jamk

Cyberattacks on critical infrastructure and corresponding countermeasures

Petri Vähäkainu

M.Eng. thesis
13.2.2023
School of Technology
Master's Degree Programme in Information Technology
Cyber Security

jamk | Jyväskylän ammattikorkeakoulu
University of Applied Sciences

jamk

Description

Vähäkainu, Petri

Cyberattacks on critical infrastructure facilities and corresponding countermeasures

Jyväskylä: JAMK University of Applied Sciences, February 2023, 62+28 pages

Degree Programme in Cyber Security. Master's thesis.

Permission for open access publication: Yes

Language of publication: English

Abstract

These days cyberattacks pose a growing risk to cyber-physical systems (CPSs) that act as a part of critical infrastructure (CI) that are vital to a nation's economy and security. These attacks can disrupt vital devices and services, paralyze whole societies, and cause even life-threatening consequences. More robust and resilient infrastructure is required to combat an ever-increasing number of incoming cyberattacks, which can be divided into denial-of-service (DoS), distributed denial-of-service (DDoS), Malware, and Phishing attack domains.

The research focus was on studying some of the most common cyberattacks studied in the chapter: "Cyberattacks Against Critical Infrastructure Facilities and Corresponding Countermeasures" by Vähäkainu et al. (2022) targeting critical infrastructure facilities, machine learning defensive mechanisms to provide additional detection and defense capabilities to extend the inadequate protection of critical infrastructure facilities against cyber threats, and reviewing the detection accuracy and best fit of these mechanisms to identify incoming cyberattacks. To reach the research objectives, information was acquired by performing database searches from various scientific databases and web pages on the internet, and comparative study methods were applied to analyze the data collected. The data gathered was used to gain an understanding of what is the best suitable machine learning classifier to detect the most common cyberattacks previously mentioned, and with what accuracy.

The results indicated that the decision tree and random forest classifiers provided an excellent performance outperforming other classifiers compared. The random forest achieved the best accuracy among all the classifiers reviewed providing 97–99 % DDoS, 89 % FDIA, 92–99 % Malware, 96–100 % Phishing, and 99 % Ransomware attack detection accuracy, reaching the best choice of the classifiers examined. The Random Forest is a well-known and extensively utilized classifier capable of preventing overfitting, and it can be applied in the domains previously mentioned. The results also showed that the Naïve Bayes classifier was able to provide only rather poor performance, 62–99 % DDoS, 89 % FDIA, 70–91 % Malware, 95 % Phishing, and 35 % Ransomware accuracy in most of the experiments, and hence, it is not advised to utilize it as a countermeasure against incoming cyberattacks, except possibly with FDIA attacks in some cases.

Keywords/tags (subjects)

Artificial intelligence, critical infrastructure, cyberattacks, cyber-physical systems, machine learning

Miscellaneous (Confidential information)

No confidential information in this thesis.

Acronyms

AES	Advanced Encryption Standard is a standard for encrypting digital data
AI	Artificial intelligence can be defined as an ability to mimic human intelligence
ANN	Artificial neural network is a computational model consisting of various processing units obtaining inputs and delivering outputs based on predefined functions
API	Application programming interface is a software intermediary allowing two applications to communicate with each other
APT	Advanced Persistent Threat is an attack campaign utilizing continuous, illicit, and sophisticated hacking techniques to obtain access to a system and stay inside
BN	Bayesian Network is a probabilistic graphical model using Bayesian inference for probability calculations
CDBN	Conditional deep belief network is a deep-learning classifier and a probability generation model
CI	Critical infrastructure consists of essential systems, networks, and assets required to remain operational to maintain security
CISA	Certified Information Systems Auditor is ISACA's standard for achievement for those who audit and assess an organization's information technology
CNN	Convolutional neural network is a form of artificial neural network (ANN) classifier, which includes convolutional layers, commonly used to analyze visual imagery
CPS	Cyber-physical systems are systems utilizing computing and communication technology to control, coordinate, and monitor the operations of physical systems
CPPS	Cyber-physical Power System is an interconnected architecture that interacts with the physical power system environment
C&W	Carlini & Wagner Attack is a method to efficiently generate adversarial examples
DDoS	Distributed denial-of-service is a malicious cyberattack by an adversary to disable a server, service, or network by flooding it with internet traffic
DL	Deep learning is a subset of artificial intelligence providing self-learning and function-improving capabilities by examining algorithms
DNN	Deep neural network is a form of ANN that includes various layers and can be utilized, for example, in image classification or text and speech recognition
DT	Decision tree is a supervised learning method, which can be utilized for classification and regression tasks

DOM	Document Object Model is a programming interface for web documents
FDIA	False data injection attack means the attack where an adversary alters/modifies the original sensor measurements affecting the control center computational capacity
FGSM	Fast gradient sign method is a method to create adversarial images
GMM-EM	Gaussian mixture model – expectation maximization is a parametric statistical model assuming that the data originates from a weighted sum of several Gaussian sources
GAN	Generative adversarial networks are a deep-learning-based generative model using two neural networks competing
GCN	Graph convolutional network is a method for learning data with a graph structure in a semi-supervised manner
HTTP	Hypertext transfer protocol is a protocol operating on the application layer that can be used to transfer web pages and other hypermedia documents like HTML
HVAC	Heating, ventilation, and air conditioning is the technology that can be used to control air quality, humidity, and temperature in a closed space
ICMP	Internet control message protocol is a protocol operating on a network layer usable for network devices to identify problems in network communication
ICS	Industrial control system is an information system used to control various industrial processes like distribution, manufacturing, and production
ICT	Information and communication technology can be defined as a set of resources and tools for creating, exchanging, sharing, storing, and transmitting information
IDS	Intrusion Detection System can be considered as a gadget or application examining a network for illicit operation or violation of rules
IP	Internet Protocol is the network protocol defined in the TCP/IP model used for sending packets from source to destination
JSMA	Jacobian-based Saliency Map Attack is a gradient-based white box -method for fooling classification models
KNN	K-nearest neighbor is a simple Machine Learning classifier based on a supervised learning technique
LDA	Linear discriminant analysis is a machine learning technique that can be used to mitigate the dimensionality of data and solve multi-class classification problems
LR	Linear regression in statistics is a linear approach for modeling the relationship between a scalar response and one or more explanatory variables

LSTM	Long short-term memory is a type of RNN used in deep learning to learn long-term dependencies, such as sequence prediction
ML	Machine learning is a subfield of artificial intelligence allowing machines to learn from data without being programmed explicitly
MLP	Multilayer perceptron is a neural network capable to learn the relationships between linear and non-linear data
NB	Naïve Bayes is a method of classification utilizing Bayer's theorem, and it can be applied to both binary and multiple-class categories classification problems
NIST	National Institute of Standards and Technology assists in promoting innovation and improving competitiveness in the industry sector
NN	Neural network is imitated by the structure of the human brain, and it is a component of ML to solve complex signal processing or pattern recognition problems
OSI	Open systems interconnection model is a framework enabling various communication systems to connect using standard protocols
OT	Operational technology is a type of computing and communicating system used to control, manage, and monitor industrial operations
PCA	Principal component analysis is a widely used unsupervised machine learning algorithm to mitigate the dimensionality of a data
PCAP	Packet Capture is an API, which captures live network packet data from layers 2-7 of the OSI model
PG	Power grid, consisting of generator stations, transmission lines, and towers, is a network for delivering electricity to consumers
PPP	Phish-prone-percentage is known as each organization's employee susceptibility to phishing attacks
RF	Random Forest is a widely used supervised machine learning method based on ensemble learning used in classification and regression problems
RNN	Recurrent neural networks are neural networks used, for example, for time series data instead of traditional feedforward networks
RSA	Rivest-Shamir-Adleman is a public-key asymmetric encryption algorithm
SCADA	Supervisory control and data acquisition is a control system architecture that uses computers, data communication networks, and a user interface to supervise machines and processes

SDN	Software-defined networking (SDN) is a method of network management that utilized software-based controllers or APIs to control the network's flow of information
SE	State Estimation is a process to estimate the electrical state of a network by eliminating inaccuracies and errors from the measurement data
SQL	Structured query language is a standard programming language for managing and operating data in relational databases
SSH	Secure shell protocol is a network communication protocol enabling two computers to securely communicate and share data over an unsecured network
SVE	State vector estimator is a method for attack detection of smart grids, or wind power generations using reservoir computing (RC)
SVM	Support vector machine is a machine learning classifier that can be used for both classification and regression tasks
TCP	Transmission control protocol is a standard, which defines the beginning, maintaining, and ending of a network conversation for exchanging application data
UDP	User datagram protocol is a communications protocol utilized to establish fast and low-tolerance connections between internet applications
URL	Uniform Resource Locator is a unique identifier that can be used to find a specific resource on the internet

Contents

Tables

1 Introduction

1.1 Background of the research

Cyberattacks against cyber-physical systems (CPSs) acting as part of critical infrastructure (CI) cause challenging situations nowadays, for example, in the form of hacking attack attempts. Hacking attacks on CI, such as power grids and telecom networks, or even governments can paralyze whole societies, produce kinetic results, and in some cases can also be life-threatening. Critical infrastructure and its assets are vital to a nation's security or economy and therefore are under high priority to be protected. Hacking attacks can also act as a military weapon, which we have seen happening in the Ukraine conflict that started in 2022 (Juutilainen, 2022). To defend against an ever-increasing number of incoming cyber-attacks, stronger and more resilient infrastructure is needed. In this thesis, the most common and the most relevant cyberattacks, such as adversarial, DoS, DDoS, False data injection (FDI), Malware, and Phishing attacks against critical infrastructure, and defense mechanisms against these attacks are studied. Critical infrastructure protection is an important step in ensuring security acting as a motivator for this study.

Cybersecurity of critical infrastructure is a known and important problem, but progress toward better security has been slow. In the past, it was commonly thought that the risk of cyberattacks against critical infrastructure was not high due to a lack of suitable internet connection, specialist knowledge of the control system configuration, and administrative operations (Mutsuo & Hirofumi, 2017). Nowadays industries are mainly vulnerable to cyberattacks because they are exposed to the internet. To cause significant damage and even destruction, an adversary must hack into the system and apply and run malicious code (for example, malware) on the computers to cut off the energy supply, and cause explosions at chemical plants, processing plants, or even nuclear power plants. The impact of successful cyberattacks on critical infrastructure can be severe, therefore, cyber defenders must anticipate how they might detect and defend against future potential developments of these kinds of attacks.

According to the World Economic Forum's 2020 Global Risk Report (World Economic Forum, 2020), cyberattacks have been ranked among the top five increasing global risks of disrupting operations and critical infrastructure. Based on Accenture estimates, the number of cyberattacks has gone up by 67 % in the past five years (Ghosh, 2019). Cyberattacks can induce significant business

disruptions and lead to great financial losses reaching hundreds of millions of dollars (Nozomi Networks, 2022). According to Erma Pte Ltd. (2022), on a global scale and in general, according to British insurance company Lloyd's, cyberattacks cost businesses as much as 400 billion USD a year, and the amount is growing. Due to the rapid digitalization of consumers' lives and enterprise records, the cost of data breaches will climb up to more than 2.1 trillion USD globally, almost four times the estimated cost of breaches in 2015.

This study is based on the chapter: "Cyberattacks Against Critical Infrastructure Facilities and Corresponding Countermeasures" by Vähäkainu et al. (2022). The chapter focused on exploring cyberattacks on critical infrastructure and defensive mechanisms to provide auxiliary detection and defense capability to enhance the insufficient protection of the smart critical facility against outsider threats. This study enhances the chapter and concentrates on examining some of the most common cyberattacks on critical infrastructure and seeking an answer to which one of the widely used machine-learning algorithms performs the best as a countermeasure on the defensive side. The research questions and objectives in the research objectives and limitations section clarify the nature of this study.

1.2 Introduction of the research article

Cyberattacks on critical infrastructure (CI) can cause life-threatening consequences, not just minor and inconvenient effects one could consider. Various countries throughout the world are increasingly under the influence of cyberattacks on critical infrastructures, such as smart power grids, water suppliers, military facilities, various cyber-physical systems, etc.

The article authored by Vähäkainu et al. (2022) concentrates on researching traditional and more sophisticated cyberattacks studied and compared in this thesis to provide an overview of the situation. In addition, the authors of the article introduce defensive mechanisms to provide additional detection and defense capability to improve the current level of protection of critical infrastructure facilities countering incoming cyber threats.

1.3 Research objectives and limitations

To make the detection of cyberattacks more actionable, effective, and scalable than traditional technologies requiring human intervention, the use of machine learning (ML) and deep learning

(DL) methods can be applied. This thesis focuses on studying some of the most common cyberattacks on critical infrastructure facilities and corresponding machine- and deep-learning defense mechanisms, and with what accuracy these defense mechanisms can identify incoming cyberattacks. This study aims to present these defensive mechanisms to provide auxiliary detection and defense capability to enhance the insufficient protections of critical infrastructures, such as the smart critical facility against outsider cyber threats. The study is independent research by the author of this study based on a chapter by the author (Vähäkainu, 2022), and there is no commissioner.

The research questions can be formulated as follows:

- With what accuracy machine learning classifiers (methods), such as decision tree (DT), support vector machine (SVM), Naïve Bayes (NB), or neural network (NN) can detect cyberattacks, such as DDoS, FDIA, malware, phishing, and ransomware?
- What is the most suitable machine learning classifier presented in the previous question to detect DoS/DDoS, FDIA, malware, and phishing attacks?

1.4 Research Methodology

The research methodology used in this study is the comparative study. According to Hantrais (1995): “Comparative research methods have been used for a long time in cross-cultural studies to identify, analyze, and explain similarities and differences across societies”. Ibsrekken (2022) states that comparative study is about looking at an object of study in relation to another. The object of study is usually compared across space and/or time. Comparative methods can be qualitative or quantitative, and in this study quantitative method is selected due to the nature of the data and machine learning classifier performance evaluation and comparison.

According to Buhari (2011), two main styles, descriptive comparison, and normative comparison form the comparative study. Descriptive aims at describing and explaining the invariances of the objects. The purpose of this style is not to generate changes in the objects but to avoid them. It can be challenging to find all potential causal influences solely based on empirical study, hence, a comprehensive literature study for finding theory and data of comparable cases is required and applied in the implementation part of the thesis. This thesis aims to provide a descriptive comparison between machine learning classifiers (models) and domains based on accuracy, precision, and recall, and therefore descriptive style is selected as a study method.

Another major style of comparative study is normative comparison. This style is required if the aim is not only to detect and explain but also to improve the present state of the object or to improve or develop a similar object in the future (Routio, 2007). Despite the comparative research methodology being mostly applied in the field of social sciences, it can also be utilized in other fields of science, such as Information Technology (Lor, 2019, 2).

The research questions in 1.3 are intended to be answered by acquiring information by performing database searches from databases, such as arXiv.org, Elsevier, IEEE Xplore Digital Library, and ScienceDirect, and by internet searches, and conducting a comparison. A comparative study was applied based on the data gathered.

1.5 Ethicality and reliability of the research

The author of this thesis has followed good scientific practice and got familiar with the guidelines of the Finnish Advisory Board on Research Integrity 2012 (Varantola et al., 2013). The theory and empirical data of the thesis have been gathered from public and trustworthy sources on the internet websites, and databases by using proper citing (APA7) to references, avoiding fabrication, falsification, misrepresentation, misappropriation, and other kinds of scientific misconduct. The gathered theoretical and empirical data does not include any personal details or illicit use of copyrighted material.

The author determined the credibility of a source document/material by researching the writer, and his/her credentials, what press published the document, how long time ago the document was created, evaluating the site's credibility, and avoiding untrustworthy internet sites, such as Wikipedia. The author favored academic sources, such as reliable scientific databases with peer-reviewed conferences or journal literature. Articles written by respected and well-known authors were favored. The author also followed the Ethical Principles for JAMK University of Applied Sciences (2018) guidelines in conducting the study, the American Psychological Association (APA) ethical code (APA, 2022), and the European Code of Conduct for Research Integrity guidelines (Allea, 2017).

According to Hirsjärvi et al. (2004, 216 – 217), the repeatability of the measures taken is directly proportional to the reliability of the research. Reliability means consistent results from data collection and analysis. For example, research is reliable if at least two researchers end up with similar results. The gathered empirical material is reliable if it does not include incoherence. Material can

be reliable even if it does not include validity, but validity is not possible without reliability. Validity means a measure is accurate and measures what it is supposed to measure.

The author states that the empirical results of this thesis are reproducible under the same conditions, which means selecting the same references, such as books, databases, web pages, and other sources used in this thesis. A possibility that two or more researchers can reach the same conclusions is required, and in addition, results of the measures are what they should have measured, it can be concluded that both reliability and validity have been reached.

1.6 Structure of the thesis

This thesis is organized as follows: The first chapter is the introduction, where the author provides background information about the research, research objectives and limitations, and methodology of the research, and discusses the ethicality and reliability of the research. The second chapter concerns theoretical framework and concepts, general information about cyber-physical systems and implementations, presents cybersecurity definitions and concepts, introduces critical infrastructure, discusses trends, explains the basics of artificial intelligence and machine learning, and presents common cyberattacks on critical infrastructure facilities and countermeasures. The third chapter comprises the implementation and results of the research, and it explains how the data was gathered, explains performance evaluation metrics, and presents the processing and analysis of data. The fourth and final chapter is a discussion and conclusion chapter in which the author summarizes the research results, discusses criticism of the research, and presents further research ideas.

2 Theoretical framework and concepts

2.1 Cyber-physical systems and implementations

Cyber-physical systems (CPS) are automated systems that link the functioning of physical reality with computing and communication infrastructures (Jazdi, 2014). Tiwari et al. (2021) described cyber-physical systems as: “Cyber-Physical Systems (CPS) are collections of physical and computer components that are integrated to operate a process safely and efficiently”. Cyber-physical systems can be considered sociotechnical systems that seamlessly merge analog, digital, physical, and human components designed to function through integrated physics and logic. (Griffor et al., 2017). In cyber-physical systems, embedded computers keep track of and monitor physical processes, typically feedback loops, in which physical processes influence computations and vice versa. (Lee, 2015). CPSs provide the foundation of critical infrastructure (CI) and ways to develop and implement novel future smart services. However, a smart building, which includes built-in IoT sensors and possibly machine-learning-based predictive controls of electrical devices, can form a cyber-physical system, which can also be a critical infrastructure facility. Cyberattacks on these kinds of facilities provide an important research context and it is conducted in this thesis.

Typically, CPSs are composed of where several interconnected agents, such as sensors, actuators, control processing units, and communication devices, that interface with the physical world. Sensors are considered devices that convert physical events and characteristics into digital signals, actuators convert digital signals into physical events and characteristics. CPSs can be data-intensive generating a huge amount of data while operating. The foundation of critical infrastructure and the ability to create and execute intelligent services that enhance the quality of life is based on CPSs and data collection. One method of implementing CPS is through feedback systems, which can be adaptive and predictive, intelligent, real-time, networked, or distributed, and may incorporate wireless sensing and actuation.

Applications of cyber-physical Systems can be seen everywhere, such as in the automotive industry, assisted living, energy conservation, HVAC (heating, ventilation, and air conditioning) (HVAC, 2023), manufacturing, medicine, military, physical security, power generation, and distribution, robotics, traffic control, and safety, water management systems, etc. Lee (2015). In smart buildings, CPS provides means to use sensors in collecting data, such as carbon dioxide, electricity, en-

ergy, humidity, inside and outside temperature, motion detection, and water consumption to adjust and control automatically, for example, HVAC systems. Control and optimization of HVAC are among the most relevant tasks of a smart building concept providing extensive influence on the quality of life of occupants and when implemented well, they can provide significant cost savings (Stamatescu, 2016).

A smart grid is a typical example of a cyber-physical application that combines physical power systems and cyber systems, including sensing, monitoring, communication, and control (Guo et al., 2017). Smart Grid includes electricity distribution services, two-way communications, intelligent sensors, automated metering, and computer systems to improve reliability, and performance, and enhance the efficiency decision of the customer and the utility provider (Forte, 2010). According to Sun et al. (2022), smart grids utilize a vast amount of information collected from the physical system to be analyzed by the cyber system, and in turn, finally affecting the operation of the physical system through economic and remedial actions. Smart grids can keep track of the grid's status in real time and use that information to run the grid securely, reliably, and stably, reducing costs and enhancing energy efficiency (Alonso et al., 2020). Integrating cyber and physical systems is critical, bringing in new types of risks in which an adversary may utilize cyber systems to initiate fake commands to damage facilities or even initiate a sequence of cascading effects (Sun et al, 2020). However, functionalities required by CPS need accurate measurement data from the physical system. Sensor, device, or communication failures provide incorrect data causing delivering important commands to a successful operation.

Artificial intelligence can be considered an "intelligent agent" that solves real-time problems in smart grids. It can be utilized to integrate renewable energy, stabilize energy networks, carry out user behavior analysis, respond to sudden changes in customer demands, power outages, sudden drops and rises in energy outputs, perform fault diagnostics, and mitigate financial risks caused by fluctuations in the infrastructure. Simulations and AI-based forecasting can be used to optimize energy systems and improve their efficiency. Developing and implementing cost-effective smart grid solutions require extensive knowledge of energy systems and their elements.

Jiao (2020) suggests using deep learning classifiers like LSTM for forecasting power load in a smart grid. LSTM is derived from RNN networks and employs memory modules to prevent the gradient from disappearing or expanding after multiple steps. This makes LSTM particularly useful for deal-

ing with and forecasting events with prolonged intervals and delays in time series data. LSTM classifier can also be applied in forecasting renewable energy, such as wind power and photovoltaic power generation. Deep learning classifiers can be useful, for instance, in identifying faults, protecting flexible equipment in power systems, and examining consumer electricity consumption. Smart grids, among other computerized systems, are prone to cyberattacks. Deep learning can be applied to automatically detect characteristics of network attacks, identify malware and intrusion, and furnish network security for power systems.

2.2 Cybersecurity definition and concepts

Cybersecurity can be perceived as "the art of protecting networks, devices, and data from unauthorized access or criminal use and the practice of ensuring confidentiality, integrity, and availability of information" (CISA, 2009). Confidentiality states that data should not be disclosed to unapproved individuals, organizations, or processes, or accessed without the appropriate authorization. Integrity indicates that the data in question must not be modified or tampered with in any way, thus ensuring the accuracy and completeness of the data is crucial. The data is expected to be accessed and modified by authorized individuals and should remain in its intended state. Availability is ensuring that information is accessible upon legitimate request and that authorized individuals can access the data when required.

Cybersecurity risks include financial loss, disruption, or damage to the reputation of the organization due to the failure of its IT systems. Cyber risk can be seen as a potential situation of exposure of business-related knowledge or data and/or communication systems to malign actors, elements, or circumstances that may cause loss or damage. Risk can be determined from the formula: asset + threat + vulnerability (Flores et al., 2017). Malign actors, such as black-hat hackers, malware authors, organized cyber-criminals, or even governmental/state actors utilize threats (attack vectors), such as DoS/DDoS, malware, phishing attacks, social engineering, and ransomware to exploit vulnerabilities to obtain, damage, or destroy assets. An asset can be, for example, data, devices, or other components of an organization's systems containing important and/or sensitive data, or an asset can be used as a tool or way to access such relevant information (NIST, n.d.).

Nowadays cyberattacks are becoming more and more sophisticated, targeted, widespread, and undetected. According to Zheng et al. (2022), the traditional cyber technologies used in cybersecu-

rity, such as access control, authentication, encryption, intrusion detection system (IDS), vulnerability scanning, and virus protection, etc. have provided a certain level of security. However, the development of e.g., diversification attacks the traditional cyber defense measures are not sufficient. Lahcen & Mohapatra (2022) states that even though artificial intelligence, and its subset, machine learning is not an omnipotent solution in the field of cybersecurity, it can still provide an efficient tool that can be exploited in various areas of information security. Security analysts are receiving a significant number of alerts and as a result, need an efficient system to evaluate them. Machine learning can bring efficiency, improve authentication, and protect against attacks. In this thesis, machine learning classifiers will be examined from the cyberattack countermeasure (attack detection) point of view.

2.3 Critical infrastructure and trends

According to Lewis (2006), certain national infrastructures are so important that if they fail or are destroyed, it will greatly affect the defense or economy of the country and are therefore called critical infrastructures. Various definitions of critical infrastructure exist, and the definition has changed over time. In 1996 President Clinton signed Executive Order EO-13010 1996, providing the first official federal definition of critical infrastructure. EO 13010 also created eight critical physical and cyber threats and assuring the continuity of their operations. Critical infrastructures include systems such as banking and finance, emergency services, government continuity, gas and oil storage and transportation, power systems, telecommunications, transportation, and water supply systems.

Critical Infrastructure can be seen as assets, systems, and networks vital for the functioning of a society and insuring citizens' well-being and industrial and economic development (Rosato et al., 2020). According to Communication from the Commission on Critical Infrastructure Protection in the Fight against Terrorism (2004), critical infrastructure can be defined as: "critical infrastructures are those physical and information technology facilities, networks, services and assets which, if disrupted or destroyed, would have a serious impact on the health, safety, security or economic well-being of citizens or the effective functioning of governments in European Union countries". In contrast, the Australian state, and territory shared the following definition of critical infrastructure: "those physical facilities, supply chains, information technologies, and communication networks which, if destroyed, degraded, or rendered unavailable for an extended period, would significantly impact the social or economic wellbeing of the nation, or affect Australia's ability to

conduct national defense and ensure national security (Australian Government)". These definitions are very similar to each other, but the Australian state, and territory bring to light Australia's ability to conduct national defense and ensure national security. These defense and security fields are vital to be secured and appropriately protected from incoming cyberattacks.

Critical infrastructure sectors are not separate, but they produce interdependent relationships meaning that a single critical infrastructure can be dependent on products and services provided by another critical infrastructure. Hence, the critical infrastructure may be depending on the products and services provided by the previous critical infrastructure. In case of a cyberattack, causing possible disruption, damage, or even destruction of critical infrastructure, intercedence between critical infrastructures may pose cascading effect in the "network" of critical infrastructures. Interdependencies can be physical, geographical, cyber, and logical. A cyber interdependency means a dependency on information and communications systems. Cyberattacks on cyberinfrastructure may lead to a significant effect on performance, reliability, security, and safety for each of the dependent infrastructures. (Alcaraz, 2014)

Critical infrastructures, such as transportation, electric power plants, communication grids, healthcare facilities, etc. are under daily attack these days. According to Microsoft Digital Defence Report (2022), cyberattacks targeting critical infrastructure increased from 20 % of all state attacks Microsoft detected to 40 %. The climb was due to Russia's aim of causing harm and destroying Ukrainian infrastructure, and espionage on Ukraine's allies. Up to 90 % of the Russian attacks detected targeted NATO member countries, and almost half of the attacks were against IT companies in NATO countries, most of them against the USA.

Attacking critical infrastructure can be life-threatening and cause significant harm to people, even loss of life. For example, damaging electric power plants, obstructing healthcare operations and patient care, poisoning drinking water, the air, etc., can cause considerable damage. The energy sector is one of the primary critical infrastructure targets of cyberattacks, along with other vulnerable sectors, such as the critical manufacturing industry, public sector services, telecommunications, and transport. According to the Microsoft Defence Report (2022), most of the cyberattacks targeted information technology (22 %), nongovernmental organizations (17 %), education (14 %), government (10 %), finance (5 %), media (4 %), and 2% for communications, healthcare, intergovernmental organizations, and transportation. Commonly used cyberattacks against critical infrastructure are DoS/DDoS, false data injection attacks (FDIA), malware, phishing, and ransomware.

Organizations are in growing need of cybersecurity and cyber-resilience plans to protect against cyberattacks and mitigate damage caused by cyberattacks concerned. Especially in the case of critical infrastructure systems, which are extremely complex and interdependent. Cyber resilience may be defined as the ability to adapt to changing conditions (CISA, 2019). Cyber resilience can be seen as the ability of an organization to protect itself from, detect, respond to, and recover from cyberattacks. Therefore, according to Rehak et al. (2018), resilience can be perceived as a characteristic that decreases the vulnerability of an element, withstands the effects of disruptive events, increases the element's ability to react and recover, and enables its adjustment to disruptive events like those encountered in the past. When being resilient, organizations can mitigate the effect of an attack, protect digital data, and systems from cyberattacks, and continue to operate effectively in case of a successful attack.

2.4 Basics of artificial intelligence and machine learning

Buczkowski (2017) states that artificial intelligence is an umbrella term that includes Machine Learning (ML), and Deep Learning (DL). Artificial intelligence can be seen as the development of smart systems or intelligent machines carrying out tasks that typically require human intelligence. Therefore, the objective of artificial intelligence is to empower computers and systems to imitate human thinking, replicate human activities, and solve problems more quickly and efficiently than humans typically can. According to SCS (2020) Machine learning provides algorithms for learning from data and makes decisions based on patterns observed. Machine learning requires human intervention when the decision is not correct. Deep learning is a subset of machine learning utilizing an artificial neural network to reach accurate conclusions without human intervention. Deep Learning uses various layers within the network structure and attempts to learn the hidden meaning of the data, which is why they are called deep.

Decision Tree (DT) represents the more conventional methods used in artificial intelligence development, but it is still the most powerful and popular tool for classification and prediction. A decision tree has many analogies in real life, and it has influenced a broad area of machine learning and has been utilized in both classification and regression tasks. The decision tree is a flowchart-like tree-structure model of decisions using the branching method to visualize every possible output for a specific input. According to Uddin et al. (2019), the decision tree has certain advantages

such as the data preparation can be easier than for other classifiers, it can generate robust classifiers, and the computational requirements are lower. It is prone to overfitting, and it does not perform as well as other classifiers presented in this thesis.

Random Forest (RF) is a method of ensemble learning that combines the output of multiple decision trees to produce more accurate results in comparison to a standard decision tree classifier. A forest of trees protects each other from unique errors and improves the final prediction. (Shahrivari et al., 2020). Random forest classifiers perform better than a single decision tree, they scale well for large datasets and avoid overfitting due to the use of multiple trees. Random forests are, though, more complex, and computationally expensive. Overfitting is possible also for random forest classifiers, but they are less prone to it compared to decision trees. Visualization of random forests is complicated, and the process of generation and analysis requires time. (Uddin, 2019) However, in general, random forests are easy to use and flexible, and they provide accurate predictions that can be used in various fields.

According to JavaTpoint (2021), Naïve Bayes (NB) is a probabilistic machine learning classifier used for classification tasks, such as sentiment analysis, spam filtering, text classification, and recommendation systems due to its simplicity, efficiency, and easy understandability. Naïve Bayes is based on Bayes' theorem, used to determine the probability of a hypothesis with prior knowledge, depending on the conditional probability. Uddin et al. (2019) state that Naïve Bayes is known for its simplicity and usefulness for large datasets, its ability to yield probabilistic predictions, its applicability for both binary and multi-class classification problems, and its ability to direct the prediction of posterior probabilities. As a disadvantage, the classifier assumes the normal distribution of numeric attributes, and the classification performance may be decreased due to the presence of dependency between attributes.

A support Vector Machine (SVM) is a supervised learning classifier used for the classification, regression, and detection of outliers. Support Vector Machine learns by example to assign labels to objects, which it can use, for example, to detect fraudulent credit card activity by exploring a significant number of fraudulent and non-fraudulent credit card activity reports (Noble, 2006). Support Vector Machines can also be applied to, for example, intrusion detection, or healthcare-related prediction and recognition tasks, such as breast cancer diagnostics or protein structure prediction. According to Uddin et al. (2019), Support Vector Machine is less prone to overfitting, it performs well in classifying semi-structured or unstructured data, such as images, text, and trees,

and it works well outside of training data. However, Support Vector Machine is computationally costly, particularly for big and complex datasets, and it does not perform well if the data set contains noise, for example, target classes overlap. Additionally, interpreting and understanding the final model, variable weights, and individual impact is difficult.

A neural network (NN) is a classifier inspired by the biology of nervous systems, used to solve problems in pattern recognition, data analysis, and control. Neural networks are endeavors of generating machines that operate similarly to the human brain by using components behaving like biological neurons (Picton, 1994). According to Uddin (2019), neural networks can be used for both classification and regression problem-solving, and there are various types of networks with unique and special strengths. For example, convolutional neural networks (CNNs) can be applied in image classification and signal processing, Recurrent Neural Networks (RNNs) could be used in text-to-speech conversation technology, etc. The disadvantage of neural networks is the lack of user access to the exact decision-making process, predictions not always in a continuous range, and the computational expenses of neural networks when training the network for a complex classification problem.

2.5 Performance evaluation metrics

The performance of the classification algorithms (ML Technique) can be evaluated by calculating the accuracy, precision, and recall based on True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) for each machine learning classification algorithm.

Accuracy (performance measure) is the ratio of correct predictions to all predictions that are calculated by dividing correct predictions with all the instances into the test data (Joshi, 2016). Accuracy is illustrated as the percentage of correct predictions over all instances. The equation is the following:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision is the ratio of true positives to all positive predictions (Joshi, 2016). Precision indicates how accurate the classifier is at identifying an attack. Precision can be determined by the number of true positive findings divided by the addition of true positive and false positive findings. The equation to calculate precision is the following:

$$Precision = \frac{TP}{TP + FP}$$

Joshi (2016) states that recall, which can also be considered sensitivity, can be indicated as the ratio of the quantity of correctly identified positive findings to the total quantity of actual positive findings in the data. Recall expresses the number of cyberattacks the classifier can detect.

The formula to calculate Recall is the following:

$$Recall = \frac{TP}{TP + FN}$$

2.6 Cyberattacks on critical infrastructure facilities and countermeasures

In this section, adversarial attacks, DoS and DDoS attacks, FDI attacks, malware attacks, phishing attacks on critical infrastructure, and corresponding defensive mechanisms are studied. This section provides fundamental information about these attacks and countermeasures forming a theoretical foundation to provide an understanding of how these attacks function and how they affect critical infrastructure facilities.

2.6.1 Adversarial attacks and defenses

Nowadays adversarial attacks on machine learning and deep learning classifiers are more and more general, causing various security concerns and in the worst case posing serious threats to critical infrastructure facilities using artificial intelligence-based models. As stated by Vähäkainu et al. (2022): “in the context of a smart building, an attacker may have a chance to deceive the ML model into causing harm, such as to create conditions for consumption spikes, when attacking the heating system guided by predictive machine learning-based feedback system”. Adversarial attacks can be considered attack vectors constructed by utilizing artificial intelligence. By using these kinds of attacks, an adversary can cause adversarial perturbations that are invisible in the eyes of the beholder but can cause adverse effects on neural network classifiers (models). An adversarial attack is a technique for creating adversarial examples, which are inputs given to a machine-learning classifier, deliberately causing disruption, and triggering the classifier or classifiers concerned to operate incorrectly or inaccurately and to make false predictions while providing a valid input in human eyes (Ibitoye et al, 2019).

According to Sagduyu et al. (2019), adversarial attacks can be causative, evasion, or exploratory type. A characteristic of a causative attack is a manipulation of training time data producing misclassification effects. Typical for evasion attacks is to concentrate on specifying the data samples that can be already misclassified by the target classifier. For example, an evasion attack can be a spam mail generator that has the capability to circumvent spam detection filters. Exploratory attacks target the testing phase of a classifier and can be considered white-box attacks, in which case an adversary has sufficient knowledge about the classifier algorithm or training data. In some cases, exploratory attacks can also consist of black-box attacks, and in that situation, the adversary has no knowledge about training data nor the classifier algorithm or type, or the detector's model parameters to analyze the vulnerability of the model (Biggio et al., 2013).

One kind of strategy to conduct an effective attack is to use a black-box attack method and train a substitute classifier mimicking the target black-box classifier, and then use white-box attack methods, such as FGSM or JSMA, on the substitute classifier concerned after the substitute classifier is trained and strong enough. According to Wiyatno et al. (2019): "concretely, the attacker first gathers a synthetic dataset, obtains predictions on the synthetic dataset from the targeted model, and then trains a substitute model to imitate the targeted model's predictions". After details of the substitute model are known, white-box methods can be applied. Whether this attack method is successful or not depends on selecting similar synthetic data samples and substitute model architecture in addition to having adequate knowledge of the target classifier.

Vähäkainu et al. (2022) state that: "a white-box attack uses the target model's gradients in producing adversarial perturbations". The Fast Gradient Sign Method (FGSM), introduced by Goodfellow et al. (2018), is a technique for generating adversarial examples against neural networks. The method applies to any machine learning classifiers that use gradients and weights and can be executed with relatively limited computational resources. The gradient is calculated by using the back-propagation method. According to Co (2018), FGSM can be efficient to execute if the learning classifier architecture is known and internal weights are identified. FGSM is one of the simplest methods to generate adversarial examples, and it produces perturbations in many dimensions, which means it can become detectable in some cases even in the eye of the beholder.

Jacobian Saliency Map Approach (JSMA) is another gradient-based white-box attack presented by Papernot et al. (2016), which can be used to deceive machine learning (or deep learning classifi-

ers), such as neural networks, by generating adversarial examples to be sent to the targeted classifier. According to Wiyathno & Xu (2018), JSMA is usually applied in image classification tasks to fool image classification models by saturating a few of the pixels in the image concerned to maximum or minimum values. As a result, the classification model misclassifies the resulting adversarial image as a specified erroneous target class. JSMA can cause even more perturbation than FSGM, despite altering just a few pixels. Due to the capability of perturbing only a few pixels when conducting an attack, JSMA is far less detectable than FSGM and it can be used for targeted misclassification attacks. However, JSMA is an iterative and therefore greedy algorithm and requires more computational resources than FGSM.

Carlini and Wagner (2017) presented an iterative gradient-based attack method (C&W attack), which can succeed against most defensive methods, such as the defense distillation, and undistilled neural networks with 100 % probability. Defensive distillation is a technique that enhances the robustness of a neural network chosen at random and greatly diminishes the effectiveness of adversarial examples creation from a 95% success rate to just 0.5%. "According to Vähäkainu et al. (2022), a C&W attack is an attempt to make a modified image as similar as possible to the original while still causing the model to classify it incorrectly. This is in alignment with the findings of Short et al. (2019). C&W attacks are generally so powerful against neural networks that they can mitigate classifier accuracy near zero percent, and they can reach high success rates against trained neural networks used in image classification tasks (Ren et al., 2020). However, the C&W method’s computational resource requirements to generate adversarial examples are eminent due to the optimization problem (Poudel, 2020), hence it is still currently one of the best algorithms to generate them.

Defending against adversarial attacks is challenging. However, one of the first methods to combat adversarial examples is adversarial training of machine learning classifier (such as DNN), making the classifier concerned more robust to counter incoming adversarial attacks. As stated by Vähäkainu et al. (2022), the robustness of a machine learning classifier can be improved by augmenting the classifier training dataset with perturbed inputs containing both non-iterative and iterative adversarial examples. However, if an adversary uses a different attack strategy, it has a mitigating effect on adversarial training accuracy. In addition to high computational cost and complexity, the robustness of adversarial training can be circumvented by initiating a joint attack with random perturbation from other models.

Papernot et al. (2015) presented a defensive mechanism called defense distillation to mitigate the impact of adversarial examples on machine learning DNN classifiers and make them less prone to exploitation. According to Goodfellow & Papernot (2017), defense distillation is a method of training a classifier (model) by having it predict the probabilities of another model that was trained before it. In other words, the distillation procedure uses knowledge from a different deep neural network, at times from a larger architecture to a smaller one, to train a selected deep neural network. Defense distillation mitigates adversarial samples crafted using the FSGM or JSMA methods, makes the crafting process more challenging, generalizes the samples outside the training set, mitigates the effectiveness of adversarial samples on deep neural networks, and increases the resilience of deep neural networks to adversarial samples. However, defense distillation can prevent weaker attacks, but more powerful attacks, such as C&W, have been proven to evade the defense-distilled defenses.

The effect of adversarial attacks has been attempted to mitigate by utilizing various defense methods, such as previously mentioned adversarial training or defense distillation. These methods among others have certain limitations when defending against adversarial attacks, such as being effective in either white-box or black-box attacks, but not in both. Defense distillation is also vulnerable to poisoning attacks where an adversary corrupts the training data The Defense-GAN approach introduced by Samangouei et al. (2018) can be used to protect classification networks against both white-box and black-box adversarial attacks, even without knowledge of how the adversarial examples were created, as stated by Samangouei et al. (2018). According to Laykaviriakul & Phaisangittisagul (2023), the method tries to re-create legitimate samples from the adversarial samples instead of the auto-encoder by using a generative adversarial network (GAN) that is trained on unperturbed data, e.g., unperturbed images. The method locates the closest output to the given image that does not include adversarial changes and passes it to the classifier. The defense-GAN method can be used to prevent the effect of novel powerful attacks, such as C&W. However, if GAN is not correctly trained, the performance of the Defense-GAN can considerably mitigate (Chakraborty et al., 2018).

2.6.2 DoS and DDoS attacks and defenses

A distributed denial-of-service (DDoS) attack is a form of denial-of-service (DoS) attack that involves using multiple systems to attack a targeted system. Nowadays these attacks concerned are

numerous, devastating, sophisticated activities, and sometimes even considerable business. Various types of DDoS attacks are identified and most of them are efficient in paralyzing communication in the networks. Adversaries may conduct these types of attacks by using a single or a network of remotely controlled, well-structured, and widely dispersed nodes (zombie computer resources) that utilize all the target's servers (for example, bank, credit card payment gateway, etc.) A DDoS attack has the capability to prevent legitimate users from accessing the resources required by overwhelming a server with malicious traffic. As a result of the attack, the target server either crashes or becomes incapable of serving legitimate requests. Detecting DDoS attacks can be challenging as they often affect slow network traffic or technical problems. Not all attacks can be detected and prevented, but various kinds of advanced countermeasures exist to mitigate these attacks.

The Open System Interconnection (OSI) model consists of 7 layers, namely: Physical, Datalink, Network, Transport, Session, Presentation, and Application. DDoS attacks can be conducted on three of these layers: network, transport, and application. Most DDoS attacks target the network and transport layers. According to Qureshi (2018), common attacks on these layers are ICMP flooding on the network layer, SYN flooding, and Smurf attacks on the transport layer. By exploiting a vulnerability in the Telnet server running on the switch, an attacker can use DDoS techniques to disrupt the availability of Telnet services at the session layer. On the application layer, a breach or vulnerability in a web application can be exploited to flood the server or database the application is using to knock it down. DDoS attacks can pose a remarkable threat to critical infrastructure sectors, such as energy and transportation, for example, by disrupting and incapacitating heating distribution systems, causing train delays, paralyzing ticket systems, and overall disruption over travel services.

Detecting and preventing DDoS attacks is extremely difficult due to their distributed nature. DDoS attacks can be detected, for example, by using statistical, knowledge-based, or soft computing methods. In the statistical method, traffic statistics are extracted including destination/source Internet Protocol (IP) addresses, Transmission Control Protocol (TCP) flags, packet sizes, flow rate, and others, to classify abnormal traffic behavior from normal behavior (Majed, 2020). Hence, statistical tests are used to examine whether new instances belong to the statistical model of normal traffic or are classified as anomalies (Kumar, 2013). Ghaben (2021) states that in the knowledge-based method of traffic or flow, patterns are matched against a set of predefined rules. If an attack fits the rules, the traffic or flow is flagged as an attack, otherwise, it is considered normal.

DoS/DDoS attacks can also be detected by using soft computing methods, such as fuzzy logic, probabilistic reasoning, neural networks, machine learning methods, etc. He et al. (2017) suggested a DoS attack detection system that employs supervised machine learning methods such as Linear Regression, SVM, Naive Bayes, and Random Forest and unsupervised methods such as K-means and GMM-EM. The authors found that supervised algorithms achieved 93% accuracy while unsupervised methods only attained 63-64% accuracy.

2.6.3 FDI attacks and defenses

False data injection (FDI) attacks are among one the top priority attacks against cyber-physical systems these days and due to their sophisticated nature, it poses a remarkable threat to the power grids, and smart grids used to provide power to critical infrastructure facilities. According to Farmanbar (2019), smart grids use information and communications technology (ICT) to create a system that is reliable, efficient, and robust for electricity transmission and distribution. It also integrates non-renewable and renewable energy sources to decrease environmental issues. The critical infrastructure of the country includes the power system; hence the safe operation of power grids is crucial to national security. There are various examples in the world of cyberattacks against power grids, such as an attack on the Ukrainian power grid in December 2015 (CISA, 2021), causing a massive blackout. False data injection attack seriously threatens the secure and smooth operation of complex information-physical coupled smart grids, therefore efficient means of False data injection attack detection are essential (Li et al., 2022).

Smart grids can be targeted through false data injection attacks, which can disrupt the balance of energy demand and supply, affect the functioning of the grid network, and alter electricity pricing. Manipulating energy demand, through the introduction of false values in the state estimation process, can lead to power outages and cause financial harm to both consumers and providers. Attacks on energy supply induce incorrect energy distributions leading to extra costs or even devastating consequences (Chen et al., 2015). According to Elmrabet et al. (2018), false data injection attacks are a form of violation that disrupt the integrity of measurements taken by devices, leading to errors and distortion. This can negatively impact the precision of state estimation, which is vital for ensuring the reliable operation of the power system and gathering accurate real-time data. Violations of SE's integrity can make the smart grid system unstable. An attacker can use a false data injection attack to manipulate smart meter data to reduce their electricity bills or target smart meters, sensors, or remote terminal units (RTU), intercept communication between sensor

networks and the SCADA system, or gain access to the SCADA system to introduce false data that closely mimics the actual states and parameters of the system. This makes false data injection attacks difficult to detect, especially if the system architecture is not well understood. As a result, it can cause prolonged power outages and wide-area power failure accidents.

There have been various attempts to detect false data injection attacks, including methods such as sparse matrix optimization, using a Kalman filter with a threshold based on the Euclidean distance metric, blockchain technology, cryptography, and learning-based techniques (Reda et al., 2021). While threshold-based detection methods have shown to be effective in identifying false data injection attacks, some attacks have been able to bypass these methods. Cryptography has been used to prevent false data injection attacks, but computing requirements are extensive. The use of blockchain to protect power generation and distribution systems helps to prevent data manipulation and guarantee data immutability (Aggarwal et al., 2021). The learning-based method is a novel and sophisticated countermeasure to detect false data injection attacks. According to Wang et al. (2019), in empirical tests, the RNN-based method reached 92.58 %, accuracy, the SVM-based method 90.06 %, Sparse optimization 86.79 %, and Euclidean detection 72.68 %. The recurrent neural network with wide components consisting of fully connected layers of neural networks reached up to 95.23 % accuracy. Another approach, conditional deep belief networks (CDBN), has been proposed as a means of detecting false data injection attacks that may evade detection by the state vector estimator (SVE) mechanism. This method aims to identify false data injections that are otherwise unobservable.

2.6.4 Malware attacks and defenses

Malware, also considered malicious software, is a tool employed by adversaries with the intent to disrupt or damage target computer functions or devices, extract sensitive information, or gain unauthorized access to private computer systems and networks. Palo Alto Networks (2022) defines malware as follows: “Malware (short for “malicious software”) is a file or code, typically delivered over a network, that infects, explores, steals, or conducts virtually any behavior an attacker wants”. Malware comes in various forms such as adware, ransomware, spyware, trojan viruses, viruses, and worms (Cisco, 2022). Amongst them, worms and trojans are the most common type of malware threats, more prevalent than traditional computer viruses. In the past years, ransomware attacks have been increasing, making it the most common form of malware impacting critical

infrastructures, such as hospitals, communications firms, railway networks, and governmental offices.

There has been a significant increase in malware attacks on critical infrastructure in recent years. The FBI internet crime complaint center found that in 2021, ransomware was a leading threat to critical infrastructure security in the USA, impacting over 600 organizations (Waldman, 2022). Dossett (2021) states that in 2021 Kaseya, which provides IT solutions for other companies, was hit by the cybercriminal group REvil in a ransomware attack. The group impacted more than 1500 organizations throughout the world and claimed thousands to multiple millions of dollars as ransoms. JBS USA, one of the largest suppliers in the US, was hit by ransomware that causes its operations to temporarily halt. JBS ended up paying 11 million USD as a ransom. America's largest "refined products" pipeline was knocked down by the Darkside group by a ransomware attack. The pipeline covers over 5500 miles and transports more than 100 million gallons of fuel per day. The attack affected gallons of gas price in the USA, and the price increased by more than 3 USD for years. Ransomware attacks like these are increasing, and posing a significant threat to critical infrastructure, organizations, and people in general in the world.

The critical infrastructure energy sector, one of the main targets of cyberattacks on critical infrastructure, has been hit by various malware attacks over the years, such as BlackEnergy on the Ukrainian electrical power industry for opening a backdoor to hackers, Shamoon on Saudi Arabia's national oil conglomerate (Saudi Aramco) for stealing passwords, wiping data, preventing rebooting, etc., and Stuxnet worm on Iranian nuclear centrifuges damaging them. Dragonfly malware campaign targeted defense and aviation companies, but later, the cybercriminal group behind it started to focus on the energy sector. According to Cyber Security Review (2017), Dragonfly 2.0 was a malware campaign targeting the critical energy sectors in the USA, Turkey, and Switzerland. In the campaign, the attackers used malicious email attachments, watering hole attacks, and Trojanized software as an initial attack vector to gain access to the target network. The attackers also utilized the Phishery toolkit to conduct email-based attacks to steal credentials. The campaign included various remote access trojans permitting access to the target computer.

Malware attacks have been increasingly targeted at healthcare organizations in the past years due to their dependency on access to relevant patient data. In 2022, healthcare organizations faced Venus Ransomware, which is capable of encrypting victims worldwide by targeting publicly exposed Remote Desktop services and by using AES and RSA encryption algorithms (HC3, 2022).

Berry (2022) claimed that two-thirds (66 %) of healthcare organizations experienced ransomware attacks in 2021, and 61 % of them reported their data is encrypted during the attacks. Healthcare organizations that paid the ransom to restore their data recovered only 65 % of their data. However, almost ¾ of organizations were able to restore encrypted data from backup files. The lowest average ransom payment was almost 200 000 USD, but some organizations ended up paying more than one million. In addition, more than 90 % of healthcare organizations experienced that the ransomware attack impacted their operating ability and 90 % of them thought that attacks caused them to lose business or revenue. In 2021, the average time for a healthcare organization to recover from a ransomware attack was around one week.

There are various ways to detect malware. Xiao et al. (2018) state that two of the most widely used methods are signature-based detection and behavioral-based heuristic detection. Signature-based detection uses an algorithm to calculate a unique numerical value (known as a malware signature) for specific types of malwares. This value can be used to identify and block malware from entering a system. The method is efficient, but it has challenges to detect zero-day or obfuscated malware. Heuristic scanning examines code for suspicious properties, looking for malware-like behavioral patterns. Heuristic scanning can detect unknown malware types in addition to encrypted, obfuscated, or polymorphic malware. According to Sprengers & Haaster (2016), another approach to detecting malware is heuristic classification, which often employs machine learning. Various machine and deep learning classifiers such as decision trees (DT), naive Bayes (NB), neural networks (NN), random forests (RF), and support vector machines (SVM) are effective in detecting malware with a high degree of accuracy. One of the disadvantages is that heuristic classification is prone to have a high false positive rate, meaning that various legitimate actions can be classified as intrusive. In addition, useful training data is needed, which is challenging to gather in a comprehensive IT environment.

2.6.5 Phishing attacks and defenses

According to Imperva (2022), phishing is a form of social engineering attack that is used to steal personal information such as login credentials or credit card numbers by bypassing technical controls in information systems. In social engineering, an adversary attempts to exploit human error or lack of knowledge to collect sensitive private information, access, or valuables. An adversary appears as a trusted entity luring a victim user to click and open an incoming email, text, or short message, and a malicious link included in the message can freeze the system, unveil critical and/or

sensitive information, provide means to identity theft, stealing funds, etc. Phishing attacks can be used as a part of more comprehensive attack scenarios, such as advanced persistent threats (APT). In the advanced persistent threat scenario, compromised victims are exploited to circumvent security perimeters, spread malware inside the information system, or obtain access to critical secured data.

Phishing is among the top cyberattacks causing data breaches with almost 14.5 billion spam emails sent daily (Ripa et al., 2021). In 2022 phishing was the most common way cybercriminals penetrated an organization. According to Cytomic (2019), up to 46% of successful cyber-attacks begin with an email phishing attack. Phishing attacks are conducted even more these days as it requires only a single employee of an organization to make a mistake to cause a significant loss of business and reputation. Adversaries are continuously updating their phishing attack procedure and utilizing advanced, and novel tools for conducting these attacks. Hence, cybersecurity professionals must continuously update their knowledge about new types of phishing attack trends to develop and implement corresponding defensive countermeasures.

According to Enisa (2022), one of the widely used and more sophisticated versions of phishing is spear phishing. According to DNI, Spear phishing is: “a type phishing campaign that targets a specific person or group or often will include information known to be of interest to the target, such as current events or financial documents”. Spear phishing exploits the human component to take an advantage of basic human traits, such as being helpful and friendly, being loyal to authority, or just curious about topical events and news. Spear phishing can be used to target specific individuals or organizations to gain unauthorized access to confidential information. An adversary can utilize publicly available information on social media, for example, Facebook or LinkedIn to customize the spear phishing email to deceive the target end-user (victim) who is then likely to react to it. While the purpose of spear phishing usually is to steal confidential data, adversaries may also utilize it to install malware on the end user’s computer to perform malicious actions.

Adversaries utilize spear-phishing attacks to target individuals, or groups of people with something in common, such as employees working in the same department having access to important information system accounts on banks in the financial services sector, which is one of the essential critical infrastructure sectors. In 2016, adversaries conducted a business email compromise (BEC) spear-phishing attack targeting a high-ranking executive (CEO) of Belgian Crelan bank gaining ac-

cess to his email account. Adversaries were then able to spoof the CEO's email account by personating the executive as the sender and asking employees to deposit money (up to $75.8 million in total) into the executive's account. The attack was detected during the internal audit process, but the identity of the adversaries remained unknown.

According to MSTIC (2022), the Microsoft Threat Intelligence Center (MSTIC) revealed that the Russian-linked Gamaredon/Actinium (primitive bear) hacker group has targeted government, military, non-government (NGO), judiciary, law enforcement, and non-profit organizations in Ukraine to steal sensitive information. Actinium, which has been considered in belonging to the Russian Federal Security Service (FSB), has been observed with objectives related to cyber espionage. The group has conducted cyber-espionage campaigns in Ukraine since at least 2014 until today. As the first attack vector, the group concerned utilizes spear-phishing emails, fooling them to come from legitimate organizations, but containing malicious macro attachments, and tracking components providing information to the adversary if the email(s) has been opened. The group uses targeted "spear-phishing" emails using remote document templates and remote macro scripts to infect only specific targets and mitigate being detected by anti-malware systems (Tung, 2022).

According to Aljofey et al. (2020), defensive measures against phishing attacks can be classified into various categories, such as list-based detection, deep learning-based detection, machine learning-based detection, heuristic-based detection, and hybrid methods. List-based detection can be divided into two sub-categories: whitelist-based and blacklist-based technologies. Whitelist-based techniques maintain a list of safe URLs and IP addresses that are permitted to access data or networks. One drawback of list-based techniques is that they may not be able to detect phishing sites if the targeted site is not included in the whitelist. Blacklist-based techniques, on the other hand, maintain a list of known malicious items that are required to be blocked. These are commonly used in anti-phishing toolbars such as Google safe browsing, providing warnings to end-users. A list of malicious phishing URLs is challenging to keep up to date as the threat situation changes continually. To enhance blacklist-based detection methods, additional information such as domain name and server details or a blacklist of signatures can be employed to detect new phishing URLs.

Heuristic-based detection methods, which evolved from list-based detection techniques, rely on extracting characteristics from a web page (potentially a phishing site) to determine its authenticity, rather than relying on pre-compiled lists. Heuristic-based detection methods extract features

from the website's URL and HTML Document Object Model (DOM). According to Bhattacharyaa et al. (2017), these features are then compared with a set of known characteristics gathered from both phishing and legitimate pages to determine the legitimacy of the website. Aljofey et al. (2020) state that an example of a heuristic method is Cantina, which uses the Google search engine to gather keywords and domain names from a website and employs research results and other heuristic rules to determine the legitimacy of a webpage; whether it can be considered benign (legitimate) or malicious (phishing). Heuristic-based detection has fewer false positives and false negatives, and it is faster than the list-based technique, but as a disadvantage, it is less accurate and it can be circumvented if the heuristic technique has been revealed (Rao et al., 2015).

According to Aljofey et al. (2020), machine learning-based methods, which enable computer systems to learn, can be used to develop techniques for mitigating phishing attacks. An extensive amount of information can improve phishing detection accuracy, but in general, both computational resources and time are limited resources, and therefore a significant number of features are not possible to extract. Features may include content-related, lexical, or WHOIS-related features. Lexical features include dots in URLs, special characters, and IP-address contained in URLs (Zhang et al., 2011). As an example of phishing detection by using ML-based methods, Jain et al. (2018) proposed URL based anti-phishing learning method by extracting 14 features of the URL to identify the website as malicious (phishing), or benign (legitimate). The authors trained their proposed system using over 30,000 malicious and benign URLs, utilizing Support Vector Machine (SVM) and Naive Bayes (NB) classifiers, achieving an accuracy rate of up to 90% in detecting malicious websites with the SVM classifier. Rao et al. (2019) proposed a more sophisticated method, CatPhish, which can predict whether a URL is legitimate or not without accessing the website's content. It extracts features from questionable URLs and uses a random forest classifier for classification.

Deep Learning-based methods, such as convolutional neural networks (CNN), deep neural networks (DNN), recurrent neural networks (RNN), and recurrent convolutional neural networks (RCNN), generally offer higher accuracy and the ability to extract features from primary data without the need for prior information or human intervention. According to Catal et al. (2022), deep learning-based algorithms, deep neural networks, recurrent neural networks, convolutional neural networks, and hybrid learning algorithms have been able to provide the best results in experiments. Surprisingly the traditional Multi-Layer Perceptron (MLP) algorithm can also provide decent results. One possible reason for the performance results can be that deep neural networks are based on multi-layer perceptron algorithms, which may be due to the widespread adoption of

these algorithms compared to other novel alternatives. Various combinations of hybrid algorithms exist, such as RNN-RNN or LSTM-LSTM, which can provide slightly better precision than solely an individual recurrent neural network or long short-term memory method. Hybrid methods combine different classification methods to combine their advantages and mitigate the disadvantages of an individual classifier providing higher accuracy, and therefore hybrid models should be considered when designing prediction models.

3 Implementation and results of the research

3.1 Data gathering

The review data (scientific articles) of this study was gathered from various databases, such as from arXiv free distribution and an open-access archive, the Institute of Electrical and Electronics Engineers (IEEE) IEEE Xplore Digital Library, SpringerLink (Springer-Verlag), ScienceDirect (Reed Elsevier), Theseus Open Repository of the Universities of Applied Sciences, and scientific journals, such as Frontiers, International Journal of Advanced Computer Science and Applications (IJACSA), International Journal of Engineering and Advanced Technology (IJEAT), International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE), Mendel Soft Computing Journal, and Scientific Reports open access journal, etc.

The data were searched for with the following search keywords: “phishing attacks machine learning detection”, “DDoS detection classification algorithms”, “an ensemble-based malware detection”, “false data injection attacks in the smart grid”, “machine learning classifiers for detecting malware”, “DDoS attack detection with Decision Tree algorithm”, “DDoS attack detection using machine learning”, “intelligent malware detection”, “convolutional neural network in malware classification”, “false data injection attack classification method”, “DDoS mitigation using machine learning”, ”ransomware detection with machine learning”, “phishing detection using machine learning”, “false data injection attacks detection in power systems”, “unknown malware detection”, “malware prediction”, “machine learning to detect unknown malware”, “machine learning performance for phishing attack detection”. The data-gathering process has been consistent, and it is reproducible.

The data gathered was then processed and presented in a form of a table describing the machine learning (ML) technique (classifier), the domain of the cyberattack, the dataset used in the research examined, reference to the research study, year of publication, and results consisting of accuracy, precision, and recall performance evaluation metrics used in conducting the research studies. ML techniques examined and compared were Decision Tree (DT), Support Vector Machine (SVM), Naïve Bayes (NB), Random Forest (RF), Neural Network (NN), and domains DDoS, FDIA, Malware Phishing, Ransomware, respectively.

3.2 Processing and analysis of the data

In this section, scientific articles obtained from various databases are studied. These articles provide rather good accuracy, recall, and precision performance measures of the machine learning classifiers examined. These machine learning classifier performances in detecting various cyberattacks are explored and compared in this section.

3.2.1 Decision Tree in detecting cyberattacks

Decision Tree classifier performance in identifying DDoS

The DDoS attack is relatively simple to conduct, impacting millions of clouds and traditional server resources worldwide each year. Detecting and preventing DDoS attacks has been under extensive research due to the popularity of the attack type concerned. Alsirhani et al. (2019) examined several machine learning classifiers, such as Naïve Bayes, Decision Tree, and Random Forest against DDoS attacks. Authors two different custom datasets with from 100 000 to two million packets. The Decision Tree classifier performed relatively well and reached 97 % accuracy, 97 % recall, and 97 % precision on DDoS attacks (Table 1). Manikumar et al. (2020) utilized the CIDDoS2019 dataset containing benign and the most up-to-date common DDoS attack consisting of real-world PCAP data. The authors tested K-Nearest Neighbor, Decision Tree, and Random Forest classifiers on DDoS attacks. The decision Tree provided an average result with 93.83 % accuracy and 94.56 % precision (Table 1), slightly less than the performance using a custom dataset in Alsirhani's research.

In Kashab et al. (2021) research, the Decision Tree classifier provided 86.74 % accuracy, 99.62 % precision, and 54.27 % recall respectively (Table 1), and the Support Vector Machine provided the lowest accuracy (71.17 %) and recall (0 %) scores. Kareem et al. (2022) utilized the CICIDS2017 dataset containing widely used and benign attacks, such as the CIDDoS2019 dataset. Their decision stump (DT) classifier resulted in only 81.58 % accuracy, 76 % precision, and 100 % recall (Table 1). Shaaban et al. (2019) utilized a custom dataset and Decision Tree classifier, which performed relatively well, providing 95 % (dataset-1) and 93 % (dataset-1) accuracy (Table 1) in detecting DDoS attacks. K-Nearest Neighbor classifier provided the weakest results in the research, which is comparable to performance results in Manikumar's and Khashab's research.

Decision Tree classifier performance in identifying FDIA

FDIA attack is a generally used method to modify the original measurements provided by sensors influencing, for example, the control system's computational resources. The Decision Tree classifier provided weaker results compared to when utilizing it to detect incoming DDoS attacks. In Lu et al. (2020) research, the Decision Tree classifier provided 81.89 % accuracy, 82.34 % precision, and 88.78 % recall (table 1).

Ashrafuzzaman et al. (2021) used the MATPOWER-generated dataset simulating the standard IEEE-bus system and presented FDI cyberattacks on the measurement data. The authors utilized the Decision Tree, Linear Regression, Naïve Bayes, Neural Network, and Support Vector Machine Classifiers in the research. The Decision Tree classifier provided 89.31 % accuracy, 99.91 % precision, and 73.02 % recall (Table 1) in the test arrangement. The authors also tested ensemble classifiers consisting of five individual and seven ensemble classifiers. In situation, the performance results of individual classifiers and ensemble classifiers did not vary much. Surprisingly, other classifiers than Decision Tree utilized in the research provided quite similar results, even compared to each other. According to the authors, this may be due to the imbalanced dataset.

Qu et al. (2021) used the ICT data set, which is CPPS data provided by Mississippi State University consisting of 15 sets of data and containing about 5000 pieces of information, such as control panel, relay records, snort logging data, etc. The authors selected six machine learning classifiers for testing: Decision Tree, K-Nearest Neighbor, Random Forest, SACS-SAE, and XGBoost. The performance of the Decision Tree classifier against the stealthy FDIA attacks before the feature selection for accuracy was 82.4 %, and precision 81.1 %, and after the feature selection 82.7 % for accuracy, and 81.5 % for precision (Table 1). The Decision Tree classifier provided an average result in the research compared to other classifiers tested. The Random Forest, SACS-SAE, and XGBoost classifiers provided slightly better (but less than 90 %) accuracy and precision performance.

Decision Tree classifier performance in identifying Malware

Malware is a common intrusive malicious software, posing a serious global security threat, and acting as an umbrella term for various malicious programs designed to intentionally cause harm or exploit to a computer, server, computer or telecommunication network, or infrastructure (for example, critical infrastructure). As traditional signature-based methods fail in detecting novel mal-

ware families, various research to identify unknown malware families have been conducted. Santos et. al. (2011) utilized machine learning classifiers, such as Bayesian Network (BN), Decision Tree, K-Nearest Neighbor, Naïve Bayes, and Support Vector Machine, to detect these kinds of novel malware families. The authors collected a malware dataset from the VxHeavens website consisting of 13189 malware executables. Results indicated that the Decision Tree classifier accuracy was 92.34 %, which was slightly higher than Naïve Bayes (89.81 %), and Bayesian Network (87.29 %) classifiers. In this research, the K-Nearest Neighbor classifier (from 93.16 % to 94.73 % accuracy) performed best with the Support Vector Machine classifier from 91.70 % to 95.80 % accuracy).

Shhadat et al. (2020) used the Bernoulli Naïve Bayes, Decision Tree, Hard Voting, K-Nearest Neighbor, Linear Regression, Random Forest, and Support Vector Machine classifiers, and collected a dataset consisting of 1156 files (984 malicious and 172 benign). In the research, Random Forest, and Decision Tree classifiers provided the best performance with 97.8 % accuracy, 87.3 % recall, and 96.4 % precision (Table 1), Naïve Bayes performed the worst with 91 % accuracy, 89.6 % recall, and 66.8 % precision (Table 3). Amer et al. (2019) gathered a dataset containing 41 324 benign, and 96724 malicious samples, and utilized various individual machine learning classifiers, such as follows: AdaBoost, Decision Tree, Extra Tree Classifier, K-Nearest Neighbor, Linear Discriminant Analysis, Multilayer Perceptron, Random Forest, Support Vector Machine, and XGBoost, and ensemble classifiers. The individual Random Forest and Decision classifiers performed the best in the research with 99.9 % accuracy, and ensemble classifiers (Proposed Ensemble Model + Random Forest + Extra Tree Classifier) provided 99.8 % accuracy respectively. However, the benefit of ensemble classifiers is that they are not prone to overfitting (unlike the decision trees) and are more generalized.

Sarah et al. (2021) selected the Drebin dataset, which consists of 215 features extracted from 15 036 applications (5560 malware, and 9476 benign) when conducting the research. The authors used four conventional machine learning classifiers, such as the Decision Tree, Gaussian Naïve Bayes, Logistic Regression, and Support Vector Machine in malware detection. In addition, the authors selected four ensemble machine learning classifies, such as the Decision Tree, Gradient Boosting, Light GBM, Random Forest, and XGBoost, respectively. The research analysis showed that ensemble classifiers reached slightly higher accuracy compared to individual machine-learning classifiers. The Decision Tree provided 99.84 % accuracy, 100 % precision, and 100 % recall (Table 1), substantially higher than Gaussian Naïve Bayes, which provided only 75.82 % accuracy and 75.9

% precision, respectively, and 82.6 % recall. However, ensemble classifiers, such as the LightGBM, provided fractionally higher performance with 99.83 % accuracy, 100 % precision, and 99.67 % recall. Azmee et al. (2020) used the Kaggle dataset consisting of malicious and benign data from PE files. The authors utilized nine classification algorithms: Adaboost, Artificial Neural Network, Decision Tree, Extra Tree Classifier, K-Nearest Neighbor, Logistic Regression, Random Forest, Support Vector Machine, and XGBoost. Artificial Neural Network provided 98.0 % accuracy, 98 % precision, 99 % recall (Table 5), Random Forest 98.1 % accuracy, 99 % precision, 98 % recall (Table 4), and XGBoost 98.6 % accuracy, 99 % precision, 99 % recall provided the best results, respectively. The Decision Tree classifier's performance was average with 97.2 % accuracy, 98 % precision, and 98 % recall (Table 1).

Liu et al. (2022) conducted their research by using the malware dataset, CICAndMal2017 consisting of 10 854 samples (4354 malware, and 6500 benign) from various sources. The authors conducted experiments on the dataset and compared it with the proposed Bidirectional Recurrent Neural Network (BIR-CNN), Convolutional Neural Network (CNN), Decision Tree, Random Forest, and Support Vector Machine classifiers. Experiments showed that the BIR-CNN classifier provided the best accuracy (99 %), precision (99 %), and recall (99 %). In this experiment, Support Vector Machine performed the worst with 89 % accuracy, 91 % precision, and 90 % recall (Table 2). The accuracy, precision, and recall of the Decision Tree was 91 %.

Li et al. (2022) collected a custom dataset (2020) consisting of 13 624 samples (6686 malware, and 6938 benign). The authors conducted their research using Convolutional Neural Network, Decision Tree, Graph Convolutional Neural Network (GCN), Naïve Bayes, Random Forest, Recurrent Neural Network, and Support Vector Machine classifiers. Research results showed that Graph Convolutional Neural Network provided the best accuracy (94.67 %), precision (94.64 %), and recall (93.21 %), Convolutional Neural Network provided 93.61 % accuracy, 93.44 % precision, and 92.59 % recall, and Decision Tree 92.45 % accuracy, 91.64 % precision, and 93.19 % recall (Table 1). Naïve Bayes provided the lowest accuracy 79.27 %, precision 81.9 %, and recall 78.47 % (Table 3).

Decision Tree classifier performance in identifying Phishing

Traditional anti-phishing techniques have not been sufficient and accurate enough, hence, the accuracy and efficiency of these techniques are essential to get improved. Meenu et al. (2019) generated custom datasets to train and test the Decision Tree, Logistic Regression, Neural Network,

and Support Vector Machine classifiers utilized in the research. Logistic Regression and Neural Network classifiers provided the best accuracy at 94.1 % for Logistic Regression, 93.65 % precision, and 93.8 % recall, and 94.31 % for Neural Network, 94.30 % precision, and 94.4 % recall (Table 5), the Decision Tree classifier provided 93.9 % accuracy, 93.3 % precision, and 93.6 % recall (Table 1), and the Support Vector Machine reached only 88.6 % accuracy, 89 % precision, and 89.6 % recall (Table 2). The authors proposed also an improved Logistic Regression classifier, which utilized the feature selection method, and reached 95.5 % accuracy.

Alam et al. (2020) a legitimate phishing dataset from Kaggle.com consisting of 32 features. The authors used Decision Trees, and Random Forest Classifiers when conducting the experiment. The Decision Tree classifier reached 91.94 % accuracy, 88.04 % precision, and 93.84 % recall (Table 1), while the Random Forest classifier provided better accuracy at 96.96 %, and 96.89 % precision, but 42.16 % recall (Table 4). According to the authors, Random Forest had less variance, and it was able to handle the over-fitting problem. Sharivari et al. (2020) gathered a dataset, which contains 11 000 sample phishing websites. However, the authors utilized only 10 % of these samples in the testing phase. The Authors utilized various classifiers, such as Ada Booster, Decision Tree, Gradient Boosting, K-Nearest Neighbor, Neural Network, Random Forest, Support Vector Machine, and XGBoost when conducting the research experiment. Random Forest and XGBoost classifiers provided the best performance with 97.26 % accuracy, 91.76 % precision, and 93.19 % recall for Random Forest (Table 4), and 98.32 % accuracy, 98.72 % precision, and 98.10 % recall for XGBoost, respectively. The Decision Tree classifier performance was only slightly lower with 96.59 % accuracy, 96.76 % precision, and 97.14 % recall (Table 1).

Siti et al. (2020) collected two datasets consisting of email (spam and legitimate messages) and SMS messages for conducting the research The first dataset was fetched from GitHub and was containing 5180 instances, and the second one was fetched from Unicamp Website and was containing 5574 instances. The authors utilized Decision Tree, K-Nearest Neighbor, Naïve Bayes, Random Forest, and Support Vector Machine classifiers when conducting the research experiment. The decision Tree classifier provided 97.78 % accuracy, and 96.7 % precision (Table 1), Naïve Bayes 98.94 % accuracy, and 98.9 % precision (Table 3), K-Nearest Neighbor and Random Forest classifiers both provided 100 % accuracy, and 100 % precision, and Support Vector Machine 99.98 % accuracy, and 99.8 % precision (Table 2). Naïve Bayes provided surprisingly high performance in phishing detection in this research as it did not perform well in detecting other attack vectors,

such as malware and ransomware. However, the Decision Tree classifier seems to provide decent performance in detecting all the cyberattacks enumerated in this thesis.

Decision Tree classifier performance in identifying Ransomware

Masum et al. (2022) gathered a ransomware dataset containing 138 047 samples with 57 features, where 70 % of samples were ransomware and 30 % were legitimate. The authors used Decision Tree, Linear Regression, Naïve Bayes, Neural Network, and Random Forest classifiers in the experiment. Random Forest classifier provided the best performance with 99% accuracy, 99% precision, and 97 % recall (Table 4). The Decision Tree was the second one with high-performance results providing 98 % accuracy, 98 % precision, and 94 % recall (Table 1). Linear regression provided 96 % accuracy, 96 % precision, and 89 % recall, and the Neural Network 97 % accuracy, 97 % precision, and 95 % recall (Table 5). Naïve Bayes provided extremely low-performance results in this research in detecting ransomware attacks with 35 % accuracy, 31 % precision, and 99 % recall (Table 3).

Table 1. Decision tree in detecting cyberattacks

ML Technique	Domain	Dataset	Reference	Year	Results		
					Accuracy	Precision	Recall
Decision Tree	DDoS	Custom	Alsirhani et al.	2019	97.00 %	97.00 %	97.00 %
		CICDDoS2019	Manikumar et al.	2020	93.83 %	95 %	-
		Custom	Khashab et al.	2021	99.11 %	98.01 %	99.01 %
		CICIDS2017	Kareem et al.	2022	81.58 %	76 %	100 %
		Custom	Shaaban et al.	2022	95.00 %	-	-
	FDIA	Custom	Lu et al.	2020	81.89 %	82.34 %	88.78 %
		Matpower (Gen.)	Ashrafuzzaman et al.	2021	89.30 %	99.91 %	73.02 %
		ICS	Qu et al.	2021	82.70 %	81.50 %	-
	Malware	Malware dataset	Santos et al.	2013	92.34 %	-	-
		Custom	Shhadat et al.	2017	97.80 %	96.40 %	87.30 %
		Custom	Amer et al.	2019	99.10 %	99.90 %	99.90 %
		Drebin	Sarah et al.	2019	99.84 %	100 %	100 %
		Kaggle (PE-dataset)	Azmee et al.	2020	97.20 %	98.00 %	98.00 %
		CICandMal2017	Liu et al.	2022	91.00 %	91.00 %	91.00 %
		Custom	Li et al.	2022	92.45 %	91.64 %	93.19 %
	Phishing	Custom	Meenu et al.	2019	93.90 %	93.30 %	93.36 %
		Kaggle (Phishing-dataset)	Alam et al.	2020	91.94 %	88.04 %	93.84 %
		Custom	Shahrivari et al.	2020	96.60 %	96.78 %	97.14 %
		Custom	Siti et al.	2020	98.05 %	98.10 %	-
	Ransomware	Custom	Masum et al.	2022	98.00 %	98.00 %	94.00 %

3.2.2 Support Vector Machine in detecting cyberattacks

Khashab et al. (2021) utilized various machine-learning classifiers in detecting DDoS cyberattacks. The Decision Tree classifier provided slightly better performance compared to the Random Forest, which provided 86.72 % accuracy, 99.45 % precision, and 52.27 % recall. However, the Random Forest provided almost as high accuracy, precision, and recall as the Decision Tree. The recall (sensitivity) of the Support Vector Machine and the Linear Regression was lower compared to other classifiers in the research indicating that the Support Vector Machine and Linear Regression did not detect as many attacks as the Decision Tree Classifier or the Random Forest. In Shaaban et al. (2019) experiment, the Support Vector Machine provided fractionally better performance for the Support Vector Machine (96.36 % accuracy) compared to the Decision Tree classifier, even though both classifiers performed relatively well.

The support vector machine (78.92 % accuracy, 81.21 % precision, 86.96 % recall) provided similar results in detecting FDIA attacks in Lu et al. (2020) research compared to the Decision Tree classifier (78.92 % accuracy, 81.21 % precision, 86.96 % recall). Surprisingly, the Ashrafuzzaman et al. (2021) experiment provided the same (89.31 % accuracy, 99.91 % precision, 73.02 % recall) performance for the Support Vector Machine and the Decision Tree classifiers. In Qu et al. (2021) research, the Decision Tree with 82.70 % accuracy, and 81.50 % precision, provided much better results compared to the Support Vector Machine with only 51.30 % accuracy and 52.40 % precision.

In detecting malware attacks, the Santos et al. (2013) experiment provided 95.80 % accuracy for the Support Vector Machine outperforming other classifiers, such as the Decision Tree (92.34 % accuracy), and Naïve Bayes (89.81 % accuracy). In another research conducted by Shhadat et al. (2017), Amer et al. (2019), Sarah et al. (2019), Azmee et al. (2020), Liu et al. (2022), and Li et al. (2022), malware detection accuracy, precision, and recall did not vary much, but in each of the research, the Decision Tree provided 1-2 percentage points better results compared to the Support Vector Machine. However, the precision in the Shadatt et al. (2017) research was significantly lower for the Support Vector Machine (88.50 %) than for the Decision Tree classifier (96.40 %).

The performance provided in these malware research experiments was similar for the Decision Tree, the Random Forest, and the Neural Network classifiers, slightly outperforming the Support Vector Machine in most cases. Not surprisingly, Naïve Bayes considerably underperformed in de-

tecting malware attacks compared to other classifiers. Naïve Bayes detected FDIA attacks with reasonable performance in the Ashrafuzzaman et al. (2021) research, but malware detection performance is weak. This may be due to the assumption that all features are independent, which may not be the real-world scenario. However, the Naïve Bayes classifier is fast and therefore can save time in solving multi-class prediction problems if the assumption of all features is independent is true. In many cases, the assumption of independent predictor features is a limiting factor.

The Decision Tree classifier outperformed the Support Vector Machine in detecting phishing attacks, even though the difference is not great. In the Meenu et al. (2019) research the Support Vector Machine provided 88.60 % accuracy, 90.40 % precision, and 93.10 % recall, and the Decision Tree provided 93.90 % accuracy, 93.30 % precision, and 93.36 % recall, the Neural Network provided 94.31 % accuracy, 94.30 % precision, and 94.40 % recall, respectively. In Siti et al. (2020) research experiment the Support Vector Machine provided almost two percentage points better accuracy, and precision compared to the Decision Tree classifier. In the experiment concerned, the Naïve Bayes classifier provided reasonable performance with 95.66 % accuracy and 96.50 % precision. Among all the classifiers compared in this thesis, the Random Forest classifier provided the best results in detecting phishing attacks, fractionally outperforming the Decision Tree, Support Vector Machine, and Neural Network classifiers.

Masum et al. (2022) conducted an experiment on detecting ransomware attacks using various classifiers, such as the Decision Tree, Naïve Bayes, Ransom Forest, and Neural Network Classifiers. The Random Forest performed the best with 99 % accuracy, 99 % precision, and 97 % recall. The Decision Tree provided almost as high performance with 98 % accuracy, 98 % precision, and 94 % recall, Neural Network provided 97 % accuracy, 97 % precision, and 95 % recall. The Naïve Bayes did not provide sufficient performance in detecting ransomware attacks either, but it provided solely 35 % accuracy, 31 % precision, and 99 % recall. Generally, the Random Forest seems to provide good results in detecting various cyberattacks, such as DDoS, Malware, Phishing, and Ransomware. The Random Forest FDIA attack detecting performance needs further research though.

Table 2. Support vector machine in detecting cyberattacks

ML Technique	Domain	Dataset	Reference	Year	Results		
					Accuracy	Precision	Recall
Support vector machine	DDoS	Custom	Alsirhani et al.	2019	-	-	-
		CICDDoS2019	Manikumar et al.	2020	-	-	-
		Custom	Khashab et al.	2021	94.99 %	97.1 %	82.81 %
		CICIDS2017	Kareem et al.	2022	-	-	-
		Custom	Shaaban et al.	2022	96.36 %	-	-
	FDIA	Custom	Lu et al.	2020	78.92 %	81.21 %	86.96 %
		Matpower (Gen.)	Ashrafuzzaman et al.	2021	89.31 %	99.91 %	73.04 %
		ICS	Qu et al.	2021	51.30 %	52.40 %	-
	Malware	Malware dataset	Santos et al.	2013	95.80 %	-	-
		Custom	Shhadat et al.	2017	96.10 %	88.50 %	86.20 %
		Custom	Amer et al.	2019	98.30 %	99.00 %	98.00 %
		Drebin	Sarah et al.	2019	98.31 %	98.00 %	98.00 %
		Kaggle (PE-dataset)	Azmee et al.	2020	96.30 %	98.00 %	97.00 %
		CICandMal2017	Liu et al.	2022	89.00 %	91.00 %	90.00 %
		Custom	Li et al.	2022	90.08 %	89.96 %	90.37 %
	Phishing	Custom	Meenu et al.	2019	88.60 %	90.40 %	93.10 %
		Kaggle (Phishing-dataset)	Alam et al.	2020	-	-	-
		Custom	Shahrivari et al.	2020	95.21 %	94.65 %	96.88 %
		Custom	Siti et al.	2020	100 %	100 %	-
	Ransomware	Custom	Masun et al.	2022	-	-	-

3.2.3 Naïve Bayes in detecting cyberattacks

The Naïve Bayes classifier utilized by Alsirhani et al. (2019) and Khashab et al. (2021) provided significantly poorer performance in detecting DDoS, Malware, and Ransomware cyberattacks compared to other classifiers reviewed in this thesis. In Alsirhani et al. (2019) research, Naïve Bayes provided only 62.20 % accuracy, 62.25 % precision, and 62.22 % recall. The Decision Tree provided 97.00 % accuracy, 97.00 % precision, and 97.00 % recall, and the Random Forest 97.20 % accuracy, 97.30 % precision, and 97.20 % recall, which were quite identical. In Khashab et al. (2021) research the Naïve Bayes classifier's accuracy in detecting DDoS attacks was 99.64 %, precision 99.97 %, and recall 98.98 %, which slightly outperforms the Decision Tree classifier, but fractionally loses to Random Forest classifier in performance reaching up to 99.76 % accuracy, 99.97 % precision, and 99.29 % recall. Hence, the Random Forest classifier was the best suitable among other classifiers in the research.

Ashrafuzzaman et al. (2021) utilized Naïve Bayes in detecting FDIA attacks. Surprisingly, Naïve Bayes provided almost identical performance results (89.31 % accuracy, 99.91 % precision, 73.04 % recall) compared to other classifiers, such as the Decision Tree, Linear Regression, Neural Network, and Support Vector Machine, examined in the research. This may be due to the usage of the imbalanced dataset as the authors stated. Qu et al. (2021) used a Random Forest classifier to detect FDIA attacks, and the classifier provided 89.60 % accuracy and 87.20 % precision by using different (ICS) datasets. In the same research, the Support Vector Machine classifier provided only 51.30 % accuracy, and 52.40 % precision, and the Decision Tree solely 82.70 % accuracy, and 81.50 % precision. Hence, the Random Forest classifier appears to be the best fit to detect FDIA attacks, especially, when considering the imbalanced dataset used in Ashrafuzzaman et al. (2021).

The Naïve Bayes classifier provided satisfactory results in detecting malware attacks in Santos et al. (2013) research with 89.81 % accuracy using the malware dataset, and Shhadat et al. (2017) using the custom dataset with 91.00 % accuracy, 66.80 % precision, and 89.60 % recall, but in other research (table 1), the Naïve Bayes provided only less than 80 % performance results, which was the lowest result compared to other classifiers reviewed. The Decision Tree, Neural Network, Random Forest, and Support Vector Machine were able to detect malware attacks with accuracy, precision, and recall higher than 90 % in most cases. The Random Forest classifier reached significantly high results with accuracy between 97-100 %, precision between 96-100 %, and recall between 87-100 % in most of the research papers examined in this thesis.

Siti et al. (2020) examined the Naïve Bayes classifier, which provided 95.66 % accuracy, and 96.50 % precision performance, by using the custom dataset gathered by the authors. The Support Vector Machine and Random Forest classifiers provided 100 % accuracy, and 100 % precision, and the Decision Tree 98.05 % accuracy, and 98.10 % precision. The Decision Tree classifier and Random Forest seem to outperform the Naïve Bayes with accuracy, and precision results between 98-100 %. Generally, the Decision Tree, Neural Network, and Random Forest provided the best performance results in detecting phishing attacks among the classifiers reviewed.

Table 3. Naïve Bayes in detecting cyberattacks

ML Technique	Domain	Dataset	Reference	Year	Results		
					Accuracy	Precision	Recall
Naïve Bayes	DDoS	Custom	Alsirhani et al.	2019	62.20 %	62.25 %	62.22 %
		CICDDoS2019	Manikumar et al.	2020	-	-	-
		Custom	Khashab et al.	2021	99.64 %	99.97 %	98.98 %
		CICIDS2017	Kareem et al.	2022	-	-	-
		Custom	Shaaban et al.	2022	-	-	-
	FDIA	Custom	Lu et al.	2020	-	-	-
		Matpower (Gen.)	Ashrafuzzaman et al.	2021	89.31 %	99.91 %	73.04 %
		ICS	Qu et al.	2021	-	-	-
	Malware	Malware dataset	Santos et al.	2013	89.81 %	-	-
		Custom	Shhadat et al.	2017	91.00 %	66.80 %	89.60 %
		Custom	Amer et al.	2019	70.01 %	70.00 %	100.00 %
		Drebin	Sarah et al.	2019	75.82 %	75.90 %	82.60 %
		Kaggle (PE-dataset)	Azmee et al.	2020	-	-	-
		CICandMal2017	Liu et al.	2022	-	-	-
		Custom	Li et al.	2022	79.27 %	81.90 %	78.47 %
	Phishing	Custom	Meenu et al.	2019	-	-	-
		Kaggle (Phishing-dataset)	Alam et al.	2020	-	-	-
		Custom	Shahrivari et al.	2020	-	-	-
		Custom	Siti et al.	2020	95.66 %	96.50 %	-
	Ransomware	Custom	Masum et al.	2022	35.00 %	31.00 %	99.00 %

3.2.4 Random Forest in detecting cyberattacks

The Random Forest classifier provided great results (accuracy > 95 %, precision >= 95 %, and recall > 97 %) in all the research experiments it was utilized, such as the research conducted by Alsirhani et al. (2029), Manikumar et al. (2020), Khashab et al. (2021), Kareem et al. (2022), and Shaaban et al. (2022). The Decision Tree was able to reach almost as good results as the Random Forest classifier, but in Kareem et al. (2022) research experiment the Random Forest (99.84 % accuracy, 99.80 % precision, 99.99 % recall) outperformed the Decision Tree classifier in accuracy and precision with a remarkable difference (81.58 % accuracy, 76 % precision, 100 % recall).

In the Qu et al. (2021) research in detecting FDIA attacks Random Forest classifier performed well providing 89.60 % accuracy, and 87.20 % precision compared to the Decision Tree (82.70 % accuracy, and 81.50 % precision), and Support Vector Machine (51.30 % accuracy, and 52.40 % precision). In the Ashrafuzzaman et al. (2021) research, performance results (90 % for both accuracy, and precision) are better, but it can be due to the Imbalanced dataset as stated before. To detect

the FDIA attacks, the Decision Tree and the Support Vector Machine were examined the most among the classifiers reviewed.

The Random Forest classifier and the Decision Tree classifiers provided almost the same performance results in the Shhadat et al. (2017), Amer et al. (2019), Sarah et al. (2019), Azmee et al. (2020), Liu et al. (2022), and Li et al. (2022) research experiments, despite of different datasets utilized. These classifiers were able to provide > 91% accuracy, and precision, and >87 % recall values. In general, the Malware detection performance provided by these classifiers was at a great level. The Support Vector Machine and Neural Network classifiers provided comparable results to these classifiers. In most research experiments examined in this thesis, at best, the Naïve Bayes classifier provided only satisfactory performance compared to more robust classifiers, such as the Decision Tree, and Random Forest.

In detecting phishing attacks, the Random Forest classifier provided the best performance in the Meenu et al. (2019), Alam et al. (2020), Shahrivari et al. (2020), and Siti et al. (2020) research experiments. The highest performance was measured in Siti et al. (2020) research, in which the Random Forest and the Support Vector classifiers provided 100 % accuracy, and 100 % precision performance results. The Decision Tree reached 98.05 % accuracy, and 98.10 % precision, and Naïve Bayes 95.66 % accuracy, and 96.50 % precision, which are both exquisite results. Neural Network classifier detected phishing attacks with 94.31 % accuracy, 94.30 % precision, and 94.40 % recall in Meenu et al. (2019) research, and with 96.98 % accuracy, 96.76 % precision, and 97.87 recall in Sharivari et al. (2020) research using custom datasets by authors. When utilizing the Kaggle phishing dataset, the Random Forest classifier provided 96.96 % accuracy, and 96.89 % precision, but only 42.16 % recall. The Decision Tree classifier did not perform as well, but it still provided good enough 91.94 % accuracy, 88.04 % precision, and 93.84 % recall performance.

Most of the classifiers examined in this thesis, such as the Decision Tree, Neural Network, and Random Forest were able to provide great performance results in detecting Ransomware attacks. In Masum et al. (2022) research, the Decision Tree provided 98.00 % accuracy, 98.00 % precision, and 94 % recall, Random Forest provided 99.00 % accuracy, 99.00 % precision, and 97.00 % recall and Neural Network provided 97.00 % accuracy, 97.00 % precision, and 95.00 % recall. The Naïve Bayes classifier was able to provide only 35.00 % accuracy, and 31.00 % precision, but 99.00 % recall performance results.

Table 4. Random Forest in detecting cyberattacks

ML Technique	Domain	Dataset	Reference	Year	Results		
					Accuracy	Precision	Recall
Random forest	DDoS	Custom	Alsirhani et al.	2019	97.20 %	97.30 %	97.20 %
		CICDDoS2019	Manikumar et al.	2020	95.19 %	95 %	-
		Custom	Khashab et al.	2021	99.76 %	99.97 %	99.29 %
		CICIDS2017	Kareem et al.	2022	99.84 %	99.80 %	99.99 %
		Custom	Shaaban et al.	2022	-	-	-
	FDIA	Custom	Lu et al.	2020	-	-	-
		Matpower (Gen.)	Ashrafuzzaman et al.	2021	-	-	-
		ICS	Qu et al.	2021	89.60 %	87.20 %	-
	Malware	Malware dataset	Santos et al.	2013	-	-	-
		Custom	Shhadat et al.	2017	97.80 %	96.40 %	87.30 %
		Custom	Amer et al.	2019	99.40 %	99.00 %	99.00 %
		Drebin	Sarah et al.	2019	99.84 %	100 %	100 %
		Kaggle (PE-dataset)	Azmee et al.	2020	98.10 %	99.00 %	98.00 %
		CICandMal2017	Liu et al.	2022	92.00 %	92.00 %	91.00 %
		Custom	Li et al.	2022	92.26 %	91.76 %	93.19 %
	Phishing	Custom	Meenu et al.	2019	-	-	-
		Kaggle (Phishing-dataset)	Alam et al.	2020	96.96 %	96.89 %	42.16 %
		Custom	Shahrivari et al.	2020	97.26 %	96.98 %	98.14 %
		Custom	Siti et al.	2020	100 %	100 %	-
	Ransomware	Custom	Masum et al.	2022	99.00 %	99.00 %	97.00 %

3.2.5 Neural network in detecting cyberattacks

The Neural Network classifier has been utilized in all the cyberattack domains reviewed in this thesis. Shaaban et al. (2022) examined various classifiers, such as the Decision Tree, Support Vector Machine, and Neural Network in detecting cyberattacks. Neural Network provided the best performance in detecting DDoS attacks with 98.03 % accuracy. The Support Vector Machine provided 96.36 % accuracy and the Decision tree 95.00 % accuracy, respectively, indicating the Neural Network classifier as the best-performing classifier in Shaaban et al. (2022) research.

In Ashrafuzzaman et al. (2021) research, which concerned detecting FDIA attacks, performance results for all the classifiers examined in this thesis provided quite similar results due to the imbalanced dataset. The Neural Network classifier was not an exception. In detecting Malware cyberattacks, the Neural Network classifier provided slightly better 98.00 % accuracy, 98.00 % precision, and 94.00 % recall in Azmee et al. (2020) research utilizing Kaggle (PE-dataset) compared to other

classifiers examined in this thesis, except the generally well-performing Random Forest classifier providing 98.10 % accuracy, 99.00 % precision, and 98.00 % recall.

In the Meenu et al. (2019) research in detecting phishing attacks, the Neural Network classifier provided 94.31 % accuracy, 94.30 % precision, and 94.40 % recall, and the Decision Tree provided 93.90 % accuracy, 93.30 % precision, and 93.36 % recall, Support Vector Machine 88.60 % accuracy, 90.40 % precision, and 93.10 % recall. Hence, the Neural Network classifier slightly provided the highest performance among these classifiers. In Shahrivari et al. (2020) research, the Random Forest classifier provided 97.26 % accuracy, 96.98 % precision, and 98.14 % recall outperforming other classifiers examined. The Neural Network classifier was able to provide almost as good a performance result as the Random Forest providing 96.98 % accuracy, 96.76 % precision, and 97.87 % recall. The Decision Tree classifier provided 96.60 % accuracy, 96.78 % precision, and 97.14 % recall, and Support Vector Machine 95.21 % accuracy, 94.65 % precision, and 96.88 % recall.

Masum et al. (2022) examined ransomware attacks and conducted research utilizing the Decision Tree, Naïve Bayes, Neural Network, and Random Forest classifiers. The Random Forest classifier provided the best performance among these classifiers with 99.0 % accuracy, 99 % precision, and 97 % recall. The Decision Tree was able to provide 98.00 % accuracy, 98.00 % precision, and 94.00 % recall, and the Neural Network 97.00 % accuracy, 97.00 % precision, and 95.00 % recall, respectively. Naïve Bayes reached only 35.00 % accuracy, and 31.00 % precision, but 99.00 % recall performance. The Random Forest classifier was able to provide the best performance compared to other classifiers reviewed, but the difference in comparison with the Decision Tree classifier is not ample. Due to the nature of Naïve Bayes, the classifier was not able to provide decent performance results in detecting phishing attacks either.

Table 5. Neural network in detecting cyberattacks

ML Technique	Domain	Dataset	Reference	Year	Results		
					Accuracy	Precision	Recall
Neural network	DDoS	Custom	Alsirhani et al.	2019	-	-	-
		CICDDoS2019	Manikumar et al.	2020	-	-	-
		Custom	Khashab et al.	2021	-	-	-
		CICIDS2017	Kareem et al.	2022	-	-	-
		Custom	Shaaban et al.	2022	98.03 %	-	-
	FDIA	Custom	Lu et al.	2020	-	-	-
		Matpower (Gen.)	Ashrafuzzaman et al.	2021	89.31 %	99.91 %	73.04 %
		ICS	Qu et al.	2021	-	-	-
	Malware	Malware dataset	Santos et al.	2013	-	-	-
		Custom	Shhadat et al.	2017	-	-	-
		Custom	Amer et al.	2019	-	-	-
		Drebin	Sarah et al.	2019	-	-	-
		Kaggle (PE-dataset)	Azmee et al.	2020	98.00 %	98.00 %	94.00 %
		CICandMal2017	Liu et al.	2022	-	-	-
		Custom	Li et al.	2022	-	-	-
	Phishing	Custom	Meenu et al.	2019	94.31 %	94.30 %	94.40 %
		Kaggle (Phishing-dataset)	Alam et al.	2020	-	-	-
		Custom	Shahrivari et al.	2020	96.98 %	96.76 %	97.87 %
		Custom	Siti et al.	2020	-	-	-
	Ransomware	Custom	Masum et al.	2022	97.00 %	97.00 %	95.00 %

4 Discussion and Conclusions

4.1 Summary

This thesis concentrated on researching the most common cyberattacks, such as DDoS, FDIA, Malware, Phishing, and Ransomware, which can be used against critical infrastructure facilities. The research clarified what kind of widely known machine learning classifiers exist that can be used as defensive mechanisms in detecting these incoming cyber-attacks, and with what accuracy. The research also introduced and explained these defensive mechanisms to provide additional detection and defense capability to improve the protection of critical infrastructure facilities in encountering incoming cyber threats. In addition, the research elucidated what is the most suitable machine learning classifier (method) that can be utilized in detecting DoS/DDoS, FDIA, Malware, and phishing attacks.

The literature review conducted in the theoretical framework chapter of the thesis presented the cybersecurity definition and concepts, explained the basics of artificial intelligence and machine learning, and discussed critical infrastructure, and trends. The literature review also presented cyber-physical systems and implementations, cyberattacks on critical infrastructure facilities, and countermeasures. The fundamental purpose of the theoretical framework was to function as a theoretical foundation to support the processing and analysis of the data chapter of this thesis and to provide a general understanding of the concepts, and previous research in the field of study.

The decision tree provided good results in detecting cyberattacks examined in this thesis. The DT was able to detect DDoS attacks with 81–99 % accuracy, and in most cases, the accuracy was higher than 90 %. The DT detection accuracy of FDIA attacks was between 81–89 % in the experiments, respectively. Malware attacks were detected by DT with 91–99 % accuracy, which is high and depends on the datasets used. Phishing attack detection was between 91–98 %. DT was able to detect ransomware attacks with 98 % accuracy. The support vector machine (SVM) detected DDoS with 94–96 %, FDIA attacks with 51–78 %, malware attacks with 89–98%, and phishing attacks with 88–100 % accuracy. Naïve Bayes (NB) had the lowest detection accuracy, and it detected DDoS with 62–99 %, FDIA attacks with 89 %, malware attacks with 70–91 %, phishing attacks with 95 %, and ransomware attacks with only 35 % accuracy. Random Forest (R)F) was able to detect DDoS attacks with 95–99 %, FDIA attacks with 89 %, malware attacks with 92–99 %, phishing attacks with 96–100 %, and ransomware attacks with 99 % accuracy. Neural network

(NN) detected DDoS attacks with 98 %, FDIA attacks with 89 %, malware attacks with 98 %, phishing attacks with 94–96 %, and ransomware attacks with 97 % accuracy.

The Decision Tree and Random Forest classifiers provided great performance results in experiments conducted by authors of the scientific research papers examined while pursuing this thesis. The Random Forest outperformed the Decision Tree and other classifiers presented and compared in the processing and analysis of the data chapter of this research, making it the best option of the classifiers examined to detect cyberattacks presented in this thesis. The Random Forest is widely used due to its combination of accuracy and explicability providing accurate and precise results, and in addition, preventing overfitting. The classifier is usable in all the domains (DDoS, FDIA, Malware, Phishing, and Ransomware) presented in this thesis to detect malicious cyberattacks. The Naïve Bayes classifier provided the lowest performance in most of the experiments, and therefore, it cannot be recommended as a defensive measure.

4.2 Criticism of the research

As stated before, the review part of this thesis is based solely on the comparison of research experiments already conducted by comparing common machine-learning classifiers in detecting various cyberattacks. The data gathered and used in comparison in the empirical part of this thesis has not been measured by the author of this thesis, and there are no custom-made research experiment arrangements, whose results could have been used in comparison with the results gathered from already published research articles. In addition, it would have been beneficial to use the same datasets (if possible, and publicly available) as used in the articles and research experiments already published to conduct a custom-made research experiment by using, for example, other well-known and powerful classifiers to measure the performance in detecting cyberattacks discussed in this thesis.

Due to limitations in technology resources and time, building the research test environment, teaching machine and deep learning models based on datasets under the domains concerned to detect incoming attacks, and conducting cyberattacks to test the accuracy, prediction, and recall of the models, weren't feasible under the circumstances. This kind of simulation is not feasible either by utilizing virtualized platforms located on the internet, such as TryHackme or HackTheBox, but requires a custom-built research test environment to conduct the experiments.

4.3 Further research suggestions

This thesis did not compare machine and deep learning classifiers in detecting cyberattacks presented in the thesis, but it focused only on comparing machine learning classifiers and presenting some deep-learning-based adversarial attacks and corresponding countermeasures, already showcased in the chapter by Vähäkainu et al. (2022). In this context, a comparison of deep learning and machine learning classifiers would have provided more profound information and added value. In addition, a more comprehensive comparison using ensemble machine learning and deep learning classifiers would provide beneficial information and raise an interesting question "Does a machine-learning, deep-learning (or a hybrid of those) combination of ensemble classifiers exist, which could provide higher accuracy, precision, and recall compared to an individual classifier?"

In the next research and in addition to the comparison of machine learning, deep learning, and ensemble classifiers, it would be interesting to conduct a practical research experiment, providing novel information on comparing these classifier techniques with a custom-made or already published dataset from public sources. It would be also interesting to find out, which one of these classifiers to be compared would be simple and cost-efficient enough to use fast and effective enough learners, and if it could provide self-learning capabilities to detect novel cyberattacks of the future without human intervention.

These days adversarial attacks may pose a significant risk to machine learning and deep learning classifiers (models). If these models are used to adjust and/or control electrical devices or systems especially, in critical infrastructure facilities, and experienced a successful adversarial attack, consequences may be unpredictable. Hence, it would be beneficial to create a virtualized test environment to simulate a control system, such as a heating system guided by a predictive machine or deep learning-based feedback system and conduct an adversarial attack campaign to obtain data on how the machine or deep learning classifier behaves for being under such attack. It would also be important to test the deep learning classifiers', ensemble classifiers', and machine learning classifiers' performance in detecting and predicting incoming adversarial cyberattacks on these critical infrastructure facilities.

Ensemble classifiers were not compared when conducting the processing and analysis of the data section of this thesis. Hence, this thesis did not answer the questions: "Do ensemble classifiers im-

prove accuracy, and if they do, is the improvement significant?", "What kind of combination of ensemble classifiers could the best improve the accuracy (if ever possible)?" or "Would an individual classifier perform better than an ensemble classifier or combination of ensemble classifiers in some cases?". As this thesis could not answer these questions, but it just focused on comparing the performance of individual classifiers. Hence, conducting further research to provide answers to these questions would possibly provide better means of detecting the cyberattacks presented in this thesis.

References

Aggarwal, S. & Kumaj, N. (2021). The Blockchain Technology for Secure and Smart Applications across Industry Verticals. *Advances in Computers, Elsevier*, 121, pp. 455–481. https://doi.org/10.1016/bs.adcom.2020.08.023

Alam, M., N., Sarma, D., Lima, F., F., Saha, I., Ulfath, R., & Hossain, S. (2020). *Phishing Attacks Detection Using Machine Learning Approach*. https://doi.org/10.1109/ICSSIT48917.2020.9214225

Alcaraz, C., & Zeadally, S. (2014). Critical infrastructure protection: Requirements and challenges for the 21st century. *International Journal of Critical Infrastructure Protection*, 8, pp. 53-66. http://dx.doi.org/10.1016/j.ijcip.2014.12.002

Aljofey, A., Jiang, Q., Qu, Q., Huang, M., & Niyigena, J-P. (2020). An Effective Phishing Detection Model Based on Character Level Convolutional Neural Network from URL. *Electronics*, 9(9), 1514. MDPI AG. https://doi.org/10.3390/electronics9091514

Alonso, M., Amaris, H., Alcala, D. & Florez, R., D., M. (2020). Smart Sensors for Smart Grid Readability. *Sensors*, 20(8), 2087. MDPI AG. https://doi.org/10.3390/s20082187

Allea. (2017). *The European Code of Conduct for Research Integrity*. Brandenburg Academy of Sciences and Humanities. https://www.allea.org/wp-content/uploads/2017/05/ALLEA-European-Code-of-Conduct-for-Research-Integrity-2017.pdf

Alsirhani, A., Sampalli, S., & Bodorik, P. (2019). DDoS Detection System: Using a Set of Classification Algorithms Controlled by Fuzzy Logic System in Apache Spark. *IEEE Transactions on Network and Service Management*, 16(1), 936-949. https://doi.org/10.1109/TNSM.2019.2929425

Amer, E. & Zelinka, I. (2019). An Ensemble-based Malware Detection Model Using Minimum Feature Set. *Mendel*, 25(2), 1-10. https://doi.org/10.13164/mendel.2019.2.001

APA. (2022). *The Apa Ethics Code*. American Psychological Association. https://apa.org/ethics

Ashrafuzzaman, M., Das, S., Chakhchoukh, Y., Duraibi, S., Shiva, S., Sheldon, F. T. (2021). Supervised Learning for Detecting Stealthy False Data Injection Attacks in the Smart Grid. In K. Daimi, H. R. Arabnia, L. Deligiannidis, MS. Hwang, & F. G. Tinetti (Eds.), *Advances in Security, Networks, and Internet of Things*. Springer. https://doi.org/10.1007/978-3-030-71017-0_21

Azmee, A., Choudhury, P., P., Alam, A., Dutta, O., & Hossain, M., I. (2020). Performance Analysis of Machine Learning Classifiers for Detecting PE Malware. *International Journal of Advanced Computer and Applications*, 11(1), 510-517. https://doi.org/10.14569/IJACSA.2020.0110163

Australian Government. Critical Infrastructure Centre. What infrastructure is critical? Retrieved 19.11.2022 from http://homeaffairs.gov.au/nat-security/files/cic-factsheet-what-is-critical-infrastructure-centre.pdf

Berry, M., D. (2022). *Ransomware attacks against healthcare organizations nearly doubled in 2021*. Thomson Reuters. Retrieved 25.11.2022 from https://www.thomsonreuters.com/en-us/posts/investigation-fraud-and-risk/ransomware-attacks-against-healthcare

Bhattacharyaa, S., Pal, C., K., & Pandey, P., K. (2017). Detecting Phishing Websites, a Heuristic Approach. *International Journal of Latest Engineering Research and Applications (IJLERA)*, 02(03), pp. 120–129.

Biggio, B., Corona, L., Maiorca, D., Nelson, B., Srndic, N., Laskov, P., Giacinto, G., & Roli, F. (2013). *Evasion Attacks Against Machine Learning at Test Time*. https://doi.org/10.48550/arXiv.1708.06131

Bukhari, S., A., H. (2011). *What is comparative study*. http://dx.doi.org/10.2139/ssrn.1962328

Buczkowski, A. (2017). *What's the Difference Between Artificial Intelligence, Machine Learning, and Deep Learning?* GEO. Retrieved from 22.11.2022 http://geoawesomeness.com/whats-difference-artificial-intelligence-machine-learning-deep-learning

Carlini, N. & Wagner, D. (2017). *Towards Evaluating the Robustness of Neural Networks*. https://doi.org/10.48550/arXiv.1608.04644

Catal, C., Giray, G., Tekinerdogan, B., Kumar, S. & Shukla, S. (2022). *Applications of deep learning for phishing detection: a systematic literature review*. Knowledge and Information Systems, 64, pp. 1457–1500. https://doi.org/10.1007/s10115-022-01672-x

Chakraborty, A., Alam, M., Dey, V., Chattopadhyay, A. & Mukhopadhyay, D. (2018). *Adversarial Attacks and Defences: A Survey*. https://doi.org/10.48550/arXiv.1810.00069

Chen, P-Y., Yang, S., McCann, J., A., Lin, J., & Yang, X. (2015). *Detection of False Data Injection Attacks in Smart-Grid Systems*. *IEEE Communications Magazine,* 53(2), pp. 206-213. https://doi.org/10.1109/MCOM.2015.7045410

CISA. (2009). Security Tip (ST04-001). *What is cybersecurity?* Retrieved 22.11.2022 from http://cisa.gov/uscert/ncas/tips/ST04-001

CISA. (2019). *A Guide to Critical Infrastructure Security and Resilience*. Retrieved 22.11.2022 from http://cisa.gov/sites/default/files/publications/Guide-Critical-Infrastructure-Security-Resilience-110819-508v2.pdf

CISA. (2021). *ICS Alert (IR-ALERT-H-16-056-01). Cyber-Attack Against Ukrainian Critical Infrastructure*. Retrieved 10.2.2023 from https://cisa.gov/uscert/ics/alerts/IR-ALERT-H-16-056-01

Cisco. (2022). *What is Malware?* Retrieved 24.11.2022 from https://www.cisco.com/c/en/us/products/security/advanced-malware-protection/what-is-malware.html

Co, K., T. (2018*). Bayesian Optimization for Black-Box Evasion of Machine Learning Systems*. Imperial College London, Department of Computing. Retrieved 4.1.2023 from https://www.imperial.ac.uk/media/imperial-college/faculty-of-engineering/computing/public/1617-pg-projects/CoK-Bayesian-Optimization-for-Black-Box-Evasion-of-Machine-Learning-Systems.pdf

Communication from the Commission on Critical Infrastructure Protection in the fight against terrorism. (2004). COM(2004) 702 final.

Cyber Security Review. (2017). *Dragonfly 2.0: Hacking Group Infiltrated European and US Power Facilities*. Retriever 24.11.2022 from https://www.cybersecurity-review.com/news-september-2017/dragonfly-2-0-hacking-group-infiltrated-european-and-us-power-facilities

Dossett, J. (2021). *A timeline of the biggest ransomware attacks*. CNET. Retrieved 24.11.2022 from https://www.cnet.com/personal-finance/crypto/a-timeline-of-the-biggest-ransomware-attacks

DNI. *Counterintelligence tips – Spear Phishing and Common Cyber Attacks*. Retrieved 13.12.2022 from https://dni.gov/files/NCSC/documents/campaign/Counterintelligence_Tips_Spearphishing.pdf

Elmrabet, Z., Kaabouch, N., Elghazi, H., Elghazi, H. (2018). Cyber-security in Smart Grid: Survey and Challenges. *Computers & Electrical Engineering*, 67, pp. 469-582.

Enisa. (2022). *Phishing/Spear phishing*. Retrieved 13.12.2022 from https://enisa.europa.eu/topics/incident-response/glossary/phishing-spear-phishing

Erma Pte Ltd. (2022). *Cyber Risk – Trends and Critical Infrastructure.* Enterprise Risk Management Academy - A Global Learning Centre for Risk Professionals. Retrieved 23.11.2022 from http://www2.erm-academy.org/publication/risk-management-article/cyber-risk-trends-and-critical-infrastructure

Ethical Principles for JAMK University of Applied Sciences. (2018). JAMK University of Applied Sciences. https://jamk.fi/en/media/34826

Farmanbar, M., Parham, K., Arild, Ø., & Rong, C. (2019). A Widespread Review of Smart Grids Towards Smart Cities. *Energies*, *12*(23), 4484. MDPI AG. http://dx.doi.org/10.3390/en12234484

Flores, C., Guasco, T., Leon-Acurio, J. (2017). A Diagnosis of Threat Vulnerability and Risk as IT Related to the Use of Social Media Sites When Utilized by Adolescent Students Enrolled at the Urban Center of Canton Canar. *International Conference on Technology Trends*, 199–214.

Forte, V., J. (2010). Smart Grid at National Grid. *2010 Innovative Smart Grid Technologies (ISGT)*, pp. 1–4. https://doi.org/10.1109/ISGT.2010.5434729

Ghaben, A., Anbar, M., Hasbullah, I., H., & Karuppayah, S. (2021). Mathematical Approach as Qualitative Metrics of Distributed Denial of Service Attack Detection Mechanisms. *IEEE Access*, 9, pp. 123012-123028. https://doi.org/10.1109/ACCESS.2021.3110586

Goodfellow, I., & Papernot, N. (2017). *Is attacking machine learning easier than defending it?* Cleverhand-blog. Retrieved 5.1.2023 from http://cleverhans.io/security/privacy/ml/2017/02/15/why-attacking-machine-learning-is-easier-than-defending-it.html

Gosh, I. (2019). *This is the crippling cost of cybercrime on corporations*. World Economic Forum. Retrieved 1.2.2023 from https://www.weforum.org/agenda/2019/11/cost-cybercrime-cybersecurity

Griffor, E., Greer, C., Wollman, D., Burns, M. (2017) *Framework for Cyber-Physical Systems: Overview*, 1. Special Publication (NIST SP) – 1500–201. https://doi.org/10.6028/NIST.SP.1500-201

Guo, Q., Hiskens, I., A., Jin, D., K., Su, W., & Zhang, L. (2017). Cyber-Physical Systems in Smart Grids: Security and Operation. *IET: Cyber-Physical Systems: Theory & Application*: 2(4), pp. 153-154. https://doi.org/10.1049/iet-cps.*2017*.0133

Hantrais, L. (1995). Social research update*. Comparative Research Methods, 13. Department of Sociology*, University of Surrey. https://sru.soc.surrey.ac.uk/SRU13.html

HC3. (2022). *HC3: Analyst Note*. Office of Information Security One HHS, Health Sector Cybersecurity Coordination Center. Retrieved 25.11.2022 from https://www.hhs.gov/sites/default/files/venus-ransomware-analyst-note.pdf

He, Z., Zhang, T., Lee, R., B. (2017). Machine Learning Based DDoS Attack Detection from Source Side in Cloud. *4th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)*, pp. 114-120. http://doi.org/10.1109/CSCloud.2017.58

Hirsjärvi, S., Remes, P. & Sajavaara, P. (2004). *Tutki ja kirjoita*. 10th ed. Tammi.

HVAC. (2023). *The Britannica Dictionary Definition of HVAC*. Retrieved 22.1.2023 from https://www.britannica.com/dictionary/HVAC

Ibitoye, O, Shafig, O, & Matrawy, A. (2019). *Analysing Adversarial Attacks Against Deep Learning for Intrusion Detection in IoT Networks*. https://doi.org/10.48550/arXiv.1905.05137

Ibsrekken, A., T. (2022). *Comparative methods*. NUPI, Norks Utenrikspolitisk Institut, Norwegian Institute of International Affairs, Oslo, Norway. Retrieved 18.11.2022 from https://www.nupi.no/en/our-research/topics/theory-and-method/comparative-methods?page=2

Imperva. (2022). *Phishing Attacks*. Retrieved 7.12.2022 from https://www.imperva.com/learn/application-security/phishing-attack-scam

Jain, A., K., & Gupta, B., B. (2018). PHISH-SAFE: URL Features-based Phishing Detection System Using Machine Learning", Cyber Security. *Advances in Intelligent systems and Computing*, 729. https://doi.org/10.1007/978-981-10-8536-9_44

JavaTpoint. (2021). *Naïve Bayes Classifier Algorithm*. Retrieved 21.11.2022 from http://javatpoint.com/machine-learning-naive-bayes-classifier

Jazdi, N. (2014). Cyber Physical Systems in the Context of Industry 4.0. *IEEE International Conference on Automation*, Quality, and Testing, Robotics, pp. 1–4. https://10.1109/AQTR.2014.6857843

Jiao, J. (2020). Application and prospect of artificial intelligence in smart grid. *IOP Conference Series: Earth and Environmental Science*. https://doi.org/10.1088/1755-1315/510/2/022012

Joshi, R. (2016). *Accuracy, Precision, Recall & F1 Score: Interpretation of Performance Measures. Exilio Solutions*. Retrieved 18.11.2022 from http://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures

Juutilainen, J. (2022*). Cyber Warfare: A Part of the Russo-Ukrainian War in 2022*. [Master's thesis, JAMK University of Applied Sciences]. https://www.theseus.fi/handle/10024/780757

Kareem, M. A., & Jasim, M. N. (2022). DDOS Attack Detection Using Lightweight Partial Decision Tree algorithm. *International Conference on Computer Science and Software Engineering (CSASE)*, 362–367, https://doi.org/10.1109/CSASE51777.2022.9759824

Khashab., F., Moubarak, J., Feghali, A., & Bassil, C. (2021). DDoS Attack Detection and Mitigation in SDN using Machine Learning. *IEEE 7th International Conference on Network Softwarization (NetSoft)*, 395–401. https://doi.org/10.1109/NetSoft51509.2021.9492558

Kumar, D., Rao, C., V. G., Singh, M., K., & Satyanarayana, G., C. (2013). A Survey on Defence Mechanisms countering DDoS Attacks in the Network. *International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE)*, 2(7), 2599–2606.

Lahcen, R., A., M. & Mohapatra, R. (2022) Challenges in CyberSecurity and Machine Learning. *Panamerican Mathematical Journal*, 32(1), pp. 14–33.

Lee, E., A. (2015). The Past, Present and Future of Cyber-Physical Systems: A Focus on Models. *Sensors*, 15(3), pp. 4837–3869. https://doi.org/10.3390/s150304837

Lewis, T. (2006). *Critical Infrastructure Protection in Homeland Security*. John Wiley & Sons, p. 474.

Li, Y., Wei, X., Li, Y., Dong, Z., & Shahidehpour, M. (2022). Detection of False Injection Attacks in Smart Grid: A Secure Federated Deep Learning Approach. *IEEE Transactions on Smart Grid*, 13(6), pp. 4862–4872. https://doi.org/10.1109/TSG.2022.3204796

Li, S., Zhou, Q., Zhou, R., & Lv Q. (2022). Intelligent malware detection based on graph convolutional network. *The Journal of Supercomputing*, 78, 4182–4198. https://doi.org/10.1007/s11227-021-04020-y

Liu, T., Zhang, H., Long, H., Shi, J., & Yao, Y. (2022) Convolution neural network with batch normalization and inception-residual modules for Android malware classification. *Scientific reports* 12(13996), 1–17. https://doi.org/10.1038/s41598-022-18402-6

Lor, P. (2019*). International and Comparative Librarianship*. De Gruyter Saur.

Lu, X., Jing, J., & Wu, Y. (2020). False Data Injection Attack Location Detection Based on Classification Method in Smart Grid. *2nd International Conference on Artificial Intelligence and Advanced Manufacture (AIAM)*, 133–136. https://doi.org/10.1109/AIAM50918.2020.00033

Majed, H., Noura, H., Salman, O., Malli, M., & Chehab, A. (2020). Efficient and Secure Statistical DDoS Detection Scheme. *17th International Joint Conference on e-Business and Telecommunications (ICETE 2020)*, pp 153–161. https://doi.org/10.5220/0009873801530161

Manikumar, D., V., V., S., & Maheswari, B., U. (2020). Blockchain Based DDoS Mitigation Using Machine Learning Techniques. *Second International Conference on Inventive Research in Computing Applications (ICIRCA)*, 795–800. https://doi.org/10.1109/ICIRCA48905.2020.9183092

Masum, M., Faruk, J., H., Shahriar, H, Qian, K., Lo, D., & Adnan M., I. (2022). Ransomware Classification and Detection with Machine Learning Algorithms. *IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*, 0316–0322. http://doi.org/10.1109/CCWC54503.2022.9720869

Meenu, & Godara, S. (2019). Phishing Detection Using Machine Learning Techniques. *International Journal of Engineering and Advanced Technology (IJEAT)*, 9(2), 3820–3829. https://doi.org/10.35940/ijeat.B4095.129219

Microsoft Digital Defence Report. (2022). Retrieved 19.11.2022 from https://query.prod.cms.rt.microsoft.com/cms/api/am/binary/RE5bcRe?culture=en-us&country=us

MSTIC. (2022*). ACTINIUM targets Ukrainian organizations*. Retrieved 14.12.2022 from https://www.microsoft.com/en-us/security/blog/2022/02/04/actinium-targets-ukrainian-organizations

Mutsuo, N. & Hirofumi, U. (2017) An Analysis of the Actual Status of Recent Cyberattacks on Critical Infrastructure. *NEC Technical Journal*, 17(2). http://nec.com/en/global/techrep/journal/g17/n02/170204.html

NIST. (n.d.). *Asset*. Information Technology Laboratory, Computer Security Resource Center. Retrieved 19.11.2022 from http://csrc.nist.gov/glossary/term/asset

Noble, W., S. (2006). What is a support vector machine? *Nature biotechnology*, 24, 1565–1567. https://doi.org/10.1038/nbt1206-1565

Nozomi Networks. (2017). *The Cost of OT Cyber Security Incidents*. Retrieved 17.11.2022 from http://nozominetworks.com/solutions/topic/cost-of-ot-cyber-security-incidents

Palo Alto Networks. (2022). *Malware | What is Malware & How to Stay Protected from Malware Attacks*. Retrieved 24.11.2022 from https://www.paloaltonetworks.com/cyberpedia/what-is-malware

Papernot, N., McDaniel, P., Wu, X., Jha, S., & Swami, A. (2015). *Distillation as a Defense to Adversarial Perturbations against Deep Neural Networks*. https://doi.org/10.48550/arXiv.1511.04508

Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celic, Z., B., & Swami, A. (2016) *Practical Black-Box Attacks against Machine Learning*. https://doi.org/10.48550/arXiv.1602.02697

Picton, P. (1994). *What is Neural Network?* In introduction to Neural Networks, pp. 1–12. Palgrave. https://doi.org/10.1007/978-1-349-13530-1_1

Poudel, B. (2020). *Explaining the Carlini & Wagner Attack Algorithm to Generate Adversarial Examples*. Retrieved 4.1.2023 from https://medium.com/@iambibek/explanation-of-the-carlini-wagner-c-w-attack-algorithm-to-generate-adversarial-examples-6c1db8669fa2

Pranpaveen, L. & Phaisangittisagul, E. (2023). Collaborative Defense-GAN for protecting adversarial attacks on classification system, *Expert Systems with Applications*, 2014. https://doi.org/10.1016/j.eswa.2022.118957

Qu, Z., Dong, Y., Qu, N., Li, H., Cui, M., Bo, X., Wu, Y., & Mugemanyi, S. (2021). False Data Injection Attack Detection in Power Systems Based on Cyber-Physical Attack Genes. *Frontiers in Energy Research*. https://doi.org/10.3389/fenrg.2021.644489

Qureshi, A., S. (2018). *How to Mitigate DDoS Vulnerabilities in Layers of OSI Model*. Retrieved 24.11.2022 from http://dzone.com/articles/how-to-mitigate-ddos-vulnerabilities-in-layers-of

Rao, R., S., & Ali, S., T. (2015). PhishShield: A Desktop Application to Detect Phishing Webpages through Heurestic Approach. *Eleventh International Multi-Conference on Information Processing (IMCIP-2015)*, 54, pp. 147–156. Https://doi.org/10.1016/j.procs.2015.06.017

Rao, R., S., Vaishnavi, T., & Pais, A., R. (2019). CatchPhish: Detection of phishing websites inspecting URLs. *Journal of Ambient Intelligence and Humanized Computing*, 11, pp. 813-825. https://doi.org/10.1007/s12652-019-01311-4

Reda, H., T., Anwar, A., Mahmood, A., N., & Tari, Z. (2021). *A Taxonomy of Cyber Defence Strategies Against False Data Attacks in Smart Grid*. https://doi.org/10.48550/arXiv.2103.16085

Rehak, D., Senovsky, R, Hromada, M, & Lovecek, T. (2019). Complex approach to assessing resilience of critical infrastructure elements. *International Journal of Critical Infrastructure Protection*, 25, pp. 125–138. https://doi.org/10.1016/j.ijcip.2019.03.003

Ren, K., Zheng, T., Qin, Z., & Liu, X. (2020). Adversarial Attacks and Defences in Deep Learning. *Engineering*, 6(3), pp. 346–360. https://doi.org/10.1016/j.eng.2019.12.012

Ripa, S., P., Islam, F., & Arifuzzaman, M. (2021). The Emergence of Phishing Attacks and The Detection Techniques Using Machine Learning Models. 2021 *International Conference on Automation, Control and Mechatronics for Industry 4.0 (ACMI)*, pp. 1–6. https://doi.org/10.1109/ACMI53878.2021.9528204

Rosato, V., Tofani, A., Di Pietro, A., Pollino, M., Giovinazzi, S., Lavalle, L., & D'Agostino, G. (2020). The European Infrastructure Simulation and Analysis Centre (EISAC) initiative and its technological assets. *43rd International Convention on Information, Communication and Electronic Technology (MIPRO)*. https://doi.org/10.23919/MIPRO48935.2020.9245340

Routio, P. (2007). *Comparative study*. Retrieved 15.11.2022 from http://www2.uiah.fi/projects/metodi/172.htm

Sagduyu, Y., Shi, E., Wang., L., & Hopcroft, J., E. (2019). Improving the Generalization of Adversarial Training with Domain Adaptation. *International Conference on Learning Representations*. https://doi.org/10.48550/arXiv.1906.00076

Samangouei, P., Kabkab, M., & Chellappa, R. (2018). Defense-GAN: Protecting Classifiers against Adversarial Attacks using Generative Models. *Sixth International Conference on Learning Representations (ICLR 2018)*. https://doi.org/10.48550/arXiv.1805.06605

Santos, I., Brezo, F., Ugarte-Pedrero, X., & Bringas, P., G. (2013). Opcode sequences as representation of executables for data-mining-based unknown malware detection. *Journal of Information Sciences*, 231, 64–82. https://doi.org/10.1016/j.ins.2011.08.020

Sarah, N., A., Rifat, F., Y., Hossain, S., & Narman, H., S. (2021). An Efficient Android Malware Prediction Using Ensemble machine learning algorithms. *Procedia Computer Science*, 191, 184–191. https://doi.org/10.1016/j.procs.2021.07.023

SCS. (2020). Simplifying the Difference: machine learning vs deep learning. Singapore Computer Society. Retrieved 21.11.2022 from http://scs.org.sg/articles/machine-learning-vs-deep-learning

Shaaban, A. R., Abd-Elwanis, E., & Hussein, M. (2019). DDoS attack detection and classification via Convolutional Neural Network (CNN). *Ninth International Conference on Intelligent Computing and Information Systems (ICICIS)*, 233–238, https://doi.org/10.1109/ICICIS46948.2019.9014826

Shahrivari, V., Darabi, M., M., & Izadi, M. (2020). *Phishing Detection Using Machine Learning Techniques.* https://doi.org/10.48550/arXiv.2009.11116

Shhadat, I., Bataineh, B., Hayajneh, A., Al-Sharif, Z., A. (2020). The Use of Machine Learning Techniques to Advance the Detection and Classification of Unknown Malware. *Procedia Computer Science*, 170, 917–922. https://doi.org/10.1016/j.procs.2020.03.110

Short, A., Pay, T., L., & Gandhi, A. (2019). *Defending Against Adversarial Examples*. Sandia Report, SAND 2019-11748. Sandia National Laboratories. Retrieved 4.1.2023 from https://www.osti.gov/servlets/purl/1569514

Siti, N., W., A., Ismail, M., A., Sutoyo, Edi, S., Shahreen, K., & Mohd, S., M. (2020). Comparative Performance of Machine Learning Methods for Classification on Phishing Attack Detection. *International Journal of Advanced Trends in Computer Science and Engineering*, 9(1.5), 349–454. http://doi.org/10.30534/ijatcse/2020/4991.52020

Sprengers, M., & van Haaster, J. (2016). Organization: Evading reputation-based malware detection. In J. van Haaster, R. Gevers, M. Sprengers (Eds.), *Cyber Guerilla*. Syngress, pp. 41–109. https://doi.org/10.1016/B978-0-12-805197-9.00003-6

Stamatescu, G., Stamatescu, I., Arghira, N. Calofir, V., & Fagarasan, I. (2016). *Building Cyber-Physical Energy Systems*. https://doi.org/10.48550/arXiv.1605.06903

Sun, C-C., Liu, C-C., & Xie, J. (2020). Cyber-Physical System Security of a Power Grid: State-of-the-ARt. *Electronics*, 5(3), 40. MDPI AG. https://doi.org/10.3390/electronics5030040

Tiwari, D., D., Naska, S., Sai, A., S., & Palleti, V., R. (2021). Attack Detection Using Unsupervised Learning Algorithms in Cyber-Physical Systems. In M. Turkay & R. Gani (Eds.), *Computer Aided Chemical Engineering*, pp. 1259–1264. Elsevier. https://doi.org/10.1016/B978-0-323-88506-5.50194-7

Tung, L. (2022). *Microsoft: These hackers are targeting emergency response and security organizations in Ukraine*. ZDNET. Retrieved 14.12.2022 from https://www.zdnet.com/article/microsoft-these-hackers-are-targeting-emergency-response-and-security-organizations-in-ukraine

Varantola, K., Launis, V., Helin, M., Spoof, S., K., & Jäppinen, S. (2013*). Responsible conduct of research and procedures for handling allegations of misconduct in Finland*. Guidelines of the Finnish Advisory Board on Research Integrity 2012. Retrieved 2.3.2023 from https://tenk.fi/sites/tenk.fi/files/HTK_ohje_2012.pdf

Vähäkainu, P., Lehto, M., & Kariluoto, A. (2022). Cyberattacks against critical infrastructure facilities and corresponding countermeasures. In M. Lehto, P. Neittaanmäki (Eds.), *Cyber Security*. Computational Methods in Applied Sciences, 56. Springer. https://doi.org/10.1007/978-3-030-91293-2_11

Wang, Y., Chen, D., Zhang, C., Chen, X., Huang, B., & Cheng, X. (2019). Wide and Recurrent Neural Networks for Detection of False Data Injection in Smart Grids. In E. Biagioni, Y. Zheng & S. Cheng (Eds.), *Wireless Algorithms, Systems, and Applications*. Lecture Notes in Computer Science, 11604. Springer. https://doi.org/10.1007/978-3-030-23597-0_27

World Economic Forum. (2020*). The Global Risks Report 2020, Insight Report 15th edition*. Retrieved 1.2.2023 from https://tinyurl.com/mu6fjad9

Wiyatno, R., & Xu, A. (2018). *Maximal Jacobian-based Saliency Map Attack*. https://doi.org/10.48550/arXig.1808.07945

Wiyatno, R., R., Xu, A., Dia, O., & De Berker, A. (2019). *Adversarial Examples in Modern Machine Learning: A Review*. https://doi.org/10.48550/arXiv.1911.05628

Xiao, F., Lin, Z., Sun, Y., Ma, Y. (2019) Malware Detection Based on Deep Learning of Behavior Graphs. *Mathematical Problems in Engineering*, 1, pp. 1–10. https://doi.org/10.1155/2019/8195395

Zhang, W., Ding, Y-X., Tang, Y., & Zhao, B. (2011). Malicious Web Page Detection Based on On-line Learning Algorithm. *2011 International Conference on Machine Learning and Cybernetics*, Guilin, pp. 1914–1919. https://doi.org/10.1109/ICMLC.2011.6016954

Zheng, Y., Li, Z., Xu, X., & Zhao, Q. (2022). Dynamic defenses in cyber security: Techniques, methods, and challenges. *Digital Communications and Networks*, 8(4), pp. 422–435. https://doi.org/10.1016/j.dcan.2021.07.006

Uddin, S., Khan, A., Hossain, M., E., & Moni, M., A. (2019). Comparing different supervised machine learning algorithms for disease prediction. *BMC Medical Informatics and Decision Making*, 19(281). https://doi.org/10.1186/s12911-019-1004-8

Appendices

Appendix 1. Cyberattacks against critical infrastructure facilities and corresponding countermeasures

Vähäkainu, P., Lehto, M., & Kariluoto, A. (2022). Cyberattacks against critical infrastructure facilities and corresponding countermeasures. In M. Lehto, P. Neittaanmäki (Eds.), *Cyber Security*. Computational Methods in Applied Sciences, 56. Springer. https://doi.org/10.1007/978-3-030-91293-2_11

Cyberattacks against critical infrastructure facilities and corresponding countermeasures

Petri Vähäkainu, Martti Lehto, Antti Kariluoto
Faculty of Information Technology, University of Jyväskylä, Jyväskylä, Finland
petri.vahakainu@jyu.fi
martti.lehto@jyu.fi
antti.j.e.kariluoto@jyu.fi

Abstract Critical infrastructure (CI) is a vital asset for the economy and society's functioning, covering sectors such as energy, finance, healthcare, transport, and water supply. Governments around the world invest a lot of effort in continuous operation, maintenance, performance, protection, reliability, and safety of CI. However, the vulnerability of CIs against cyberattacks and technical failures has become a significant concern nowadays. Sophisticated and novel cyberattacks, such as adversarial attacks, may deceive physical security controls, for example, smart locks, providing a perpetrator an illicit entry to the smart critical facility (e.g., smart building). The same adversarial attacks may be used to deceive predictive machine learning (ML) based classifier, which automatically adjusts the heating, ventilation, and air conditioning (HVAC) of a smart building. Additionally, false data injection attacks have been used against smart grids. Traditional and widely used cyberattacks utilizing malicious code, such as DoS/DDoS, malware, phishing, and ransomware, are able to cause remarkable physical damage, such as blackouts and disruptions to energy production or the entire city's water supply when used as attack vectors to manipulate critical infrastructure controls of defense, energy providers, financial services, healthcare databases, power grids, etc. In order to detect incoming attacks and mitigate the performance of these attacks, we introduce defensive mechanisms to provide auxiliary detection and defense capability to enhance the insufficient protections of the smart critical facility against outsider threats.

Keywords adversarial attacks · critical infrastructure · cyberattacks · cyber-physical system · defensive mechanisms

1 Introduction

Cyber-physical systems (CPS) are sociotechnical systems seamlessly integrating analog, digital, physical, and human components engineered for function through integrated physics and logic (Griffor et al., 2017). Cyber-physical systems can be considered as integrations of computation, networking, and physical processes. CPSs can be implemented as feedback systems that are adaptive and predictive, intelligent, real-time, networked, or distributed, possibly with wireless sensing and actuation. In CPSs, physical processes are controlled and monitored by embedded computers and networks with feedback loops where physical processes influence computations and contrarily. These kinds of systems provide the foundation of critical infrastructures (CI), providing means to develop and implement smart services of the future, and improving quality of life in various areas. Cyber-physical systems interact directly with the physical world; thus, they are able to provide advantages to our daily lives in the form of automatic warehouses, emergency response, energy networks, factories, personalized health care, planes, smart buildings, traffic flow management, etc.

Critical infrastructure refers to infrastructure that is vital in providing community and individual functions. It can include buildings, e.g., airports, hospitals, power plants, schools, town halls, and physical facilities as roads, storm drains, potable water pipes, or sewer systems. (Planning for Hazards) CI can be considered a subset of the cyber-physical system, which includes smart buildings (Miller, 2014). Smart buildings utilize technology aiming to create a safe and healthy environment for its occupants. Smart building technology, which is still in the early stages of growth and adoption, increases moderately and is becoming a significant business around the world.

Cyber threats against critical infrastructures raise concerns these days, and cyber-physical systems must operate under the same assumption that they might become a target. For example, in the case of an adversarial attack, a perpetrator could fool the Machine Learning (ML) model and gain entry to a building causing significant security threats. The perpetrator may also use the predictive deep learning neural network (DNN) used to adjust the HVAC system by conducting adversarial attacks to cause challenging situations in the form

of energy consumption spikes causing high costs. The impact is not negligible as the cost of power spikes has a long payback time; in some cases, several years. Defensive countermeasures against these kinds of attacks are not always straightforward, but adversarial training, defensive distillation, or defense-GAN methods can be utilized in certain cases.

DoS/DDoS, Malware, and Phishing, are traditional attacks capable of causing a considerable threat to critical infrastructure sectors, such as energy and transportation. Perpetrators have utilized DDoS attacks in disrupting the heating distribution system by incapacitating the controlling computers used to heat buildings. This type of attack has also been used when attacking transportation services to cause delays and disruptions over travel services, such as communications, internet services, ticket sales, etc. (Metropolitan, 2016). Perpetrators may also conduct False Data Injection Attacks (FDIA) to cause a significant threat to, for example, smart grids (SG). They may disrupt energy and supply figures to cause false energy distribution resulting in additional costs (Chen et al., 2015) often with destructive consequences, or they may conduct the attack towards the smart meter of the power grid to lower one's own electricity bills (Elmbrabet et al., 2018). If the perpetrator initiates an attack against the power-line connections of the power grid, he or she may be able to separate nodes from the power grid to fool the energy distribution system, which may result in power defects or increased energy transmission costs. In order to provide efficient countermeasures against FDIA attacks, detection methods, such as blockchain, cryptography, and learning-based methods, can be considered.

In the past years, the utilization of malicious software (malware) when conducting attacks towards critical infrastructures have increased. In 2012, Shamoon malware was used to attack the Saudi Arabian national petroleum company, Aramco, by wiping hard disk drives (Alelyan & Kumar, 2018). In 2016, BlackEnergy malware was used to cause disruptions to the Ukrainian electrical grid (Santos, 2016). Petya malware infected websites of Ukrainian organizations, banks, ministries, newspapers, and electrical utilities (OSAC, 2018). Phishing attacks bring in a human component in which a perpetrator exploits human error and manipulates user behavior, for example, to obtain access to a target system. These kinds of attacks could be detected with deep learning (DL) methods.

In this chapter, the authors briefly introduced the concepts of critical infrastructure, cyber-physical systems, and topical attack vectors against critical infrastructure and countermeasures, respectively. In chapter 2, the authors explain critical infrastructure and resilience concepts in more detail. Chapter 3 addresses the cyber-physical system and presents some relevant CPS sectors these days. Chapter 4 defines cybersecurity and explains the intertwined concepts of cybersecurity, threat, vulnerability, and risks in more detail. Chapter 5 describes artificial intelligence and machine learning and discusses the most common and sophisticated deep learning methods. In chapter 6, we showcase well-known cyberattacks utilized against critical infrastructure facilities, such as smart buildings. Chapter 7 focuses on reviewing the defense mechanisms utilized in combating cyberattacks towards critical infrastructure facilities. Lastly, chapter 8 concludes the study.

2 Critical Infrastructure and resilience

Critical infrastructure (CI) is the body of systems, networks, and assets that are so essential that their continued operation is required to ensure the security of the state, nation, its economy, and the public's health and safety (Connecticut State, 2020). Critical infrastructure provides services crucial for everyday life, e.g., banking, communication, energy, food, finance, health, transport, and water. Infrastructure, which is resilient and secure, is a backbone in supporting productivity and economic growth. Disturbances in critical infrastructure can cause harmful consequences for businesses, communities, and governments affecting service continuity and supply security. (Australian Government Department of Home Affairs, 2020) Disruptions to critical infrastructure can be caused by, for example, real-world cyber-attacks, which may include environmental damage, financial loss, and even substantial personal injury.

In Finland, critical infrastructure has not been defined in legislation, but the Finnish Government discussed Finnish supply security objectives in 2013. The Finnish Government's decision on supply security objectives contains information about integral threats against the performance of society's vital functions. The decision divides critical infrastructure protection as follows: "Energy production and distribution systems, financial

services, infrastructure and communications systems, transport and logistics, information and communication systems, networks and services, transport and logistics, waste management in special situations and water supply." (Valtioneuvosto, 2013)

European Parliament adopted the directive on security of network and information systems (the NIS directive) on 6 July 2016, aiming to bring cybersecurity capabilities at the same level of development in all the EU Member States and ensure that exchanges of information and cooperation are efficient, including at the cross border level. The directive increases and facilitates strategic cooperation and the exchange of information among the EU Member States.

(European Commission, 2016) The core idea of the NIS directive is that relevant service operators and digital service providers shall ensure their information infrastructure is secure, ensure business continuity in case of adverse information security disruptions, and report any substantial information security breaches to authorities. (Cagla, 2018) According to (EUR-Lex, 2016), NIS sectors, according to the directive, are the following: 1) Banking, 2) Digital infrastructure, 3) Drinking water supply, 4) Energy, 5) Financial market infrastructure, 6) Health, 7) Transport.

In the United States, there are 16 critical infrastructure sectors whose assets, systems, and networks are so vital to the country that operational incapability or destruction would have a harmful impact on security, economic security, public health, or safety. These 16 sectors are the following: 1) Chemicals, 2) Business, 3) Communications, 4) Critical manufacturing, 5) Damns, 6) Defense industry, 7) Emergency services, 8) Energy, 9) Financial services, 10) Food and agriculture, 11) Government facilities, 12) Healthcare and public health, 13) Information technology, 14) Nuclear reactors, materials, and waste, 15) Transportation systems, 16) Water and wastewater systems. (CISA, 2020)

Table. 1 Critical infrastructure sectors in Finland, EU and United States.

NIS-directive	**Supply security**	**United States**
Banking	Energy	Chemicals
Digital infrastructure	Financial services	Business
Drinking water supply	Infrastructure construction and maintenance	Communications
Energy	Information and communication	Critical manufacturing
Financial market infra.	Transport and logistics	Damns
Health	Waste management in special situations	Defense industry
Transport	Water supply	Emergency services
		Energy
		Financial services
		Food and agriculture
		Government facilities
		Healthcare and public health
		Information technology
		Nuclear reactors, materials and waste
		Transportation systems
		Water and wastewater systems

Critical infrastructure is facing various threats that may lead to the appearance of disruptive events causing disruption or failure of the services provided. Minimizing the impact of disruptions and ensuring continuity of services is often cost-effective and the most resilient way, which can be approached with strengthening the resilience. Resilience in the CI system can be seen as a quality that mitigates vulnerability, minimizes the effects of threats, accelerates response and recovery, and facilitates adaptation to a disruptive event. (Rehak et al., 2018). According to Berkeley et al. (2010), resilience is a fundamental strategy that makes the business

stronger, communities better prepared, and nations more secure. Hence, resilience is an ability to absorb, adapt to, and quickly recover from a disruptive event (Rehak et al., 2018).

In cybersecurity, (cyber) resilience denotes the ability to plan, respond, and recover from cyber-attacks and possible data breaches and continue to operate efficiently. An organization can be cyber resilient if it can safeguard itself against cyberattacks, provide expedient risk control for information protection, and assure continuity of operation within and after a cyber incident. For an organization, cyber resilience aims to preserve the ability to deliver goods and services concerned, such as the ability to restore common mechanisms, change or modify mechanisms according to the need during a crisis or after a security breach. (Teceze, 2018) These kinds of attacks, such as cybersecurity breaches or cyberattacks, are able to cause companies significant damage attempting to destroy, expose, or obtain unauthorized access to computer networks, personal computer devices, or computer information systems (RSI Security, 2019).

Cyber resilience consists of four elements (Nathan, 2018), which are the following: 1) Manage and protect, 2) Identify and detect, 3) Respond and recover, and 4) Govern and assure. Manage and protect consists of the capability to identify, analyze, and handle security threats associated with networks and information systems; third and fourth-party vendors included. Identify and detect consists of continuous security monitoring and surface management of threats to detect anomalies and data breaches in addition to leaks before they cause significant problems. Respond and recover concerns incident response planning in order to assure continuity of functions (e.g., business) even in case of a cyberattack. Govern and assure confirms that the cyber resilience scheme is supervised as usual through the whole organization.

3 Cyber-physical systems

NIST (2013) described cyber-physical systems (CPS) as "smart systems that encompass computational (i.e., hardware and software) and physical components, seamlessly integrated and closely interacting to sense the changing state of the real world" (NIST, 2013). Rajkumar, Lee, Sha, & Stankovic (2010) instead characterized cyber-physical systems as "physical and engineered systems whose operations are monitored, controlled, coordinated, and integrated by a computing and communications core." While according to Griffor et al. (2017), cyber-physical systems are sociotechnical systems seamlessly integrating analog, digital, physical, and human components engineered for function through integrated physics and logic (Griffor et al., 2017). These definitions have many similarities, especially; they agree on CPS systems having a physical part, seamless integration of the devices, and controlling software. Compared to the NIST definition, on the one hand, the definition by Rajkumar et al. (2010) impress the need for monitoring, controlling, and coordinating the functioning of the engineered system. On the other hand, the definition by Griffor et al. (2017) includes the human aspect and the need for the system to have a reason to exist in the first place. However, the most general definition the authors have come across is the one by Legatiuk and Smarsly (2018); all CPSs include both computational (cyber) part, which controls the system, and a physical part, which includes sensors, actuators, and the frame.

There are various definitions of cyber-physical systems as introduced above. Therefore, the authors settle for defining a cyber-physical system as a cohesive group of computational devices capable of communication; and controlling, coordinating, and monitoring software, engineered and closely integrated aiming to solve the common problem the physical frame or the users of the physical frame might come across during operation of the entire system under uncertainties related to the physical frame and agents. The agents refer to hardware (e.g., sensors, actuators, or other devices) and software (e.g., ML-based access control, energy consumption control programs, etc.) that generate or process the data in any way, including humans. One should understand that different definitions of CPS serve a specific need, and every cyber-physical system might not fit the said definition even though it might be a cyber-physical system.

CPSs can be implemented as feedback systems that are adaptive and predictive, intelligent, real-time, networked, or distributed, possibly with wireless sensing and actuation. In CPSs, physical processes are controlled and monitored by embedded computers and networks with feedback loops where physical processes influence computations and contrarily. CPSs are data-intensive, generating a lot of data during their use. For example, sensors may be able to collect air pressure, CO2, humidity, motion detection, temperature, etc. These kinds of systems provide the foundation of critical infrastructures (CI), providing means to develop and implement smart services of the future, and improving quality of life in various areas. Cyber-physical systems

interact directly with the physical world; thus, they are able to provide advantages to our daily lives in the form of automatic warehouses, emergency response, energy networks, factories, personalized health care, planes, smart buildings, traffic flow management, etc.

Feedback system refers to programs having the capacities to accept and use data both from previous time steps and current time step in the calculation of how the program should change the state of its comprising components or, in other words, how the actuators should be adjusted to implement changes to the system's flow. For example, the program might try to decide how the valve of the HVAC cooling device should be adjusted to save the maximum amount of energy with the least amount of changes made to the device's state. Without this knowledge of previous events or data by the system, it can be difficult to make intelligent choices that affect the future state of the network.

CPS can utilize, for example, the interconnected network of various embedded Internet-of-Things (IoT) sensors, devices, and actuators, which observe a small portion of the physical world and, based on the decisions made by the guiding program, change the actuators behavior and thus, cause change to the behavior of the surroundings. The change in physical surroundings might have large scale effects for the whole system's operation, such as advancements of indications to impending and unavoidable service breaks. Therefore, the software program attempts to harmonize the totality of the ensemble of sensors and actuators under the challenges brought upon by the system and the real-world. One of these challenges can be, for example, the replacement of an old actuator with a new one. If the new actuator has capacities beyond the old device, recognizes a different protocol, or stores data in some other format than the old one, then the program might not be able to communicate with the device, and it may cause an error to the system holistically, and thus, the CPS may need calibration or human intervention to correct.

Cyber-physical systems are becoming more and more widespread in the future. For example, even though smart building technology is still in the early stages of growth, its adoption throughout the world is increasing, and it is becoming a remarkable business. For example, the value of smart cities (another embodiment of CPS) is expected to reach over USD 820 billion in the year 2025 (marketsandmarkets, 2020). The same could be said about smart grid technology used to manage energy consumption in energy networks. According to a whitepaper by Business Finland (2016), the energy clusters' yearly turnover just in Finland has reached EUR 4.4 billion (Business Finland, 2016).

A smart building concept can be defined as a set of communication technologies enabling different objects, sensors, and functions within a building to communicate and interact with each other and be managed, controlled, and automated in a remote way (European Commission, 2017). It can measure information, such as the temperature of a room or state of windows (open or closed), by utilizing sensors located in the building. The building can become smart if it can obtain such information. An actuator can be used to open a door or to increase the heating temperature of buildings. Intelligent sensors provide significant amounts of information, which must be gathered, processed, and utilized to enable smart functionalities. CPS provides means to utilize sensors to collect data from smart buildings to adjust and control automatically, for example, heating, ventilation, and air conditioning (HVAC) systems. Relevant variables, such as energy, electricity, water consumption, inside and outside temperature, humidity, carbon dioxide, and motion detection, can be utilized in controlling the functions of smart buildings.

Automation and digitalization have become important topics in the energy sector these days, as modern energy systems (e.g., smart grids) increasingly rely on communication and information technology to combine smart controls with hardware infrastructure. The smart grid is another complex example of a cyber-physical system, which continuously evolves and expands. These technologies leveraged the intelligence level of the SG by enabling the adoption of a wide variety of simultaneous operation and control methods into it, such as decentralized and distributed control, multi-agent systems, sensor networks, renewable energy resources, electric vehicle penetration, etc. (Mohammad et al. 2018) In brief, smart grids are electric networks that employ advanced monitoring, control, and communication technologies to deliver reliable and secure energy supply, enhance operational efficiency for generators and distributors, and provide flexible choices for prosumers by integrating the physical systems (power network infrastructure) and cyber systems (sensors, ICT, and advanced technologies) (Xinghuo, 2016).

4 Cybersecurity

The history of cybersecurity dates back to the 1970s when ARPANET (The Advanced Research Projects Agency Network) was developed during a research project. At this time, concepts of ransomware, spyware, viruses, or worms did not yet exist. These days due to active cybercrime, these concepts are frequently mentioned in the headlines of newspapers. Cybersecurity has become a preference for organizations worldwide, especially concerning critical infrastructure. The question is not if the system will be under attack, but the question is when it will happen. Hence, proper measures to detect and prevent malicious cyberattacks are required in order to secure essential assets for the functioning of a society or economy.

The concept of cybersecurity can be defined in various ways. Cambridge dictionary defines cybersecurity as follows: “things that are done to protect a person, organization, or country and their computer information against crime or attacks carried out using the internet.” Gartner defined cybersecurity as the combination of people, policies, processes, and technologies employed by an organization to protect its cyber assets. Cybersecurity can also be thought of as a practice of protecting systems, networks, and programs from digital attacks (Cisco). Furthermore, cybersecurity can be defined subsequently: “cybersecurity refers to the preventative techniques used to protect the integrity of networks, programs, and data from attack, damage, or unauthorized access.” (Paloaltonetworks, 2020).

The main purpose of cybersecurity is to ensure information confidentiality, integrity, and availability, which form the well-known CIA triangle. Confidentiality means that data should not be exposed to unauthorized individuals, entities, and processes or to be read without proper authorization. Integrity means that the data concerned is not to be modified or compromised in any way; therefore, maintaining the accuracy and completeness of the data is crucial. The data is assumed to be accessed and modified by authorized individuals, and it is anticipated to remain in its intended state. Availability means that information must be available upon legitimate request, and authorized individuals have unobstructed access to the data when required. (Nweke, 2017)

In the field of cybersecurity, threat, vulnerability, and risk are intertwined concepts. The risk is located in the intersection of an asset, threat, and vulnerability, being a function of threats exploiting vulnerabilities to obtain, damage, or destroy assets. Threats may exist, but if there are no vulnerabilities, there is no risk, or the risk is relatively small. The formula to determine risk is the following: risk = asset + threat +vulnerability. (Flores et al., 2017) The generic definition of risk is the following: “risk is a description of an uncertain alphanumeric expression (objective or subjective), which describes an outcome of an unfavorable uncertain event, which might degrade the performance of a single (or community of) civil infrastructure asset (or assets).” (Ettourney, 2016). Assets denotes what to be protected, a threat is a target to be protected against, and vulnerability can be experienced as a gap or weakness in protection efforts. Threats (attack vectors), especially in cybersecurity alludes to cybersecurity circumstances or events with prospective means to induce harm by way of their outcome. Attack surface sums up all attack vectors (penetration points), where a perpetrator can attempt to gain entry into the target system. Common types of intentional threats are, for example, DoS/DDoS attacks, malware, phishing attacks, social engineering, and ransomware. General vulnerabilities are, e.g., SQL injections, cross-site scripting, server misconfigurations, sensitive data transmitted in plain text, respectively.

Measures in the field of cybersecurity are associated with risk management, vulnerability patching, and system resiliency improvements (Lehto, 2015, 3-29). Cybersecurity risk management uses the concept of real-world risk management and applies it to the cyber world by identifying risks and vulnerabilities and applying administrative means and solutions to sufficiently protect the organization. Reducing one or more of the following components (Riskviews, 2013) is an integral part of the risk management process: threat, vulnerability, and consequence. In order to improve system resiliency, improving one or more of the following components is required to be improved: robustness, resourcefulness, recovery, and redundancy. Robustness includes the concept of reliability and alludes to the capability to adopt and endure disturbances and crises. Redundancy involves having excess capacity and back-up systems, enabling the maintenance of core functionality in case of disturbances. Resourcefulness denotes the capability to adjust to crises, respond resiliently, and, when possible, to change a negative impact into a positive one. Response means the capability to mobilize quickly prior to crises, and recovery denotes the capability to regain a degree of normality after a crisis or event.

The important question is to detect the challenges of cybersecurity and to counter them expediently. Cyberattacks cannot be prevented entirely. Hence, an integral part of cybersecurity is to preserve the capability to function under a cyberattack, stop the attack and restore the organization's functions to the previous regular state before the incident took place (Limnell, Majewski & Salminen, 2014, 107). In order to counter cyber threats, appropriate measures are important to be taken care of in addition to building adequate protection against the harmful impact of the threats. For example, organizations may utilize an incident response plan (IRP) to detect and react to computer security incidents, determine their scope and risk, respond appropriately to the incident, communicate the results and risks, and reduce the likelihood of the incident from reoccurring (Carnegie Mellon, 2015).

5 Artificial Intelligence and Machine Learning

Artificial intelligence is a mathematical approach to estimate a function, and it can be expressed with mathematical terms as f(x): Rn -> Rm, where f(x) is the function to model, Rn represents the real multidimensional input values, and Rm represents the possible real multidimensional output values. The machine learning research field is needed to make AI models and systems more capable of handling new situations (Jordan, & Mitchell, 2015) because resources might have been limited during initial training, and the occurring circumstance might be from outside the original input or output domain that was used for training of the model. Deep Learning (DL) is a subfield of ML, where the learning is done with models that have multiple layers within their structure. The additional depth can help the models to learn more complex associations within the given data than regular AI models (LeCun et al., 2015); hence DL models are called deep.

Artificial intelligence is a very enticing choice for many different use cases, where the function to be estimated either unknown or difficult to implement in practice, such as machine translations. In practice, the quality and quantity of data, the structure of the model, and training time, as well as the training method, affect how any AI learns to make its choices. Especially, the data quality is an important aspect of the training of an AI. In a case where there is no connection between given inputs and expected outputs, the outcome of the trained model will not reflect reality. In other cases, the poor quality of data may cause the model to gain no insights into the intended use. In a worse case, the model passes the production inspections and winds up in a live situation where it just does not function properly. The malfunction is even worse if it hides itself to take place only under certain specific situations or if the model's use case is of high importance. Therefore, the implementation of artificial intelligence requires, if not expert knowledge of the field where it is intended to be applied to, but rather clear, innate relation between the inputs and the outputs, and rigorous documenting, testing, and follow up after the implementation.

Ensemble methods refer to grouping different ML models together to process inputs, or according to Valle, Saravia, Allende, Monge, and Fernández (2010), to the manner, the data is to be used in the training phase of these models. Either the definition, both typically consider the ensemble as some version of two different structures, which either process the inputs in sequence or in parallel (that in the case of model training are both resource inefficient and inaccurate, respectively (Valle et al., 2010)). With the utilization of ensembles, it is possible to improve ML models' performance. Imagine that you have similar ML models, which have been trained for the same problem domain, but the data they have been trained with were from different patches or data sources. Hence, it is not probable that these models have had the same learning experience and that they would calculate exactly the same predictions with the same prediction confidences based on the same inputs. In an ensemble, the performance scores may rise as the result of the ensembled models' outputs, and confidence scores are compared against each other. The errors stemming from individual models' states get mitigated, thus lessening the effect of any bias within the models. The process can be thought of like voting, where the most endorsed output becomes the actual final output, or more commonly, the final output is some weighted combination of the predicted outputs.

Decision trees (DTs) represent the more traditional algorithms used in artificial intelligence development, and their popularity is mostly related to the ease of interpretation of the results. The interpretation is simpler because these models' behavior is well defined, forming decision rules or paths from the data systematically. A decision tree is a flowchart-like tree structure where an internal node represents a feature or attribute, the branch represents a decision rule, each leaf node represents the outcome, and the first node in a DT is known as the root node. It learns to partition based on the attribute value partitioning the tree recursively and

providing the tree classifier a higher resolution to process different kinds of numerical or categorical datasets. (Shahrivari et al., 2020) Depending on the decision criteria, the algorithm chooses which part of the input data is most significant at each iteration until the conclusion criteria have been filled. It can model nonlinear or unconventional relationships. In other words, DTs can be used to explain the data and their behavior. In addition, many coding libraries have visualization capacities of these paths. However, the decision tree's performance suffers from unbalanced data, overgrowing decision paths, which may also hinder the model's interpretation, and updating a DT by new samples is challenging (Shahrivari et al., 2020).

Random Forest (RF) includes a significant number of decision trees forming a group to decide the output. Each tree specifies the class prediction resulting in the most predicted class in DTs. RF trees protect each other from distinct errors, and if a single tree predicts incorrectly, other trees will correct the final prediction. RFs can reduce overfitting, deal with a huge number of variables in a dataset, estimate the lost data, or estimate the generalization error. RFs experience challenges in reproducibility and interpreting the final model and results. RFs are swift, straightforward to implement, extremely accurate, and relatively robust in dealing with noise and outliers. RFs are not fit for all the datasets as they tend to induce randomness into the training and testing data. (Shahrivari et al., 2020)

Neural network (NN) is a popular base model used in the development of AI solutions. The model has three layers: an input layer, a hidden layer, and an output layer, where data flows from the input layer through the hidden layer consisting of multiple layers, and the result is produced to the output layer. NNs are a collection of structured, interjoined nodes whose values are comprised of all the weights of the connections coming to each node. Every value of a node is inputted to an activation function, such as a rectified linear unit (ReLu). The activation function is typically the same for all the nodes in the same layer.

NN may require a lot of quality data. The need is formed based on the difficulty of the problem, suitability of the data, and the chosen structure and size of the model. In case there are a limited amount of quality data available, it can be beneficial to attempt using two competing neural networks to generate the missing training data. According to Probst (2015), the general way is to have the first model to generate new values based on the original data, and the second model tries to classify the original and generated inputs (the outputs of the first model) from each other. The results of the classifier are then used as feedback for improving the generator and the classifier. Eventually, the generated outputs' distributions move closer and closer to the real inputs. This machine learning method is called Generative Adversarial Neural networks (GAN) (Probst, 2015).

Long-Short Term Memory neural network (LSTM) is a special case of Recurring Neural Network (RNN) (Lipton et al., 2015), which retains output information from previous timesteps as part of the input information. The extra information can be helpful, i.e., when forecasting with sequential data. Because NNs can suffer from the problems of vanishing and exploding gradients, which likely will increase with the growth of sequence size, LSTMs have three gates within each node that are used to control the information going through them (Lipton et al., 2015). These logical gates use sinh and tanh activation functions to control the flow and size of internal representations of the inputs and outputs. RNN, LSTM, and their various variants have been used, for example, in machine translation tasks (Zhang, Liu & Song, 2018), predicting the smart grid stability (Alazab et al., 2020), and classifying malware (Athiwaratkun & Stokes, 2017).

Even though NN models suffer from data issues and it can be more difficult to interpret how models have reached their conclusions, they are perceived to attain more accurate results than some of the traditional algorithms, such as decision trees. In addition, Zhang, Yang, Ma, and Wu (2019) used DTs to interpret the predictions of a Convolutional Neural Network (CNN) model, thus explaining the model's behavior (Zhang, Yang, Ma & Wu, 2019). A convolutional neural network is a neural network that has special layers within its hidden layers. These layers group the inputs systematically from the previous layer and calculate a value for each of these groups, which they then output for the next layers as inputs (Albawi, Mohammed, & Al-Zawi, 2017); consequently, reducing the layer's dimensions. The field of research focused on explaining and interpreting these malleable algorithms for human experts in an easily understandable form is called explainable artificial intelligence (XAI) (Barredo Arrieta et al., 2019).

6 Cyberattacks against critical infrastructure facilities

6.1 Adversarial attacks

An adversarial attack is an attack vector created using artificial intelligence. These attacks are adversarial disruptions constructed purposely by the attacker. The disruptions are imperceptible in the human eyes but generally adversely impact neural network models. These days, adversarial attacks towards machine learning models are becoming more and more common, bringing out noticeable security concerns. For example, in the context of smart building (CPS), an attacker may have a chance to deceive the ML model into causing harm, such as to create conditions for consumption spikes, when attacking the heating system guided by predictive machine learning-based feedback system.

An adversarial attack happens when an adversarial example is sent as an input to a machine-learning model. An adversarial example can be seen as an instance to the input with features that deliberately cause a disturbance in an ML-model to deceive the ML-model into acting incorrectly and into making false predictions (Ibitoye et al. 2019). Deep learning applications are becoming more critical each day, but they are vulnerable to adversarial attacks. Szegedy et al. (2013) argue that making tiny changes in an image can allow someone to cheat a deep-learning model to classify the image incorrectly. The changes can be minimal and invisible to the human eye and can eventually lead to considerable differences in results between humans and trained ML-models.

The effectiveness of these attacks is determined based on the amount of information the perpetrator has concerning the model. In a white-box attack, a perpetrator has total knowledge about the model (f) used in classification, and she knows the classifier algorithm or training data. She is also aware of the parameters (θ) of the fully trained model architecture. The perpetrator then has a possibility to identify the feature space where the model may be vulnerable (e.g., where the model has a high error rate). The model can then be exploited by modifying an input using an adversarial example crafting method. (Chakraborty et al., 2018)

There may be indirect ways to obtain an adequate amount of knowledge about a learned model to apply a successful attack scenario. For example, in case of a malware evasion attack, a set of features may be public through published work. Datasets used to train the detector might be public, or there might be similar ones publicly available. The learner might use a standard learning algorithm to learn the model, such as deep neural networks, random forest, or Support Vector Machine (SVM), by using standard techniques to adjust hyperparameters. This may lead to the situation that the perpetrator can get a similar working detector as the actual one (Vorobeychik & Kantarcioglu 2018).

In the case of Black-box attacks, the perpetrator does not know the type of the classifier, detector's model parameters, classifier algorithm, or have any knowledge about the training data in order to analyze the vulnerability of the model. (Biggio et al. 2013) For example, in an oracle attack, the perpetrator exploits a model by providing a series of carefully crafted inputs and observing outputs. In model inversion type of an attack, the perpetrator cannot directly access the target model, but she can indirectly learn information, such as model structure and parameters, about the model by querying the interface system and gather the responses. (Chakraborty et al., 2018) Papernot et al. (2017) presented a strategy (Papernot-attack) to produce synthetic inputs by using some collected real inputs. Many studies are focusing on research utilizing images as datasets (MNIST or CIFAR). In such a case, the perpetrator can, for example, fetch several pictures of the target dataset and use the augmentation technique for each of the pictures to find new inputs that should be labeled with the API. The next step is to train a substitute by sequentially labeling and augmenting a set of training inputs. After the substitute is accurate enough, the perpetrator can launch white-box adversarial attacks, such as FGSM (Fast Gradient Sign Method) or JSMA (Jacobian Saliency Map Approach), to produce adversarial examples to be transferred to the targeted model (Goodfellow, McDaniel & Papernot 2018).

Jacobian-based saliency map algorithm (JSMA) was presented by Papernot et al. to optimize L0 distance. JSMA attack can be used for fooling classification models, for example, neural network classifiers, such as DNNs in image classification tasks. The algorithm can induce the model to misclassify the adversarial image concerned as a determined erroneous target class. (Wiyatno & Xu, 2018). JSMA is an iterative process, and in each iteration, it saturates as few pixels as possible by picking the most important pixel on the saliency map in a given image to their maximum or minimum values to deceive the classifier. (Pawlak, 2020) Even though the attack alters a small number of pixels, the perturbation is more significant than $L\infty$ attacks, such

as FGSM (Ma et al., 2019). The method is reiterated until the network is cheated or the maximal number of altered pixels is achieved. JSMA can be considered as a greedy attack algorithm for crafting adversarial examples, and it may not be useful with high dimension input images, such as images from the ImageNet dataset (Ma et al., 2019).

The JSMA attack can cause the predictive model to output more erroneous predictions, which can, eventually, make the controlling model either complacent or too reactive. Both choices could be monetarily crippling. For example, Papernot et al. (2016) were able to perturb both categorical and sequential RNNs with JSMA adversarial attack. Therefore, the chance exists that the perpetrator could, if given enough time and resources, afflict damage to both AI models, namely the cybersecurity AI model and the controlling AI model.

A white-box attack uses the target model's gradients in producing adversarial perturbations. FGSM was introduced by Goodfellow et al. (2018) to generate adversarial examples against NN. FGSM can be used against any ML-algorithms using gradients and weights, thus providing low computational cost. The gradient needed can be calculated by using backpropagation. If internal weights and learning algorithm architecture is known, with backpropagation FGSM is efficient to execute (Co, 2018). FGSM fits well for crafting many adversarial examples with major perturbations, but it is also easier to detect than JSMA; therefore, JSMA is a stealthier perturbation, but the drawback is higher computational cost than FGSM. Defense mechanisms can prevent a relatively considerable number of FGSM and JSMA attacks. (Goodfellow et al., 2018).

Carlini and Wagner (C&W attack) has been presenting one of the most powerful iterative gradient-based attacks towards Deep Neural Networks (DNNs) image classifiers due to its ability to break undefended and defensively distilled DNNs on which, for example, the Limited-Memory-Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) and DeepFool attacks fail to find the adversarial samples. In addition, it can reach significant attack transferability. C&W attacks are optimization-based adversarial attacks, which can generate L0, L2, and L∞ norm measured adversarial samples, also known by CW0, CW2, and CW∞, respectively. The attack attempts to minimize the distance between a valid and perturbed image while still causing the perturbed image to be misclassified by the model (Short et al., 2019). In many cases, it can decrease classifier accuracy near to 0 %. According to Ren et al. (2020), C&W attacks reach a 100 % success rate on naturally trained DNNs for image datasets, such as MNIST, CIFAR-10, and ImageNet. C&W algorithm is able to generate powerful adversarial examples, but computational cost is high due to the formulation of the optimization problem.

Gradient-based and gradient-free adversarial attacks mentioned in this chapter, such as C&W, FGSM, and JSMA, can perturb the input data in such a way that the inputs seem valid for a human but mess maliciously with, e.g., a machine-learning model that can automatically adjust HVAC and other heating devices of smart buildings. This kind of model may gather data from local measurement units (IoT sensors) and external data from the weather database, including data from social media accounts. Data can then be properly merged and cleaned to be utilized in training the predictive model. The predictive model may use, e.g., LSTM neural networks to perform energy load forecasts and calculate the need for new commands to be sent to the actuators.

This kind of a classification-oriented LSTM neural network can be attacked, for example, by using the mentioned JSMA attack method. It then perturbs the input in the desired direction to selectively make the model misclassify to an appropriate output class (Anderson et al., 2016). Deep neural networks can be deceived by adding even minor perturbations, such as flawed pixels, to form an image classification problem and to be used to deceive sophisticated DNNs in the testing or deploying stage. The vulnerability of adversarial examples is an ample and ever-growing risk, especially when the field of critical infrastructure is concerned. Fooling the predictive deep neural network used to adjust the HVAC system of a cyber-physical system can cause challenging situations in the form of energy consumption spikes causing increasing operational costs.

6.2 DoS and DDoS attacks

Denial of service (DoS), and its variant (DDoS), is one of the major threats, and it can cause disastrous consequences because of its distributive nature. These attacks conducted by a perpetrator may use single or even multiple computers known as zombies in order to consume the victim's resources so that the server cannot provide a requested service to a legal or legitimate user. The perpetrator utilizes the advantage of the internet, network bandwidth, and connectivity to target the open points and initialize floods of thousands or even

millions of packets to knock off the victim's server. The server either crashes or becomes incapable of serving all of the incoming requests, and it cannot serve the legitimate clients who are trying to use the service provided by the server concerned. These attacks' main targets can be, for example, default gateways, personal computers, web servers, etc.

Perpetrators aim to look for the path they can use to gather the secret information they are after. This denotes compromising confidentiality. The second phase, which compromises the integrity, is to gain access to the confidential information to alter it. The third phase is to compromise the availability, which is the main target of perpetrators as compromising confidentiality and integrity are more challenging, requiring more advanced technical skills in order to succeed. Administrative privileges on the target system are not needed when availability is compromised. Perpetrators can compromise the service's availability by exhausting the resources to make the service unavailable for legitimate users, as mentioned earlier.

DoS/DDoS attacks can be conducted in many ways using different kinds of program codes and tools, and they can be initiated from different OSI model layers. OSI has seven layers, which are physical (layer 1) covering transmission and reception of the unstructured raw bit stream over a physical medium, data-link (layer 2) responsible for conducting an error-free transfer, network (layer 3) handles routing of the data, transport (layer 4) responsible for the packetization and delivery of data, session (layer 5) taking care of establishment, coordination, and termination of sessions, presentation (layer 6) handles data translation and sending it to the receiver, and application (layer 7) where communication partners are identified. All the messages and creating packets initiate at this level. (Obaid, 2020)

DDoS attacks may cause physical destruction, obstruction, manipulation, or malfunction of physical assets on the physical layer. MAC flooding attack floods the network switch with data packets, which usually happens on the Data-link layer. Internet Control Message Protocol (ICMP) flooding utilizes ICMP messages to overload the targeted network's bandwidth, a network layer 3 infrastructure attack method. SYN flood and Smurf attacks are transport layer 4 attack methods. In an SYN attack, series of "SYN" (synchronize) messages are sent to a computer, such as a web server, after communication between two systems over TCP/IP has been established (TechTerms, 2020). A smurf attack is an old DoS attack, which uses a great number of ICMP packets to flood a targeted server. SYN attack utilizes TCP/IP communication protocol to bombard a target system with SYN requests to overwhelm connection queues and force a system to become unresponsive to legitimate requests. On the session layer, a perpetrator can use DDoS to exploit a vulnerability in a Telnet server running on the switch, forcing Telnet services to become unavailable. On the presentation layer, the perpetrator can also use malformed SSL requests as inspecting SSL encryption packets is resource-intensive. Vulnerabilities to DDoS attacks on the application layer are, e.g., use of PDF GET requests, HTTP GET, HTTP POST methods on website forms, when logging in, uploading photo/video or submitting feedback, etc. (Qureshi, 2018)

Perpetrators may utilize botnets, which can be described as a network of several or a large number of computers or internet-enabled devices that have been taken over remotely, to launch numerous types of attacks, such as DDoS, spamming, sniffing and keylogging, identity theft, ransom and extraction attacks, etc. Botnet (zombies) target vulnerabilities in different layers of the open systems interconnection. These attacks can be divided, e.g., in the following way: 1. Application layer attacks, 2. Protocol attacks, and 3. Volumetric attacks. Application layer attacks are the most primitive form of DDoS mimicking normal server requests. This type of attack was explained in detail at the beginning of this chapter. Protocol attacks exploit the way servers process the data to overload and overwhelm the intended target. One way to conduct this type of attack is to send data packets, which cannot be reassembled, resulting in overwhelming the server's resources. Volumetric attacks are similar to application attacks, but in this type of DDoS attack, the whole server's available bandwidth is used by botnet requests. A high amount of traffic or request packets to a targeted network will be sent in order to slow down or stop the target services. (Porter, 2019)

DDoS attacks are able to cause a significant threat to critical infrastructure sectors, such as energy and transportation. The DDoS attack disrupted the heating distribution system, at least in two properties in the city of Lappeenranta, eastern Finland, in 2016. In the incident, attacks incapacitated the controlling computers of heating in the buildings concerned. The attack lasted from late October to November 3, causing inconvenience and potentially hazardous situations as the outside temperature was below freezing. During the attack, the system tried to respond by rebooting the main control circuit, which was then continuously repeated, making heating incapable of working. Unfortunately, building automation security is often neglected,

and housing companies are often reluctant to invest in firewalls and other security measures in order to improve the general security situation. (Metropolitan, 2016)

DDoS attacks have been conducted against transportation services, causing train delays and disruption over travel service. Swedish transportation system experienced such an attack on October 11 in 2017, via two internet service providers, TDC and DGC. The DDoS attack crashed the train location monitoring IT system, guiding operators to go and stop the train. The attack also knocked out the federal agency's email system, road traffic maps, and website services. As a result of the attack, train traffic and other services had to be operated manually by utilizing back-up processes. (Barth, 2017) In 2018 Danish rail travelers experienced trouble while buying tickets due to a paralyzing DDoS attack on Denmark's largest DSB railway company's ticket system. The attack made it impossible to buy a ticket via the DSB app, on the website, ticket machines, and kiosk stations. Additionally, the attack also restricted communications, telephone systems, and internal mail was also affected. Paganini (2018) In order to communicate delays to customers, the company had to utilize social media and ground staff (McCreanor, 2018). The Freedom of Information Data states that up to 51 % of critical infrastructure organizations in the UK are potentially vulnerable to these attacks due to incapability of detecting and mitigating short-duration DDoS attacks on their networks, and as a result, 5 % of these operators experienced DDoS attacks in 2017 (Reo, 2018). CI operators, such as transport agencies, cannot leave DDoS attack protection at the chance; they are required to build and improve resilience in combatting these attacks.

6.3 False data injection (FDI) attacks

False data injection (FDI) attack poses a significant threat towards the traditional power grid (PG), and in these days, smart grid, technologies that provide power to be used, for example, in cyber-physical systems, such as smart buildings. Smart grids are electrical grids, which utilize information and communication (ICT) technology in providing reliable, efficient, and robust electricity transmission and distribution. Hence, smart grids are not solely well-known power lines in traditional “dumb” energy infrastructures, but they represent a relatively new type of energy distribution system standing among the key relevant concepts in supporting sustainable energy city. SGs are connected to smart meters, which can be installed in entities, such as smart factories, hospitals, schools, etc. include components, which enable predictive analytic services in order to balance the production and consumption in the grid system. Advanced services, such as real-time pricing, provide consumers and suppliers relevant information to manage their energy demands and supplies. The service allows energy distribution to be performed in a dynamic and effective manner. (Chen et al., 2015) In addition, SGs merge the non-renewable and renewable energy resources into each other, reducing environmental problems (Farmanbar et al., 2019).

FDI attacks are typically utilized when conducting attacks against the functionality of smart grids in order to disrupt, for example, real energy and supply figures causing erroneous energy distributions resulting in additional costs or destructive consequences (Chen et al., 2015). According to Elmrabet et al. (2018), the perpetrator can, for instance, use these attacks to modify the smart meter data to lower her electricity bill or target remote terminal unit (RTU) to inject false data to the control center resulting in an increased outage time. FDI attacks can be considered as a type of integrity violation aiming to pose arbitrary errors and distortion to the device’s measurements, influencing the state estimate (SE) precision. SE is a vital service for system monitoring in ensuring reliable operation in the power system and in addition to the energy management system (EMS), which processes real-time data gathered by the SCADA system. Smart meters are able to further infer state estimations (e.g., energy demands and supplies) and to make initial decisions, for example, concerning data fusion before the estimations reach control centers. The information provided can be utilized to optimize energy distribution with regard to power grid performance metrics in order to maximize the network utility and energy efficiency while minimizing energy transmission costs. Hence, FDI attacks violate SE’s integrity making the smart grid system unstable in the worst-case scenario.

The perpetrator may inject the false monitoring data into the smart grid by using, e.g., the following ways: 1. compromising the smart meters, sensors or RTUs, 2. capturing the communication between sensor networks and SCADA system, 3. penetrating the SCADA system resulting in an incorrect estimate of the smart grid state, which may eventually lead even to large-area power failure accidents. According to Sargolzaei et al. (2020), the perpetrator’s aim is not solely to inject false information to distract the solid operation of the

target system but also to inject incorrect data, which keeps the system's controller and detection mechanism in the shadows concerning the incident. The perpetrator may also utilize means to gather side-information, such as to perform particular analyses and techniques to collect knowledge about the nominal state values of the agents, concerning the structure of the target system to conduct FDI attacks to increase the destructive power of the attack. In order to conduct the malicious attack, the perpetrator may need to inject "realistic" false data, which is close enough to the nominal states and parameters of the system to various sensors at the same time. This procedure makes FDI difficult to detect, especially if system architecture is known.

The perpetrator can conduct attacks against one or multiple of the following FDI attack surfaces: energy demand, energy supply, grid-network states, and electricity pricing. Attacking against energy demand can cause fraudulent values of the state estimation raising financial costs to both the energy users and providers due to extra cost of power transmission or waste energy. It may also lead to power outage situations, in which energy requests to the smart grid is less than the energy demand that nodes (representing the average energy demand/supply, e.g., a town) of the grid require. Energy-supply nodes provide the value of SE, and an FDI attack can secretly mitigate the amount of energy supplied, leading to an energy shortage situation of energy-demand nodes as the nodes cannot receive the required energy. In the opposite situation, an increase in wasted energy can occur. (Chen et al., 2015)

Grid-network states represent the configurations and conditions of power grids, for example, grid topologies and power-lines capacities. The perpetrator can use FDI to attack power-line connections in order to isolate nodes from the power grid deceiving the energy distribution system and leading to power shortages or energy transmission costs. Dynamic electricity pricing helps in balancing the power loads between peak and off-peak periods and reduce consumer electricity bills. The perpetrator can lower her electricity price causing loss of company revenue or lower prices during peak hours, leading to the grid system eventually overloading. Hence, fake pricing causes remarkable damage to the financial and physical subsystem, obliterating the advantages of optimum supply efficiencies. (Chen et al., 2015)

6.4 Malware attacks

Malware and software-enabled crime is not a new concept but dates back to the year 1986, when the first malware, Brain. A., appeared for a PC computer. The appearance of malware proved that PC is not a secure platform, and safety measures should be considered. Malware or malicious software is software created and possibly used by perpetrators to disrupt computer functions, collect sensitive information, damage the target device, or obtain access to a private computer system. The form of malware can be, for example, active content, code, scripts, or another kind of software. Malware incorporates adware, computer viruses, dialers, keyloggers, ransomware, rootkits, spyware, trojan horses, worms, and other types of malicious computer programs. In general, most of the common malware threats are worms or trojans instead of regular and ordinary computer viruses. (Milošević, 2013) Since 2018 Ransomware attacks have been showing signs of growth. Malware attacks can occur on all kinds of devices and operating systems, such as Android, iOS, macOS, Microsoft Windows, etc.

Malware attacks against critical infrastructure have been increased during the past several years. In 2012 Iran conducted a destructive retaliation wiper Shamoon malware attack towards Saudi Arabia's national oil conglomerate, Saudi Aramco. The functionality of Shamoon is to wipe out all data from hard disks, and it was used to overwrite hard drives of 30 000 computers in the Aramco -case. (Alelyan & Kumar, 2018) In 2016, a trojan type of malware called BlackEnergy was used to cause disruptions to the Ukrainian electrical grid. BlackEnergy is a modular backdoor that can be utilized to conduct DDoS, cyber espionage, and information destruction attacks towards ICS/Scada, government, and energy sectors worldwide. BlackEnergy malware family has been present since 2007, and initially, it started as an HTTP-based botnet for DDoS attacks. Later on, the second version, BlackEnergy2, was developed, which was a driver component-based rootkit installed as a backdoor. The above mentioned version of the backdoor predominantly spread via targeted phishing attacks by email, including the malware installer. The later version is BlackEnergy3, which was used to attack against Ukrainian electrical power industry. This version can be used when conducting phishing attacks containing Microsoft Office Files packed with malicious obfuscated VBA macros to infect target systems. (Santos, 2016)

Another type of malware that appeared in 2015 and which have been used in attacking healthcare sector critical infrastructure facilities is known as DragonFly. The malware specifically targets industrial control system (ICS) field devices in the energy sector in Europe and in the US. Utilization of the DragonFly remarkably grew during the year 2017. Perpetrators have been interested in learning how energy facilities operate and also how to gain access to operational systems themselves. The malware uses different sorts of infection vectors to obtain access to a victim's network. These vectors include malicious emails, trojan software, and watering hole attacks to leak the victim's network credentials and exfiltrate them to an external server. Hijacked device contacts a command and control server, which is controlled by perpetrators providing a back door to the infected device. (Biasi, 2018)

Stuxnet malware (worm) increased awareness of cybersecurity and related issues in the world after it was detected in 2010. The worm was targeting centrifuges used in the uranium enrichment process in a nuclear plant in Natanz in Iran. Governments around the world had to face the fact the critical infrastructures were vulnerable to cyberattacks with a possibility to cause catastrophic effects. The aim of this malware was to sabotage centrifuges in the power facilities in order to stop or delay the Iranian nuclear program. It is believed that the malware was uploaded to the power plant's network by using an infected USB drive. (Baezner & Robin, 2017)

Stuxnet is larger than other comparable worms, and it is implemented by using various programming languages with encrypted components. It used four zero-day exploits when infecting computers, which are a connection with shared printers, and vulnerabilities concerning privilege escalation, allowing the worm to run the software in computers during lock-down. The worm caused damage to the centrifuges by making them alternate between high and low speeds and by masking the change of speed to look normal. Due to the procedure, Iran had to replace 10 % of its centrifuges yearly. The incident showed critical infrastructure could be targeted by cyber threats, and even networks separated from each other did not protect against the malware. It is integral to increase protection against this kind of malware and, in addition, to improve resilience during cyberattacks. (Baezner & Robin, 2017)

Duqu followed the well-known Stuxnet malware worm and was detected by the Laboratory of Cryptographic and System Security at the Budapest University in Hungary in 2011. The similarity of the malware structure to Stuxnet is so, which indicates that it was developed and implemented by Stuxnet authors or developers who have had access to the source code. Unlike Stuxnet, Duqu was mainly implemented for cyber espionage purposes to obtain a deeper understanding of network structures in order to detect vulnerabilities to exploit and develop better attack methods to penetrate the defenses. (Benchsáth et al., 2012) Duqu is an information stealer rootkit targeting MS Windows-based computers collecting keystrokes and other relevant information, which could be used when conducting attacks against critical infrastructures, such as power plants or water supply around the world. After penetrating the defenses, Duqu injects itself into one of four general Windows processes: Explorer.exe, IEExplore.exe, Firefox.exe, or Pccntmon.exe, downloads and installs an information-stealing component to gather information from the infected target system, encrypts the data, and uploads it to the perpetrator's system. Smart grid with smart meters, substations, intelligent monitors, and sensors provide an attractive attack surface to perpetrators' exploitation of critical infrastructure systems in their minds. (Westlund & Wright, 2012)

Triton is among the most hazardous malware spreading over the networks worldwide, targeting critical infrastructure facilities utilizing automated processes. The malware was first detected in 2017 during the malicious attack towards Tasnee-owned petrochemical plant facility using Schneider Electric's Triconex Safety Instrumented System (SIS), which then experienced a sudden shutdown. The malware was deployed in emergency safety devices, which are required to be started in case of plant toxic gas leaks and during emergency situations. Triton, among other dangerous malicious attacks, can cause safety mechanisms to experience physical damage due to the incapability of operating during emergency situations. It can be used to target industrial control systems (ICS) and to use a secure shell (SSH) based tunnel to deliver attack tools to the victim system and running remote commands of the malware program. A perpetrator accesses information technology (IT)- and operational technology (OT) -networks, installs back doors in the computer network, and accessing the safety instrumentation system (SIS) controller in the OT network in order to secure and maintain the target's networks using attack tools. (Myung & Hong, 2019)

6.5 Phishing attacks

Phishing is a social engineering technique that can be utilized to override technical controls designed and implemented to mitigate security risks in information systems. Social engineering is a manipulation technique exploiting human error to obtain sensitive private information, access, or valuables. The weakest link in the security program is us, the humans. In cybercrime, perpetrators exploit the human component to deceive end-users of the system by manipulating user behavior to expose data, spread malware infections, or provide entry to the restricted system. Attacks can be conducted online, in-person, or via other means. In addition to manipulation of user behavior, perpetrators can exploit a user's lack of knowledge, e.g., "drive-by-download," which infers to installing malicious programs to devices without the user's approval. (Kaspersky, 2020)

Phishing takes advantage of this weakness and exploits the vulnerability of human nature to obtain access to a target system. (Rader et al., 2013) Even though organizations have been long increasing employee awareness of cybersecurity threats, phishing is still among the starting points for various cyberattacks. According to surveys, up to 46 % of successful cyber attacks started with a phishing email sent to an employee. (Cytomic, 2019) According to Abdullah & Mohd (2019), the attack can be used to steal user's confidential information, such as passwords, social security numbers, and banking information, and takes place when cybercriminals disguise as a trusted entity and fool users to click on fake links included in the email received. In addition, cybercriminals also target organizations belonging to the target country's critical infrastructure sector (e.g., telecommunications or defense subsector) by utilizing the special form of phishing, a spear-phishing.

Spear phishing is a certain type of phishing, in which the context and victim are examined, and which utilize custom-made email message that can be sent to the victim. As mentioned before, received email messages can include a malicious link or email attachment to deliver malware payload to direct a benevolent individual to counterfeit websites. These websites can then be used to inquire, e.g., login credentials or ask to download malicious (malware) software to the victim's device. The perpetrator is then able to utilize the credentials or infected devices in order to obtain entry to the network, steal information, and in many cases, stay inconspicuous for a prolonged amount of time. (Bossetta, 2018)

Spear phishing attacks used to conduct attacks towards critical infrastructure occurred in 2014 when a perpetrator initiated a spear-phishing attack against Korea Hydro and Nuclear Power (KHNP). The attack resulted in the leak of personal details of 10 000 KHNP workers, designs and manuals, nuclear reactors, estimates of radiation exposures among residents, etc. During only a few days, the perpetrator managed to send almost 6000 phishing email messages, which included malicious codes to more than 3000 employees. The catch was to demand money for not leaking sensitive classified information to other countries or not to be published in social media on the internet. Luckily, the server containing the information was isolated from the intranet; therefore, the perpetrator managed to cause only confusion in Korean Society. However, cyberattacks towards nuclear power plants may pose a significant risk and damage to all living organisms and the environment over a wide area. Hence, extensive security countermeasures should be developed to mitigate these risks. (Seok & Kim, 2018) Additionally, it is suspected that the Ukrainian power grid was initially attacked with a phishing attack followed by BlackEnergy malware, leaving hundreds of thousands of homes without electricity for six hours (Allianz, 2020).

7 Defensive mechanisms against cyberattacks

7.1 Adversarial attacks

Adversarial examples are maliciously perturbed inputs designed to deceive a machine learning model at test time, posing a significant risk to the ML models. These inputs can transfer across models meaning that the same adversarial example is generally misclassified by various models. Adversarial examples can be countered with adversarial training of ML model classifier, which is one of the earliest and well-known defense methods in combatting adversarial example crafting (e.g., FGSM). The adversarial training method has reached the de-facto standard status in providing robust models (Stutz et al., 2019). Robustness can be improved by augmenting the ML model training dataset with perturbed inputs in case of the training set is the same as the perpetrator uses (Samangouei et al., 2018). Robustness can be reached by adversarial training based on the strength of the adversarial examples utilized. Hence, training a model by using fast non-iterative

FGSM produces robust protection towards non-iterative attacks, such as JSMA. Defending against iterative adversarial examples also requires training to be done with iterative adversarial examples. (Shafahi et al., 2019) If a perpetrator uses a different kind of attack strategy, the efficiency of the adversarial training will decrease (Samangouei et al., 2018).

This method can be applied to large datasets when perturbations are crafted using fast single-step methods. Adversarial training generally attains adversarial examples by utilizing an attack, such as FGSM, and tries to build adequate defense targeting such an attack. The trained model can indicate poor generalization capability on adversarial examples originated from other adversaries. When combining adversarial training on FGSM with unsupervised or supervised domain adaptation, the robustness of the defense could be improved. Unfortunately, the robustness of adversarial training is possible to evade by applying a joint attack with indiscriminate perturbation from other models. (Song et al., 2019) In addition, utilization of adversarial training as a robust defense method is limited in real-life situations due to extensive computational complexity and cost (Shafahi et al., 2019).

Defensive distillation can be considered as an adversarial defense method to counter adversarial attacks, such as FGSM or JSMA. The method is one of the adversarial training techniques, which provides flexibility to an algorithm's process, making it less susceptible to exploitation. According to Zhang et al. (2019), the idea behind defensive distillation is to generate smooth classifiers that are more resilient to adversarial examples by mitigating the sensitivity of the DNN to the input perturbation. The technique also improves the generalization ability as it does not alter the neural network architecture, and in addition, it has low training overhead and no testing overhead.

Papernot et al (2016b) investigated the defensive distillation and introduced a method that can reduce the input variations making the adversarial crafting process more challenging, providing means to DNN to generalize the samples outside the training set and mitigating the effectiveness of adversarial samples on DNN. The defensive distillation reflects a strategy to pass the information from one architecture to another by reducing the size of DNN. The distillation method provides a dynamic method demanding less human intervention and the advantage of being adaptable with yet not known threats. In general, effective adversarial defense training requires a long list of known vulnerabilities of the system and possible attack vectors. Utilization of defensive distillation decreases the success rate of the adversarial crafting process and is also effective against adversarial attacks, such as JSMA.

As a disadvantage, if a perpetrator has a lot of computing power available and the proper fine-tuning, she can utilize reverse engineering to find fundamental exploits. Defense distillation models are also vulnerable to poisoning attacks in which a malicious actor corrupts a preliminary training database. (DeepAI) Defensive distillation can be evaded by the black-box approach (Papernot et al., 2016) and also with optimization attacks (Szegedy et al., 2013). Carlini & Wagner (2017) proved that defensive distillation failed against their L0, L2, and L∞ attacks. These new attacks succeed in finding adversarial examples for 100 % of images on defensively distilled networks. Previously known weaker attacks can be stopped by defensive distillation, but it cannot resist more powerful attack techniques.

Defense-GAN (Generative Adversarial Networks) is a feasible defense strategy providing advanced defense mechanisms against white-box and black-box adversarial attacks posing a threat towards machine learning classifiers. Defense-GAN is trained to model the distribution of unperturbed images, and before sending the given image to the classifier, the image is projected onto the generator by minimizing the reconstruction error and passing the resulting construction to the classifier. Training the generator to model the unperturbed training data distribution reduces potential adversarial noise. Defense-GAN can be used in conjunction with any ML classifier without a need to alter the classifier structure or re-train it, and utilization of the Defense-GAN mechanism should not significantly decrease the performance of the classifier. The mechanism can be used to combat any attack as it does not presume an attack model, but it can utilize the generative efficiency of GANs to reconstruct adversarial examples. (Samangouei et al., 2018)

Defense-GAN overcomes adversarial training as a defense method, and when conducting adversarial training using FGSM in generating adversarial examples against, for example, the C&W attack, adversarial training efficiency is not sufficient. In addition, adversarial training does not generalize well against different attack methods. Increased robustness gained by using adversarial training is reached when the attack model used to generate the augmented training set is the same as that used by the perpetrator. Hence, as mentioned, adversarial training endures inefficiently against the C&W attack; therefore, a more powerful defense

mechanism should be utilized. Training GANs is a remarkably challenging task, and if GANs are not trained correctly and hyperparameters are chosen incorrectly, the performance of the defensive mechanism may significantly mitigate. (Samangouei et al., 2018)

7.2 DoS and DDoS attacks

Distributed Denial of Services (DDoS) attacks have been increasing, contributing to the majority of overall network attacks. Detecting and preventing DDoS attacks is a challenging task, and practically designing and implementing a DDoS defense is incredibly difficult. DDoS attack and defense issues have been under intensive research, and various research has been conducted in the field of the subject concerned. The purpose of a traditional DDoS detection system is to separate malicious packet traffic from abnormal traffic (Mirkovic & Reiher, 2004). Under the traditional network environment, methods for defense against DDoS attacks mainly consist of attack detection and attack response. Attack detection bases on attack signatures, congestion patterns, protocols, and source addresses, forming an efficient DDoS detection mechanism. (Cheng et al., 2018)

The detection model has two categories: misuse-based detection and anomaly-based detection. Misuse-based detection utilizes feature-matching algorithms and matches the gathered and extracted user behavior features with the known feature database of DDoS attacks to detect if an attack has been conducted earlier. An attack in a system is detected wherever the sequence of activities in the network matches with a known attack signature. Anomaly-based detection has been used with monitoring systems in order to determine if the states of the target systems and user's activities differ from the normal profile, and it can then deduct if an attack is taking place. The following step is for an attack response to appropriately filter or limit the network traffic as much as possible after the DDoS attack has been commenced. (Cheng et al., 2018)

Artificial intelligence and its subfield of machine learning have been applied to cybersecurity in recent years, and it has affected the development of an ML-based attack detection model. Machine learning is able to gather relevant information from the data and integrate previously collected knowledge to discriminate and predict new data. Hence, ML-based methods can provide better detection accuracy in comparison to traditional detection methods. As a drawback, data generated by the DDoS attacks are usually burst and diverse. In addition, background traffic size may also have an impact on the detection model, mitigating the model's detection accuracy. (Cheng et al., 2018)

Various studies have been conducted to address the prevention and detection of cyberattacks, such as DDoS attacks, and numerous of them are utilizing ML-based methods, such as support vector machine (SVM), Random Forest, and Naïve Bayes. As an example, Pei et al. (2019) conducted research in order to detect DDoS attacks by using Random Forest and SVM ML-methods. Authors of the research trained random forest model with the training data set and mixed the remaining set of attack data packets with the normal traffic as the test set of the model, cross-sampled normal traffic and attack traffic, calculated behavior of each sample, and controlled the sampling flow period to control the ration of normal traffic to attack traffic. LIBSVM library was then utilized to detect the data of the SVM algorithm and compared it with the random forest model detection results. The research results showed that both Random Forest and SVM methods provided significant (93 % - 99 %, depending on the sampling period) DDoS attack detection accuracy against TCP, UDP, and ICMP flood attacks.

He et al. (2017) proposed a prototype DOS attack detection system on the source side in the cloud, based on machine learning techniques. The prototype was implemented under a real cloud setting, and it included six servers (S0...S5), each server running multiple virtual machines. The authors launched four different kinds of DDoS attacks (SSH brute-force, DNS reflection, ICMP flooding, and TCP SYN attacks) on virtual machines from the S0 server. The victim was a virtual machine on another server S1 running web service. Authors deployed their defense system on the server launching virtual machines running the attacks. Other virtual machines on servers (except S0 and S1) request web service, simulating the legitimate users. The data utilized in the experiment was gathered of network packages coming in and going out of the attacker virtual machines for nine hours. Supervised learning algorithms, such as Linear Regression (LR), SVM (linear, RBF, or polynomial kernels), Decision Tree, Naïve Bayes, and Random forest, were evaluated. For unsupervised algorithms, such as k-means, Gaussian Mixture Model for Expectation-Maximization (GMM-EM), were evaluated, respectively. Supervised algorithms all achieved over 93 % accuracy (Random Forest had the best accuracy with 94.96 %), but unsupervised ones reached only 63 - 64 % accuracy.

Haider et al. (2020) presented a novel deep learning framework for the detection of DDoS attacks in Software Defined Networks (SDNs), which is a prevalent networking paradigm decoupling the control logic from the forwarding logic. SDNs consist of applications (applications running on physical or virtual hosts), control (operating system), and forward planes (network constructed through programmable switches). The framework utilizes ensemble CNN models for improved detection of Flow-based data being critical attributes to SDNs. The authors evaluated the proposed framework with the Flow-based dataset CICIDS2017, which is a public, fully labeled dataset comprised of at least 80 features of network traffic, including both benign and multiple types of attack traffic. The proposed approach provided 99.45 % detection accuracy and minimal computational complexity in detecting DDoS attacks with reasonable testing and training time.

7.3 False data injection (FDI) attacks

FDI attack was introduced in the smart grid domain causing remarkable security challenges to the operation of power systems and can be utilized to circumvent conventional state estimation bad data detection security measures implemented in the power system control room (Ayad et al., 2018). FDIA detection problem has been attempted to solve by using various kinds of optimization methods, such as sparse matrix optimization problem, which can be solved by using the combination of a nuclear norm minimization and low-rank matrix factorization methods. In order to mitigate the resources required in the FDIA detection process, threshold-based comparisons have been commonly utilized. An experimental study shows that the usage of the Euclidean distance metric with a Kalman filter with the selected threshold helps to identify FDIA better than many other metrics. In addition, comparing residual signals with a predefined threshold can be used to detect the FDIA in a networked cyber-physical system. Nonetheless, a progressive number of FDIA attacks have been able to override threshold-based detection methods. (Wang et al., 2019) In order to efficiently combat FDIA attacks, more advanced detection methods, such as blockchain, cryptography, and learning-based methods, can be utilized.

Shen et al. (2016) presented a prevention technique for FDI attack, which guarantees the integrity and availability of the measurement units (measuring the smart power grid's status) and during their transmission to the control center even with the existence of compromised units. McEliece public-key cryptography system is able to guard the integrity of the smart power grid data measurements and prevent the impact of FDIA. As a drawback, cryptographic algorithms require a substantial amount of computing resources due to computational complexity. One of the common buzzwords these days, a blockchain, has been examined by Ahmed et al. (2020) to generate a shield and protect the data authenticity. The authors empirically demonstrated that the blockchain-based security framework is capable of securing healthcare images from false image injection attacks. The blockchain-based security framework introduced by the authors is decentralized as in nature, provided cryptographic authentication and consensus mechanism in order to counter FDIA attacks more efficiently than other previous methods.

Learning-based methods provide a novel and more sophisticated way of countering FDIA attacks. Esmalifalak et al. (2017) proposed an FDIA detector mechanism by utilizing the principle component analysis (PCA) and supervised learning -based support vector machine (SVM) model to statistically separate normal operations of power networks from the case under stealthy attacks. Methods mentioned were utilized to combat a new type of FDIA attacks, such as stealth attacks, which cannot be detected by conventional bad data detection using state estimation. The detection performance of the SVM-based method was relatively high, with 90.06 % accuracy in comparison to Euclidean detector's 72.68 % and Sparse Optimization 86.79 % (Wang et al., 2019). Wang et al. (2019) utilized wide and recurrent neural networks (RNN) model to learn the state variable measurement data and identify the FDIA. The wide component consists of a fully connected layer of neural networks, and the RNN component includes two LSTM layers. The wide component is able to learn the global knowledge and the RNN component has a capability to catch the sequential correlations from state variable measurement data. Wide component accuracy reached 75.13 % and RNN model 92.58 %, respectively. The proposed combination of Wide and RNN models detection performance reached up to 95.23 % accuracy, which outperforms the previously mentioned learning-based detection methods.

He et al. (2017) presented Conditional Deep Belief Network (CDBN) in order to analyze the temporal attack patterns that are presented by the real-time measurement data from the distributed sensors/meters. The aim is to efficiently reveal the high-dimensional temporal behavior features of the unobservable FDI

attacks, which are able to bypass the State Vector Estimator (SVE) mechanism. According to Niu et al. (2019), no prior studies have been conducted on the dynamic behavior of FDI attacks. Detecting FDI attacks is considered a supervised binary classification problem, which is not able to detect dynamically evolving cyber threats and changing the system configuration. The authors developed an anomaly detection framework based on a neural network in order, to begin with, the construction of a smart grid specific intrusion detection system (IDS). The framework utilizes a recurrent neural network with LSTM cell to capture the dynamic behavior of the power system and a convolutional neural network (CNN) to balance between two input sources. In case a residual between the observed and the estimated measures is greater than a given threshold, an attack is launched.

7.4 Malware attacks

Malware infections have been significantly increasing in the past years, and large quantities of malware are automatically created each day. According to AtlasVPN (2020), almost 10 million malware infection cases have occurred per day during the first quartal in 2020, and 64 % of the malicious attacks were targeting educational institutions. These days, there are nearly one billion malware programs out there, and up to 350 000 new pieces of malware are detected each day (Jovanovic, 2019). The number of cybercriminals conducting vicious acts such as malicious attacks has been increasing quickly. The exponential growth of malware has been causing a remarkable threat in our daily life, sneaking in stealth to the computer system without revealing an adverse intent to disrupt the computer operations. Due to the enormous number of malware, it is impossible to deal with the malware solely by human engineers and security experts, but advances and sophisticated detection methods are required.

Malware detection methods can be categorized in various ways depending on the point of view. One possible way is to divide malware detection methods into signature-based and behavioral (heuristic) -based methods. Signature-based detection has been the most widely utilized way method in antivirus programming. This method extracts a unique signature from a malware file and utilizes it in order to detect similar malware. (Xiao et al., 2018) Signature-based detection can be efficiently used to detect the already known type of malware, but it has challenges in detecting zero-day malware and can also be easily defeated by malware that uses obfuscation techniques. Obfuscation techniques include, for example, dead code insertion, register reassignment, instruction substitution, and code manipulation (Sihwail et al., 2018). Additionally, signature-based detection requires prior knowledge of malware samples (Xiao et al., 2018).

In behavior (heuristic or anomaly) -based detection, malware sample behaviors are analyzed during execution in the training (learning) phase in order to label the file as malicious or benign (legitimate) during the testing phase. In contrast to signature-based detection, behavior-based detection is also able to detect the unknown type of malware in addition to malware utilizing encryption, obfuscation, or polymorphism. A significant number of false positives and considerable monitoring time requirement can be seen as the downsides of the method concerned. (Sihwail et al., 2018) The method incorporates a virtual machine (VM) and function call monitoring, information flow tracking, dynamic binary instrumentation, and Windows Application Programming Interface (API) call Graph. Behavior detection method benefits of utilization of traditional machine learning methods, such as Decision Tree (DT), K-Nearest Neighbor (KNN), Naïve Bayes (NB), and Support Vector Machine (SVM) to comprehend the behaviors of running files. (Xiao et al., 2019)

Deep learning is a subset of machine learning utilizing multiple layers of neural networks with the capability to perform better on unstructured data (Mathew et al., 2021). DL has been shown to include various advantages over traditional machine learning in areas such as speech recognition, computer vision, and natural language processing. Deep learning enables computational models to learn high-level features from original data at multiple levels. As a drawback, DL requires more computation time to train and retrain the models, which is a common phase in the malware detection process as new malware types continuously emerge. In contrast, traditional machine learning algorithms are fast but not necessarily accurate enough. (Cakir & Dogdu, 2018) The deep learning model is able to learn complicated feature hierarchies and include steps of the malware detection process into one model, which can be then trained end-to-end with all of the components simultaneously (Kaspersky, 2020).

Deep learning has been adopted for the development of Malware Detection Systems (MDSs) due to its success when utilized in other relevant areas. In the beginning, a single deep learning model was applied to

the whole dataset, which ended up causing problems as the model experience challenges in dealing with increasingly complicated data distribution of the malware samples. A Group of deep learning models has been used in conjunction (ensemble approach) in order to solve the issue, but the utilization of multiple models have ended up in similar problems. Zhong & Gu (2019) presented a multi-level deep learning system for malware detection. The system can manage more complicated data distributions utilizing tree structure in order to provide means for each DL model to learn the unique data distribution for one group of a malware family. The authors demonstrated that their system improves the performance of malware detection systems compared to SVM, decision tree, the single deep learning model, and the ensemble-based approach. The system also provides more precise detection in less time to efficiently identify malware threats. (Zhong & Gu, 2019)

Kolosnjaji et al. (2016) presented a hybrid Deep Learning-based neural network model for the classification of malware system call sequences. Authors combined two convolutional and one recurrent (LSTM) neural network layers into one neural network architecture in order to increase malware classification performance. The malware classification process initiates with a malware zoo, which included open source-based Cuckoo Sandbox, where acquired malware binaries can be executed in a protected environment. Results of the executions are then preprocessed to obtain numerical feature vectors, which are sent to neural networks. Neural networks act as a classifier classifying the malware into one of the predefined malware families. Malware data samples with labels were gathered from Virus Share, Maltrieve, and private collections, which provided a large and diverse number of samples. Authors utilized Tensorflow and Theano frameworks providing GPU utilization when constructing and training the neural networks. The proposed Deep Learning-based hybrid model endures simpler neural network models and, in addition, even more sophisticated and broadly used Hidden Markov Models and Support Vector machines and provided an average accuracy, precision, and recall of over 90 % for most malware families.

7.5 Phishing attacks

Phishing can be counted as one of the most challenging problems in the cyber-world, causing financial worries for industries and individuals, and detecting phishing attacks accurately enough can be difficult. Phishing websites may look similar in appearance compared to equivalent legitimate websites implemented to fool users into believing they are visiting the correct and safe website. (Jain & Gupta, 2017) Though there are several anti-phishing software and techniques for detecting potential phishing attempts in emails and detecting phishing contents on websites, phishers utilize new and hybrid techniques to circumvent the available software and techniques (Basnet et al., 2008). According to Oluwatobi et al. (2015), phishing detection techniques tend to suffer relatively low detection accuracy and may induce an extensive number of false alarms, in particular, if novel and sophisticated phishing approaches have been utilized. Traditional phishing detection techniques utilized, such as the blacklist-based method, is not efficient enough countering these kinds of attacks nowadays due to easier registering of domains making blacklist databases quickly outdated.

Phishing detection techniques can be classified into the following approaches: user awareness and software detection. User awareness includes user training concerning phishing threats in order to lead users into correctly identifying phishing and non-phishing messages and mitigating the threat level. Relying on user training in the mitigating effect of phishing attacks is challenging due to human weaknesses. According to Khonji et al. (2013), end-users failed to detect 29 % of phishing attacks even after training. However, phishing detection techniques are usually evaluated against so-called bulk phishing attacks, which can affect the performance with regards to targeted forms of phishing attacks. Using, e.g., proper simulated phishing platform, organization's Phish-Prone percentage (PPP) indicating how many of their employees are likely to fall for phishing or social engineering scam, could be used as a training method. User training can be an effective method, but human errors still exist, and people are prone to forget their training. Training also requires a significant amount of time, and it is not much appreciated by non-technical users.

Machine learning can be utilized as an effective tool in phishing detection due to the classification problem nature of phishing. Traditional ML classifiers, such as decision trees and random forest, can be considered as effective techniques what comes to computational time and accuracy.

Deep-learning-based methods have been recently proposed in the phishing website detection domain. Adebowale & Hossain (2020) introduced an intelligent phishing detection system (IDPS), which uses the image, frame, and text content of a web page to detect phishing activities by utilizing deep learning methods, such as a convolutional neural network (CNN) and the long short-term memory (LSTM) to build a hybrid classification model. The proposed model was built by training the CNN and LSTM classifiers by using 1m universal resource locators and over 10 000 images. Various types of features have been extracted from websites to predict phishing activities. The knowledge model is used to compare the extracted features to determine whether the websites are phishing, suspicious, or legitimate. Phishing websites are indicated as red, suspicious as yellow, and legitimate as green color. The experimental results showed that the model achieved an accuracy rate of 93.28 % and an average detection time of 25 seconds.

8 Conclusion

In this paper, the authors reviewed the concepts of cybersecurity, cyber threats, cyber-physical systems, and artificial intelligence in critical infrastructure. The critical infrastructure field includes systems, networks, assets, services, and infrastructure essential for the continued operation of everyone from citizens to the country. Examples of these high-importance necessities include banking and business services, digital infrastructure, drinking water supply, energy, health, transport and logistics, etc. It can be argued that cyber-physical systems are the future way to guarantee the operation of these services in the modern world because they offer accessibility and ease of use in a near real-time fashion with continuous automation of tedious and arduous processes. Some of the processes can be improved utilizing artificial intelligence, for example, in the access control service of smart buildings or the energy consumption optimization of the smart grid and the local smart buildings.

The attacks towards CPSs are various, and many different attack vectors were identified, out of which the most concerning ones being adversarial attacks, false data injection attacks, malware attacks, and phishing attacks. These malicious attacks all rely on fooling humans on some level, having the capacity to harm the system itself and the human users. Especially, the malware attacks towards nuclear power plants are detesting. The DoS/DDoS attacks do not attempt to deceive human users as the other mentioned attacks; however, they too are harmful, as the case of Metropolitan (2016) proved. The attack caused financial losses and disgruntlement in the smart building occupants in the Lappeenranta region.

In essence, the defense methods against these attacks focused on the second and fourth attribute of the cyber resilience concept, namely, “Identify and detect” and “Govern and assure.” These attacks can be defended against with machine learning methods, and in the case of phishing attacks, users can be trained to detect some of the attack attempts. The authors recommend utilizing combinations of different ML models and frameworks to mitigate the risks associated with these attacks. For example, having a layered protective structure to first mitigate the DoS/DDoS attacks with trained artificial intelligence model, such as proposed by Pei et al. (2019), and then in conjunction a more optimized ensemble structure introduced in, for example, by Zhong & Gu (2019) could improve protection for the cyber-physical systems. The authors recommend that one uses defensive distillation and defense-GAN in the training of the ensemble models when applicable in order to enhance the defensive capabilities of the algorithms. Unfortunately, there exists no perfect solution to mitigate these threats. The CNN model introduced by Adebowale & Hossain (2020) should be utilized when people governing the CI have an elevated risk of encountering phishing attacks, or those attacks are geared towards the system.

References

Abdullah, SA, Mohd M (2019) Spear Phishing Simulation in Critical Sector: Telecommunications and Defense Sub-sector. International Conference on Cybersecurity (ICoCSec). DOI: 10.1109/ICoCSec47621.2019.8970803.

Abdallah A, Shen XS (2016) Efficient Prevention Technique for False Data Injection Attack in Smart Grid. In: IEEE International Conference on Communications (ICC), pp. 1-6, DOI: 10.1109/ICC.2016.7510610.

Adebowale M A, Lwin K T, Hossain M A (2020). Intelligent Phishing Detection Scheme Using Deep Learning Algorithms. In: Journal of Enterprise Information Management: DOI: 10.1108/JEIM-01-2020-0036.

Ahmed M, Pathan, Al-SK (2020) Blockchain: Can It Be Trusted? In: Computer, 53(4), pp. 31-35, DOI: 10.1109/MC.2019.2922950.

Alazab M, Khan S, Krishnan SSR, Pham QV, Reddy MPK, Gadekallu TR (2020) A Multidirectional LSTM Model for Predicting the Stability of a Smart Grid. *IEEE Access*, *8*, pp.85454-85463.

Albawi S, Mohammed TA, Al-Zawi S (2017) Understanding of a convolutional neural network. In: 2017 International Conference on Engineering and Technology (ICET) (pp. 1-6). IEEE.

Alelyani S, Kumar GR (2018) Overview of Cyberattack on Saudi Organizations. JISCR - Naif Arab University for Security Sciences. DOI: 10.26735/16587790.2018.004.

Allianz (2020). Cyber attacks on critical infrastructure. News & Insights, Expert Risk Articles. http://agcs.allianz.com/news-and-insights/expert-risk-articles/cyber-attacks-on-critical-infrastructure.html. Accessed 4.10.2020

Anderson M, Bartolo A, Pulkit T (2016) Crafting Adversarial Attacks on Recurrent Neural Networks. arXiv:1604.08275v1 [cs.CR].

Athiwaratkun B, Stokes JW (2017), March. Malware classification with LSTM and GRU language models and a character-level CNN. In: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 2482-2486). IEEE.

AtlasVPN (2020) Over 400 Million Malware Infections Detected in Last 30 Days, More Than 10 Million Daily. https://atlasvpn.com/blog/nearly-404-million-malware-infections-detected-in-last-30-days-more-than-10-million-daily. Accessed 22.10.2020

Australian Government Department of Home Affairs (2020) Security coordination: Critical infrastructure resilience. https://www.homeaffairs.gov.au/about-us/our-portfolios/national-security/security-coordination/critical-infrastructure-resilience. Accessed 10.9.2020

Ayad A, Farag HEZ, Youssef A, El-Saadany EF (2018). Detection of False Data Injection Attacks in Smart Grids Using Recurrent Neural Networks. In: IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), pp. 1-5. DOI: 10.1109/ISGT.2018.8403355.

Barredo Arrieta A, Díaz-Rodríguez N, Del Ser J, Bennetot A, Tabik S, Barbado A, Garcia S, Gil-Lopez S, Molina D, Benjamins R, Chatila R, Herrera F (2020) Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. In: Information Fusion, volume 58, 2020, pages 82-115, ISSN 1566-2535, https://doi.org/10.1016/j.inffus.2019.12.012.

Barth (2017) DDoS Attacks Delay Trains, Stymie Transportation Services in Sweden. SC Media – Security News. https://www.scmagazine.com/home/security-news/cybercrime/ddos-attacks-delay-trains-stymie-transportation-services-in-sweden. Accessed 22.9.2020

Basnet R, Mukkamala S, Sung A (2008). Detection of Phishing Attacks: A Machine Learning Approach. In: Prasad B (eds) Soft Computing Applications in Industry. Studies in Fuzziness and Soft Computing, 226. Springer, Berling, Heidelberg. DOI: 10.1007/978-3-540-77465-5_19.

Bencsáth B, Pék G, Buttyán L, Félegyházi M (2012) The Cousins of Stuxnet: Duqu, Flame, and Gauss. Future Internet 2012, 4(4), 971-1003. DOI: 10.3390/fi/4040971.

Berkeley AR, Wallace M (2010) A Framework for Establishing Critical Infrastructure Resilience Goals. Final Report and Recommendations by the Council. National Infrastructure Advisory Council. https://www.dhs.gov/xlibrary/assets/niac/niac-a-framework-for-establishing-critical-infrastructure-resilience-goals-2010-10-19.pdf. Accessed 11.9.2020

Baezner M, Robin P (2017) Hotspot Analysis: Stuxnet. CSS Cyber Defense Project. Risk and Resilience Team, Center of Security Studies, ETH Zurich. https://css.ethz.ch/content/dam/ethz/special-interest/gess/cis/center-for-securities-studies/pdfs/Cyber-Reports-2017-04.pdf. Accessed 21.9.2020

Biasi J (2018) Malware Attacks on Critical Infrastructure Security are Growing. Burns & McDonnel. http://amplifiedperspectives.burnsmcd.com/post/ malware-attacks-on-critical-infrastructure-security-are-growing. Accessed 18.9.2020

Biggio B, Corona I, Maiorca D, Nelson B, Srndic N, Laskov P, Giacinto G, Roli F (2013). Evasion Attacks Against Machine Learning at Test Time. arXiv:1708.06131v1 [cs.CR].

Bossetta M (2018) The Weaponization of Social Media: Spear Phishing and Cyberattacks on Democracy. In: Journal of International Affairs, 71(2), pp. 97-106.

Bugra C, Dogdu E (2018) Malware Classification Using Deep Learning Methods. In: Proceedings of the ACMSE Conference, 10, pp. 1-5. DOI: 10.1145/3190645.3190692.

Business Finland (2016) MARKET OPPORTUNITIES IN THE SMART GRID SECTOR IN FINLAND 2016. https://www.businessfinland.fi/48cd02/globalassets/julkaisut/invest-in-finland/white-paper-smart-grid.pdf. Accessed 30.11.2020

Carlini N, Wagner D (2017) Towards Evaluating the Robustness of Neural Networks. arXiv:1608.04644v2 [cs.CR].

Cambridge dictionary (2020) Cybersecurity. http://dictionary.cambridge.org/us/dictionary/english/cybersecurity. Accessed 17.9.2020

Carnegie Mellon (2015) Computer Security Incident Response Plan. http://cmu.edu/iso/governance/procedures/docs/incidentresponseplan1.0.pdf. Accessed 17.9.2020

Chakraborty A, Alam M, Dey V, Chattopadhyay A, Mukhopadhyay D (2018) Adversarial Attacks and Defences: A Survey. arXiv:1810.00069v1[cs.LG].

Chen P-Y, Yang S, McCann JA, Lin J, Yang X (2015) Detection of False Data Injection Attacks in Smart-Grid Systems. IEEE Communications Magazine, 53(2), pp. 206-213. DOI: 10.1109/MCOM.2015.7045410.

Cheng J, Zhang C, Tang X, Sheng V, Dong Z, Li J (2018) Adaptive DDoS Attack Detection Method Based on Multiple-Kernel Learning. In: Security and Communication Networks, Article ID 5198685. DOI: 10.1155/2018/5198685.

Cisco (2020) What is Cybersecurity? https://www.cisco.com/c/en/us/products/security/what-is-cybersecurity.html. Accessed 17.9.2020

Co KT (2018). Bayesian Optimization for Black-Box Evasion of Machine Learning Systems. Imperial College London, Department of Computing.

Cytomic (2019) The cybercriminal protagonists of 2019: ransomware, phishing and critical infrastructure. http://www.cytomic.ai/trends/protagonists-crybercrime-2019. Accessed 4.10.2020

DeepAI (2019) What is Defensive Distillation? https://deepai.org/machine-learning-glossary-and-terms/defensive-distillation. Accessed 9.10.2019

Esmalifalak M, Liu L, Nguyen N, Zheng R, Han Z (2017) Detecting Stealthy False Data Injection Using Machine Learning in Smart Grid. In: IEEE Systems Journal, 11(3), pp. 1644-1652. DOI: 10.1109/JSYST.2014.2341597.

Ettourney (2016) Resilience and Risk Management. Building Innovation Conference & Expo. https://cdn.ymaws.com/www.nibs.org/resource/resmgr/Conference2016/BI2016_0113_ila_ettouney.pdf. Accessed 18.9.2020

Eur-Lex (2016) ANNEX II: Types of entities for the purposes of point (4) of article 4. Directive (EU) 2016/1148 of the European Parliament and of the Council of 6 July 2016 concerning measures for a high common level of security of network and information systems across the Union. https://eur-lex.europa.eu/eli/dir/2016/1148/oj. Accessed 10.9.2020

European Commission (2016) Shaping Europe's digital future. The Directive on security of network and information systems (NIS Directive). https://ec.europa.eu/digital-single-market/en/news/directive-security-network-and-information-systems-nis-directive. Accessed 10.9.2020

European Commission (2017) Digital Transformation Monitor. Smart Building: Energy Efficiency Application. https://ec.europa.eu/growth/tools-databases/dem/monitor/sites/default/files/DTM_Smart%20building%20-%20energy%20efficiency%20v1.pdf. Accessed 10.11.2020

Flores C, Guasco T, Leon-Acurio J (2017) A Diagnosis of Threat Vulnerability and Risk as IT Related to the Use of Social Media Sites When Utilized by Adolescent Students Enrolled at the Urban Center of Canton Canar. International Conference on Technology Trends, 199-214.

CISA (2020) Critical Infrastructure Sectors. https://www.cisa.gov/critical-infrastructure-sectors. Accessed 10.9.2020

Connecticut State (2020) Critical Infrastructure. Connecticut's Official State Website, division of Emergency Management and Homeland Security. https://portal.ct.gov/DEMHS/Homeland-Security/Critical-Infrastructure. Accessed 10.9.2020

Farmanbar M, Parham K, Arild Ø, Rong C (2019) A Widespread Review of Smart Grids Towards Smart Cities. Energies 2019, 12(23), 4484. DOI: 10.3390/en12234484.

Gartner (2020) Cybersecurity. https://www.gartner.com/en/information-technology/glossary/cyber-security. Accessed 17.9.2020

Goodfellow I, McDaniel P, Papernot N (2018) Making Machine Learning Robust Against Adversarial Inputs. Communications of the ACM, 61(7), 56 – 66.

Griffor E, Greer C, Wollman D, Burns M (2017) Framework for Cyber-Physical Systems: Volum 1, Overview. Special Publication (NIST SP) – 1500-201. DOI: 10.6028/NIST.SP.1500-201.

Elmrabet Z, Kaabouch N, Elghazi H, Elghazi H (2018) Cyber-security in Smart Grid: Survey and Challenges. Computers & Electrical Engineering, 67, pp. 469-482.

Haider S, Akhunzada A, Mustafa I, Patel TB, Fernandez A, Choo K-WR, Iqbal J (2020) A Deep CNN Ensemble Framework for Efficient DDoS Attack Detection in Software Defined Networks. In: IEEE Access, 8, pp. 53972-53983, DOI: 10.1109/ACCESS.2020.2976908.

He Z, Zhang T, Lee RB (2017) Machine Learning Based DDoS Attack Detection from Source Side in Cloud. In: 4th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud), New York, NY, pp. 114-120. DOI: 10.1109/CSCloud.2017.58.

Ibitoye O, Shafiq O, Matrawy A (2019) Analysing Adversarial Attacks Against Deep Learning for Intrusion Detection in IoT Networks. arXiv:1905.05137 [cs.NI].

Jain A K, Gupta B B (2017). Phishing Detection: Analysis of Visual Similarity Based Approaches. Article ID 5421046. DOI: 10.1155/2017/5421046.

Jordan MI, Mitchell TM (2015) Machine Learning: Trends, Perspectives, and Prospects. Science, 349(6245), pp. 255-260.

Jovanovic B (2019) Malware Statistics – You'd Better Get Your Computer Vaccinated. DataProt. http://www.dataprot.net/statistics/malware-statistics. Accessed 22.10.2020

Kaspersky (2020) Machine Learning Methods for Malware Detection. http://media.kaspersky.com/en/enterprise-security/Kaspersky-Lab-Whitepaper-Machine-Learning.pdf. Accessed 23.10.2020

Kaspersky (2020) What is Social Engineering? – Social Engineering Definition. http://kaspersky.com/resource-center/definitions/what-is-social-engineering. Accessed 7.10.2020

Khonji M, Iraqi Y, Jones A (2014). Phishing Detection: A Literature Survey. In: IEEE Communications Surveys & Tutorials. DOI: 10.1109/SURV.2013.032213.00009.

Kolosnjaji B, Zarras A, Webster G, Eckert C (2016) Deep Learning for Classification of Malware System Call Sequences. In: Kang B, Bai Q (eds) AI 2016: Advances in Artificial Intelligence. AI 2016. Lecture Notes in Computer Science, 9992. Springer, Cham. DOI: 10.1007/978-3-319-50127-7_11.

LeCun Y, Bengio Y, Hinton G (2015) Deep learning. nature, 521(7553), 436-444.

Legatiuk D, Smarsly K (2018) 'An abstract approach towards modeling intelligent structural systems', 9th EWSHM 2018, Creative Commons CC-BY-NC licence, viewed 7 December 2020, <https://creativecommons.org/licenses/by-nc/4.0>.

Lehto M, Neittaanmäki, P (eds) Phoenomena in the Cyber World (2015) Cyber Security: Analytics, Technology and Automation. Berlin: Springer.

Limnell J, Majewski K, Salminen M (2014) Kyberturvallisuus. Saarijärvi: Docendo.

Lipton ZC, Berkowitz J, Elkan C (2015) A critical review of recurrent neural networks for sequence learning. arXiv preprint arXiv:1506.00019.

Ma S, Liu Y, Tao G, Lee WC, Zhang X (2019) NIC: Detecting Adversarial Samples with Neural Network Invariant Checking. Network and Distributed Systems Security (NDSS) Symposium 2019, San Diego, CA, USA. DOI: 10.14722/ndss.2019.23415.

MarketsandMarkets (2020) Smart Cities Market worth $820.7 billion by 2025. https://www.marketsandmarkets.com/PressReleases/smart-cities.asp Accessed 30.11.2020

Mathew A, Arul A, Sivakumari S (2021) Deep Learning Techniques: An Overview. In: Advanced Machine Learning Technologies and Applications. DOI: 10.1007/978-981-15-3383-9_54.

Metropolitan (2016). DDoS Attack Halts Heating in Finland Amidst Winter. Metroplitan.fi – News from Finland in English. http://metropolitan.fi/entry/ddos-attack-halts-heating-in-finland-amidst-winter. Accessed 22.9.2020

McCreanor N (2018) Danish Rail Network DSB Hit by Cyber Attack. https://www.itgovernance.eu/blog/en/danish-rail-network-dsb-hit-by-cyber-attack. Accessed 22.9.2020

Miller WB (2014) Classifying and Cataloging Cyber-Security Incidents Within Cyber-Physical Systems. Master's Thesis. School of Technology, Brigham Young University, USA.

Milošević N (2013) History of Malware. arXiv:1302.5392[cs.CR].

Mirkovic J, Reiher P (2004) A Taxonomy of DDoS Attack and DDoS Defense Mechanisms. In: ACM SIGCOMM Computer Communication Review, 34(2). DOI: 10.1145/997150.997156.

Mohammad OA, Youssef T, Ibrahim A (2018) Special Issue "Smart Grid Networks and Energy Cyber Physical Systems". Accessed 12.11.2020 https://www.mdpi.com/journal/sensors/special_issues/ smart_grid_networks.

Myung JW, Hong S (2019). ICS Malware Triton Attack and Countermeasures. International Journal of Emerging Multidisciplinary Research, 3(2). DOI: 10.22662/IJEMR.2019.3.2.0.13.

Nathan S (2018) What Is Cyber Resilience? Why It Is Important? Teceze blog. https://www.teceze.com/what-is-cyber-resilience-why-it-is-important. Accessed 11.9.2020.

NIST (2013) Foundations for Innovation in Cyber-Physical Systems – Workshop Report. Energetics Incorporated. Prepared for National Institute of Standards and Technology. Accessed 12.11.2020 https://www.nist.gov/system/files/documents/el/CPS-WorkshopReport-1-30-13-Final.pdf.

Niu X, Li J, Sun J, Tomsovic K (2018) Dynamic Detection of False Data Injection Attack in Smart Grid using Deep Learning. In: IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), Washington, DC, USA, pp. 1-6. DOI: 10.1109/ISGT.2019.8791598.

Nweke LN (2017). Using the CIA and AAA Models to Explain Cybersecurity Activities. PM World Journal, 6(12).

Obaid HS (2020) DoS and DDoS Attacks at OSI Layers. International Journal of Multidisciplinary Research and Publications, 2(8), pp. 1-9. DOI: 10.5281/zenodo.3610833.

Oluwatobi A A, Amiri I S, Fazeldehkordi E (2015). A Machine-Learning Approach to Phishing Detection and Defense. Elsevier Inc. DOI: 10.1016/C2014-0-03762-8.

OSAC (2018) Ukraine 2018 Crime & Safety Report. Accessed 6.11.2020 http://www.osac.gov/Country/Ukraine.

Paganini P (2018). Massive DDoS Attack Hit the Danish State Rail Operator DSB. https://securityaffairs.co/wordpress/72530/hacking/rail-operator-dsb-ddos.html. Accessed 22.9.2020

Paloaltonetworks (2020) What is Cybersecurity? https://www.paloaltonetworks.com/cyberpedia/what-is-cyber-security. Accessed 17.9.2020

Papernot N, McDaniel P, Goodfellow I, Jha S, Celik ZB, Swami A (2016) Practical Black-Box Attacks against Machine Learning. arXiv: 1602.02697[cs.CR].

Papernot N, McDaniel P, Wu X, Jha S, Swami A (2016b) *Distillation as a defense to adversarial perturbations against Deep Neural Networks*, Ithaca, NY, US, arXiv.org > cs > arXiv:1511.04508.

Papernot N, McDaniel P, Goodfellow I, Jha S, Celik ZB, Swami A (2017) Practical Black-box Attacks Against Deep Learning Systems Using Adversarial Examples. In Proceedings of the ACM Asia Conference on Computer and Communications Security, UAE. New York: ACM Press.

Pawlak A (2020) Adversarial Attacks for Fooling Deep Neural Networks. https://neurosys.com/article/adversarial-attacks-for-fooling-deep-neural-networks. Accessed 31.7.2020

Pei J, Chen Y, Ji W (2019) A DDoS Attack Detection Method Based on Machine Learning. In: Journal of Physics: Conference Series, 1237(3), Series 1237 042040.

Planning for Hazards. Land Use Tool: Critical Infrastructure Protection. Accessed 6.11.2020 http://planningforhazards.com/critical-infrastructure-protection.

Porter E (2019) What is a DDoS Attack and How to Prevent One in 2020. SafetyDetectives. http://www.safetydetectives.com/blog/what-is-a-ddos-attack-and-how-to-prevent-one-in/#what. Accessed 22.9.2020

Probst M (2015) Generative adversarial networks in estimation of distribution algorithms for combinatorial optimization. arXiv preprint arXiv:1509.09235.

Qureshi AS (2018) How to Mitigate DDoS Vulnerabilities in Layers of OSI Model. DZone. http://dzone.com/articles/how-to-mitigate-ddos-vulnerabilities-in-layers-of. Accessed 22.9.2020

Rader MA, Syed S, Rahman M (2015) Exploring Historical and Emerging Phishing Techniques and Mitigating the Associated Security Risks. In: International Journal of Network Security & Its Applications (IJNSA), 5(4).

Rehak D, Senovsky P, Slivkova S (2018) Resilience of Critical Infrastructure Elements and Its Main Factors. Systems 2018, 6(21). DOI: 10.3390/systems6020021.

Reo J (2018) DDoS Attacks on Sweden's Transit System Signal a Significant Threat. Corero – the DDoS Blog. https://www.corero.com/blog/ddos-attacks-on-swedens-transit-system-signal-a-significant-threat. Accessed 22.9.2020

Ren K, Zheng T, Qin Z, Liu X (2020) Adversarial Attacks and Defences in Deep Learning. Engineering, vol. 6, issue 3, pp. 346-360. DOI:10.1016/j.eng.2019.12.012.

Riskviews (2013) Five components of resilience – robustness, redundancy, resourcefulness, response and recovery. Commentary of Risk and ERM. http://riskviews.wordpress.com/2013/01/24/five-components-of-resilience-robustness-redundancy-resourcefulness-response-and-recovery. Accessed 18.9.2020

RSI Security (2019) What Is Cyber Resilience and Why Is It Important? Cybersecurity Solutions. https://blog.rsisecurity.com/what-is-cyber-resilience-and-why-is-it-important. Accessed 11.9.2020

Salmensuu C (2018) NIS directive in the Nordics: Finnkampen in the air? tieto EVRY. https://www.tietoevry.com/en/blog/2018/09/nis-directive-in-the-nordics-finnkampen-in-the-air. Accessed 10.9.2020

Samangouei P, Kabhab M, Chellappa R (2018) Defense-GAN: Protecting Classifiers against Adversarial Attacks using Generative Models. arXiv:1805.06605v2 [cs.CV].

Santos N (2016) BlackEnergy APT Malware. http://community.rsa.com/thread/186012. Accessed 18.9.2020

Sargolzaei A, Yazdani K, Abbaspour A, Crane CD, Dixon WE (2019) Detection and Mitigation of False Data Injection Attacks in Networked Control Systems. IEEE Transactions on Industrial Informatics, 16(6).

Seok I, Kim SJ (2018). Cyber Security for Nuclear Power Plants. In: Gluschke, G., Casin, M., H. & Macori, M. (eds) Cyber Security Policies for Critical Energy Infrastructures in Korea, Institute for Security and Safety GmbH, Germany.

Shahrivari V, Darabi MM, Izadi M. (2020). Phishing Detection Using Machine Learning Techniques. arXiv:2009.11116 [cs.CR].

Short A, Pay TL, Gandhi A (2019) Defending Against Adversarial Examples. Sandia Report, SAND 2019-11748. Sandia National Laboratories.

Sihwail R, Omar K, Ariffin KAZ (2018) A Survey on Malware Analysis Techniques: Static, Dynamic, Hybrid and Memory Analysis. In: International Journal on Advanced Science Engineering and Information Technology, 8(4-2), 1662. DOI: 10.18517/ijaseit.8.4-2.6827.

Szgedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow I, Fergus R (2013). Intriguing Properties of Neural Networks. arXiv preprint arXiv:1312.6199.

TechTerms (2020). SYN Flood Definition. Accessed 6.11.2020 http://www.techterms.com/definition/syn_flood.

Valle C, Saravia F, Allende H, Monge R, Fernández C (2010) Parallel Approach for Ensemble Learning with Locally Coupled Neural Networks. Neural Process Lett 32:277–291 DOI 10.1007/s11063-010-9157-6

Valtioneuvosto (2013) Valtioneuvoston päätös huoltovarmuuden tavoitteista, 857/2013, Helsinki 5.12.2013.

Vorobeychik Y, Kantarcioglu M (2018). Adversarial Machine Learning. Synthesis Lectures of Artificial Intellignece and Machine Learning. Morgan & Claypool, USA.

Wang Y, Chen D, Zhang C, Chen X, Huang B, Cheng X (2019) Wide and Recurrent Neural Networks for Detection of False Data Injection in Smart Grids. In: Biagioni, E., Zheng, Y & Cheng, S. (eds) Wireless Algorithms, Systems, and Applications. Lecture Notes in Computer Science, 11604. Springer, Cham. DOI: 10.1007/978-3-030-23597-0_27.

Westlund D, Wright A (2012) Newsletter of the Northeast Public Power Association, NEPPA. http://www.naylornetwork.com/ppa-nwl/articles/index-v5.asp?aid=163517&issueID=23606. Accessed 21.9.2020

Wiyatno R, Xu A (2018) Maximal Jacobian-based Saliency Map Attack. arXiv:1808.07945v1 [cs.LG].

Xiao F, Lin Z, Sun Y, Ma Y (2019) Malware Detection Based on Deep Learning of Behavior Graphs. In: Mathematical Problems in Engineering, 1, 1-10. DOI: 10.1155/2019/8195395.

Xinghuo Y, Yusheng X (2016) Smart Grids: A Cyber-Physical Systems Perspective. In: Proceedings of the IEEE, 104(5), pp. 1058-1070. DOI: 10.1109/JPROC.2015.2503119.

Zhang J, Li C (2019) Adversarial Examples: Opportunities and Chanllenges. In: IEEE Transactions on Neural Networks and Learning Systems. arXiv: 1809.04790v4 [cs.LG].

Zhang, Y., Liu, Q. and Song, L., 2018. Sentence-state lstm for text representation. arXiv preprint arXiv:1805.02474.