Cyberattacks Against Critical Infrastructure Facilities and Corresponding Countermeasures

Petri Vähäkainu, Martti Lehto, and 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 CI to cyberattacks and technical failures has become a major concern nowadays. Sophisticated and novel cyberattacks, such as adversarial attacks, may deceive physical security controls providing a perpetrator an illicit entry to the smart critical facility. Adversarial attacks can be used to deceive a classifier based on predictive machine learning (ML) that automatically adjusts the heating, ventilation, and air conditioning (HVAC) of a smart building. False data injection attacks have also been used against smart grids. Traditional and widespread cyberattacks using malicious code can cause remarkable physical damage, such as blackouts and disruptions in power production, as attack vectors to manipulate critical infrastructure. To detect incoming attacks and mitigate the performance of those attacks, we introduce defensive mechanisms to provide additional detection and defense capabilities to enhance the inadequate protection of a smart critical facility from external.

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. (Colorado, 2020) 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. (Janita, 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 (El Mbrabet 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 (NortonSantos, 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 Sect. 2, the authors explain critical infrastructure and resilience concepts in more detail. Section 3 addresses the cyber-physical system and presents some relevant CPS sectors these days. Section 4 defines cybersecurity and explains the intertwined concepts of cybersecurity, threat, vulnerability, and risks in more detail. Section 5 describes artificial intelligence and machine learning and discusses the most common and sophisticated deep learning methods. In Sect. 6, we showcase well-known cyberattacks utilized against critical infrastructure facilities, such as smart buildings. Section 7 focuses on reviewing the defense mechanisms utilized in combating cyberattacks towards critical infrastructure facilities. Lastly, Sect. 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, 2020). Critical infrastructure provides services crucial for everyday life, e.g., banking, communication, energy, food, finance, health, transport, and water (Table 1). 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. (GOV-AU, 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 (Valtioneuvosto, 2013):

1. Energy production, transmission and distribution systems,

2. Information and communication systems, networks and services,
3. Financial services,
4. Transport and logistics,
5. Water supply,
6. Infrastructure construction and maintenance,
7. Waste management in special situations.

European Parliament adopted the directive on security of network and information systems (NIST, 2013) 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. (EC, 2016)

The core idea of the NIS directive is that relevant service operators and digital service providers must ensure the security of their information infrastructure is secure, ensure business continuity in case of adverse information security disruptions, and report any substantial information security breaches to authorities. (Salmensuu, 2018) According to (EU, 2016, ANNEX-II), NIS sectors are the following:

1. Energy,
2. Transport,
3. Banking,
4. Financial market infrastructures,
5. Health sector,
6. Drinking water supply and distribution,
7. Digital infrastructure.

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 (CISA, 2020)

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.

Table 1. Critical infrastructure sectors in Finland (Valtioneuvosto, 2013), EU (EU, 2016) and United States (CISA, 2020)

Finland	EU	United States
Energy production	Energy	Chemicals
Transmission and distribution systems	Transport	Business
Information and communication systems, networks and services	Banking	Communications
Financial services	Financial market infrastructures	Critical infrastructure manufacturing
Transport and logistics	Health sector	Damns
Water supply	Drinking water supply and distribution	Defense industry
Infrastructure, construction and maintenance	Digital infrastructure	Emergency services
Waste management in special situations		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 and Wallace (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, 2019).

Cyber resilience consists of four elements (Nathan, 2020), which are the following:

1. Manage and protect,
2. Identify and detect,
3. Respond and recover,
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.” Rajkumar et al. (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.

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 number 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 (Markets, 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.

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 (EC, 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) (Yu & Xue, 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.” (Cambridge, 2020) Gartner Glossary defined cybersecurity as the combination of people, policies, processes, and technologies employed by an organization to protect its cyber assets (Gartner, 2020). Cybersecurity can also be thought of as a practice of protecting systems, networks, and programs from digital attacks (Cisco, 2020). 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.” (Paloalto, 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 alpha-numeric 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).” (Ettouney & Alampalli, 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 (Limnéll et al., 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, 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 $f(x)$: $R^n \rightarrow R^m$, where *f(x)* is the function to model, R^n represents the real multidimensional input values, and R^m 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 et al. (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 et al., 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 et al. (2019) used DTs to interpret the predictions of a Convolutional Neural Network (CNN) model, thus explaining the model's behavior. 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 et al., 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., 2020).

6 Cyberattacks Against Critical Infrastructure Facilities

This section introduces and discusses well-known cyberattacks, such as adversarial, DoS and DDoS, False data injection (FDI), malware and phishing attacks from a critical infrastructure perspective and illustrates utilization of attacks mentioned with real world case-examples.

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., 2017). 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 et al., 2018).

Jacobian-based saliency map algorithm (JSMA) was presented by Papernot et al. (2016a) 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∞ 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. (2016a) 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 (2017) has been presenting C&W attack, 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 (Christensson, 2013). 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,
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. (Janita, 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 El Mrabet 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. (NortonSantos, 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, 2020b)

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 and 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. (Oh Il & 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

This section focuses on reviewing possible detection and prevention mechanisms that could be utilized in combating previously mentioned cyberattacks threatening critical infrastructure facilities.

7.1 Defending Against 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, 2019) Defensive distillation can be evaded by the black-box approach (Papernot et al., 2016a) and also with optimization attacks (Szegedy et al., 2013). Carlini and 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 Defending Against 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 Defending Against 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.

Addallah and Shen (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 Pathan (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 Defending Against Malware Attacks

Malware infections have been significantly increasing in the past years, and large quantities of malware are automatically created each day. According to Anton (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, 17 million malware programs are registered montly, and up to 560 000 new pieces of malware are detected each day (Jovanovic, 2021). 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 malwares, 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., 2019) 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., 2019).

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 and 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 Defending Against 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 et al. (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 Janita (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 and 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 et al. (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

Abdallah A, Shen XS (2016). Efficient prevention technique for false data injection attack in smart grid. In *2016 IEEE International Conference on Communications (ICC)*, pp. 1-6. IEEE. Doi: 10.1109/ICC.2016.7510610.

Abdullah SA, Mohd M (2019). Spear phishing simulation in critical sector: telecommunications and defense sub-sector. In *2019 International Conference on Cybersecurity (ICoCSec)*, pp. 26-31. IEEE. Doi: 10.1109/ICoCSec47621.2019.8970803.

Adebowale MA, Lwin KT, Hossain MA (2020). Intelligent phishing detection scheme using deep learning algorithms. *Journal of Enterprise Information Management*. Doi: 10.1108/JEIM-01-2020-0036. Published online.

Ahmed M, Pathan ASK (2020). Blockchain: Can it be trusted? *Computer*, 53(4), 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*, 85454-85463. Doi: 10.1109/ACCESS.2020.2991067.

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. Doi: 10.1109/ICEngTechnol.2017.8308186.

Alelyani S, Kumar H (2018). Overview of cyberattack on Saudi organizations. *Journal of Information Security & Cybercrimes Research*, 1(1), 32-39. Doi: 10.26735/16587790.2018.004.

Allianz (2020). Cyber attacks on critical infrastructure. Allianz Global Corporate & Specialty (AGCS), http://agcs.allianz.com/news-and-insights/expert-risk-articles/cyber-attacks-on-critical-infrastructure.html. Accessed 4 October 2020.

Anderson M, Bartolo A, Tandon P (2016). Crafting adversarial attacks on recurrent neural networks. https://stanford.edu/~bartolo/assets/crafting-rnn-attacks.pdf. Accessed 29.6.2021.

Anton P (2020). Over 400 million malware infections detected in last 30 days, more than 10 million daily. AtlasVPN, https://atlasvpn.com/blog/nearly-404-million-malware-infections-detected-in-last-30-days-more-than-10-million-daily.

Athiwaratkun B, Stokes JW (2017). 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. Doi: 10.1109/ICASSP.2017.7952603.

Ayad A, Farag HEZ, Youssef A, El-Saadany EF (2018). Detection of false data injection attacks in smart grids using Recurrent Neural Networks. In *2018 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1-5. Doi: 10.1109/ISGT.2018.8403355.

Baezner M, Robin P (2017). Hotspot analysis: Stuxnet. CSS Cyber Defense Project, Center for 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.

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, 58, 82-115, DOI: 10.1016/j.inffus.2019.12.012.

Barth B (2017). DDoS attacks delay trains, stymie transportation services in Sweden. SC. https://www.scmagazine.com/home/security-news/cybercrime/ddos-attacks-delay-trains-stymie-transportation-services-in-sweden.

Basnet R, Mukkamala S, Sung AH (2008). Detection of phishing attacks: A machine learning approach. In: Prasad B (ed.) *Soft Computing Applications in Industry*. Studies in Fuzziness and Soft Computing, 226. Springer, Berling. 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*, 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, Washington, DC, https://www.dhs.gov/xlibrary/assets/niac/niac-a-framework-for-establishing-critical-infrastructure-resilience-goals-2010-10-19.pdf.

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.

Biggio B, Corona I, Maiorca D, Nelson B, Srndic N, Laskov P, Giacinto G, Roli F (2017). Evasion attacks against machine learning at test time. arXiv:1708.06131v1.

Bossetta M (2018). The weaponization of social media: Spear phishing and cyberattacks on democracy. *Journal of International Affairs*, 71(1.5), 97-106.

BusinessFinland (2016). Market opportunities in the smart grid sector in Finland 2016. Business Finland, https://www.businessfinland.fi/48cd02/globalassets/julkaisut/invest-in-finland/white-paper-smart-grid.pdf.

Cakir B, Dogdu E (2018). Malware classification using deep learning methods. In *ACMSE '18: Proceedings of the ACMSE 2018 Conference*, Article 10, pp. 1-5. Doi: 10.1145/3190645.3190692.

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

Carlini N, Wagner D (2017). Towards evaluating the robustness of neural networks. arXiv:1608.04644v2.

Carnegie (2015). Computer security incident response plan. Carnegie Mellon, http://cmu.edu/iso/governance/procedures/docs/incidentresponseplan1.0.pdf.

Chakraborty A, Alam M, Dey V, Chattopadhyay A, Mukhopadhyay D (2018). Adversarial attacks and defences: A survey. arXiv:1810.00069v1.

Chen PY, Yang S, McCann JA, Lin J, Yang X (2015). Detection of false data injection attacks in smart-grid systems. *IEEE Communications Magazine*, 53(2), 206-213. Doi: 10.1109/MCOM.2015.7045410.

Cheng J, Zhang C, Tang X, Sheng VS, Dong Z, Li J (2018). Adaptive DDoS attack detection method based on multiple-kernel learning. *Security and Communication Networks*, 2018, Article 5198685. Doi: 10.1155/2018/5198685.

Christensson P (2013). SYN flood definition. TechTerms http://www.techterms.com/definition/syn_flood.

CISA (2020). Critical infrastructure sectors. Cybersecurity & Infrastructure Security Agency, https://www.cisa.gov/critical-infrastructure-sectors. Accessed 10 September 2020.

Cisco (2020). What is cybersecurity? Cisco Systems, San Jose, CA, https://www.cisco.com/c/en/us/products/security/what-is-cybersecurity.html. Accessed 17 September 2020.

Co KT (2017). Bayesian optimization for black-box evasion of machine learning systems. Master's thesis, Imperial College London.

Colorado (2020). Critical infrastructure protection. Planning for Hazards: Land Use Solutions for Colorado, http://planningforhazards.com/critical-infrastructure-protection. Accessed 6.11.2020.

Connecticut (2020). Critical Infrastructure. Connecticut State, Division of Emergency Management and Homeland Security, https://portal.ct.gov/DEMHS/Homeland-Security/Critical-Infrastructure. Accessed 10 September 2020.

Cytomic (2019). The cybercriminal protagonists of 2019: Ransomware, phishing and critical infrastructure. Cytomic, https://www.cytomic.ai/trends/protagonists-cybercrime-2019/.

DeepAI (2019). What is defensive distillation? DeepAI, https://deepai.org/machine-learning-glossary-and-terms/defensive-distillation. Accessed 9 October 2019.

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

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

Esmalifalak M, Liu L, Nguyen N, Zheng R, Han Z (2017). Detecting stealthy false data injection using machine learning in smart grid. *IEEE Systems Journal*, 11(3), 1644-1652. Doi: 10.1109/JSYST.2014.2341597.

Ettouney MM, Alampalli S (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

EU (2016). 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. The European Parliament and the Council of the European Union.

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

Farmanbar M, Parham K, Arild Ø, Rong C (2019). A widespread review of smart grids towards smart cities. *Energies*, 12(23), 4484. Doi: 10.3390/en12234484.

Flores C, Flores C, Guasco T, León-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. In *Technology Trends: Proceedings of the Third International Conference, CITT 2017*, pp. 199-214. Springer, Cham.

Gartner (2020). Cybersecurity. Gartner Glossary, https://www.gartner.com/en/information-technology/glossary/cybersecurity. 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.

GOV-AU (2020). Critical infrastructure resilience. Australian Government, https://www.homeaffairs.gov.au/about-us/our-portfolios/national-security/security-coordination/critical-infrastructure-resilience. Accessed 10 September 2020.

Griffor ER, Greer C, Wollman DA, Burns MJ (2017). Framework for cyber-physical systems: Volume 1, overview. NIST Special Publication 1500-201, National Institute of Standards and Technology. Doi: 10.6028/NIST.SP.1500-201.

El Mrabet Z, Kaabouch N, El Ghazi H, El Ghazi H (2018). Cyber-security in smart grid: Survey and challenges. *Computers and Electrical Engineering*, 67, 469-482.

Haider S, Akhunzada A, Mustafa I, Patel TB, Fernandez A, Choo KKR, Iqbal J (2020). A deep CNN ensemble framework for efficient DDoS attack detection in software defined networks. *IEEE Access*, 8, 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 *2017 IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud)*, pp. 114-120. IEEE. Doi: 10.1109/CSCloud.2017.58.

Ibitoye O, Shafiq O, Matrawy A (2019). Analyzing adversarial attacks against deep learning for intrusion detection in IoT networks. arXiv:1905.05137.

Jain AK, Gupta BB (2017). Phishing detection: Analysis of visual similarity based approaches. *Security and Communication Networks*, 2017, Article 5421046. DOI: 10.1155/2017/5421046.

Janita (2016). DDoS attack halts heating in Finland amidst winter. Metropolitan.fi, http://metropolitan.fi/entry/ddos-attack-halts-heating-in-finland-amidst-winter.

Jordan MI, Mitchell TM (2015). Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245), 255-260.

Jovanovic B (2021) A Not-So-Common Cold: Malware Statistics in 2021. DataProt. https://dataprot.net/statistics/malware-statistics. Accessed 29.6.2021.

Kaspersky (2020a). Machine learning methods for malware detection. http://media.kaspersky.com/en/enterprise-security/Kaspersky-Lab-Whitepaper-Machine-Learning.pdf. Accessed 23.10.2020

Kaspersky (2020b). What is Social Engineering? Kaspersky, http://kaspersky.com/resource-center/definitions/what-is-social-engineering. Accessed 7 October 2020.

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: Proceedings of the 29th Australasian Joint Conference*, pp 137-149. Lecture Notes in Computer Science, 9992. Springer, Cham. Doi: 10.1007/978-3-319-50127-7_11.

Khonji M, Iraqi Y, Jones A (2013). Phishing detection: A literature survey. *IEEE Communications Surveys & Tutorials*, 15(4), 2091-2121. 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. In *9th European Workshop on Structural Health Monitoring*. NDT.net.

Lehto M (2015). Phenomena in the cyber world. In Lehto M, Neittaanmäki P (eds.), *Cyber Security: Analytics, Technology and Automation*, pp. 3-29. Springer, Berlin.

Limnéll J, Majewski K, Salminen M (2014). *Kyberturvallisuus*. Docendo.

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

Ma S, Liu Y, Tao G, Lee WC, Zhang X (2019). NIC: Detecting adversarial samples with neural network invariant checking. In *26th Annual Network and Distributed System Security Symposium, NDSS 2019*. The Internet Society. Doi: 10.14722/ndss.2019.23415.

Markets (2020). Smart cities market worth $820.7 billion by 2025. Exclusive Report by MarketsandMarketsTM, https://www.marketsandmarkets.com/PressReleases/smart-cities.asp Accessed 30 November 2020.

Mathew A, Amudha P, Sivakumari S (2021). Deep learning techniques: An overview. In: *Advanced Machine Learning Technologies and Applications: Proceedings of AMLTA 2020*, pp 599-608. Springer, Singapore. DOI: 10.1007/978-981-15-3383-9_54.

McCreanor N (2018). Danish rail network DSB hit by cyber attack. IT governance, https://www.itgovernance.eu/blog/en/danish-rail-network-dsb-hit-by-cyber-attack. Accessed 22.9.2020.

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

Miller WB (2014). Classifying and cataloging cyber-security incidents within cyber-physical systems. Master's thesis, Brigham Young University.

Milošević N (2013). History of malware. arXiv:1302.5392.

Mirkovic J, Reiher P (2004). A taxonomy of DDoS attack and DDoS defense mechanisms. *ACM SIGCOMM Computer Communication Review*, 34(2), 39-53. DOI: 10.1145/997150.997156.

Mohammad OA, Youssef T, Ibrahim A, eds. (2018). Special issue "Smart Grid Networks and Energy Cyber Physical Systems". Issue information, MDPI, 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), 13-17. DOI: 10.22662/IJEMR.2019.3.2.0.13.

Nathan S (2020). What is cyber resilience? Why it is important? Teceze, 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. National Institute of Standards and Technology, https://www.nist.gov/system/files/documents/el/CPS-WorkshopReport-1-30-13-Final.pdf.

Niu X, Li J, Sun J, Tomsovic K (2019). Dynamic detection of false data injection attack in smart grid using deep learning. In *2019 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1-6. IEEE. Doi: 10.1109/ISGT.2019.8791598.

NortonSantos (2016). Blackenergy APT malware. RSA Link, http://community.rsa.com/thread/186012. Accessed 18.9.2020.

Nweke LO (2017). Using the CIA and AAA models to explain cybersecurity activities. *PM World Journal*, 6(12).

Obaid HS, Abeed EH (2020). DoS and DDoS attacks at OSI layers. *International Journal of Multidisciplinary Research and Publications*, 2(8), 1-9. DOI: 10.5281/zenodo.3610833.

Oh IS, Kim SJ (2018). Cyber security policies for critical energy infrastructures in Korea focusing on cyber security for nuclear power plants. In Gluschke G., Casin, MH, Macori M (eds.), *Cyber*

Security Policies and Critical Infrastructure Protection, pp. 77-95. Institute for Security and Safety, Potsdam.

OSAC (2018). Ukraine 2018 crime & safety report. Overseas Security Advisory Council, U.S. Department of State, Washington, DC, http://www.osac.gov/Country/Ukraine.

Paganini P (2018). Massive DDoS attack hit the Danish state rail operator DSB. Security Affairs, https://securityaffairs.co/wordpress/72530/hacking/rail-operator-dsb-ddos.html.

Paloalto (2020). What is cybersecurity? Palo Alto Networks, https://www.paloaltonetworks.com/cyberpedia/what-is-cyber-security. Accessed 17 September 2020.

Papernot N, McDaniel P, Goodfellow I, Jha S, Celik ZB, Swami A (2016a). Practical black-box attacks against machine learning. arXiv:1602.02697.

Papernot N, McDaniel P, Wu X, Jha S, Swami A (2016b). Distillation as a defense to adversarial perturbations against deep neural networks. arXiv:1511.04508.

Papernot N, McDaniel P, Goodfellow I, Jha S, Celik ZB, Swami A (2017). Practical black-box attacks against machine learning. In *ASIA CCS '17: Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, pp. 506-519. ACM, New York.

Pawlak A (2020). Adversarial attacks for fooling deep neural networks. NeuroSYS, https://neurosys.com/article/adversarial-attacks-for-fooling-deep-neural-networks.

Pei J, Chen Y, Ji W (2019). A DDoS attack detection method based on machine learning. *Journal of Physics: Conference Series*, 1237(3), 032040.

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: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, Rahman SM (2013). Exploring historical and emerging phishing techniques and mitigating the associated security risks. *International Journal of Network Security & Its Applications*, 5(4), 23-41.

Rehak D, Senovsky P, Slivkova S (2018). Resilience of critical infrastructure elements and its main factors. *Systems*, 6(2), 21. Doi: 10.3390/systems6020021.

Reo J (2018). DDoS attacks on Sweden's transit system signal a significant threat. Corero, https://www.corero.com/blog/ddos-attacks-on-swedens-transit-system-signal-a-significant-threat/.

Ren K, Zheng T, Qin Z, Liu X (2020). Adversarial attacks and defenses in deep learning. *Engineering*, 6(3), 346-360. Doi: 10.1016/j.eng.2019.12.012.

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

RSI (2019). What is cyber resilience and why is it important? RSI Security, 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? TietoEVRY, https://www.tietoevry.com/en/blog/2018/09/nis-directive-in-the-nordics-finnkampen-in-the-air. Accessed 10.9.2020

Samangouei P, Kabkab M, Chellappa R (2018). Defense-GAN: Protecting classifiers against adversarial attacks using generative models. arXiv:1805.06605v2.

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), 4281-4292.

Shahrivari V, Darabi MM, Izadi M. (2020). Phishing detection using machine learning techniques. arXiv:2009.11116.

Short A, La Pay T, Gandhi A (2019). Defending against adversarial examples. Sandia report, SAND 2019-11748, Sandia National Laboratories, Albuquerque, NM.

Sihwail R, Omar K, Ariffin KAZ (2018). A survey on malware analysis techniques: Static, dynamic, hybrid and memory analysis. *International Journal on Advanced Science, Engineering and Information Technology*, 8(4-2), 1662-1671. Doi: 10.18517/ijaseit.8.4-2.6827.

Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow I, Fergus R (2013). Intriguing properties of neural networks. arXiv:1312.6199.

Valle C, Saravia F, Allende H, Monge R, Fernández C (2010). Parallel approach for ensemble learning with locally coupled neural networks. Neural Processing Letters, 32, 277–291. Doi 10.1007/s11063-010-9157-6.

Valtioneuvosto (2013). Valtioneuvoston päätös huoltovarmuuden tavoitteista. Säädös 857/2013, Oikeusministeriö.

Vorobeychik Y, Kantarcioglu M (2018). *Adversarial machine learning*. Morgan & Claypool.

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, pp. 335-345. Springer, Cham. Doi: 10.1007/978-3-030-23597-0_27.

Westlund D, Wright A (2012). Duqu, son of Stuxnet, increases pressure for cyber security at all utilities. Newsletter of the Northeast Public Power Association, http://www.naylornetwork.com/ppa-nwl/articles/index-v5.asp?aid=163517&issueID=23606.

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

Xiao F, Lin Z, Sun Y, Ma Y (2019). Malware detection based on deep learning of behavior graphs. *Mathematical Problems in Engineering*, 2019, Article 8195395, 10 pp. Doi: 10.1155/2019/8195395.

Yu X, Xue Y (2016). Smart grids: A cyber-physical systems perspective. *Proceedings of the IEEE*, 104(5), 1058-1070. Doi: 10.1109/JPROC.2015.2503119.

Zhang Q, Yang Y, Ma H, Wu YN (2019). Interpreting CNNs via decision trees. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6254-6263. IEEE. Doi: 10.1109/CVPR.2019.00642.

Zhang Y, Liu Q, Song L (2018). Sentence-state LSTM for text representation. arXiv:1805.02474.

Zhong W, Gu F (2019). A multi-level deep learning system for malware detection. *Expert Systems with Applications*, 133, 151-162.