Penetrating the Power Grid: Realistic Adversarial Attacks on Smart Grid Intrusion Detection Systems

Nelson Makau Mutua[1][0000−0001−6079−711X], Simin Nadjm-Tehrani[2][0000−0002−1485−0802], and Petr Matoušek[1][0000−0003−4589−2041]

[1] Brno University of Technology, Bozetechova 1, 612 66 Brno, Czech Republic
{imutua,matousp}@fit.vutbr.cz
[2] Linköping University, SE-581 83 Linköping, Sweden
simin.nadjm-tehrani@liu.se

Abstract. The widespread adoption and use of Machine Learning-Based Intrusion Detection Systems (ML-IDS) has increased the flexibility and efficiency of automated cyber attack detection in smart grid systems. However, the introduction of such IDSes has created a new attack vector against the learning models commonly known as adversarial attacks. Such attacks could have serious consequences in smart grid systems, as adversaries can evade detection by the IDS. This could lead to delayed attack detection. From the existing literature, a lot of research proposes threat models that are inappropriate for generating realistic adversarial attacks. In this research, we model realistic adversarial attacks with a focus on real attacker capabilities and circumstances required by attackers to launch feasible and successful adversarial attacks. We demonstrate how adversarial learning can be used to target ML models by using the Fast Gradient Sign Method (FGSM) and Jacobian-based Saliency Map Attack (JSMA). A power system dataset generated from a smart grid testbed was used for testing the models. Overall, the classification performance of three widely used classifiers Random Forest, XGBoost and Naive Bayes decreased when adversarial samples were present. The outcomes of this paper are useful for helping researchers model realistic scenarios to avoid dealing with hypothetical problems.

Keywords: Intrusion Detection Systems· Adversarial Attacks· Critical Infrastructure· Machine Learning· Smart Grid Systems.

1 Introduction

Smart electrical grids play a critical role in the digital age of hyper-connected Critical Infrastructures (CIs), offering benefits such as more effective grid resilience, efficient energy distribution, and smart load and response management [19]. The adoption of technology enablers, such as Machine Learning (ML), the Internet of Things (IoT), 5G and Artificial Intelligence (AI), plays a significant role in the life cycle of smart grids. However, this technological advancement

raises severe cybersecurity issues that can result in disastrous consequences, especially in the energy domain. Given the significance of these systems, they have become a desirable target for attackers. By the fact that these systems control physical processes, cyber-attacks may have far-reaching effects on the environment, in which they operate and their users [4].

Multi-step attacks and Advanced Persistent Threats (APTs) against CIs, like the smart electrical grid, can cause service failures, financial losses, and even tragic accidents. Examples of APT [4] campaigns include Industroyer, SolarWinds (Sunburst), Hafnium, and the Lazarus Group. Industroyer caused a widespread blackout in Ukraine in 2015. The NotPetya ransomware caused significant financial damage for various energy-related organizations, making it a notable cybersecurity incident. A more recent CI attack was reported in Denmark in May 2023 where attackers compromised 22 energy organizations in the largest coordinated attack against Denmark's CI [32]. To launch the attacks, hackers exploited multiple vulnerabilities in the firewall for initial access, executing code and gaining complete control over the impacted systems. The attackers successfully compromised 11 energy organizations by executing commands on the vulnerable firewall to obtain device configurations and usernames and thus access to the CI behind it. Security concerns about such systems have become a global issue. Developing robust, safe, and efficient techniques to identify and protect against cyber attacks in smart grid networks is critical.

Although various security methods exist for traditional IT systems, integrating them into smart grid networks is difficult because the monitoring devices have limited resources and do not support modern security measures. As a result, alternative security measures such as passive security monitoring are more promising. This has resulted in a significant rise in research into more tailor-made IDSes that monitor network or sensor data to detect attacks and anomalies that could disrupt the operation of CIs [20]. Due to the efficiency of IDSes in detecting attacks, there has been a significant growth of integration with ML. However, the introduction of such systems has created a new attack vector; trained models may also be vulnerable to attacks. Adversarial Machine Learning (AdvML) refers to deploying attacks against ML systems. Small perturbations can be applied automatically to unseen data points that can result the model crossing a decision boundary and classifying malicious data as normal. Consequently, the effectiveness of the model can be reduced.

The existence of such dynamics implies that CI such as smart grid systems that use ML-IDSes may be exposed to cyber attacks. AdvML can be used to manipulate data from the Intelligent Electronic Devices (IEDs) that are responsible for switching the circuit breakers or other devices by introducing perturbations that cause malicious data to be classed as benign, hence circumventing the IDS. This could result in delayed attack detection, information leaks, financial losses, and even casualties. As ML-based detection methods grow more prevalent, attackers may have a stronger motive to target them. As a result, they require extensive evaluation against AdvML attacks.

1.1 Motivation and Contribution

Our research is motivated by the recognition that many research papers design, develop, and evaluate IDS in adversarial settings without considering the realism of the proposed attacks or explaining how they can be launched in reality. Most of the proposed research work assumes a threat model and proceed to analyze the effects of the attack with none or insufficient considerations about the feasibility of the considered perturbation. Moreover, some general techniques are applied to generate adversarial attacks to manipulate the network features in a way that is inconsistent with actual network traffic [3].

For instance, some researchers assume adversaries with full knowledge of the target system [15], while others suppose that an attacker can perform an unlimited number of trials against the Network Intrusion Detection Systems (NIDS) without detection [30]. Although investigating the effectiveness of adversarial attacks against any ML is an important goal for creating more robust detectors, cybersecurity scenarios should always deal with realistic issues and adversaries. Failing to do so could misinform defenders to allocate resources against false cases or hypothetical problems, potentially diverting attention from more critical issues. The abundance of research on adversarial attacks might inadvertently give the impression that any ML-IDS is an unreliable defensive system, contrary to the actual scenario. Additionally, deceiving an ML model is not guaranteed to be a successful cyber-attack in a real communication network.

Therefore, this paper proposes a realistic approach to modeling adversarial attacks against ML-IDS for smart grid communication by identifying the capabilities and conditions that are necessary for the attacker to carry out such attacks. More importantly, this research recreates a realistic attack model and assumptions as well as a realistic dataset collected from a power system testbed. The contributions of this paper are summarised as follows:

- An in-depth analysis of the feasibility constraints necessary for generating valid adversarial perturbations of data used as input to an ML-IDS while maintaining the underlying logic of the network attack.
- Generating evasion attacks for smart grid network communication capable of evading ML-IDS detection with limited knowledge of the target NIDS.
- Demonstrate the effectiveness of the evasion attack on ML-IDS.

2 Background and Related Work

This section first outlines the fundamental principles of adversarial machine learning. Then, we discuss related work that has employed adversarial evasion techniques to illustrate their effectiveness in evading and reducing the performance of IDS models. Lastly, we describe the network traffic constraints that should be maintained to generate valid adversarial flow.

2.1 Adversarial Machine Learning

Adversarial attacks involve the application of small and undetectable alterations to an ML detector [29]. The modified samples should only differ minimally from their initial form, while still maintaining the basic malicious logic without triggering other detection methods. In this research, we focus on evasion attacks. To perform evasion attacks, the adversary manipulates the inputs to deceive the model and induce misclassification decisions. There are several approaches to generating adversarial samples. The approaches differ in complexity, speed of generation, and performance. A simple method for creating such samples is to manually change the input data points. Manually perturbing huge datasets is time-consuming and may provide inaccurate results. More sophisticated approaches include automatically analyzing and identifying features that best discriminate between target values.

Goodfellow et al. [12] and Papernot et al. [28] introduced the Fast Gradient Sign Method (FGSM) and Jacobian-based Saliency Map Attack (JSMA) as popular methods for creating perturbed samples automatically. Both techniques rely on the concept, that when adding small perturbations (δ) to the original sample *(X)*, the resulting sample *(X*)* can exhibit adversarial characteristics (X* = X + δ) such that X^* will be classified differently by the targeted model.

2.2 Fast Gradient Sign Method (FGSM)

FGSM method for creating adversarial instances is based on the gradient sign method with back propagation. It is an untargeted attack approach used to obtain max-norm constrained perturbation (η) expressed in Eq. 1. Here (θ) represents the model parameter, x is the input vector to the model, y is the associated label of the input and J(θ,x,y) is the cost function. FGSM generates perturbation samples with a small noise parameter ϵ [12] .

$$adv\_x = x + \epsilon * \text{sign}\left(\nabla_x J(\theta, x, y)\right) \tag{1}$$

2.3 Jacobian-Based Saliency Map Attack (JSMA)

On the other hand, the JSMA approach is based on the Jacobian matrix and seeks to calculate the forward derivative of the cost function *f(x)*. The Jacobian of the overall neural network function F to the input X is calculated as follows:

$$JF = \frac{\partial F(X)}{\partial X} \tag{2}$$

Unlike the FGSM, JSMA operates differently from other adversarial attacks by leveraging saliency maps. These maps visually represent the prediction process of a classification model for each pixel, illustrating how each pixel influences the model prediction of a specific class. JSMA, like other adversarial attacks also has advantages and disadvantages. One advantage of using JSMA is its ability to make small perturbations while maintaining high success rates. These minimal

changes make it easier to control the intensity of the attack within a specific ML-IDS. However, JSMA is more computationally intensive than FGSM [28].

2.4 Major Adversarial Attacks Against the NIDS

This section reviews previous research that used adversarial evasion techniques to reduce the performance of ML-IDS models. The existing literature identifies detection methods as vulnerable to generic evasive adversarial attacks, which are considered significant threats. However, the previous research failed to evaluate the effectiveness of generated adversarial traffic for real-world attacks.

Warzyński and Kołaczek [37] demonstrated that an FGSM attack completely degraded a Deep Neural Networks (DNN) binary classifier on the NSL-KDD[3] dataset [35]. They confirmed that the FGSM attack, originally created for image recognition, can also be applied to network traffic. Clements et al. [8] evaluated the resilience of Kitsune, a lightweight IDS for Internet of Things (IoT) networks, to FGSM attacks using the Mirai[4] dataset. Wang [36] discovered that FGSM attacks achieve various degrees of success and use different feature patterns. The author suggested that perturbing specific features may increase the vulnerability of IDS to adversarial traffic. However, the study did not analyze how these features had been manipulated to verify whether the perturbations resulted in consistent traffic instances.

Peng et al. [30] demonstrated a drop in the performance of DNN, Support Vector Machine (SVM), Random Forest (RF), and Logistic Regression (LR) classifiers against Momentum Iterative Fast Gradient Sign Method (MI-FGSM) attacks over the NSL-KDD dataset. Ibitoye et al. [17] compared the performance of Self-Normalizing Neural Networks (SNN) and DNNs under the FGSM using the BoT-IoT dataset[5]. The authors concluded that while DNNs outperformed SNNs in the accuracy rate, the SNNs were more resilient to adversarial attacks.

Asimopoulos et al. [6] presented an AI powered IDS for IEC 60870-5-104 protocol. In their research, the authors utilize four ML methods: (a) Decision Tree, (b) RF, (c) eXtreme Gradient Boosting (XGBoost), and (d) Multilayer Perceptron (MLP) to test the model. The authors investigated how adversarial attacks could affect the detection performance of IDS using the FGSM and a Conditional Tabular Generative Adversarial Network (CTGAN) adversarial attack generator. The performance of the tested models Decision Trees (DT), XGBoost, Random Forest, and MLP was better on the FGSM adversarial datasets when compared to the CTGAN datasets. However, the authors did not discuss the realistic implementation of adversarial attacks in their case studies. Additionally, they did not explain how to set the optimum level of perturbations that could trigger an attack.

Huang et al. [16] assessed the efficiency of three port-scan attack-detecting models for Software Defined Networking (SDN) environments: MLP, Convolutional Neural Network (CNN), and Long Short-Term memory (LSTM) under

[3]See https://www.unb.ca/cic/datasets/nsl.html [May 2024].
[4]See https://ieee-dataport.org/documents/nss-mirai-dataset [May 2024].
[5]See https://ieee-dataport.org/documents/bot-iot-dataset [May 2024].

the FGSM attack. Martins et al. [23] demonstrated a deterioration in the mean performance of RF, SVM, Decision Trees (DT), Naïve Bayes (NB) and Neural Network (NN) classifiers under FGSM attacks. Sriram et al. [34] analyzed the performance of DNN, RF, Support Vector Machine (SVM), NB and DT classifiers against FGSM attack using the NSL-KDD dataset[3]. Debicha et al. [9] concluded that the FGSM attacks significantly deteriorated the performance of a DNN detection model.

The existing literature highlights the vulnerability of detection models to evasion adversarial attacks, which are considered substantial threats. The literature concentrates on compromising IDS by employing generic evasion adversarial attacks. While these attacks may be demonstrated with high evasion rates, the realism and effectiveness of the generated adversarial traffic have not been taken into account for real-world attacks. Moreover, the majority of research has concentrated on the consequences of adversarial attacks within conventional IP networks [9, 10, 18, 24, 31]. Conversely, it is imperative to evaluate security threats in other networking landscapes like smart grids given their critical role in the digital age of hyper-connected Critical Infrastructures (CIs).

Based on the previous research, we did not find any research that has verified the realism of adversarial attacks in smart grid networks. Therefore, this paper proposes a realistic approach to modeling adversarial attacks against ML-IDS for smart grid communication by identifying the capabilities and conditions that are necessary for an attacker to carry out such attacks. More importantly, this research recreates a realistic attack model and assumptions as well as a realistic dataset collected from a power system testbed.

2.5 Limitations of Previous Research Studies

The previously published studies had three significant flaws. First, they overlooked the importance of adhering to traffic domain constraints when crafting adversarial attacks to uphold the validity and functionality of attack traces. Second, they assumed that the adversary could manipulate any number of features without restraint, potentially disrupting the semantic connections between interdependent features. In real-world scenarios, this assumption may not be relevant in some contexts as the adversary may be an outsider or unfamiliar with the detailed workings of an IDS. Lastly, they operated under the assumption of a white-box threat model, wherein the adversary had access to all the parameters of the targeted model, which may not be feasible in many real-world scenarios.

3 Case Study

For our case study, we use publicly available power system datasets implemented by Mississippi State University and Oak Ridge National Laboratory[6]. Fig. 1 details the power system framework configuration and the components utilized

[6]See https://sites.google.com/a/uah.edu/tommy-morris-uah/ics-data-sets [05/24].

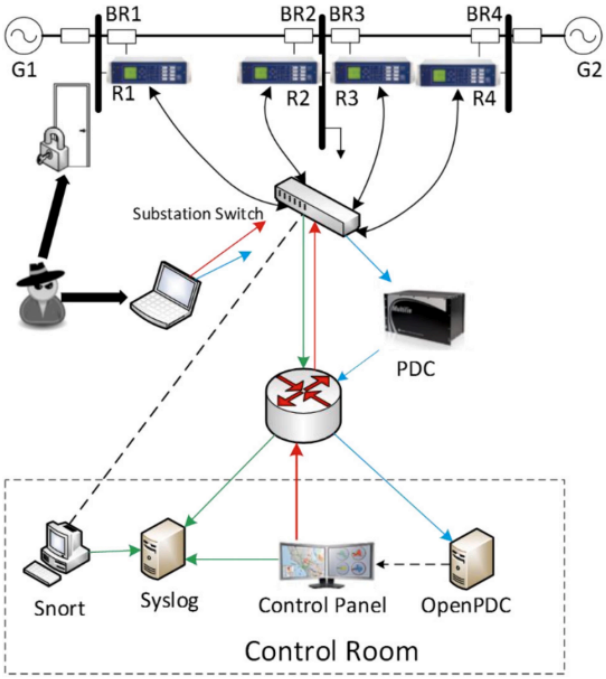

Fig. 1. Power System Framework Testbed used for generating the datasets [2].

to generate the datasets that enabled the experiments presented in this research. More specifically, the components of the power system include:

- G1 and G2 are the primary generators.
- The Intelligent Electronic Devices (IEDs) R1, R2, R3, and R4 switch the breakers (BR1, BR2, BR3, BR4), which automatically protect electrical circuits from overload or short circuits.
- Each IED controls a single breaker (e.g., R1 controls BR1, R2 controls BR2).
- IEDs use a distance protection method to trip the breaker on detected faults, regardless of validity, as they lack internal validation to differentiate them.
- Operators can send commands to the IEDs to manually trip the breakers. Manual override is used for maintenance on lines or system components.
- The testbed includes additional network monitoring and detection tools, like SNORT and Syslog servers.

3.1 Dataset Description

The original dataset contains 128 features from two categories: 1) Phasor Measurement Unit (PMU) and 2) Control Room logs. There are four PMUs, and 29 measurements are taken from each PMU, contributing to a total of 116 features.

The control room logs are divided into three categories: control panel, SNORT, and relay logs, each with four features, i.e. a total of 12 features. The dataset column-naming convention helps in understanding of each feature. Each PMUs measurement is denoted by R# - Signal Reference, where R# is the PMU number (R1, R2, R3, and R4) and Signal Reference as specified in Table 1. Information in the table was adapted from the original dataset description document [2].

Table 1. Feature Description

Feature	Description
PA1-PA3:VH	PA1:VH-PA3:VH Phase A
PM1: V-PM3:V	C Voltage Phase Angle
PA4:IH-PA6:IH	Phase A-C Current Phase Angle
PM4: I-PM6: I	Phase A-C Current Phase Magnitude
PA7:VH-PA9:VH	Pos.-Neg.-Zero Voltage Phase Angle
PM7: V-PM9: V	Pos.-Neg.-Zero Voltage Phase Magnitude
PA10:VH-PA12:VH	Pos.- Neg.-Zero Current Phase Angle
PM10: V - PM1	Pos.-Neg.-Zero Current Phase Magnitude
F	Frequency for relays
DF	Frequency Delta (dF/dt) for relays
PA:Z	Appearance Impedance for relays
PA:ZH	Appearance Impedance Angle for relays
S	Status Flag for relays

3.2 Simulated Attacks

A dataset comprising both benign and malicious data was generated from the testbed. The data is classified into three primary categories: instances with 'no events', instances with 'natural events', and instances with 'attack events'. Both instances of 'no event' and 'natural event' are grouped together to indicate benign activity. Five different attack scenarios were used to target the power system in order to generate attacks. These attacks are described as follows:

- ***Short-circuit fault.*** This is a short circuit in a power line and can occur in various locations along the line. The location is indicated by the percentage range.
- ***Line maintenance.*** One or more relays are disabled on a specific line to do maintenance for that line.
- ***Remote tripping command injection attack.*** This is an attack that sends a command to a relay which causes a breaker to open. It can only be done once an attacker has penetrated outside defences.
- ***Relay setting change attack.*** Relays are configured with a distance protection scheme. The attacker changes the setting to disable the relay function so that the relay will not trip for a valid fault or a valid command.

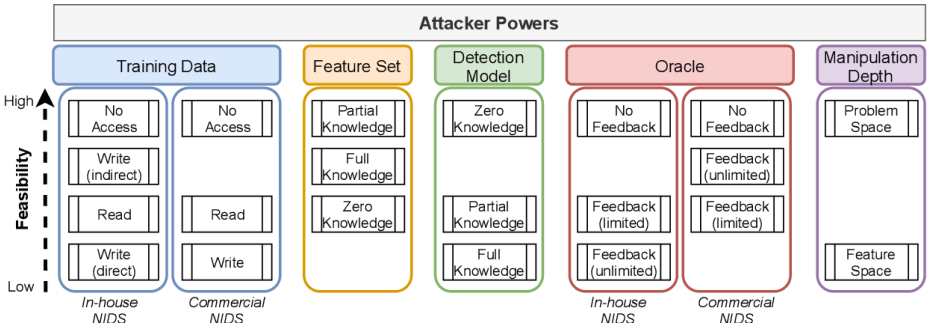

Fig. 2. Feasibility of each power available to the attacker [5].

– ***Data injection attack.*** A valid fault is imitated by changing values to parameters such as the current, voltage, and sequence components. This attack aims to blind the operator and cause a blackout.

3.3 Attacker Capabilities

In this research, we model realistic adversarial attacks against Machine Learning NIDS (ML-NIDS) by adopting the taxonomies of Apruzzese et al. [5]. To model them, we take into consideration the realistic capabilities of an attacker, which denotes how much control the attacker has on the target detection system. The attacker can have access to the following five elements as highlighted in Fig. 2.

– ***Training Data*** represents the ability to access the dataset used to train the ML-NIDS. It can come in the form of read, write, or no access at all.
– ***Feature Set*** refers to the knowledge of the features analyzed by the ML-NIDS to perform its detection. It can come in the form of none, partial, or full knowledge.
– ***Detection Model*** describes the knowledge of the (trained) ML model integrated into the NIDS that is used to perform the detection. This knowledge may be none, partial or full.
– ***Oracle*** is an element which denotes the possibility of obtaining feedback from the output produced by the ML-NIDS to an attacker's input. This feedback can be limited, unlimited, or absent.
– ***Manipulation Depth*** describes the nature of the adversarial manipulation, that may modify the analyzed traffic (problem space) traffic level or one or more features (feature space).

3.4 Threat Model

In this paper, we examine the risk posed by an insider threat actor with administrative access privileges to the network systems of the smart grid network. Insider threats represent a significant yet often overlooked danger to CI [11].

More specifically, as insiders reside behind the enterprise-level security defense mechanisms and often have privileged access to the network, detecting and preventing insider threats is a complex and challenging problem [22]. According to the German Federal Office for Information Security, insider threats include those with potentially privileged access to IT components, services, installations, documents, or any other critical information about the infrastructure and its components. In particular, the following groups are considered as insider threats [1]:

- A person with direct physical access to control systems (e.g., operators, engineers).
- A person with privileged rights (e.g., administrators).
- People with indirect access (e.g., to the office network or administration buildings).
- External service providers (e.g., maintenance or software development), suppliers, etc.

Such adversaries can deploy a range of attacks such as:

- Social engineering can be employed to plan follow-up attacks. This can be accomplished through determining weak employees, understanding industrial processes, and mapping the IT infrastructure.
- Unauthorized acquisition or alteration of confidential data may occur through gaining access to file servers, historians, and data storage media. Primarily the motives of such attack is industrial espionage and whistle-blowing.
- Deliberate acts of sabotage against the company. Motivated by political or economic motives, this might include modifying control components or insertion of malware or spyware into the system.

As shown in Fig. 3, our work is based on a realistic scenario involving an insider threat actor with physical and administrative access privileges to the network systems of the smart grid network. Within the network, a ML-NIDS model is present to detect any form of attack on the network. In high-speed networks environment and considering the difficulties of analyzing each individual packet, it is realistic to consider that the NIDS is a flow-based system rather than a packet-based system. This NIDS therefore analyzes the flow data generated by the router based on the traffic outgoing/entering the network. All flows first pass through a flow exporter, which extracts the network features for pre-processing and classification.

In the power system scenario discussed in Section 3 and given the capabilities of the attacker as discussed in Section 3.3, it is presumed that the adversary is interested in launching an evasion attack. Given the adversary position, it is presumed that he/she knows the features that the IDS is using for the classification; nevertheless, he/she does not know the specific algorithm configuration of the detector. The attacker's primary objective is to identify how to circumvent the NIDS. This will allow him/her to either launch more damaging attacks in the future or exploit the organization for personal gain by selling this information

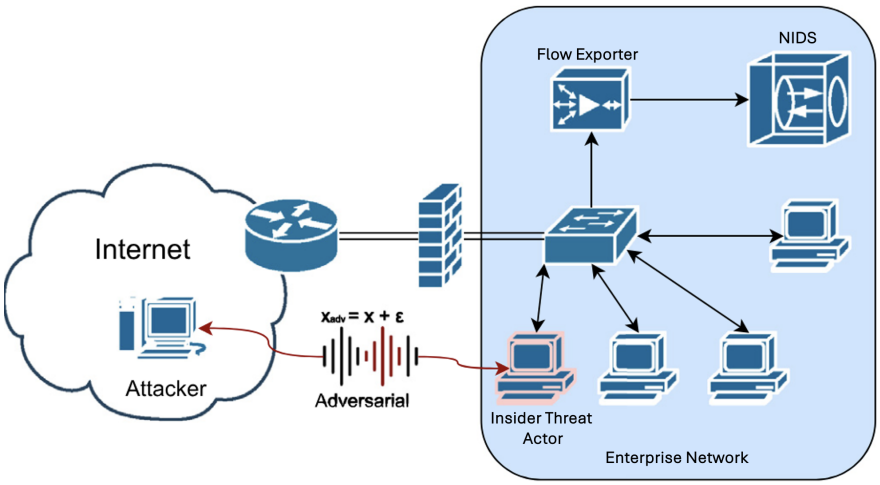

Fig. 3. Illustration of the considered threat scenario

to competitors, ultimately leaving the organization exposed and susceptible to harm. Due to the knowledge acquired by the adversary, this type of attack can be classified as a grey box attack. This threat scenario presented in Fig. 3 was used to generate adversarial data for testing on trained ML model as presented in Section 5.

4 Attack Generation

This research investigates the use of JSMA and FGSM techniques in a grey-box scenario where the attacker has access to the full dataset and features but has no knowledge of the target model. Despite not knowing the target model, we can approximate samples that will cause the target model to declassify it using another model since adversarial samples are transferable across machine learning models.

As shown in Fig. 4, there are four steps in the process of creating adversarial traffic. During step 1, the attacker generates adversarial traffic that is specifically designed to bypass the surrogate models that the attacker previously trained using sniffed traffic. The attacker then receives and analyzes the adversarial traffic that managed to avoid detection by the surrogate models during step 2. During step 3, the attacker uses the transferability property to send adversarial traffic to the defender NIDS. In step 4, the adversarial traffic that successfully bypassed the defender NIDS will arrive at the insider threat actor machine. In the context of this research, the attacks were implemented through the Adversarial

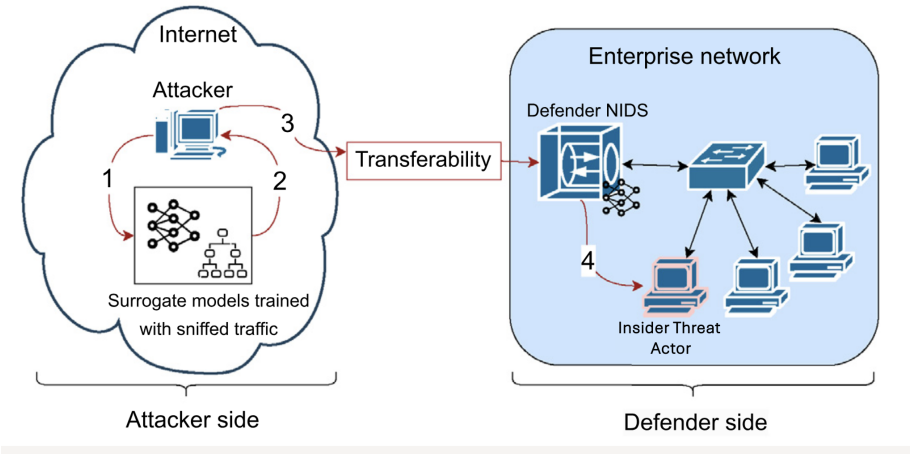

Fig. 4. Illustration of the adversarial traffic generation.

Robustness Toolbox (ART)[7]. ART is a Python tool, which can generate a variety of adversarial attacks.

4.1 Machine Learning based NIDS

ML methods are increasingly used in the context of the NIDS. To explore how well the supervised classification methods can learn to detect cyber attacks in smart grid environment, some classical ML algorithms are used to train the classification model and evaluated in this research. On the defender side, the defender uses Random Forest (RF), Naive Bayes (NB) and XGBoost (XGB) algorithms as a model for the NIDS. We selected these three ML models for our work because of their wide usage by the research community [25] [21]. Additionally, the selected methods are easy to implement, less computational cost is needed and work well with annotated data making them a suitable choice for our NIDS. These algorithms adhere to the same training and testing procedure, as demonstrated in Figure 5. The experimental arrangement provides a common platform to compare the performance and help decide the best-performing model.

4.2 Hyper Parameters Optimization

XGboost and Naive Bayes models were trained using the default parameters provided by the scikit-learn framework[8]. To ensure optimal performance of RF as

[7]See https://github.com/Trusted-AI/adversarial-robustness-toolbox [05/24].
[8]See https://scikit-learn.org/stable/ [05/24].

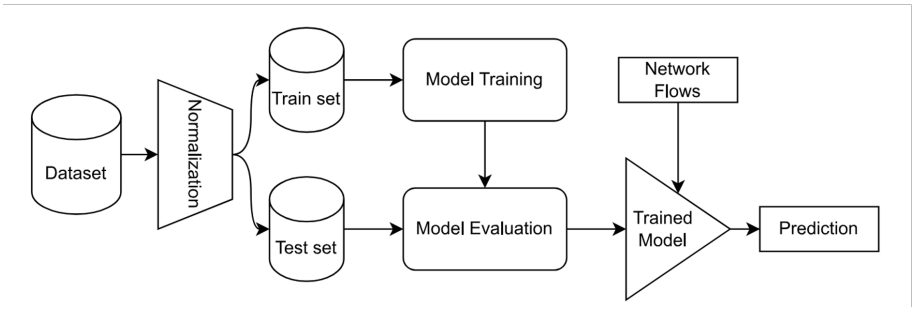

Fig. 5. The training and testing pipeline for an attacker and a defender.

recommended by Zhu et al. [38] key tunable hyperparameters were applied: number of trees (100), split method (Gini), and minimum number of samples required to split (2). Hyperparameter tuning finds the best value for the algorithm's parameters from the search space. This study did not perform hyperparameter tuning; however, this is a possible area of research for future work.

4.3 Model Training and Testing

To ensure the usability of our research, the dataset was divided into subsets and stratified according to the labels. The data subsets are equivalent in terms of size and distribution. The first subset is used for training and evaluation of the NIDS model (defender). The second data subset is used by the attacker to train a surrogate model as shown on Fig. 5. Considering the insider threat scenario described in Section 3, the attacker can obtain this data by sniffing the network. The datasets for each side (defender and attacker) are split into a training dataset and a testing dataset for validation, with proportions of 70% and 30% respectively. Both training and testing data subsets are evenly split in terms of malicious and benign traffic. The datasets are separated in this manner to have the most balanced representation to avoid the problem of unbalanced data.

While this is a requirement to get the most of our envisaged IDSes, it is not essential to the actual claim of the paper. We are aware of the fact that real traffic data may be rather unbalanced and tuning the IDS to work in those contexts may overcome that problem. However, our evasion attack methodology is not dependent on this aspect. The threat model follows the same training and testing process, as shown in Fig. 5. The performance of the models was evaluated using the standard ML evaluation metrics.

4.4 Performance Metrics

To evaluate the performance of IDS models, different evaluation metrics can be used [14]. All of them are based on the confusion matrix represented in Table 2.

Table 2. IDS Confusion Matrix

	Predicted Class	
Actual Class	**Anomaly**	**Normal**
Anomaly	True Positive (TP)	False Negative (FN)
Normal	False Positive (FP)	True Negative (TN)

– Recall, also known as the "detection rate,". Recall measures the proportion of actual positive instances that are correctly identified by the model.

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

– Precision evaluates the accuracy of the positive predictions made by a model. Specifically, precision measures the proportion of predicted positive instances that are correct.

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

– F1 score gives the performance of the combined recall and precision evaluation metrics—it is the harmonic mean of both. It provides the system with the capacity to give relevant results and refuse the others.

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \tag{5}$$

5 Evaluation

This section details the experimental findings. Section 5.1 discusses the results from the performance of the defender and an attacker model based on different metrics, namely precision, recall, and F1 score. Further, under Section 5.2, the results regarding the performance of the models in an adversarial context are discussed. Lastly, a detailed analysis is carried out to determine the difference in perturbation between the initial malicious instance and the adversarial instance.

5.1 Initial performance of ML-IDS models in clean settings

To assess the initial performance of ML-IDS models trained in non-adversarial (clean) settings for both the attacker and defender perspectives, several metrics are utilized. These include recall (Eq. 3), precision (Eq. 4), and F1-score (Eq. 5). In clean settings, the ML-IDS models performed binary classification to distinguish between malicious and benign traffic. As shown in Table 3, the results of the initial performance of the trained ML models, XGBoost performed better compared to Naive Bayes and the Random Forest models. These initial results show good performance of the Random Forest and XGBoost in general. The average F1-scores achieved by the classifiers were 0.845, 0.567 and 0.925 respectively.

Table 3. Performance of the trained ML models on clean settings

Classifier	Accuracy	Precision	Recall	F1 Score	Time (s)
Random Forests	0.8473	0.8655	0.8473	0.8454	30
XGBoost	0.9464	0.9075	0.9464	0.9252	45
Naive Bayes	0.5742	0.5831	0.5742	0.5673	24

5.2 Performance of ML-IDS models in adversarial settings

To reiterate, adversarial attacks aim to automatically introduce perturbations to the unseen data points to evade detection of the trained model. To explore how different combinations of the FGSM parameters affect the performance of the trained classifiers, different adversarial samples were generated from the testing data using epsilon (ϵ) values ranging between 0 to 0.45. Although the current literature does not recommend a standard value for ϵ, in our research, we adopted a range between 0 to 0.45 to test attack success rates as suggested by Goodfellow et al. [13].

The adversarial dataset was then generated using different (ϵ) values. To determine how the detection performance of the aforementioned ML models could be affected, adversarial samples were then joined with the benign testing data and subsequently presented to the trained model. Figure 6 shows the overall performance for different adversarial combinations. As the (ϵ) values increased, the model accuracy decreased further. For instance XGBoost performance decreased from 94.64% at (ϵ)= 0 to 72.03% at (ϵ)= 0.45. On the other hand, Random Forest performance decreased from 84.73% at (ϵ)= 0 to 68.02% at (ϵ)= 0.45. Lastly, the performance for Naive Bayes decreased from 57.42% at (ϵ)= 0 to 32.05% at (ϵ)= 0.45. For FGSM adversarial attack, the attack success rate increased with higher (ϵ) values hence the accuracy declined because to the ML model was deceived by the attack.

Selecting an appropriate (ϵ) value that will control the pertubation size is very crucial as higher (ϵ) value may increase attack success rates but may also increase the detectability of the adversarial samples. Small pertubations are more ideal to launch a realistic attack and remain undetected by the IDS. For instance, when $\epsilon = 0.05$, the XGBoost accuracy dropped from 94.64% to 88.04% while Random Forest accuracy dropped from 84.73% to 78.43% and the Naive Bayes accuracy dropped from 57.42% to 52.89%. To consider another instance, when $\epsilon = 0.001$, XGBoost accuracy dropped from 94.64% to 92.36%, the Random Forest accuracy dropped from 84.73% to 82.67% and Naive Bayes accuracy dropped from 57.42% to 55.32% as detailed in Table 4. As per the adversarial performance, all the metrics declined in comparison to the performance of the original datasets in a clean setting.

In comparison to Random Forest and XGBoost, the Naive Bayes achieved a higher decrease in performance. This may indicate that Naive Bayes is more sensitive, subsequently misclassifying malicious data. Conversely, the classification performance of XGBoost achieved a better performance. This may indicate

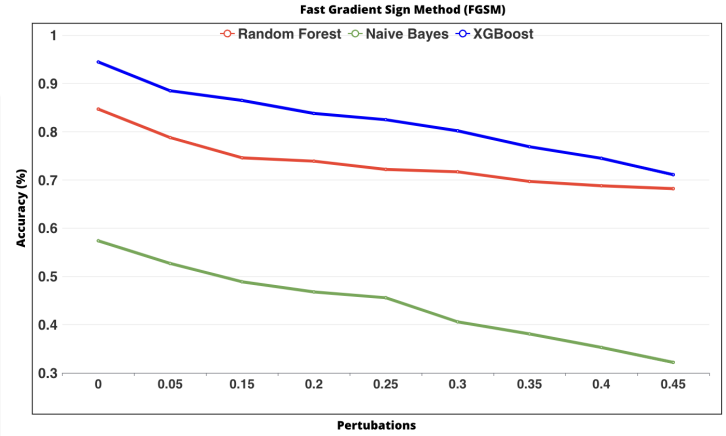

Fig. 6. Performance of trained models in adversarial settings when increasing perturbations (ϵ).

Table 4. Degradation of trained models in FGSM adversarial settings (ϵ=0.001).

Classifier	Accuracy	Precision	Recall	F1 Score
Random Forests	0.8267	0.6667	0.8267	0.6656
XGBoost	0.9236	0.7690	0.9236	0.7493
Naive Bayes	0.5532	0.2808	0.5532	0.2976

that XGBoost may be more a more robust classifier in discriminating between malicious and benign data points correctly. To study the impact of adversarial instances generated in our evasion attack, as well as the effectiveness of transferring the adversarial instances created by the attacker to the model trained by the defender, the first experiment focuses on JSMA as shown in Table 5.

To measure the impact of adversarial instances, the detection rate metric known as recall is used. It measures the rate of adversarial instances detected by the ML-IDS as malicious. To accomplish this, the attacker initially generates adversarial instances for each model trained on his/her side (i.e., Random Forest, XGBoost and Naive Bayes). The adversarial instances created for one model are then sent to the other models to evaluate the transferability property between the models trained by the attacker.

To summarise, our results make it evident that the adversarial attacks generated through FGSM and JSMA techniques achieved a high misclassification rate against the ML classifiers. The classification performance for all the classifiers degraded on adversarial setting. This demonstrates that FGSM is an efficient adversarial attack technique that leverages gradient information to create perturbations. Selecting a suitable epsilon value, which regulates the perturbation size, is critical because larger epsilon values enhance attack success rates while

Table 5. Degradation of trained models in JSMA adversarial settings.

Classifier	Accuracy	Precision	Recall	F1 Score
Random Forests	0.6416	0.6605	0.6434	0.5980
XGBoost	0.7090	0.7289	0.7355	0.7234
Naive Bayes	0.2934	0.2789	0.2978	0.2784

also increasing the detectability of adversarial samples. Second, evaluation using JSMA, the classification performance of XGBoost degrades from 94.64% to 70.90% while Random Forest degrades from 84.73% to 64.16% and Naive Bayes degrades from 57.42% to 29.34%.

6 Conclusion

The use of the NIDS based on ML algorithms presents intriguing security challenges. Despite their impressive performance, these ML models are vulnerable to various adversarial attacks, particularly evasion attacks. This paper demonstrated the importance of realistic threat modeling in the context of adversarial attacks on smart grid systems. By highlighting real attacker capabilities and feasible attack scenarios, this research provides a more practical and applicable perspective compared to the existing literature, which often deals with hypothetical or idealized models. Moreover, this research performs an empirical evaluation using a power system dataset generated from a smart grid testbed, which adds significant value, grounding the theoretical insights in real-world data.

To the best of our knowledge, this is the first realistic approach that aims to evade the NIDS by leveraging on the transferability property without relying on any query methods and with very limited knowledge of the target NIDS. This approach operates within the traffic space and adheres to domain constraints.

This paper demonstrates a realistic adversarial approach designed to generate valid and realistic adversarial network traffic by introducing minor perturbations. This allows for bypassing the NIDS protection with a high probability while preserving the core logic of the underlying model. The experiments detailed in this research have shown that evasion attacks can be successfully generated using JSMA and FGSM methods, impacting the classification performance of Random Forest, Naive Bayes and the XGBoost ML models.

Furthermore, our results show that the same set of adversarial examples that managed to deceive one classifier also succeeded in deceiving the other classifiers. For instance, the adversarial samples generated by FGSM managed to decrease the performance of XGBoost from 94.64% to 72.03%, Random Forest from 84.73% to 68.02% and Naive Bayes from 57.42% to 32.05%. This observation can be considered additional evidence for the transferability phenomenon first alluded to by Papernot et al. [27] within the image recognition domain and by Sheatsley et al. [33] within the network intrusion detection domain. Our work in the smart grid domain makes it clear that all three classifiers are vulnerable to adversarial perturbations.

6.1 Future Work

Although the experiments in our research demonstrate that adversarial attacks can successfully be generated using JSMA and FGSM and affect the classification performance of state-of-the-art supervised models, it is noteworthy that there are other techniques for generating adversarial attacks to be considered such as Carlini Wagner (CW) and Generative Adversarial Network (GAN). As a part of future work, this research can be extended to observe different adversarial techniques as a source for adversarial attacks. Moreover adversarial attacks should be investigated against other ML models.

The adversarial attacks against ML models are not limited to the domain of IDS systems, but to all systems where ML techniques are implemented. Awareness, defence and mitigation of adversarial attacks against ML is an important direction for future research, for example in federated learning [26, 7]. Therefore, it would be interesting to assess the applicability of the proposed model in a distributed setting and evaluate its applicability. As mentioned before, there is a great need for research on suitable mitigation techniques against adversarial threats.

Acknowledgments. The first and third author were supported by the Brno University of Technology project "Smart information technology for a resilient society", 2023-2025, code FIT-S-23-8209. The second author was supported by the RICS centre financed by the Swedish Civil contingencies agency (MSB) and project AIR^2 supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP), funded by the Knut and Alice Wallenberg Foundation.

References

1. Industrial Control System Security. Insider threat. https://www.allianz-fuer-cybersicherheit.de/, accessed: 22-01-2024
2. Power system attack datasets. http://www.ece.uah.edu/thm0009/icsdatasets, accessed: 30-1-2024
3. Alatwi, H.A., Morisset, C.: Realism versus performance for adversarial examples against DL-based NIDS. In: Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing. pp. 1549–1557 (2023)
4. Alshamrani, A., Myneni, S., Chowdhary, A., Huang, D.: A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities. IEEE Communications Surveys & Tutorials **21**(2), 1851–1877 (2019)
5. Apruzzese, G., Andreolini, M., Ferretti, L., Marchetti, M., Colajanni, M.: Modeling realistic adversarial attacks against network intrusion detection systems. Digital Threats: Research and Practice (DTRAP) **3**(3), 1–19 (2022)
6. Asimopoulos, D.C., Radoglou-Grammatikis, P., Makris, I., Mladenov, V., Psannis, K.E., Goudos, S., Sarigiannidis, P.: Breaching the defense: Investigating FGSM and CTGAN adversarial attacks on IEC 60870-5-104 AI-enabled intrusion detection systems. In: Proceedings of the 18th International Conference on Availability, Reliability and Security. pp. 1–8 (2023)
7. Bouacida, N., Mohapatra, P.: Vulnerabilities in federated learning. IEEE Access **9**, 63229–63249 (2021). https://doi.org/10.1109/ACCESS.2021.3075203

8. Clements, J., Yang, Y., Sharma, A.A., Hu, H., Lao, Y.: Rallying adversarial techniques against deep learning for network security. In: Symposium Series on Computational Intelligence (SSCI). pp. 01–08. IEEE (2021)
9. Debicha, I., Debate, T., Dricot, J.M., Mees, W.: Adversarial training for deep learning-based intrusion detection systems. In: The Sixteenth International Conference on Systems (ICONS) (2021)
10. Fu, X., Zhou, N., Jiao, L., Li, H., Zhang, J.: The robust deep learning–based schemes for intrusion detection in internet of things environments. Annals of Telecommunications **76**(5-6), 273–285 (2021)
11. Gollmann, D.: From insider threats to business processes that are secure-by-design. In: Third International Conference on Intelligent Networking and Collaborative Systems. pp. 627–627 (2011). https://doi.org/10.1109/INCoS.2011.175
12. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. Advances in neural information processing systems **27** (2014)
13. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples (2015), https://arxiv.org/abs/1412.6572
14. Han, J., Kamber, M., Pei, J.: Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 3rd edn. (2011)
15. Hashemi, M.J., Cusack, G., Keller, E.: Towards evaluation of NIDSs in adversarial setting. In: Proceedings of the 3rd ACM CONEXT Workshop on Big Data, Machine Learning and Artificial Intelligence for Data Communication Networks. pp. 14–21 (2019)
16. Huang, C.H., Lee, T.H., Chang, L.h., Lin, J.R., Horng, G.: Adversarial attacks on SDN-based deep learning IDS system. In: International Conference on Mobile and Wireless Technology (ICMWT). pp. 181–191. Springer (2019)
17. Ibitoye, O., Shafiq, O., Matrawy, A.: Analyzing adversarial attacks against deep learning for intrusion detection in IoT networks. In: IEEE global communications conference (GLOBECOM). pp. 1–6. IEEE (2019)
18. Jeong, J., Kwon, S., Hong, M.P., Kwak, J., Shon, T.: Adversarial attack-based security vulnerability verification using deep learning library for multimedia video surveillance. Multimedia Tools and Applications **79**, 16077–16091 (2020)
19. Kravchik, M., Shabtai, A.: Detecting cyber attacks in industrial control systems using convolutional neural networks. In: Proceedings of the Workshop on Cyber-Physical Systems Security and Privacy. p. 72–83. CPS-SPC '18, Association for Computing Machinery (2018)
20. Linda, O., Vollmer, T., Manic, M.: Neural network based intrusion detection system for critical infrastructures. In: International joint conference on neural networks. pp. 1827–1834. IEEE (2009)
21. Liu, C., Gu, Z., Wang, J.: A hybrid intrusion detection system based on scalable k-means+ random forest and deep learning. IEEE Access **9**, 75729–75740 (2021)
22. Liu, L., De Vel, O., Han, Q.L., Zhang, J., Xiang, Y.: Detecting and preventing cyber insider threats: A survey. IEEE Communications Surveys & Tutorials **20**(2), 1397–1417 (2018)
23. Martins, N., Cruz, J.M., Cruz, T., Abreu, P.H.: Analyzing the footprint of classifiers in adversarial denial of service contexts. In: 19th EPIA Conference on Artificial Intelligence, Proceedings, Part II. pp. 256–267. Springer (2019)
24. Merzouk, M.A., Cuppens, F., Boulahia-Cuppens, N., Yaich, R.: A deeper analysis of adversarial examples in intrusion detection. In: Risks and Security of Internet and Systems: 15th International Conference, CRISIS 2020, Revised Selected Papers 15. pp. 67–84. Springer (2021)

25. Min, E., Long, J., Liu, Q., Cui, J., Chen, W.: Anomaly-based intrusion detection through text-convolutional neural network and random forest. Security and Communication Networks (2018)
26. Nguyen, T.D., Rieger, P., Miettinen, M., Sadeghi, A.R.: Poisoning attacks on federated learning-based IoT intrusion detection system. In: Proc. Workshop Decentralized IoT Syst. Secur.(DISS). vol. 79 (2020)
27. Papernot, N., McDaniel, P., Goodfellow, I.: Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. arXiv preprint arXiv:1605.07277 (2016)
28. Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z.B., Swami, A.: The limitations of deep learning in adversarial settings. In: European symposium on security and privacy (EuroS&P). pp. 372–387. IEEE (2016)
29. Papernot, N., McDaniel, P., Sinha, A., Wellman, M.P.: Sok: Security and privacy in machine learning. In: European Symposium on Security and Privacy (EuroS&P). pp. 399–414. IEEE (2018)
30. Peng, X., Huang, W., Shi, Z.: Adversarial attack against DOS intrusion detection: An improved boundary-based method. In: 31st International Conference on Tools with Artificial Intelligence (ICTAI). pp. 1288–1295. IEEE (2019)
31. Peng, Y., Su, J., Shi, X., Zhao, B.: Evaluating deep learning based network intrusion detection system in adversarial environment. In: 9th International Conference on Electronics Information and Emergency Communication (ICEIEC). pp. 61–66. IEEE (2019)
32. SectorCERT: The attack against Danish, critical infrastructure. https://sektorcert.dk/wp-content/uploads/2023/11/SektorCERT-The-attack-against-Danish-critical-infrastructure-TLP-CLEAR.pdf, accessed: 01-01-2024
33. Sheatsley, R., Papernot, N., Weisman, M., Verma, G., McDaniel, P.: Adversarial examples in constrained domains (2022), https://arxiv.org/abs/2011.01183
34. Sriram, S., Simran, K., Vinayakumar, R., Akarsh, S., Soman, K.: Towards evaluating the robustness of deep intrusion detection models in adversarial environment. In: International Symposium on Security in Computing and Communication. pp. 111–120. Springer (2019)
35. Tavallaee, M., Bagheri, E., Lu, W., Ghorbani, A.A.: A detailed analysis of the KDD CUP 99 data set. In: Symposium on Computational Intelligence for Security and Defense Applications. pp. 1–6. IEEE (2009). https://doi.org/10.1109/CISDA.2009.5356528
36. Wang, Z.: Deep learning-based intrusion detection with adversaries. IEEE Access **6**, 38367–38384 (2018)
37. Warzyński, A., Kołaczek, G.: Intrusion detection systems vulnerability on adversarial examples. In: Innovations in Intelligent Systems and Applications (INISTA). pp. 1–4. IEEE (2018)
38. Zhu, N., Zhu, C., Zhou, L., Zhu, Y., Zhang, X.: Optimization of the random forest hyperparameters for power industrial control systems intrusion detection using an improved grid search algorithm. Applied Sciences **12**(20), 10456 (2022)