) claim in their article on STRIDE for cyber-physical systems that tampering and spoofing generally pose the greatest threats to a cyber-physical system. *"Spoofing and tampering* *are especially critical and they impact the operations of other elements (particularly in the physical domain) resulting in more severe consequences for the system."* Khan et al. (2017). A similar conclusion can be drawn by reviewing the spoofing and tampering threats highlighted above.

Two spoofing threats are selected for review. Spoofing the SCADA server allows an attacker to issue commands to the IED controlling the OLTC. This can have severe implications for power availability. Based on the use case, this causes the distribution part of the grid to be disconnected from the rest of the grid. One can imagine this to be even more severe if the windmill is attacked simultaneously. Spoofing the IED may cause the SCADA server to send commands to the attacker instead of the IED. This scenario has two implications. The first is that the attacker discovers the protocols and commands used to control the IED and OLTC. This may function as reconnaissance and be exploited on a later occasion. We argue that this is less critical. Actors motivated to attack the grid are often nation-states or nation state-backed groups. Such actors can be expected to be able to acquire this knowledge either way. The Stuxnet attack, discussed by Falliere et al. (2011), the Crashoverride attack discussed by Slowik (2018), and the Triton attack discussed by Johnson et al. (2017) all give the impression of actors with thorough knowledge of industrial control systems and their ways of operation. The second implication is that it may cause the OLTC to become unavailable if its commands do not reach their intended target. This is a more serious attack but still not critical, given the assumption of adequate backup solutions to handle the loss of availability.

One tampering threat is selected for review. Successfully tampering with the communication may create a similar scenario like the one obtained from successfully spoofing the SCADA server, namely, to send malicious commands to the IED. Tampering the communication would likely involve intercepting the message, for instance, through a MITM attack. Regardless of the method used, successful tampering the communication can have severe implications for power availability.

Elevation of privilege threats are not highlighted as particularly critical by Khan et al. (2017). This deviates from this thesis. Many of the elevation of privilege threats in the smart grid template must be regarded as critical to the grid. In the example of the smart grid use case, the threat of malware execution on the IED poses a severe threat to power availability. Execution of malware gives the attacker all possibilities regarding attacks. The attacker may deny, tamper or learn the content of all communication. The attacker may also set up a backdoor in order to conduct attacks later. The presence of such a backdoor into critical infrastructure may be the worst-case scenario. The threat of exploiting remote update functionality has the same consequence.

While the use case under study would be severely impacted by spoofing, tampering, and elevation of privilege threats, the argument can be made that this is less so for information disclosure and repudiation. As discussed in chapter 4, information disclosure is considered less critical in the OT domain. One may argue that the same applies to repudiation. Repudiation is the possibility of denying involvement in an action. Consequently, repudiation is only relevant after an attack has occurred. While the ability to attribute an attack to an actor with full certainty would prove very advantageous, it is difficult to achieve in practice.

Based on this discussion of varying criticality of STRIDE categories, an attempt can be made to classify the threats. In this use case example, discarding the information disclosure and repudiation threats would reduce the total number of threats from 355 to 278. If power availability is the primary objective, the repudiation threats can be neglected. If it is assumed that an attacker possesses the necessary OT related knowledge, the same can be concluded for some of the information disclosure threats. The information disclosure threats related to the transmission and storage of credentials may be regarded as extra critical. Obtaining credentials would make an attack much easier to conduct. According to Slowik (2018), the Crashoverride attack relied in part on stolen credentials from the affected systems.

It should be noted that the conclusions drawn from the discussion above do not necessarily apply to the IT part of the grid. The IT part of the grid is, amongst others, characterized by the importance of confidentiality and non-repudiation. Availability is regarded as less important, for instance, in the transmission of smart meter data.

The assumptions about the underlying grid is another factor affecting the relevance of threats. For this thesis, it was assumed that denial of service could not cause a blackout. For use cases where this does not hold true, the denial of service threats may have to be given a high priority instead of medium. The analysis dependability on underlying assumptions may not pose a great problem, as grid engineers have the knowledge to make such assumptions.

We believe that these results highlight several advantages of using our template. The first is an increase in efficiency. Threat modeling of a system is often performed by iterating across all elements or interactions in the model and manually identifying threats. This is not only a laborious process but may also result in threats being forgotten. The template we have developed in this thesis analyses threats according to a predefined logic and presents the threats in a structured way.

The second advantage is that we believe a part of the threat analysis can be carried out by power grid personnel not specialized in cyber security or the smart grid. During the design phase of part or the whole grid, power grid specialist can model the planned system with the template and conduct the automatic threat generation. The power grid specialists can then proceed to perform a first sorting of the identified threats according to their organization’s risk tolerance and security standards. The ten threats chosen for a closer review above can be used to illustrate this. After having performed an initial classification, as performed above in this thesis, the power grid experts may decide to investigate all threats with a priority of medium or high. If needed, these threats may then be forwarded to smart grid cyber security experts.

A final advantage of the template is that it creates a documentation for all identified threats. This documentation is in the form of an HTML file and indicates the threat’s priority, location, mitigation difficulty, the risk of it leading to a blackout, and whether it affects the IT or OT system. An example of part of such a documentation is provided in Appendix B. This documentation gives future security efforts an excellent starting point as it indicates what threats have been identified, how they are addressed and justification for the choices made.

8 Chapter 8

Conclusions, Discussion, and Recommendations for Further Work

8.1 Summary and Conclusions

This thesis applies threat modeling to the smart grid. This is done by developing a generic smart grid template in the Microsoft TMT and applying the developed template to a use case from the smart grid. The process of creating the template involves a study of the components and threats relevant to the smart grid. The use case represents a real-life scenario from the smart grid, realized in a lab environment at SINTEF. The threat modeling of the use case results in 355 identified threats. The STRIDE threat categories are discussed considering the use case and a subset of the identified threats are selected for closer review.

The applied use case demonstrates how our template can be used to model and generate relevant threats for the smart grid. The template includes methods for classifying and sorting threats according to affected components and STRIDE category. More thorough classification of threats demands review of the individual threats. This is a challenge as even use cases of moderate complexity generate many threats. The Microsoft TMT offers great flexibility for defining new templates. This made it possible to adopt a primarily software-focused threat modeling method and develop a template for the smart grid. The tool offers adequate functionality for such threat modeling, but certain aspects are not ideal.

The first task of the thesis objective is to identify and describe the ICT components of the smart grid. This task is partially achieved and is presented in chapter 2. Several components are identified. These are used as input for the smart grid process stencils in the template. The components are not described in detail. This was not done due to limited time. Instead, the assumption is made that all smart grid processes have the capability to execute software and communicate over a network.

The second task of the thesis objective is to identify and describe smart grid cyber threats. This task is achieved and is presented in section 4.3. This section presents threats grouped by the components they affect. The result of the survey is used as input for the threats we included in the template.

The third task of the thesis objective is to describe threat modeling methods and some widely used tools. Apart from STRIDE, Attack trees, PASTA, and LINDDUN are briefly described and discussed. No threat modeling tools apart from the Microsoft TMT are discussed. The Microsoft TMT is discussed in section 7.2.

The fourth task of the thesis objective is to study the Microsoft TMT and techniques. The tool largely relies on STRIDE-per-Interaction and DFDs. The practical use of the tool was obtained through working with the template, modeling of the use case, and through an introduction by SINTEF. STRIDE-per-Interaction and DFDs are discussed in chapter 3.

The fifth task of the thesis objective is to develop a smart grid template. This is achieved through input from literature, existing templates, and previous cyber-attacks on OT systems. Several cyber threats are included in the template. The developed template contains 43 threats spread across the STRIDE threat categories. These threats are presented in section 5.7. The template contains many stencils representing processes, data stores, external interactors, and data flows in the smart grid.

The final task of the thesis objective is to develop and apply a threat model for the smart grid use case. The smart grid use case is introduced in chapter 6. A presentation and discussion of the use case results are given in chapter 7. A total of 355 threats are identified. Based on a small subset of ten threats selected for closer review, we argue that Spoofing, Tampering, and Elevation of Privilege threats are the most critical. This result is dependent on the use case, use case assumptions, and the subset of selected threats. We believe that the result may change, particularly if the IT part of the grid is analyzed.

These six tasks achieve the objective of the thesis to assist in improving cybersecurity management by developing a threat modeling template for the smart grid. The template is made available for download and may be used, extended or changed by others. In order to help in this process, the thesis introduces threat modeling, the smart grid, smart grid components and threats, discusses advantages and disadvantages, and provides suggestions for future developments.

8.2 Discussion

Not all tasks as listed in the thesis proposal are completed. Components in the smart grid are not described in detail. A more detailed study of their inner workings could have enabled the threat generating expressions to be more precise. As an example, the smart grid template assumes that an IED can launch a flooding type of denial of service attack on a SCADA server if it is compromised. A closer study of the components in the smart grid may have provided a better foundation for such assumptions.

Except for the Microsoft TMT, no tools for automated threat modeling are discussed in the thesis. To our knowledge, at the time of writing, few such tools for automated threat modeling exist. We additionally believed that a discussion of any such tools would become overly technical and would have added little to the thesis.

A weakness of the smart grid template is that it is tested on a limited number of scenarios, mainly the use case in chapter 6. This may have led to biased development of the template, resulting in a template better suited for threat modeling of the use case in chapter 6 than arbitrary smart grid use cases. Another weakness is that it was not tested for any IT-based use cases. A relevant IT use case could have been an AMI-related use case.

The template and its usefulness are validated through feedback from security experts at SINTEF and CINELDI. This took the form of presentations of the template. An interesting exercise would have been to perform a manual threat modeling of a use case, in parallel with a threat modeling performed using the template we developed. We believe that comparing the result of these approaches could have provided interesting insight with regards to the template. This was not done because of a lack of time.

A strength of the results is that they demonstrate the applicability of threat modeling and the Microsoft TMT to the smart grid domain. This may hopefully encourage others to adopt threat modeling into cyber-security in the smart grid. The template may additionally serve as a starting point for others to adapt or expand. The developed smart grid templates are the latest addition to a couple of templates for cyber physical threat modeling. The others are the templates for vehicles and medical devices. Hopefully the smart grid template developed in this thesis can inspire others to develop additional templates for new domains.

Threats in the smart grid template have in most cases been generalized beyond the components or systems they are reported to affect in literature, exiting templates, or in previous cyber-attacks. It is difficult to verify the correctness of these assumptions. As a remedy for this problem, each of the threats included in the template and presented in section 5.7 states what assumptions were made.

Several threats are adapted or included from the default and Azure templates. We have evaluated their applicability to the smart grid, but there is a risk that errors made in these templates propagate into our template.

8.3 Recommendations for future work

8.3.1 Extensions of the Smart Grid template

The template can be extended further to include a greater number of threats. More specific threats are easier to mitigate or dismiss as not applicable. A disadvantage of including more threats is that the threat modeling process becomes more time-consuming. An important class of threats not considered by the template is malicious insiders. Malicious insiders, either in the form of disgruntled employees or motivated by other factors, pose a threat to the system. One can imagine such an employee issuing malicious commands to the system, for instance, opening circuit breakers.

The template can be extended further by increasing the number of stencils in the model. To harvest the benefits from this, stencil specific threats should be included. If such threats are not included, the same result can be obtained from using generic stencils. The same argument applies to the inclusion of more stencil configurations. Including more configurations has few advantages if no threats rely on the newly included properties.

Another extension of the template would be to include a focus on the market, service provider, and generation domains. These domains are largely not covered in the existing template. Many of the threats and stencils are applicable to the generation domain. Threat modeling of the market and service provider domains can benefit from more IT-related threats and stencils. Desktop computers and cloud infrastructure are examples of stencils not included in the present template. The domains would additionally benefit from including threats such as spear phishing and other forms of social engineering.

The template does not propose threat mitigation techniques. A list of tested and proven mitigation techniques for each threat may further increase the value of the template to asset owners and security specialists.

Future work should consider if stencils should be closely aligned with the physical components of the smart grid. This may be combined with creating individual DFDs for each component in the model, as proposed by Khan et al. (2017). In the current Microsoft TMT, this would likely have to take the form of several separate threat models. A threat model could be created for each component in the system, along with a top-level model like the one shown in Figure 20. A potential starting point for this approach may be to merge our smart grid template with the default template from Microsoft.

8.3.2 Further advances toward automated smart grid threat modeling

This section discusses functionality and methods believed to be beneficial but which cannot be found in the existing Microsoft TMT.

Threat modeling in the smart grid can benefit from support for different levels of abstraction. Swiderski and Snyder (2004, p. 88) argue that a double-circled process stencil can be used to represent DFDs of subsystems. By incorporating this functionality, the threat modeling can investigate threats to processes and data stores to a greater level of detail. By encapsulating the complexity, threats to subsystems can be studied while the subsystem interacts with the entire system in a realistic way. This functionality would have been useful for the threat modeling in chapter 6. The windmill found to the right in Figure 20 could have been made into a subsystem containing further RTUs, Measurement Units, and possibly SCADA servers. Functionality should be included such that a list of threats can be generated for the subsystem, the parent system, or both. The idea is shown in Figure 21.

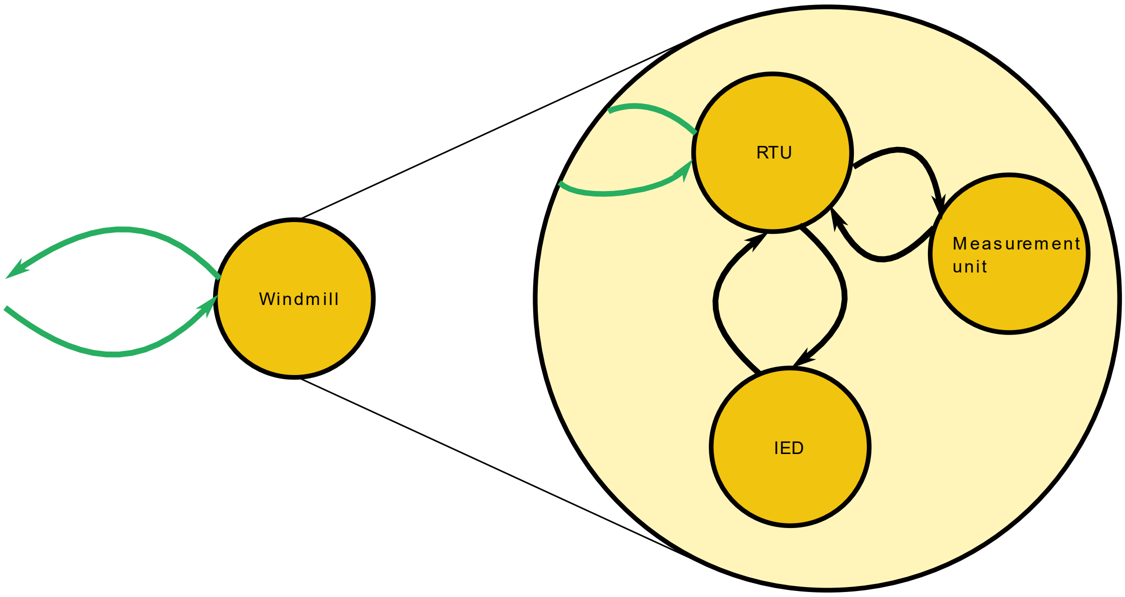


Figure 21: Proposed hierarchical threat modeling.

In the case that existing systems are threat modeled, connecting the threat modeling program to a database of publicly disclosed vulnerabilities on IT and OT systems would be beneficial. An example of such a database could be the Common Vulnerabilities and Exposure (CVE) Database. This would provide concrete warnings to asset owners of exploitable vulnerabilities present in their infrastructure. Simultaneously it would provide a way of prioritizing threats. Such functionality would likely require an extensive database of equipment and software versions. Failure to update such a database may cause a false sense of security.

Greater data sharing between the actors performing threat modeling can facilitate knowledge sharing across the smart grid domain. One way this can be done is for the tool to collects data on how the criticality of various threats is evaluated. This data can, in turn, be used to provide suggestions and feedback to users possibly less experienced in smart grid threat modeling. Ideas on mitigation tactics are another example of beneficial knowledge sharing that a tool could facilitate.

The tool does not support a notion of state. Such a notion of state could have helped to organize the threats according to a timeline. For t = 1, all attacks that can be launched from outside the system can be listed. It is then assumed that the attacks are successful and that systems have been compromised. The properties of the compromised systems are then updated to reflect the new state. An example would be that all outgoing data flows from the compromised system now is set to no longer provide confidentiality. The assumption here is that encryption keys at the compromised system is now in the hands of the attacker.

For t = 2, a new threat analysis is run with the updated properties due to successful attacks for t = 1. As an example, it can be assumed that a system not reachable from the outside is compromised. Properties can be updated to reflect the state of the overall system, and the process is repeated for t = 3. The idea is illustrated in Figure 22. Existing functionality is illustrated in green. The proposed notion of state and timesteps is illustrated by the purple circle and time-controlled transitions. This analysis could run until a defined criterion is violated. A blackout in all or parts of the grid may be an example of such a criterion.

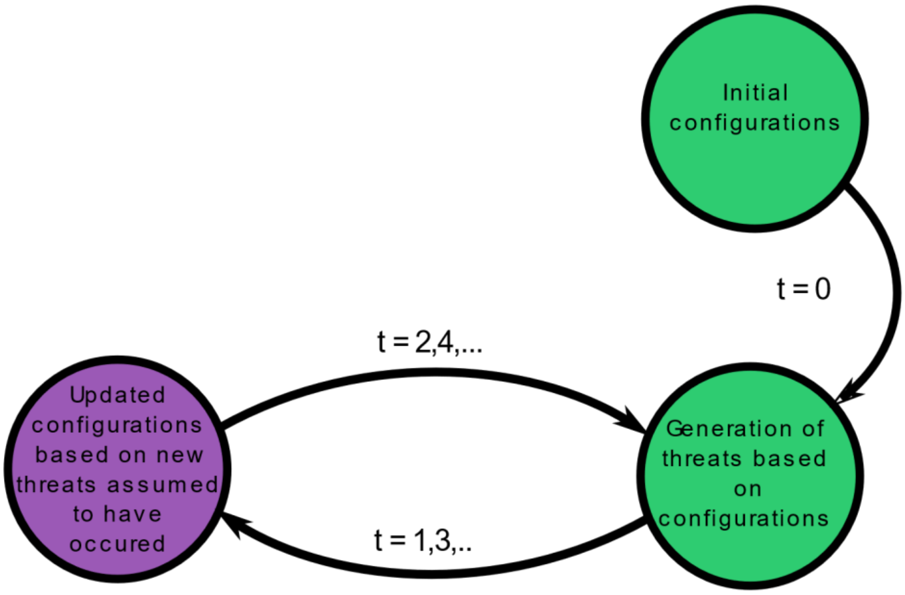

Initial configurations
t = 0
t = 2,4,...
Updated configurations based on new threats assumed to have occured
Generation of threats based on configurations
t = 1,3,..

Figure 22: Proposed state-based threat modeling

9 Appendix A

9.1 Smart Grid Component Table

Generation including DER	**Transmission**	**Distribution**	**Customer**	**Operations**	**Service**	**Market**
PLC/RTU Sayed and Gabbar (2017)	**PLC/RTU** Cosovic et al. (2017; Sallam and Malik (2011) Pillitteri and Brewer (2014)	**PLC/RTU** Bentarzi et al. (2018; Cosovic et al. (2017; Dhend and Chile (2015; Gopstein et al. (2020; Sayed and Gabbar (2017) Pillitteri and Brewer (2014)	**Smart Meter** Barai et al. (2015; Dhend and Chile (2015; Gopstein et al. (2020; Nielsen et al. (2017; Petruševski et al. (2014; Tom and Sankaranarayanan (2017; Yan et al. (2012) Pillitteri and Brewer (2014)	**SCADA server** Bentarzi et al. (2018; Gopstein et al. (2020; Sallam and Malik (2011) Pillitteri and Brewer (2014)	**MDMS** Barai et al. (2015; Nielsen et al. (2017; Petruševski et al. (2014)	**Market platforms** Gopstein et al. (2020)
Edge Computing Cosovic et al. (2017)	**Edge Computing** Cosovic et al. (2017)	**Edge Computing** Cosovic et al. (2017; Tom and Sankaranarayanan (2017)	**Data Concentrator** Barai et al. (2015; Cosovic et al. (2017; Mak and Farah (2012; Petruševski et al. (2014)	**MDMS / AMI headend** Barai et al. (2015; Mak and Farah (2012) Pillitteri and Brewer (2014)	**Cloud** Fang et al. (2012)	**Cloud** Fang et al. (2012)
PMUCosovic et al. (2017)	**PMU** Bentarzi et al. (2018; Cosovic et al. (2017;	**PMU** Bentarzi et al. (2018; Flerchinger et al. (2018; Nielsen et al. (2017)	**Electric Vehicle** Gopstein et al. (2020) Pillitteri and Brewer (2014)	**Cloud** Fang et al. (2012)		

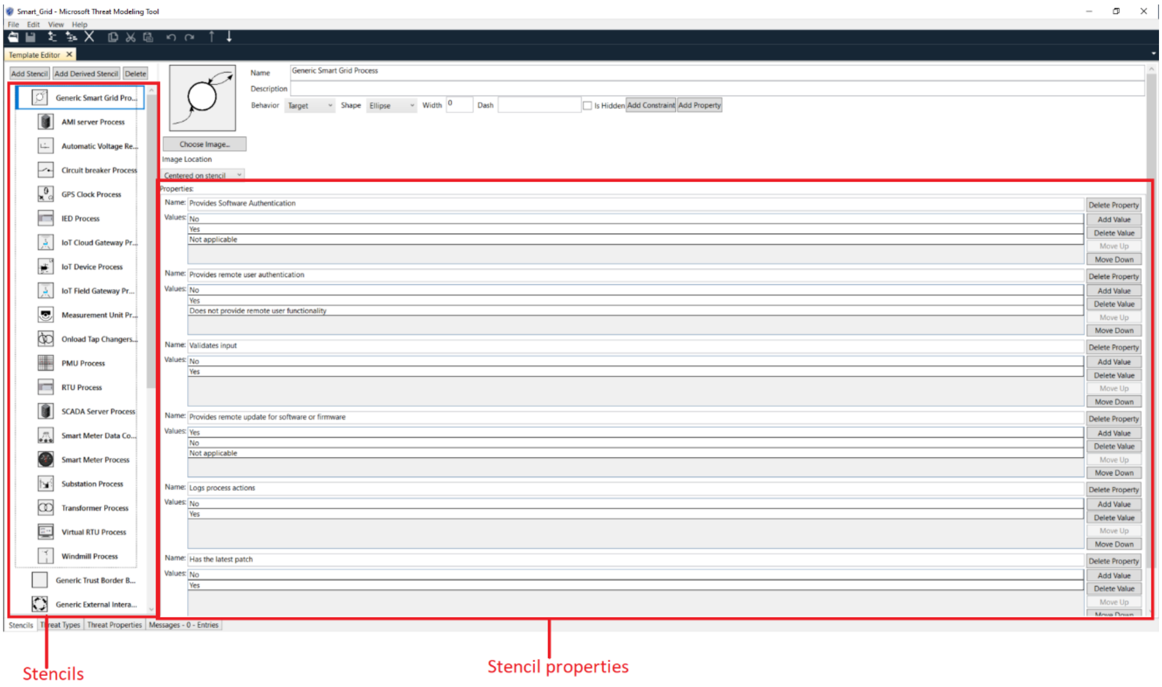
	Gopstein et al. (2020; Nielsen et al. (2017) Pillitteri and Brewer (2014)					
Base stationCosovic et al. (2017)	**Protection relays** Gopstein et al. (2020)	**Base station** Cosovic et al. (2017; Tom and Sankaranarayanan (2017) Nielsen et al. (2017) Flerchinger et al. (2018)	**Appliances** Gopstein et al. (2020)			
Cloud Cosovic et al. (2017; Fang et al. (2012)	**Power quality monitors** Gopstein et al. (2020)	**Cloud** Cosovic et al. (2017; Fang et al. (2012; Tom and Sankaranarayanan (2017)	**Automation systems** Gopstein et al. (2020)			
Circuit Switches Gopstein et al. (2020)	**Line Sag monitors** Gopstein et al. (2020)	**Circuit Switches** Gopstein et al. (2020)	**Base station** Cosovic et al. (2017)			
Sensors Gopstein et al. (2020; Sayed and Gabbar (2017)	**Fault recorders** Gopstein et al. (2020)	**Sensors** Gopstein et al. (2020; Tom and Sankaranarayanan (2017) Pillitteri and Brewer (2014)	**Cloud** Cosovic et al. (2017; Fang et al. (2012)			
	Base station Cosovic et al. (2017)		**DER** Greer et al. (2014) Pillitteri and Brewer (2014)			
	Cloud Cosovic et al. (2017; Fang et al. (2012)					

	Substation meter Gopstein et al. (2020)					
	DER Gopstein et al. (2020)	**DER** Gopstein et al. (2020)				
	Sensors Sayed and Gabbar (2017)					

10 Appendix B

Microsoft Threat Modelling Tool Screenshots

Template creation



Generic Smart Grid Pro...
AMI server Process
Automatic Voltage Re...
Circuit breaker Process
GPS Clock Process
IED Process
IoT Cloud Gateway Pr...
IoT Device Process
IoT Field Gateway Pr...
Measurement Unit Pr...
Onload Tap Changers...
PMU Process
RTU Process
SCADA Server Process
Smart Meter Data Co...
Smart Meter Process
Substation Process
Transformer Process
Virtual RTU Process
Windmill Process
Generic Trust Border B...
Generic External Intera...

Name: Generic Smart Grid Process
Description:
Behavior: Target
Shape: Ellipse
Width: 0
Dash:
□ Is Hidden
Image Location: Centered on stencil

Properties:

Name: Provides Software Authentication
Values: No / Yes / Not applicable

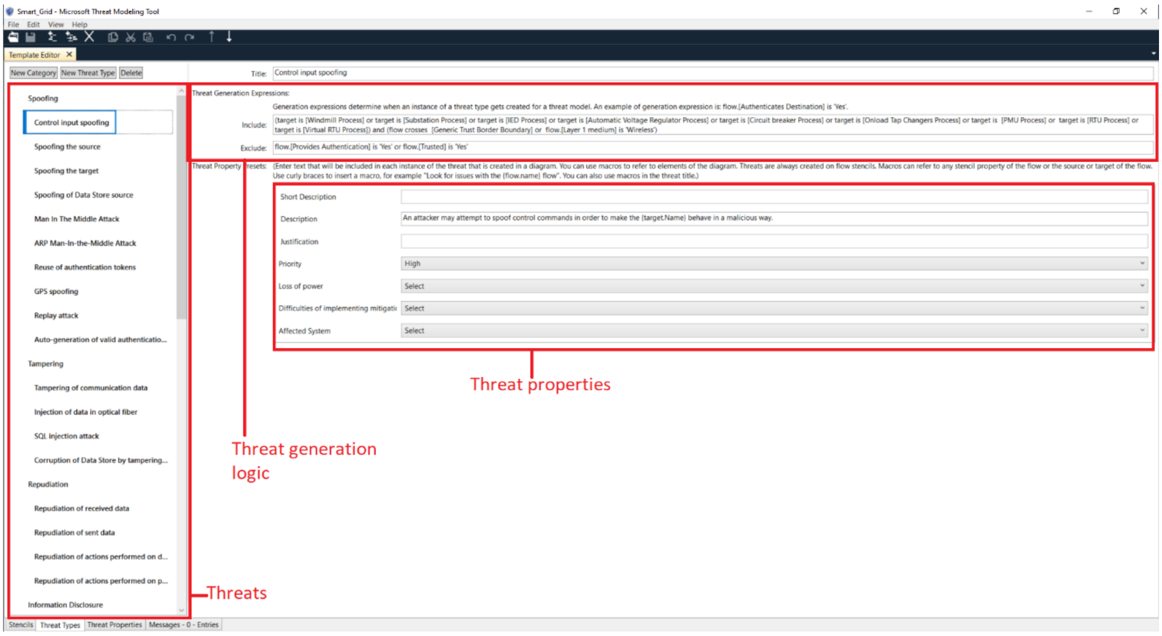
Name: Provides remote user authentication
Values: No / Yes / Does not provide remote user functionality

Name: Validates input
Values: No / Yes

Name: Provides remote update for software or firmware
Values: Yes / No / Not applicable

Name: Logs process actions
Values: No / Yes

Name: Has the latest patch
Values: No / Yes



Spoofing
- Control input spoofing
- Spoofing the source
- Spoofing the target
- Spoofing of Data Store source
- Man In The Middle Attack
- ARP Man-In-the-Middle Attack
- Reuse of authentication tokens
- GPS spoofing
- Replay attack
- Auto-generation of valid authenticatio...

Tampering
- Tampering of communication data
- Injection of data in optical fiber
- SQL injection attack
- Corruption of Data Store by tampering...

Repudiation
- Repudiation of received data
- Repudiation of sent data
- Repudiation of actions performed on d...
- Repudiation of actions performed on p...

Information Disclosure

Title: Control input spoofing

Threat Generation Expressions:

Generation expressions determine when an instance of a threat type gets created for a threat model. An example of generation expression is: flow.[Authenticates Destination] is 'Yes'.

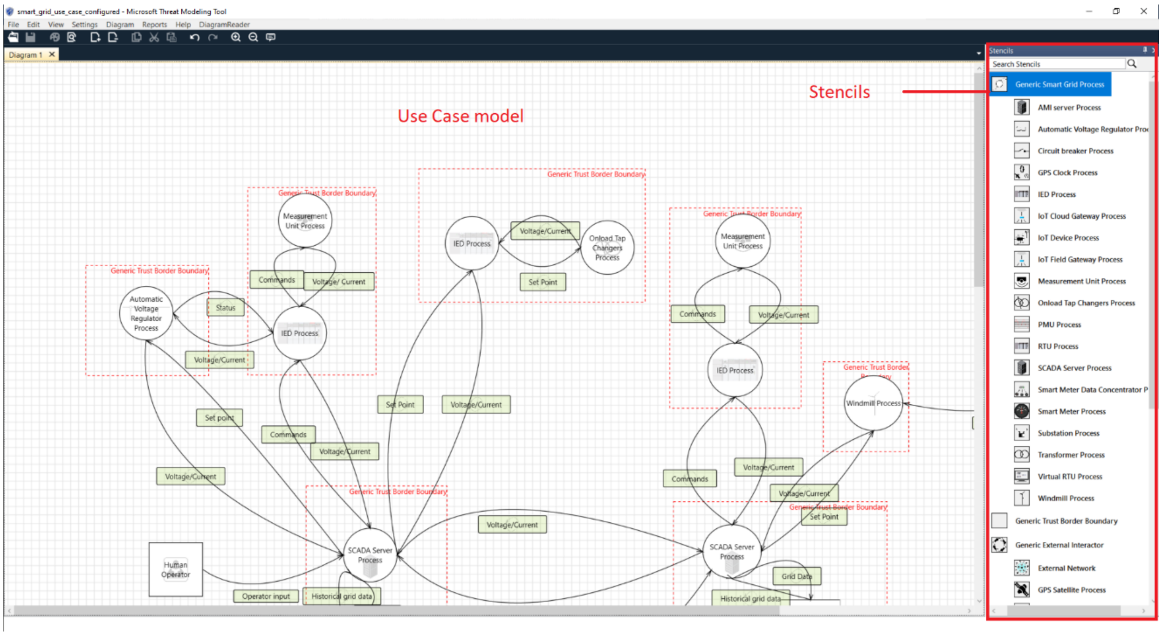
Include: (target is [Windmill Process] or target is [Substation Process] or target is [IED Process] or target is [Automatic Voltage Regulator Process] or target is [Circuit breaker Process] or target is [Onload Tap Changers Process] or target is [PMU Process] or target is [RTU Process] or target is [Virtual RTU Process]) and (flow crosses [Generic Trust Border Boundary] or flow.[Layer 1 medium] is 'Wireless')

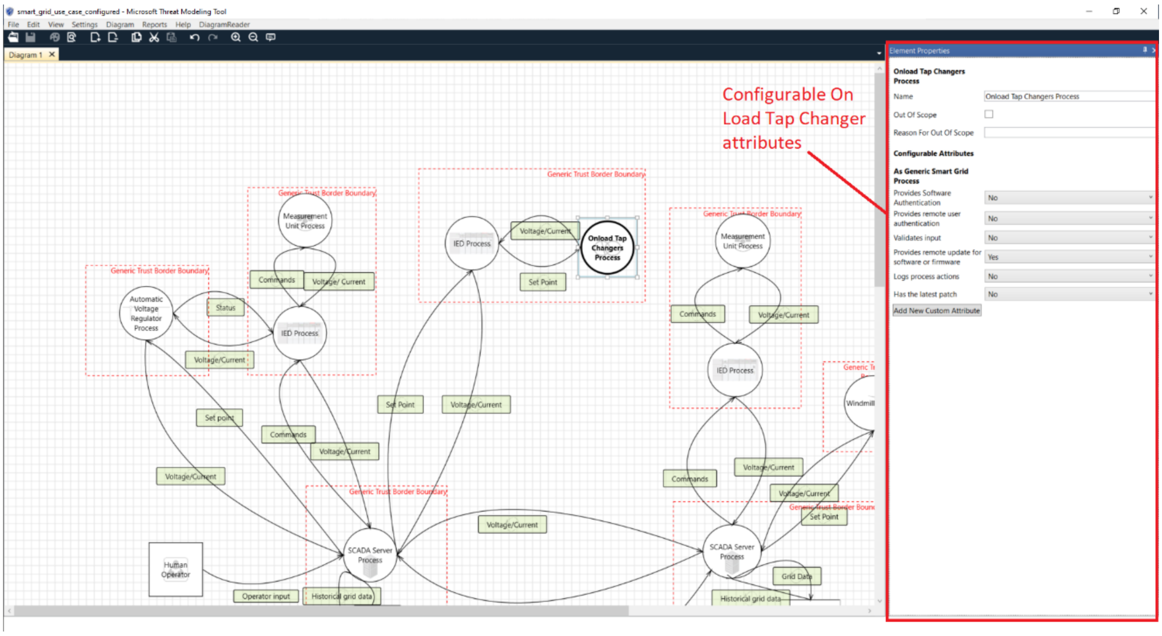
Exclude: flow.[Provides Authentication] is 'Yes' or flow.[Trusted] is 'Yes'

Threat Property resets: (Enter text that will be included in each instance of the threat that is created in a diagram. You can use macros to refer to elements of the diagram. Threats are always created on flow stencils. Macros can refer to any stencil property of the flow or the source or target of the flow. Use curly braces to insert a macro, for example "Look for issues with the {flow.name} flow". You can also use macros in the threat title.)

Short Description	
Description	An attacker may attempt to spoof control commands in order to make the {target.Name} behave in a malicious way.
Justification	
Priority	High
Loss of power	Select
Difficulties of implementing mitigatic	Select
Affected System	Select

Threat Analysis




smart_grid_use_case_configured - Microsoft Threat Modeling Tool
File Edit View Settings Diagram Reports Help DiagramReader
Diagram 1
Configurable On Load Tap Changer attributes
Element Properties
Generic Trust Border Boundary
Measurement Unit Process
Automatic Voltage Regulator Process
Status
Commands
Voltage/ Current
IED Process
Voltage/Current
Set point
Onload Tap Changers Process
Set Point
Human Operator
Operator input
Historical grid data
SCADA Server Process
Windmill
Grid Data

Onload Tap Changers Process

Name: Onload Tap Changers Process

Out Of Scope ☐

Reason For Out Of Scope

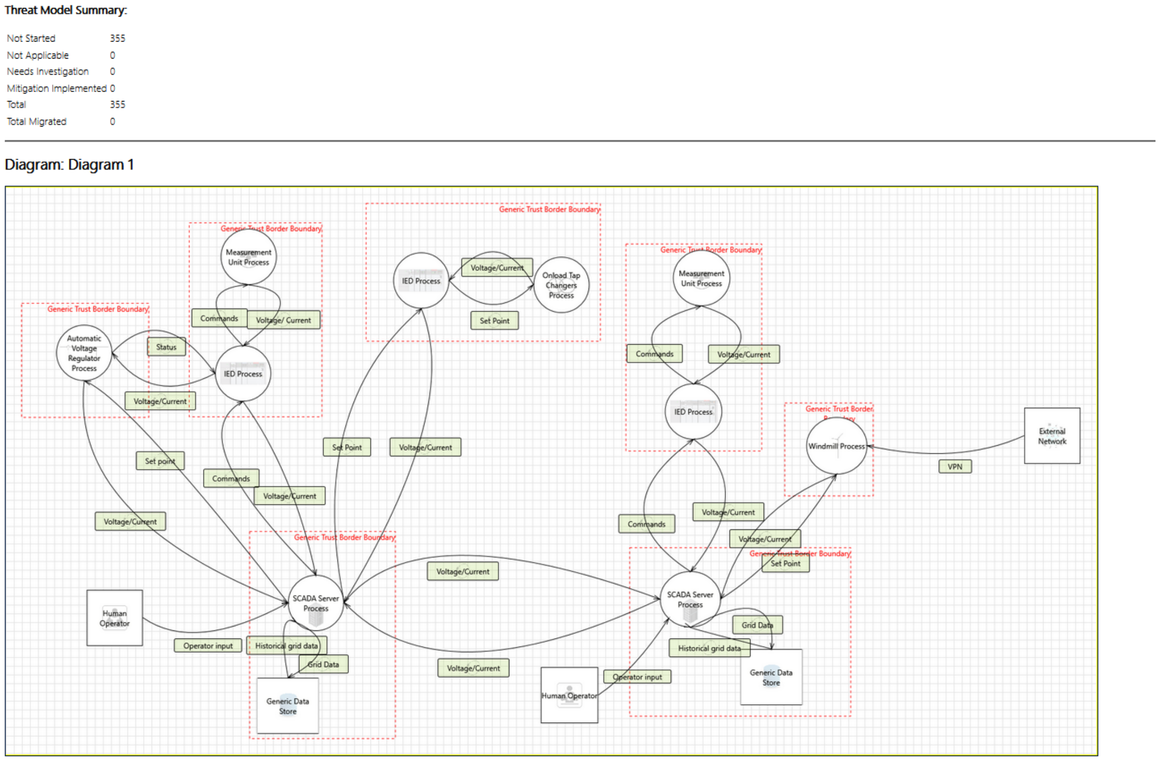
Configurable Attributes

As Generic Smart Grid Process

Provides Software Authentication	No
Provides remote user authentication	No
Validates input	No
Provides remote update for software or firmware	Yes
Logs process actions	No
Has the latest patch	No

Add New Custom Attribute

Examples of generated threat documentation



Threat Model Summary:

Not Started	355
Not Applicable	0
Needs Investigation	0
Mitigation Implemented	0
Total	355
Total Migrated	0

Diagram: Diagram 1

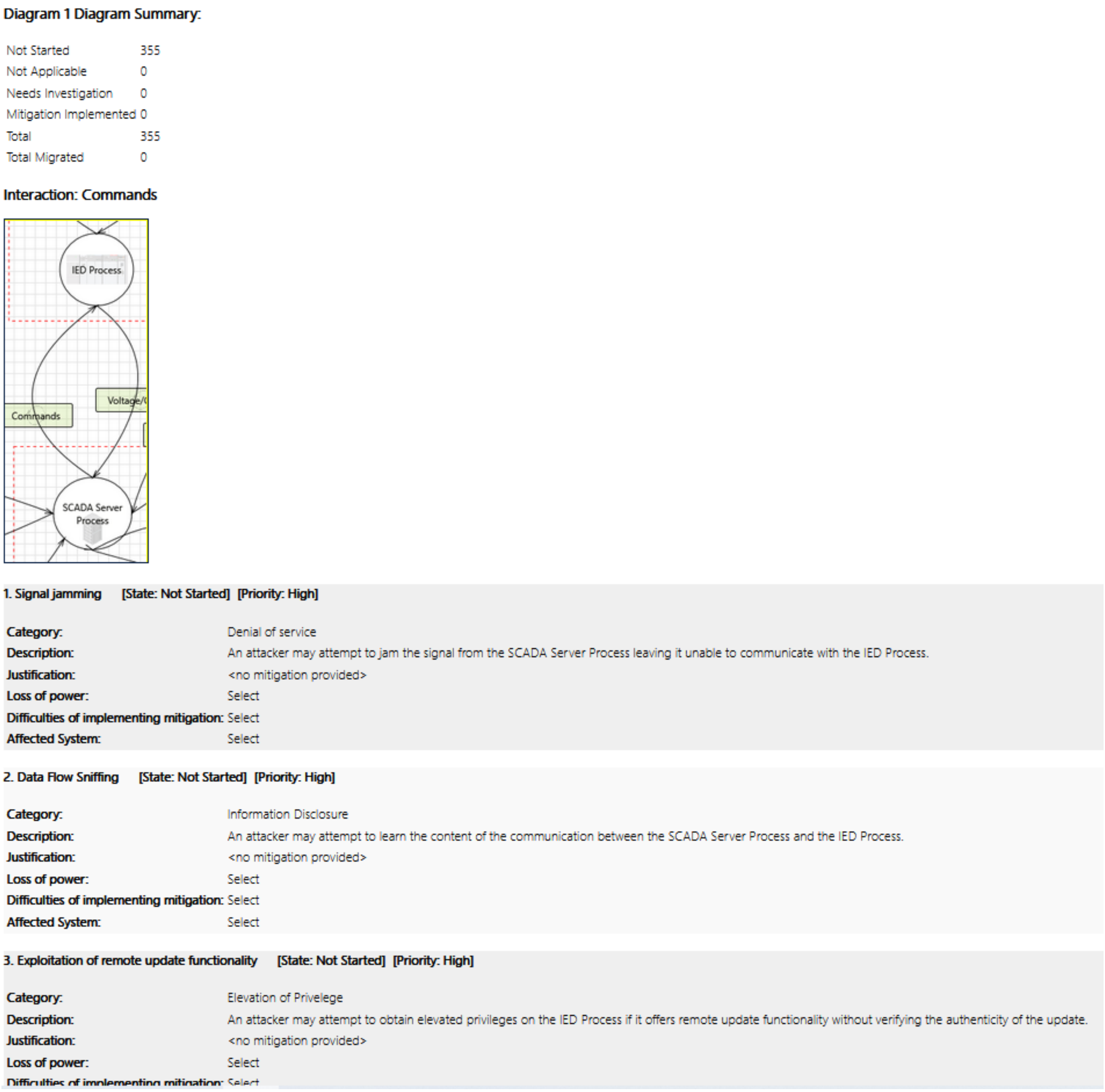


Diagram 1 Diagram Summary:

Not Started	355
Not Applicable	0
Needs Investigation	0
Mitigation Implemented	0
Total	355
Total Migrated	0

Interaction: Commands

1. Signal jamming [State: Not Started] [Priority: High]

Category:	Denial of service
Description:	An attacker may attempt to jam the signal from the SCADA Server Process leaving it unable to communicate with the IED Process.
Justification:	<no mitigation provided>
Loss of power:	Select
Difficulties of implementing mitigation:	Select
Affected System:	Select

2. Data Flow Sniffing [State: Not Started] [Priority: High]

Category:	Information Disclosure
Description:	An attacker may attempt to learn the content of the communication between the SCADA Server Process and the IED Process.
Justification:	<no mitigation provided>
Loss of power:	Select
Difficulties of implementing mitigation:	Select
Affected System:	Select

3. Exploitation of remote update functionality [State: Not Started] [Priority: High]

Category:	Elevation of Privelege
Description:	An attacker may attempt to obtain elevated privileges on the IED Process if it offers remote update functionality without verifying the authenticity of the update.
Justification:	<no mitigation provided>
Loss of power:	Select
Difficulties of implementing mitigation:	Select

11 References

Aloul, F., Al-Ali, A., Al-Dalky, R., Al-Mardini, M., and El-Hajj, W. (2012). Smart grid security: Threats, vulnerabilities and solutions. *International Journal of Smart Grid and Clean Energy*, *1*(1), 1-6.

Ancillotti, E., Bruno, R., and Conti, M. (2013). The role of communication systems in smart grids: Architectures, technical solutions and research challenges. *Computer Communications*, *36*(17-18), 1665-1697.

Barai, G. R., Krishnan, S., and Venkatesh, B. (2015). Smart metering and functionalities of smart meters in smart grid-a review. In *2015 IEEE Electrical Power and Energy Conference (EPEC)* (pp. 138-145). IEEE.

Bentarzi, H., Tsebia, M., and Abdelmoumene, A. (2018). PMU based SCADA enhancement in smart power grid. In *2018 IEEE 12th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG 2018)* (pp. 1-6). IEEE.

Borgaonkar, R., and Jaatun, M. G. (2019). 5G as an Enabler for Secure IoT in the Smart Grid. In *2019 First International Conference on Societal Automation (SA)* (pp. 1-7). IEEE.

Cardenas, A. A., Roosta, T., and Sastry, S. (2009). Rethinking security properties, threat models, and the design space in sensor networks: A case study in SCADA systems. *Ad Hoc Networks*, *7*(8), 1434-1447.

Cosovic, M., Tsitsimelis, A., Vukobratovic, D., Matamoros, J., and Anton-Haro, C. (2017). 5G mobile cellular networks: Enabling distributed state estimation for smart grids. *IEEE Communications Magazine*, *55*(10), 62-69.

D'Antonio, S., Coppolino, L., Elia, I. A., and Formicola, V. (2011). Security issues of a phasor data concentrator for smart grid infrastructure. In *Proceedings of the 13th European Workshop on Dependable Computing* (pp. 3-8).

Deng, M., Wuyts, K., Scandariato, R., Preneel, B., and Joosen, W. (2011). A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements. *Requirements Engineering*, *16*(1), 3-32.

Dhend, M. H., and Chile, R. H. (2015). Innovative scheme for smart grid distribution SCADA system. In *2015 IEEE 2nd International Future Energy Electronics Conference (IFEEC)* (pp. 1-6). IEEE.

Ding, J., Atif, Y., Andler, S. F., Lindström, B., and Jeusfeld, M. (2017). Cps-based threat modeling for critical infrastructure protection. *ACM SIGMETRICS Performance Evaluation Review*, *45*(2), 129-132.

European Network and Information Security Agency. (2012). *Smart Grid Security Annex II. Security aspects of the smart grid*. https://www.enisa.europa.eu/topics/critical-information-infrastructures-and-services/smart-grids/smart-grids-and-smart-metering/ENISA_Annex%20II%20-%20Security%20Aspects%20of%20Smart%20Grid.pdf/view

Falliere, N., Murchu, L. O., and Chien, E. (2011). W32. stuxnet dossier. *White paper, Symantec Corp., Security Response*, *5*(6), 29. https://pax0r.com/hh/stuxnet/Symantec-Stuxnet-Update-Feb-2011.pdf

Fang, X., Misra, S., Xue, G., and Yang, D. (2012). Managing smart grid information in the cloud: opportunities, model, and applications. *IEEE network*, *26*(4), 32-38.

Flerchinger, W., Ferraro, R., Steeprow, C., Mills-Price, M., and Knapek, J. (2018). Third-Generation Cellular and Wireless Serial Radio Communications: Field Testing for Smart Grid Applications. *IEEE Industry Applications Magazine*, *24*(5), 10-17.

Gopstein, A., Nguyen, C., O'Fallon, C., Wollman, D., and Hasting, N. (2020). *NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 4.0*. NIST. https://www.nist.gov/el/smart-grid/smart-grid-framework

Greer, C., Wollman, D. A., Prochaska, D. E., Boynton, P. A., Mazer, J. A., Nguyen, C. T., FitzPatrick, G. J., Nelson, T. L., Koepke, G. H., and Hefner Jr, A. R. (2014). *Nist framework and roadmap for smart grid interoperability standards, release 3.0*.

Guo, Y., Ten, C.-W., Hu, S., and Weaver, W. W. (2015). Modeling distributed denial of service attack in advanced metering infrastructure. In *2015 IEEE power & energy society innovative smart grid technologies conference (ISGT)* (pp. 1-5). IEEE.

Howard, M., and Lipner, S. (2006). *The Security Development Lifecycle*. Microsoft Press, Redmond, Wash, USA.

ICS-CERT U.S Department of Homeland Security. (2016). Cyber-attack against Ukrainian critical infrastructure. Alert (IR-ALERT-H-16-056-01).

IEC 60870-5-104. (2006). *Telecontrol equipment and systems - Part 5-104: Transmission protocol - Network acess for IEC 60870-5-104 unsig standard transport profiles*. Geneva: International Electrotechnical Commission.

IEC 62443-1-1. (2009). *Security for industrial automation and control systems—Models and concepts*. Geneva: International Electrotechnical Commission.

IEC 62443-3-2. (2020). *Security for industrail automation and control systems*. Geneva: International Electrotechnical Commission.

Irmak, E., and Erkek, İ. (2018). An overview of cyber-attack vectors on SCADA systems. In *2018 6th International Symposium on Digital Forensic and Security (ISDFS)* (pp. 1-5). IEEE.

ISO 7498-2. (1989). *nformation processing systems — Open Systems Interconnection — Basic Reference Model — Part 2: Security Architecture*. Geneva: International Organization for Standardization.

ISO 27000. (2018). *Information technology - Security techniques - Information security management systems - Overview and vocabulary*. Geneva: International Organization for Standardization.

ISO 27005. (2018). *Information technology - Security techniques - Information security risk management*. Geneva: International Organization for Standardization.

Isozaki, Y., Yoshizawa, S., Fujimoto, Y., Ishii, H., Ono, I., Onoda, T., and Hayashi, Y. (2015). Detection of cyber attacks against voltage control in distribution power grids with PVs. *IEEE Transactions on Smart Grid*, *7*(4), 1824-1835.

Jiang, R., Lu, R., Wang, Y., Luo, J., Shen, C., and Shen, X. (2014). Energy-theft detection issues for advanced metering infrastructure in smart grid. *Tsinghua Science and Technology*, *19*(2), 105-120.

Johnson, B., Caban, D., Krotofil, M., Scali, D., Brubaker, N., and Glyer, C. (2017). *Attackers Deploy New ICS Attack Framework "TRITON" and Cause Operational Disruption to Critical Infrastructure*. https://www.fireeye.com/blog/threat-research/2017/12/attackers-deploy-new-ics-attack-framework-triton.html

Kalluri, R., Mahendra, L., Kumar, R. S., and Prasad, G. G. (2016). Simulation and impact analysis of denial-of-service attacks on power SCADA. In *2016 national power systems conference (NPSC)* (pp. 1-5). IEEE.

Kalluri, R., Mahendra, L., Senthil Kumar, R. K., Ganga Prasad, G. L., and Bindhumadhava, B. S. (2018). Analysis of Communication Channel Attacks on Control Systems—SCADA in Power Sector. In (pp. 115-131). Springer Singapore.

Karimi, B., and Namboodiri, V. (2013). On the capacity of a wireless backhaul for the distribution level of the smart grid. *IEEE Systems Journal*, *8*(2), 521-532.

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

Klick, J., Lau, S., Marzin, D., Malchow, J.-O., and Roth, V. (2015). Internet-facing PLCs as a network backdoor. In *2015 IEEE Conference on Communications and Network Security (CNS)* (pp. 524-532). IEEE.

Lai, C., Cordeiro, P., Hasandka, A., Jacobs, N., Hossain-McKenzie, S., Jose, D., Saleem, D., and Martin, M. (2019). Cryptography considerations for distributed energy resource systems. In *2019 IEEE Power and Energy Conference at Illinois (PECI)* (pp. 1-7). IEEE.

Liang, G., Zhao, J., Luo, F., Weller, S. R., and Dong, Z. Y. (2016). A review of false data injection attacks against modern power systems. *IEEE Transactions on Smart Grid*, *8*(4), 1630-1638.

Liu, X., Zhu, P., Zhang, Y., and Chen, K. (2015). A collaborative intrusion detection mechanism against false data injection attack in advanced metering infrastructure. *IEEE Transactions on Smart Grid*, *6*(5), 2435-2443.

Mak, S. T., and Farah, N. (2012). Synchronizing SCADA and smart meters operation for advanced smart distribution grid applications. In *2012 IEEE PES Innovative Smart Grid Technologies (ISGT)* (pp. 1-7). IEEE.

Maynard, P., McLaughlin, K., and Haberler, B. (2014). Towards understanding man-in-the-middle attacks on iec 60870-5-104 scada networks. In *2nd International Symposium for ICS & SCADA Cyber Security Research 2014 (ICS-CSR 2014) 2* (pp. 30-42).

Moulinos, K., and Mattioli, R. (2016). *Communication network interdependencies in smart grids*.

Nagaraju, V., Fiondella, L., and Wandji, T. (2017). A survey of fault and attack tree modeling and analysis for cyber risk management. In *2017 IEEE International Symposium on Technologies for Homeland Security (HST)* (pp. 1-6). IEEE.

Niedermaier, M., Malchow, J.-O., Fischer, F., Marzin, D., Merli, D., Roth, V., and Von Bodisco, A. (2018). You snooze, you lose: measuring {PLC} cycle times under attacks. In *12th {USENIX} Workshop on Offensive Technologies ({WOOT} 18)*.

Nielsen, J. J., Ganem, H., Jorguseski, L., Alic, K., Smolnikar, M., Zhu, Z., Pratas, N. K., Golinski, M., Zhang, H., and Kuhar, U. (2017). Secure real-time monitoring and management of smart distribution grid using shared cellular networks. *IEEE Wireless Communications*, *24*(2), 10-17.

Olayemi, O., Antti, V., Keijo, H., and Pekka, T. (2017). Security issues in smart homes and mobile health system: threat analysis, possible countermeasures and lessons learned.

Petruševski, I., Živanović, M., Rakić, A., and Popović, I. (2014). Novel AMI architecture for real-time Smart Metering. In *2014 22nd Telecommunications Forum Telfor (TELFOR)* (pp. 664-667). IEEE.

Pillitteri, V. Y., and Brewer, T. L. (2014). *Guidelines for smart grid cybersecurity*.

Rizzetti, T. A., Wessel, P., Rodrigues, A. S., da Silva, B. M., Milbradt, R., and Canha, L. N. (2015). Cyber security and communications network on SCADA systems in the context of smart grids. In *2015 50th International Universities Power Engineering Conference (UPEC)* (pp. 1-6). IEEE.

Sallam, A. A., and Malik, O. P. (2011). Scada systems and smart grid vision.

Sayed, K., and Gabbar, H. A. (2017). SCADA and smart energy grid control automation. In *Smart Energy Grid Engineering* (pp. 481-514). Elsevier.

Scandariato, R., Wuyts, K., and Joosen, W. (2015). A descriptive study of Microsoft’s threat modeling technique. *Requirements Engineering*, *20*(2), 163-180.

Schneier, B. (1999). Attack trees. *Dr. Dobb’s journal*, *24*(12), 21-29.

Sgouras, K. I., Birda, A. D., and Labridis, D. P. (2014). Cyber attack impact on critical smart grid infrastructures. In *ISGT 2014* (pp. 1-5). IEEE.

Shepard, D. P., Humphreys, T. E., and Fansler, A. A. (2012). Evaluation of the vulnerability of phasor measurement units to GPS spoofing attacks. *International Journal of Critical Infrastructure Protection*, *5*(3-4), 146-153.

Shevchenko, N., Chick, T. A., O'Riordan, P., Scanlon, T. P., and Woody, C. (2018). *Threat modeling: a summary of available methods*. Carnegie Mellon University Software Engineering Institute Pittsburgh United States. https://resources.sei.cmu.edu/asset_files/WhitePaper/2018_019_001_524597.pdf

Shostack, A. (2014). *Threat modeling: Designing for security*. John Wiley & Sons.

Slowik, J. (2018). *Anatomy of an Attack: Detecting and Defeating CRASHOVERRIDE*. VB2018. https://www.bgp4.com/wp-content/uploads/2018/10/CRASHOVERRIDE2018.pdf

Slowik, J. (2019). *CRASHOVERRIDE: Reassessing the 2016 Ukraine electric power event as a protection-focused attack*. Dragos. https://www.dragos.com/wp-content/uploads/CRASHOVERRIDE.pdf

Staggs, J., Ferlemann, D., and Shenoi, S. (2017). Wind farm security: attack surface, targets, scenarios and mitigation. *International Journal of Critical Infrastructure Protection*, *17*, 3-14.

Stefanov, A., and Liu, C.-C. (2012). Cyber-power system security in a smart grid environment. In *2012 IEEE PES Innovative Smart Grid Technologies (ISGT)* (pp. 1-3). IEEE.

Suleiman, H., Alqassem, I., Diabat, A., Arnautovic, E., and Svetinovic, D. (2015). Integrated smart grid systems security threat model. *Information Systems*, *53*, 147-160.

Sundararajan, A., Chavan, A., Saleem, D., and Sarwat, A. I. (2018). A survey of protocol-level challenges and solutions for distributed energy resource cyber-physical security. *Energies*, *11*(9), 2360.

Swiderski, F., and Snyder, W. (2004). *Threat Modeling*. Microsoft Press.

Tom, R. J., and Sankaranarayanan, S. (2017). IoT based SCADA integrated with fog for power distribution automation. In *2017 12th Iberian Conference on Information Systems and Technologies (CISTI)* (pp. 1-4). IEEE.

Tøndel, I. A., Jaatun, M. G., and Line, M. B. (2013). Threat modeling of AMI. In *Critical Information Infrastructures Security* (pp. 264-275). Springer.

Wang, W., and Lu, Z. (2013). Cyber security in the smart grid: Survey and challenges. *Computer networks*, *57*(5), 1344-1371.

Xiong, W., and Lagerström, R. (2019). Threat modeling–A systematic literature review. *Computers & Security*, *84*, 53-69.

Yan, J., Liu, C.-C., and Govindarasu, M. (2011). Cyber intrusion of wind farm SCADA system and its impact analysis. In *2011 IEEE/PES Power Systems Conference and Exposition* (pp. 1-6). IEEE.

Yan, Y., Qian, Y., Sharif, H., and Tipper, D. (2012). A survey on smart grid communication infrastructures: Motivations, requirements and challenges. *IEEE communications surveys & tutorials*, *15*(1), 5-20.

Yang, Y., McLaughlin, K., Littler, T., Sezer, S., Im, E. G., Yao, Z., Pranggono, B., and Wang, H. (2012). Man-in-the-middle attack test-bed investigating cyber-security vulnerabilities in smart grid SCADA systems.