Journal Pre-proof

SoK: Realistic adversarial attacks and defenses for intelligent network intrusion detection

João Vitorino, Isabel Praça and Eva Maia

PII: S0167-4048(23)00343-7
DOI: https://doi.org/10.1016/j.cose.2023.103433
Reference: COSE 103433

To appear in: *Computers & Security*

Received date: 5 June 2023
Revised date: 28 July 2023
Accepted date: 13 August 2023

Please cite this article as: J. Vitorino, I. Praça and E. Maia, SoK: Realistic adversarial attacks and defenses for intelligent network intrusion detection, *Computers & Security*, 103433, doi: https://doi.org/10.1016/j.cose.2023.103433.

SoK: Realistic Adversarial Attacks and Defenses for Intelligent Network Intrusion Detection

João Vitorino[a,*], Isabel Praça[a,*], Eva Maia[a]

[a]*Research Group on Intelligent Engineering and Computing for Advanced Innovation and Development (GECAD), School of Engineering, Polytechnic of Porto (ISEP/IPP), 4249-015, Porto, Portugal*

Abstract

Machine Learning (ML) can be incredibly valuable to automate anomaly detection and cyber-attack classification, improving the way that Network Intrusion Detection (NID) is performed. However, despite the benefits of ML models, they are highly susceptible to adversarial cyber-attack examples specifically crafted to exploit them. A wide range of adversarial attacks have been created and researchers have worked on various defense strategies to safeguard ML models, but most were not intended for the specific constraints of a communication network and its communication protocols, so they may lead to unrealistic examples in the NID domain. This Systematization of Knowledge (SoK) consolidates and summarizes the state-of-the-art adversarial learning approaches that can generate realistic examples and could be used in real ML development and deployment scenarios with real network traffic flows. This SoK also describes the open challenges regarding the use of adversarial ML in the NID domain, defines the fundamental properties that are required for an adversarial example to be realistic, and provides guidelines for researchers to ensure that their future experiments are adequate for a real communication network.

Keywords: realistic adversarial examples, adversarial robustness, cybersecurity, intrusion detection, machine learning

*Corresponding author

Email addresses: jpmvo@isep.ipp.pt (João Vitorino), icp@isep.ipp.pt (Isabel Praça), egm@isep.ipp.pt (Eva Maia)

1. Introduction

Modern organizations can benefit from the digital transformation to re-engineer their business processes, integrating control and information systems and automating decision-making procedures. Nonetheless, as organizations become more and more dependent on digital systems, the threat posed by a cyber-attack skyrockets [1]. Every novel technology adds hidden vulnerabilities that can be exploited in multiple attack vectors to disrupt the normal operation of a system. This is particularly concerning for organizations that deal with confidential information and sensitive personal data, or manage critical infrastructure, such as the healthcare and energy sectors [2].

The disruptions caused by a successful cyber-attack can be extremely costly for an organization. In 2022, the average cost of a data breach was reported to be 4.35 million US dollars, an increase of 12.7% since 2020 [3]. This continued growth of both the number of successful cyber-attacks and their associated costs in various sectors and industries denotes that modern organizations face tremendous security challenges. Furthermore, since monitoring a system to detect suspicious activity is not a trivial process and small enterprises commonly fall short of security best practices, most go out of business within 6 months of a breach [4].

With financial security and business continuity on the line, it is essential for organizations to improve the way they perform Network Intrusion Detection (NID). This is where Artificial Intelligence (AI), and more specifically Machine Learning (ML), can be incredibly valuable [5]. ML models can originate from numerous algorithms, including tree-based algorithms and deep learning algorithms based on Artificial Neural Networks (ANNs), and can be trained to automate several tasks, from the recognition of patterns and anomalies in network traffic flows to the classification of complex cyber-attacks. The adoption of intelligent cybersecurity solutions can improve resilience and shorten the time required to detect and contain an intrusion by up to 76 days, leading to cost savings of up to 3 million US dollars [3].

However, despite the benefits of ML to tackle the growing number and increasing sophistication of cyber-attacks, it is highly susceptible to adversarial examples: cyber-attack variations specifically crafted to exploit ML models [6]. Even though the malicious purpose of a cyber-attack causes it to have distinct characteristics that could be recognized in a thorough analysis by security practitioners, an attacker can generate specific data perturbations in a network traffic flow to evade detection from intelligent security systems.

ML engineers and security practitioners still lack the knowledge and tools to prevent such disruptions, so adversarial attacks pose a major threat to ML and to the systems that rely on it [7, 8].

In recent years, a wide range of adversarial attacks have been developed and researchers have worked on various defenses to protect ML models, but most were not intended for the specific requirements of a communication network and the utilized communication protocols, so they may lead to unrealistic data perturbations in the NID domain. Even though several reviews have been published with comparisons of the strengths and weaknesses of multiple methods [9, 10], they do not address a key aspect: whether or not they could be used in a real communication network. Therefore, there is a lack in the current literature of a systematization of the approaches capable of generating realistic adversarial examples in the NID domain.

This Systematization of Knowledge (SoK) consolidates and summarizes the state-of-the-art adversarial learning approaches that could be applied in real ML development and deployment scenarios with real network traffic flows, and provides guidelines for future research to better address realism. The main Research Question (RQ) to be investigated was:

- How can adversarial cyber-attack examples be realistically used to attack and defend the ML models utilized in NID?

To provide more specific directions for the research, the main RQ was divided into three narrower sub-questions:

RQ1: What are the main perturbation crafting processes?

RQ2: What are the most realistic attack methods?

RQ3: What are the most reliable defense strategies?

By consolidating the main constraints and limitations of adversarial ML in the NID domain, this SoK intends to guide ML engineers and security practitioners to improve their methods and strategies according to the constraints of their specific communication networks. For that purpose, it is organized into multiple sections. Section 2 describes the adopted research methodology. Sections 3, 4, and 5, summarize and discuss the findings of RQ1, RQ2, and RQ3, respectively. Section 6 describes the open challenges regarding the main RQ and provides guidelines for future research. Finally, Section 7 presents the concluding remarks.

Table 1: Defined search terms.

Scope	Terms
Adversarial	*adversarial*
Learning	(*learning* OR *example* OR *perturbation* OR *attack* OR *defense*)
Network	(*network* OR *wireless* OR *IoT*)
Intrusion	(*intrusion* OR *anomaly* OR *cyber-attack*)
Detection	(*detection* OR *classification*)

2. Research Methodology

The research was based on the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) [11], which is a standard reporting guideline that aims to improve the transparency of literature reviews. Search terms were used in reputable bibliographic databases, and several inclusion and exclusion criteria were defined to screen the found publications. Since screening the titles and abstracts of the publications was sufficient to assess their eligibility, their full texts were directly reviewed without further exclusion rounds being necessary.

After a careful initial analysis of the literature, several search terms were chosen to address the formulated RQs. To ensure a comprehensive coverage of relevant publications within the scope of adversarial ML, the *adversarial* keyword was combined with other suitable terms like *perturbation*, *attack*, and *defense*. Concepts closely related to NID, such as *anomaly detection*, *cyber-attack classification*, *wireless* and *IoT* networks, were also considered. Table 1 provides an overview of the defined search terms.

The primary search source was Science Direct [12], which is a large bibliographic database of scientific journals and conference proceedings provided by the internationally recognized publisher Elsevier. Due to their relevance for scientific literature of ML, computing, software engineering, and information technology, the search also included the digital libraries of the Association for Computing Machinery (ACM) [13], the Institute of Electrical and Electronics Engineers (IEEE) [14], and the Multidisciplinary Digital Publishing Institute (MDPI) [15]. It is important to note that the PRISMA backward snowballing process of checking the references of the findings led to additional

Table 2: Defined inclusion and exclusion criteria.

Inclusion Criteria	Exclusion Criteria
IC1: Peer-reviewed journal article or conference paper	EC1: Duplicated publication
IC2: Available in the English language	EC2: Not applied to NID
IC3: Published from 2017 onwards	EC3: Not a novelty
IC4: Addressed adversarial ML for NID	EC4: Full text not available

records that were not directly obtained from these databases.

Since adversarial ML is an active research field, the search was limited to peer-reviewed publications in journals or conference proceedings from 2017 onwards. It included recent works addressing the use of adversarial ML in the NID domain, as well as surveys and reviews that addressed key developments, which led to additional publications. The findings that were duplicated in multiple databases were removed, and those that were not directly applied to the NID domain or did not introduce a novel method or strategy were excluded. Table 2 provides an overview of the inclusion and exclusion criteria that were defined to screen the found publications.

A total of 936 records were initially retrieved by applying the query to the contents of the publications stored in the selected databases. After removing duplicates and performing the screening phase, 139 records were excluded because they mentioned NID but were not directly applied to the NID domain. Furthermore, another 703 records were excluded because they did not present novel approaches. Despite performing experiments with different datasets and different contexts, these records used previously published methods and strategies without relevant modifications. The 703 records correspond to over 75% of the found publications, which demonstrates that it is difficult for researchers to find innovative approaches in a regular search in these databases, and further highlights the necessity for a systematization of the most relevant advances in this research field.

The remaining 82 records, which correspond to only 9% of the found publications, provided relevant aspects for the use of adversarial ML in the NID domain. Their content and references were checked and 16 additional publications were found by performing backward snowballing. Therefore, 98 publications were included in the review (see Figure 1). The publications

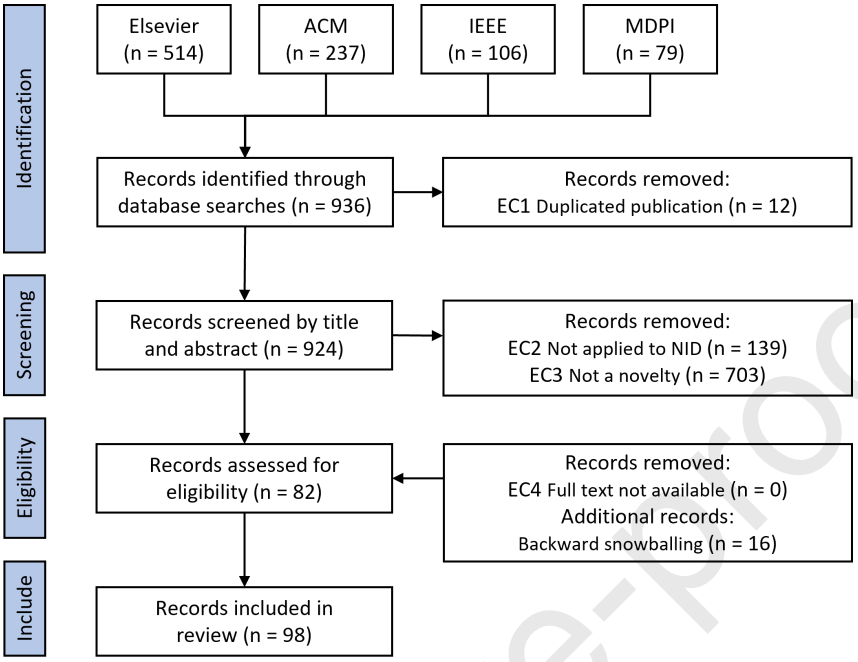

Figure 1: PRISMA search process.

were independently reviewed by each author of this SoK to extract the key developments and key takeaways for NID, and then their notes were consolidated and systematized to create this manuscript.

3. Data Perturbations

ML has been increasingly used to make digital systems more intelligent, but it is not flawless. For instance, if an ML model is trained with non-representative data that has missing or biased information, it may become underfit, performing poorly on both its training data and new data, or even overfit, performing very well on its training data but still poorly on previously unseen testing data [16]. These generalization errors can be quickly noticed during the development of an intelligent system with ML models, and better results can be achieved by improving data quality and fine-tuning the models [17]. However, even if a model generalizes well to the testing data, it is not guaranteed to always have a stable performance. During the inference phase, when it is deployed to make predictions on live data, it may sometimes behave unexpectedly with seemingly ordinary data samples [18, 19].

In a set of very similar samples of the same class, a model may correctly classify all but one. That specific sample may be assigned to a completely different class with a high confidence score because the model wrongly considers that it is different from the others. Ultimately, this unexpected behavior is caused by some unnoticed generalization errors during the model's training phase [20]. Since a training set does not cover all the samples that a model will encounter in its inference phase when deployed in a real system, the model will inevitably learn some simplifications of the decision boundaries that lead to incorrections in its internal reasoning [21, 22]. These incorrections can be hard to notice because the intricate mechanics of ML models cause the misclassifications to only occur in very specific samples, which are designated as **adversarial examples** [6].

An adversarial example may have very subtle perturbations that are almost imperceptible to humans but make it significantly different from regular samples to an ML model. Such adversarial perturbations can occur naturally in faulty data recordings with incorrect readings, but they can also be specifically crafted with specialized inputs to exploit the generalization errors [23, 24]. Even though all ML models are inherently susceptible to adversarial examples, different models will learn distinct simplifications of the target domain and create distinct decision boundaries. Hence, some models may be more vulnerable to perturbations in a certain feature than others, presenting model-specific edge cases that are hard to detect and address [25, 26].

Due to the advances in computer vision technologies and their increasing use across various industries, the major developments in adversarial ML have been focused on the image classification domain and are then adapted to other domains [27, 28]. In adversarial images, the perturbed features are pixels with a value freely assigned from 0 to 255, but it is pertinent to understand how these research efforts can be applied in cybersecurity solutions and if the concepts are transferable to a NID system in a real communication network. In the current literature, the perturbations that turn a regular sample into an adversarial example can be crafted using two main concepts: an **adversarial patch** that heavily modifies a few features, and an **adversarial mask** that slightly modifies many or all features [29].

Adversarial patches are the most straightforward way to disrupt a cyber-physical system. Since live data from a physical environment is not easily controllable, there is a greater risk for an ML model to be affected by faulty input data, either naturally occurring or purposely created [30]. For instance, for a model trained to classify street signs, a perturbed sample of a stop sign

Figure 2: Adversarial perturbation via a patch, based on [31].

with small black and white patches can be misclassified as a completely unrelated sign (see Figure 2) [31]. These patches are devised to cause the model to make a mistake when it encounters the sign at a certain angle, although a human would still recognize a stop sign [32].

Despite being harder to apply adversarial masks in physical environments, they are very well-suited for digital systems. For instance, for a model that performs handwritten digit recognition, a picture of a digit with a subtle change to several pixels can be misclassified as another digit (see Figure 3) [33]. Such model can have a wide range of applications, from certified documents and bank check processing to authentication via a picture of an identification document. If a person applies a filter that has a built-in adversarial mask before submitting the requested document, the automated verification systems that rely on this model can be deceived [34]. Furthermore, there are even some adversarial masks that exploit the intrinsic vulnerabilities of ML models and turn every image of well-established datasets into an adversarial image, which denotes that adversarial examples might not be as difficult to create as previously thought [35].

Even though most developments in the adversarial ML area of research have addressed image classification, the susceptibility of ML models to these examples has also been noticed in other domains with different data types, such as audio, text, tabular data, and time series [25, 36]. For the NID domain, adversarial perturbations must follow a tabular format, where each feature is a categorical or numerical variable representing a characteristic of network traffic [37, 38]. This tabular data format requires more complex perturbations, but they can also be based on the concepts utilized for images. For a tabular classification model, a patch-like perturbation could fully replace the values of categorical variables, which may include the communi-

Figure 3: Adversarial perturbation via a mask, based on [33].

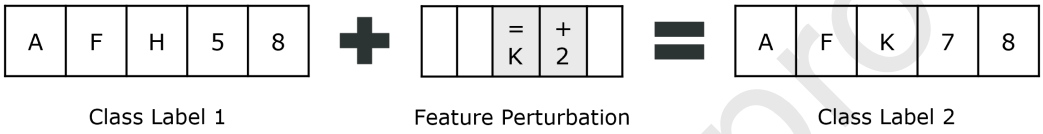

Figure 4: Adversarial perturbation on tabular data, based on [39].

cation protocol or the endpoint port number, and a mask-like perturbation could slightly increase or decrease the values of numerical variables, such as the amount of sent packets or the download to upload ratio (see Figure 4) [39, 40]. Nonetheless, not all perturbations are suitable for the NID domain because there are specific constraints that must be complied with.

In contrast with the pixels of an image, each tabular feature may have a different range of possible values, according to the characteristic it represents. Furthermore, a feature may also be highly correlated to several others, being required to exhibit specific values depending on the other characteristics of a sample [41]. For instance, a Slowloris is a Denial-of-Service (DoS) attack that attempts to overwhelm a web server by opening multiple connections and maintaining them as long as possible. A flow utilized in this cyber-attack must use the Transmission Control Protocol (TCP) and the Push (PSH) flag to keep the connection open on the port number 80, its endpoint [42].

A very relevant characteristic of this flow is its packet Inter-Arrival Time (IAT), which represents the elapsed time between the arrival of two subsequent packets and may be represented as two features: the minimum and maximum IAT. The flow may have a varying IAT between 20 and 30 seconds to appear as arbitrary benign traffic instead of scheduled packets just to keep the connection open. In a certain network, a longer IAT cannot be

used because the web server being attacked is configured with a timeout to close connections after 30 seconds of inactivity, which is a very common web application security measure [43].

However, throughout the literature, various studies provide adversarial cyber-attack examples crafted via patch-like and mask-like perturbations as direct input to an ML model without questioning if they are viable for a real communication network [9, 44]. This may result in misleading evaluations where the ML models are tested against unrealistic examples that they will not encounter in a real deployment scenario with real network traffic.

Due to their lack of constraints, it is very difficult to transfer the perturbation crafting processes of the image classification domain to the NID domain. A patch-like crafting process could be performed, changing the flow from a TCP connection to another protocol, or from port number 80 to another port, but these modifications would not be useful for a lengthy DoS. The communication protocol, the connection flag, and the port must remain the same, otherwise the crafted example will no longer be a flow of the Slowloris class [45, 46]. Likewise, a mask-like crafting process may increase or decrease the values of the minimum and maximum IAT, but not all perturbations will be suitable for a real communication network.

Three diferent examples may be generated for the considered Slowloris flow: the first with a minimum IAT of 22 and a maximum IAT of 28 seconds, the second with 18 and 32, and the third with 26 and 24. Even though all three may deceive an ML model and be misclassified as belonging to the benign class, only the first is a harmful Slowloris flow. The second example is actually a harmless flow because the considered web server will terminate the connection at the 31st second, before a packet is received at the 32nd second, preventing the functionality of this type of DoS [47]. In turn, the third example would not even be possible in a real communication network because a flow with packets at least every 26 seconds cannot also have packets at most every 24 seconds [39] (see Figure 5).

Despite all examples following similar mask-like perturbations of increasing and decreasing some numerical variables of the flow, they would lead to very different outcomes in a communication network and only one example could be used against a real NID system. Therefore, a successful adversarial attack is not guaranteed to be a successful cyber-attack. [48].

To ensure that an adversarial example represents a real network traffic flow that can be transmitted through a real communication network, the constraints of the utilized communication protocols and the malicious pur-

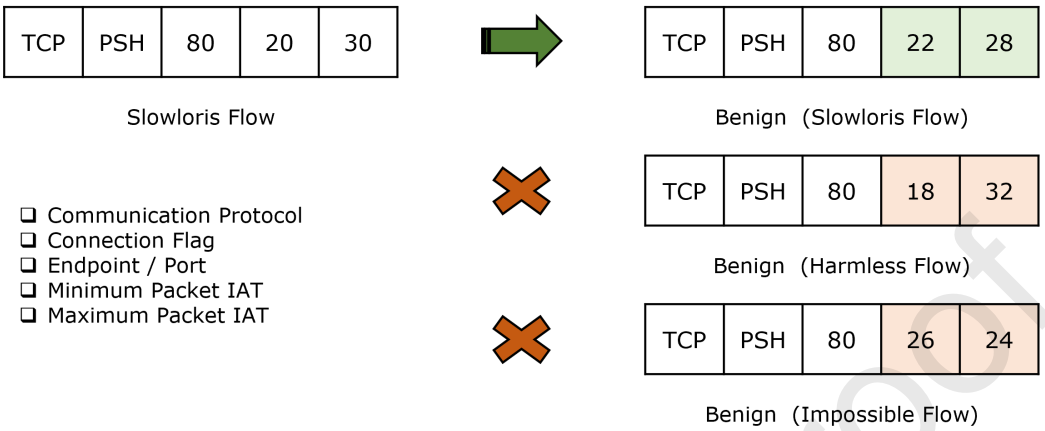

Figure 5: Adversarial perturbation on a network traffic flow, based on [48].

pose and functionality of a cyber-attack must be taken into account when generating the perturbations [49, 50]. Nonetheless, despite the current difficulty in creating realistic adversarial cyber-attack examples, the growing popularity of adversarial ML is leading to the development of novel methods to attack various types of algorithms, which is very concerning for the security of intelligent systems [9, 37, 8].

4. Attack Methods

The susceptibility of ML to adversarial examples can be exploited for diverse malicious purposes with methods that automatically generate the required perturbations. An attacker targeting an intelligent system may use multiple methods to perform a wide range of attacks, which can be divided into two primary categories: **poisoning attacks** during a model's training phase, and **evasion attacks** during the inference phase [51].

Poisoning attacks inject adversarial examples in a model's training data to compromise its internal reasoning and decision boundaries. These attacks can perform **model corruption** that make it completely unusable, or even introduce **hidden backdoors** that make it exhibit a biased behavior in specific samples, which is difficult to detect and explain because the model only deviates from its expected behavior when triggered by very specific perturbations [52, 53]. This is a serious security risk for organizations that heavily rely on third-party datasets or outsource their cybersecurity solutions, such

as the development of facial recognition models for biometric authentication systems [54, 55]. Nonetheless, since NID systems are commonly developed in secure environments with thoroughly verified network traffic data, an external attacker does not usually have access to an ML model to compromise it during its training phase [49, 10].

On the other hand, evasion attacks use adversarial examples to deceive a vulnerable model after it has been deployed. The misclassifications caused by these attacks can be directly used to **evade detection** from an intelligent security system, or for more complex goals, such as **membership inference** and **attribute inference** to check if a model was trained with a certain sample and certain features, **model inversion** to reconstruct a training set, and **model extraction** to steal its functionality and replicate it in a substitute model [56, 57, 58]. If confidential or proprietary information is used to train a model, an attacker can cause significant damage to an organization by gathering that information during the inference phase [59, 60]. Even though a model must be queried many times to obtain the information, advances in wireless and IoT technologies are making NID systems process larger and larger amounts of network traffic, which substantially increases query opportunities and therefore the feasibility of evasion attacks [61, 62, 63].

In recent years, numerous methods have been created to automate the misclassification attempts for evasion attacks. A method may require access to a model in one of three possible settings: **black-box**, **gray-box**, and **white-box**. The first is model-agnostic and solely queries a model's predictions, whereas the second may also require knowledge of its architecture or the utilized features, and the third needs full access to its internal parameters [64, 30]. Additionally, a black-box or gray-box method may solely use class predictions, a **decision-based** approach, or require a model to output the confidence scores of the predictions, a **score-based** approach [65, 10]. These characteristics affect the choice of an adversarial method because it must be able to attack the targeted model and system, while also being useful to the fulfillment of the goals of the attacker.

Since the focus of adversarial ML has been image classification, the common attack approach is to freely exploit the internal gradients of an ANN in a white-box setting [51, 36]. Consequently, most state-of-the-art methods do not support other settings nor other models, which severely limits their applicability to other domains. Considering that a deployed NID system is securely isolated, having full access to a model and its feedback is highly unlikely, and an attacker will only know if a certain example evades

detection if the entire cyber-attack is successfully completed. This interaction corresponds to a decision-based approach in black-box or gray-box settings, depending on the available system information about the utilized model and feature set [49, 34]. Furthermore, various other types of ML models can be used for classification tasks with tabular data. For instance, tree-based algorithms and ensembles like Random Forest (RF) are remarkably well-established for NID, but are also susceptible to adversarial attacks [66, 16, 67, 68]. Therefore, an attacker will have to resort to methods that support these models and all the specificities of NID.

Various adversarial evasion attack methods have been made open-source software and have started being used to target the ML models of intelligent NID systems. Table 3 summarizes the characteristics of the most relevant methods of the current literature that have been used in NID, noting if they could potentially fulfill the constraints of complex communication networks. Even though some methods were introduced as suitable for a black-box setting, they require knowledge of the utilized feature set to determine how which feature will be perturbed, so they were categorized as being in the gray-box setting. The 'Scores' keyword corresponds to models that can output confidence scores for a score-based approach. In turn, the 'Gradients' keyword corresponds to models that provide full access to their internal loss gradients, which includes ANNs.

Several methods initially developed for the generation of adversarial images have been adapted to generate adversarial network traffic flows. However, most do not account for the constraints of the utilized communication protocols nor the functionalities of the cyber-attacks, so only a few could potentially generate realistic examples [10, 68].

Both the Jacobian-based Saliency Map Attack (JSMA) [72] and the One Pixel attack [78] were developed to attack image classification models, but their perturbation crafting processes could be used to preserve the structure of a traffic flow. The former minimizes the number of modified pixels, requiring full access to the internal gradients of an ANN in a white-box setting, whereas the latter only modifies a single pixel, based on the confidence scores of a model in a black-box setting. These methods only perturb the most appropriate features in a decision boundary without affecting the remaining ones, which can preserve the correlations between most features of a flow.

Nonetheless, these methods freely generate the perturbations for the few modified features. When adapted to network flows, this lack of constraints could lead to values that are incompatible with the remaining features, which

Table 3: Characteristics of relevant adversarial evasion attack methods.

Method	Setting	Models	Constraints	Reference
BIM	White-box	Gradients	✗	[24]
C&W	White-box	Gradients	✗	[69]
DeepFool	White-box	Gradients	✗	[70]
FGSM	White-box	Gradients	✗	[23]
Hierarchical	White-box	Gradients	✗	[41]
Houdini	White-box	Gradients	✗	[71]
JSMA	White-box	Gradients	✓	[72]
PGD	White-box	Gradients	✗	[73]
Structured	White-box	Gradients	✗	[74]
GSA-GAN	Gray-box	Scores	✗	[75]
IDS-GAN	Gray-box	Scores	✗	[46]
Polymorphic	Gray-box	Scores	✓	[45]
A2PM	Gray-box	Any	✓	[47]
DoSBoundary	Gray-box	Any	✓	[40]
BFAM	Black-box	Scores	✗	[76]
BMI-FGSM	Black-box	Scores	✗	[77]
OnePixel	Black-box	Scores	✓	[78]
RL-S2V	Black-box	Scores	✗	[79]
WGAN	Black-box	Scores	✗	[80]
ZOO	Black-box	Scores	✗	[81]
Boundary	Black-box	Any	✗	[82]
CGAN	Black-box	Any	✗	[83]
CVAE	Black-box	Any	✗	[84]
GADGET	Black-box	Any	✗	[85]
HopSkipJump	Black-box	Any	✗	[86]
Optimization	Black-box	Any	✗	[87]

would result in mostly harmless or impossible flows and only a few occasional realistic examples created by chance. To generate high-quality examples on a more regular basis, some methods have been specifically developed to tackle the constraints of the NID domain.

The Polymorphic attack [45] addresses the preservation of original class characteristics to create examples compatible with a cyber-attack's purpose. A feature selection algorithm is applied in a gray-box setting to obtain the most impactful features for the distinction between benign traffic and cyber-attack classes in a dataset. Then, the remaining features, which are considered non-relevant for the functionality of a cyber-attack, are perturbed by a Wasserstein Generative Adversarial Network (WGAN) [80]. Despite the WGAN not accounting for the constraints of the remaining features, the most important features of each class are not modified, so the main characteristics required for a successful cyber-attack may be preserved.

The distinction between benign traffic and cyber-attack classes was further explored in the DoSBoundary attack [40] and the Adaptative Perturbation Pattern Method (A2PM) [47]. Both iteratively optimize the perturbations that are performed on each feature of a traffic flow according to the constraints of a communication network and the functionality of each cyber-attack class. The former requires expert knowledge to manually configure the specific perturbations of each feature, whereas the latter only needs to know the utilized feature set and relies on adaptative patterns to learn the characteristics of each feature. Despite these methods requiring many queries to a model and knowledge of the feature set, which corresponds to a gray-box setting, they can generate constrained adversarial examples that preserve the correlations between the features of a network traffic flow.

Due to the different characteristics of existing methods and diverse goals of attackers, efforts are being made to systematize the possible attack vectors in the Adversarial Threat Landscape for Artificial-Intelligence Systems [88] knowledge base, and to complement it with case studies and demonstrations based on real-world observations. As novel adversarial methods continue to be developed, it is becoming essential to raise awareness of the diverse strategies that attackers can use to exploit ML models and the security risks they pose to modern organizations.

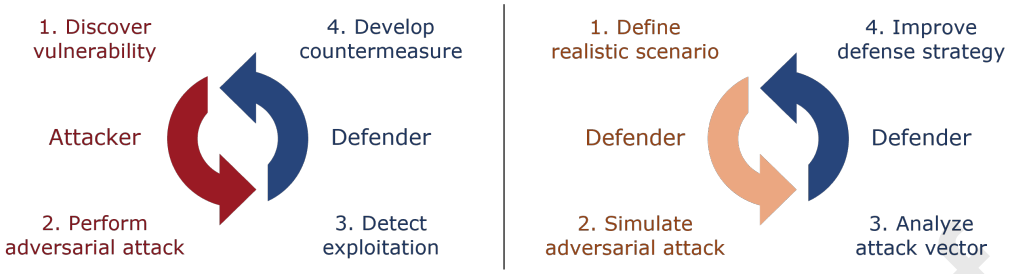

Figure 6: Adversarial arms race (left) and security-by-design (right), based on [90].

5. Defense Strategies

The growing ML attack surface led to a never-ending arms race where attackers continuously exploit newly discovered vulnerabilities and defenders develop countermeasures against each novel threat. However, the defenders are always a step behind because it can take a long time until the effects of an attack are detected, and then it is difficult to retrace it and develop a specific countermeasure [89]. To get ahead of attackers, organizations should follow a security-by-design development approach and proactively search for vulnerabilities themselves. By simulating adversarial attacks in realistic scenarios and analyzing entire attack vectors, ML engineers and security practitioners can anticipate possible threats and use that knowledge to preemptively revise and improve their defense strategy (see Figure 6) [90].

A defense strategy can combine multiple techniques to address different security concerns. Due to their proven value against several adversarial attacks, some defenses have been standardized across the scientific literature, divided into two primary categories: **proactive defenses** during a model's training phase, and **reactive defenses** during the inference phase [36].

Regarding reactive defenses, they attempt to mitigate the effects of corrupted data on a model's predictions by safely processing its input and output data. These defenses can rely on several preprocessing techniques, such as **data denoising** and **feature squeezing** to reduce the search space for an attack, and postprocessing techniques, such as mechanisms that deal with **model uncertainty** and require predictions with **high confidence** scores [64, 57, 91]. Even though reactive defenses can be valuable against both erroneous data and adversarial attacks purposely exploiting a model, they represent an additional software layer that attempts to encapsulate a vulnerable model. This layer is always needed for a NID system to convert the

recorded network traffic into the utilized feature set and then convert the predictions of one or more models into relevant warnings and alerts, but it does not fully protect those models [49, 19].

On the other hand, proactive defenses tackle the susceptibility of ML to adversarial examples, aiming to reduce the vulnerabilities and intrinsically improve a model's robustness against adversarial examples during its training phase. These defenses include several techniques, such as **adversarial training** with perturbed samples, **regularization** to better calibrate the learning process, and **defensive distillation** to create smaller models less sensitive to data variations [23, 9, 92, 28]. It is not yet clear how to completely resolve this susceptibility and achieve an adversarially robust generalization in a classification task, but progress is being made in robustness research with regularization and optimization techniques [93, 94, 95]. This gives ML engineers and security practitioners better tools to address ML security during the entire lifecycle of an intelligent system, including its development, testing, deployment, and maintenance phases.

Most proactive defenses are focused on improving the robustness of deep learning algorithms based on ANNs against evasion attacks with adversarial images [96, 97, 98, 99], although some also take measures against backdoors [53, 55]. Despite ANN defenses being difficult to apply to other models and domains, the protection of tree-based algorithms has been drawing attention for intelligent cybersecurity solutions [100, 101]. Some defenses have been developed to improve the robustness of entire tree ensembles at once [102, 103], whereas others address each individual decision tree at a time [104, 105]. Still, proactive defenses often trade-off some performance on regular samples to improve performance on adversarial examples. This trade-off affects the choice of a defense strategy because the utilized techniques need to balance adversarial robustness and generalization to regular network traffic.

Defense strategies continue to be enhanced with better techniques, but the most effective and widespread defense is still adversarial training because it anticipates the data variations that an ML model may encounter when it is deployed [106, 107, 108]. Augmenting a training set with examples created by one or more adversarial attack methods enables a model to learn additional characteristics that the samples of each class can exhibit, so it becomes significantly harder to deceive it. This augmented training data with more data variations can improve a model's robustness not only against attack methods similar to the utilized ones, but also against a wide range of attacks that perform different data perturbations [93, 109, 110].

Nonetheless, augmenting a model's training set with the examples created by adversarial attack methods may not be as beneficial as it seems. Even though it is meant to improve robustness, training with unrealistic samples will make a model learn distorted characteristics that will not be exhibited by regular samples [20]. This raises a major security concern because including unrealistic data in a training set can not only be detrimental to a model's generalization, but also lead to accidental data poisoning and to the introduction of hidden backdoors that leave a model even more vulnerable [53]. Therefore, to improve a model's robustness to adversarial data without deteriorating its generalization to regular network traffic flows, it is essential to ensure that adversarial training is performed with realistic examples that could be transmitted through a communication network and preserve the functionality and malicious purpose of a cyber-attack [34, 48].

6. Future Directions

Adversarial data perturbations are very concerning for ML security and reliability. Despite the current difficulty in the transferability of the patch-like and mask-like crafting processes of the image classification domain to the NID domain, there are more sophisticated approaches being specifically designed for communication networks. Nonetheless, an adversarial example that successfully deceives an ML model is not guaranteed to be a successful cyber-attack in a real communication network.

By inspecting the third example of Figure 5, it can be observed that the reason it is impossible is because it does not comply with the inherent data structure of a network traffic flow. On the other hand, the reason that the second example is harmless is because it does not comply with the intended functionality of the Slowloris class of DoS attacks. Therefore, it is possible to define two fundamental properties that are required for an adversarial example to resemble a real data sample:

- Validity: Compliance with the constraints of a domain, following its inherent data structure.
- Coherence: Compliance with the constraints of a specific class, following the characteristics that distinguish it from other classes.

Even though validity was already taken into account in the reviewed adversarial methods, it is imperative to address it together with coherence to

fully achieve adversarial realism. Hence, to ensure that their experiments are realistic, researchers must use examples that represent valid network traffic capable of being transmitted through a real communication network, as well as coherent cyber-attack flows capable of fulfilling their intended functionality and the malicious purposes of an attacker. Novel approaches should support more rigorous configurations to address the complex constraints of the tabular data format and of the time-related characteristics of network traffic flows, creating constrained adversarial examples capable of evading detection while preserving their realism.

As more realistic perturbation crafting processes are developed, they may be used by data scientists and engineers to improve their AI applications with high-quality data, but they may also be used by attackers with malicious intents to disrupt the critical business processes of an organization. An attacker will usually have limited knowledge of the model and feature set utilized in a NID system, corresponding to a black-box or gray-box setting, and will only be able to interact with it in a decision-based approach, without direct feedback and only knowing if a certain example deceived a model if the entire cyber-attack is successfully completed. This limits the possible attack vectors and hinders the feasibility of most adversarial methods of the current literature, but novel attack methods may be developed to address the constraints of complex communication networks and deceive the ML models of NID systems in these scenarios.

To tackle the growing ML attack surface and counteract the disruptions caused by the known attacks, it is becoming essential to enforce a security-by-design approach throughout the entire lifecycle of intelligent systems by simulating realistic evasion attack vectors to assess a system's resilience in edge cases. Reactive and proactive defenses should be combined to attempt to encapsulate ML models in secure software layers and also improve their intrinsic robustness against faulty input data.

Still, efforts to improve a model's robustness against all possible adversarial examples that might occur should not disregard the importance of the model's generalization to regular network traffic that it will definitely encounter. It may not be possible to fully safeguard ML models from adversarial cyber-attack examples, but ML engineers and security practitioners should stay up-to-update with security best-practices and preemptively test their ML models against the novel threats they may face.

7. Conclusion

This work investigated the adversarial ML approaches of the current scientific literature that use realistic adversarial examples and could be applied in real ML development and deployment scenarios in the NID domain, fulfilling the requirements of the diverse cyber-attack classes and of the tabular data format and time-related characteristics of network traffic flows.

From the 936 records initially retrieved, over 75% did not present novel methods nor strategies, which demonstrates that it is difficult for researchers to find innovative approaches in a regular search. The 82 records that remained after the screening phase, which correspond to approximately 9% of the found publications, presented relevant advances in the use of adversarial ML in the NID domain, and were included in the review together with 16 additional records found through backward snowballing.

The information present in the 98 publications was consolidated and combined into three sections, each discussing its respective RQ, systematizing the approaches, and highlighting the most relevant aspects. An additional section described the open challenges regarding the main RQ, defined the fundamental properties that are required for an adversarial example to be realistic, and provided guidelines for researchers to ensure that their future experiments are adequate for a real communication network.

A security-by-design approach throughout the entire ML lifecycle is becoming essential to tackle the growing ML attack surface, and research efforts continue to be made to better protect various types of algorithms with reactive and proactive defenses. However, there is still a lack of realism in the state-of-the-art perturbation crafting processes, which hinders the development of secure defense strategies. It is pertinent to continue the research efforts to improve the robustness and trustworthiness of ML and of the intelligent cybersecurity solutions that rely on it.

Author Contributions. Conceptualization, J.V.; methodology, J.V. and E.M.; validation, E.M. and I.P.; investigation, J.V., E.M. and I.P.; writing, J.V.; supervision, I.P.; project administration, I.P.; funding acquisition, I.P. All authors have read and agreed to the published version of the manuscript.

Funding. The present work was partially supported by the Norte Portugal Regional Operational Programme (NORTE 2020), under the PORTUGAL 2020 Partnership Agreement, through the European Regional Development

Fund (ERDF), within project "Cybers SeC IP" (NORTE-01-0145-FEDER-000044). This work has also received funding from UIDB/00760/2020.

Conflicts of Interest. The authors declare no conflict of interest.

References

[1] European Union Agency for Cybersecurity, N. Christoforatos, I. Lella, E. Rekleitis, C. V. Heurck, A. Zacharis, Cyber Europe 2022: After Action Report, Tech. rep. (2022). doi:10.2824/397622.

[2] Verizon, Data Breach Investigations Report 2022, Tech. rep. (2022).
URL `https://www.verizon.com/business/resources/reports/dbir/`

[3] IBM Security, Ponemon Institute, Cost of a Data Breach Report 2022, Tech. rep. (2022).
URL `https://www.ibm.com/reports/data-breach`

[4] S. Mansfield-Devine, Sophos: The State of Ransomware 2022, Computer Fraud & Security 2022 (5) (2022). doi:10.12968/s1361-3723(22)70573-8.

[5] European Union Agency for Cybersecurity, I. Lella, E. Tsekmezoglou, R. S. Naydenov, C. Ciobanu, A. Malatras, M. Theocharidou, ENISA Threat Landscape 2022, Tech. rep. (2022). doi:10.2824/764318.

[6] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, R. Fergus, Intriguing properties of neural networks, in: Proceedings of the 2nd International Conference on Learning Representations, ICLR 2014, 2014, pp. 1–10. doi:10.48550/ARXIV.1312.6199.
URL `https://arxiv.org/abs/1312.6199`

[7] European Union Agency for Cybersecurity, A. Malatras, G. Dede, AI Cybersecurity Challenges: Threat Landscape for Artificial Intelligence, Tech. rep. (2020). doi:10.2824/238222.

[8] R. S. Siva Kumar, M. Nyström, J. Lambert, A. Marshall, M. Goertzel, A. Comissoneru, M. Swann, S. Xia, Adversarial Machine Learning-Industry Perspectives, in: 2020 IEEE Security and Privacy Workshops (SPW), 2020, pp. 69–75. doi:10.1109/SPW50608.2020.00028.

[9] N. Martins, J. M. Cruz, T. Cruz, P. Henriques Abreu, Adversarial Machine Learning Applied to Intrusion and Malware Scenarios: A Systematic Review, IEEE Access 8 (2020) 35403–35419. doi:10.1109/ACCESS.2020.2974752.

[10] K. He, D. D. Kim, M. R. Asghar, Adversarial machine learning for network intrusion detection systems: A comprehensive survey, IEEE Communications Surveys & Tutorials 25 (1) (2023) 538–566. doi:10.1109/COMST.2022.3233793.

[11] D. Moher, et al., Preferred reporting items for systematic review and meta-analysis protocols (PRISMA-P) 2015 statement, Systematic Reviews 4 (1) (2015) 1. doi:10.1186/2046-4053-4-1.
URL https://doi.org/10.1186/2046-4053-4-1

[12] Elsevier, ScienceDirect Search Source (3 2023).
URL https://www.sciencedirect.com/search

[13] Association for Computing Machinery, ACM Digital Library Search Source (3 2023).
URL https://dl.acm.org/search/advanced

[14] Institute of Electrical and Electronics Engineers, IEEE Xplore Search Source (3 2023).
URL https://ieeexplore.ieee.org/search/advanced

[15] Multidisciplinary Digital Publishing Institute, MDPI Search Source (3 2023).
URL https://www.mdpi.com/search

[16] H. Liu, B. Lang, Machine learning and deep learning methods for intrusion detection systems: A survey, Applied Sciences (Switzerland) 9 (20) (2019). doi:10.3390/app9204396.

[17] R. L. Alaoui, E. H. Nfaoui, Deep Learning for Vulnerability and Attack Detection on Web Applications: A Systematic Literature Review, Future Internet 14 (4) (2022). doi:10.3390/fi14040118.
URL https://www.mdpi.com/1999-5903/14/4/118

[18] O. Salman, I. H. Elhajj, A. Kayssi, A. Chehab, A review on machine learning–based approaches for Internet traffic classification, Annals of Telecommunications 75 (11) (2020) 673–710. doi:10.1007/s12243-020-00770-7.
URL https://doi.org/10.1007/s12243-020-00770-7

[19] A. Thakkar, R. Lohiya, A Review on Machine Learning and Deep Learning Perspectives of IDS for IoT: Recent Updates, Security Issues, and Challenges, Springer Netherlands, 2020. doi:10.1007/s11831-020-09496-0.
URL https://doi.org/10.1007/s11831-020-09496-0

[20] D. Stutz, M. Hein, B. Schiele, Disentangling Adversarial Robustness and Generalization, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 6969–6980. doi:10.1109/CVPR.2019.00714.

[21] A. Fawzi, O. Fawzi, P. Frossard, Fundamental limits on adversarial robustness, in: Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, 2015.

[22] S. Sabour, Y. Cao, F. Faghri, D. J. Fleet, Adversarial Manipulation of Deep Representations, in: Proceedings of the 4th International Conference on Learning Representations, ICLR 2016, 2016. doi:10.48550/ARXIV.1511.05122.
URL https://arxiv.org/abs/1511.05122

[23] I. J. Goodfellow, J. Shlens, C. Szegedy, Explaining and harnessing adversarial examples, in: Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015, 2015, pp. 1–11. doi:10.48550/ARXIV.1412.6572.
URL https://arxiv.org/abs/1412.6572

[24] A. Kurakin, I. Goodfellow, S. Bengio, Adversarial examples in the physical world, in: Proceedings of the 5th International Conference on Learning Representations, ICLR 2017, 2017, pp. 1–14. doi:10.48550/ARXIV.1607.02533.
URL https://arxiv.org/abs/1607.02533

[25] N. Papernot, P. McDaniel, I. Goodfellow, Transferability in Machine Learning: from Phenomena to Black-Box Attacks using Adversarial Samples, arXiv (2016). doi:10.48550/ARXIV.1605.07277.

[26] P. Tabacof, E. Valle, Exploring the Space of Adversarial Images, in: 2016 IEEE International Joint Conference on Neural Networks (IJCNN), 2016. doi:10.48550/ARXIV.1510.05328.
URL https://arxiv.org/abs/1510.05328

[27] K. Ren, T. Zheng, Z. Qin, X. Liu, Adversarial Attacks and Defenses in Deep Learning, Engineering 6 (3) (2020) 346–360. doi:10.1016/j.eng.2019.12.012.

[28] J. Zhang, C. Li, Adversarial Examples: Opportunities and Challenges, IEEE Transactions on Neural Networks and Learning Systems 31 (7) (2020) 2578–2593. doi:10.1109/TNNLS.2019.2933524.

[29] R. R. Wiyatno, A. Xu, O. Dia, A. de Berker, Adversarial Examples in Modern Machine Learning: A Review, arXiv (2019). doi:10.48550/ARXIV.1911.05268.

[30] J. Li, Y. Liu, T. Chen, Z. Xiao, Z. Li, J. Wang, Adversarial Attacks and Defenses on Cyber-Physical Systems: A Survey, IEEE Internet of Things Journal 7 (6) (2020) 5103–5115. doi:10.1109/JIOT.2020.2975654.

[31] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, D. Song, Robust Physical-World Attacks on Deep Learning Visual Classification, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 1625–1634. doi:10.1109/CVPR.2018.00175.

[32] T. Brown, D. Mane, A. Roy, M. Abadi, J. Gilmer, Adversarial Patch, in: Advances in Neural Information Processing Systems, 2017. doi:10.48550/ARXIV.1712.09665.
URL https://arxiv.org/pdf/1712.09665.pdf

[33] D. Edwards, D. B. Rawat, Study of Adversarial Machine Learning with Infrared Examples for Surveillance Applications, Electronics 9 (8) (2020). doi:10.3390/electronics9081284.
URL https://www.mdpi.com/2079-9292/9/8/1284

[34] I. Rosenberg, A. Shabtai, Y. Elovici, L. Rokach, Adversarial Machine Learning Attacks and Defense Methods in the Cyber Security Domain, ACM Comput. Surv. 54 (5) (2021). doi:10.1145/3453158.
URL https://doi.org/10.1145/3453158

[35] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, P. Frossard, Universal Adversarial Perturbations, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 86–94. doi:10.1109/CVPR.2017.17.

[36] X. Yuan, P. He, Q. Zhu, X. Li, Adversarial Examples: Attacks and Defenses for Deep Learning, IEEE transactions on neural networks and learning systems 30 (9) (2019) 2805–2824. doi:10.1109/TNNLS.2018.2886017.

[37] P. Papadopoulos, O. Thornewill von Essen, N. Pitropakis, C. Chrysoulas, A. Mylonas, W. J. Buchanan, Launching Adversarial Attacks against Network Intrusion Detection Systems for IoT, Journal of Cybersecurity and Privacy 1 (2) (2021) 252–273. doi:10.3390/jcp1020014.

[38] M. J. Hashemi, G. Cusack, E. Keller, 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, Association for Computing Machinery, New York, NY, USA, 2019, p. 14–21. doi:10.1145/3359992.3366642.

[39] M. A. Merzouk, F. Cuppens, N. Boulahia-Cuppens, R. Yaich, Investigating the practicality of adversarial evasion attacks on network intrusion detection, Annals of Telecommunications (2022). doi:10.1007/s12243-022-00910-1.
URL https://doi.org/10.1007/s12243-022-00910-1

[40] X. Peng, W. Huang, Z. Shi, Adversarial attack against dos intrusion detection: An improved boundary-based method, in: 2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI), 2019, pp. 1288–1295. doi:10.1109/ICTAI.2019.00179.

[41] X. Zhou, W. Liang, W. Li, K. Yan, S. Shimizu, K. I.-K. Wang, Hierarchical Adversarial Attacks Against Graph-Neural-Network-Based IoT Network Intrusion Detection System, IEEE Internet of Things Journal 9 (12) (2022) 9310–9319. doi:10.1109/JIOT.2021.3130434.

[42] T. Shorey, D. Subbaiah, A. Goyal, A. Sakxena, A. K. Mishra, Performance Comparison and Analysis of Slowloris, GoldenEye and Xerxes DDoS Attack Tools, 2018 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2018 (2018) 318–322doi:10.1109/ICACCI.2018.8554590.

[43] Z. Al-Qudah, M. Rabinovich, M. Allman, Web Timeouts and Their Implications, in: A. Krishnamurthy, B. Plattner (Eds.), Passive and Active Measurement, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, pp. 211–221.

[44] J. Vitorino, T. Dias, T. Fonseca, I. Praça, E. Maia, Constrained Adversarial Learning and its applicability to Automated Software Testing: a systematic review, arXiv (2023). doi:10.48550/ARXIV.2303.07546.

[45] R. Chauhan, U. Sabeel, A. Izaddoost, S. Shah Heydari, Polymorphic Adversarial Cyberattacks Using WGAN, Journal of Cybersecurity and Privacy 1 (4) (2021) 767–792. doi:10.3390/jcp1040037.

[46] Z. Lin, Y. Shi, Z. Xue, IDSGAN: Generative Adversarial Networks for Attack Generation Against Intrusion Detection, in: J. Gama, T. Li, Y. Yu, E. Chen, Y. Zheng, F. Teng (Eds.), Advances in Knowledge Discovery and Data Mining, Springer International Publishing, Cham, 2022, pp. 79–91.

[47] J. Vitorino, N. Oliveira, I. Praça, Adaptative Perturbation Patterns: Realistic Adversarial Learning for Robust Intrusion Detection, Future Internet 14 (4) (2022) 108. doi:10.3390/fi14040108.
URL https://www.mdpi.com/1999-5903/14/4/108

[48] J. Vitorino, I. Praça, E. Maia, Towards Adversarial Realism and Robust Learning for IoT Intrusion Detection and Classification, Annals of Telecommunications (3 2023). doi:10.1007/s12243-023-00953-y.

[49] G. Apruzzese, M. Andreolini, L. Ferretti, M. Marchetti, M. Colajanni, Modeling Realistic Adversarial Attacks against Network Intrusion Detection Systems, Digital Threats: Research and Practice 1 (1) (2021). doi:10.1145/3469659.

[50] A. McCarthy, P. Andriotis, E. Ghadafi, P. Legg, Feature Vulnerability and Robustness Assessment against Adversarial Machine Learning Attacks, in: Proceedings of the 2021 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA), 2021, pp. 1–8. doi:10.1109/CyberSA52016.2021.9478199.

[51] N. Pitropakis, E. Panaousis, T. Giannetsos, E. Anastasiadis, G. Loukas, A taxonomy and survey of attacks against machine learning, Computer Science Review 34 (2019) 100199. doi:10.1016/j.cosrev.2019.100199.
URL https://doi.org/10.1016/j.cosrev.2019.100199

[52] T. Gu, B. Dolan-Gavitt, S. Garg, BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain, arXiv (2017). doi:10.48550/ARXIV.1708.06733.

[53] Y. Li, Y. Jiang, Z. Li, S.-T. Xia, Backdoor Learning: A Survey, IEEE Transactions on Neural Networks and Learning Systems (2022) 1–18doi:10.1109/TNNLS.2022.3182979.

[54] X. Chen, C. Liu, B. Li, K. Lu, D. Song, Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning, arXiv (2017). doi:10.48550/ARXIV.1712.05526.

[55] B. Wang, Y. Yao, S. Shan, H. Li, B. Viswanath, H. Zheng, B. Y. Zhao, Neural Cleanse: Identifying and Mitigating Backdoor Attacks in Neural Networks, in: 2019 IEEE Symposium on Security and Privacy (SP), 2019, pp. 707–723. doi:10.1109/SP.2019.00031.

[56] M. Fredrikson, S. Jha, T. Ristenpart, Model Inversion Attacks That Exploit Confidence Information and Basic Countermeasures, in: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, CCS '15, Association for Computing Machinery, New York, NY, USA, 2015, pp. 1322–1333. doi:10.1145/2810103.2813677.
URL https://doi.org/10.1145/2810103.2813677

[57] S. Qiu, Q. Liu, S. Zhou, C. Wu, Review of Artificial Intelligence Adversarial Attack and Defense Technologies, Applied Sciences 9 (5) (2019). doi:10.3390/app9050909.
URL https://www.mdpi.com/2076-3417/9/5/909

[58] R. Shokri, M. Stronati, C. Song, V. Shmatikov, Membership Inference Attacks Against Machine Learning Models, in: 2017 IEEE Symposium on Security and Privacy (SP), 2017, pp. 3–18. doi:10.1109/SP.2017.41.

[59] B. Hitaj, G. Ateniese, F. Perez-Cruz, Deep Models Under the GAN: Information Leakage from Collaborative Deep Learning, arXiv (2017). doi:10.48550/ARXIV.1702.07464.

[60] M. Veale, R. Binns, L. Edwards, Algorithms that remember: model inversion attacks and data protection law, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 376 (2133) (2018) 20180083. doi:10.1098/rsta.2018.0083.
URL https://royalsocietypublishing.org/doi/10.1098/rsta.2018.0083

[61] J. Aiken, S. Scott-Hayward, Investigating Adversarial Attacks against Network Intrusion Detection Systems in SDNs, in: 2019 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), 2019, pp. 1–7. doi:10.1109/NFV-SDN47374.2019.9040101.

[62] B. Flowers, R. M. Buehrer, W. C. Headley, Evaluating Adversarial Evasion Attacks in the Context of Wireless Communications, IEEE Transactions on Information Forensics and Security 15 (2020) 1102–1113. doi:10.1109/TIFS.2019.2934069.

[63] O. Ibitoye, O. Shafiq, A. Matrawy, Analyzing Adversarial Attacks against Deep Learning for Intrusion Detection in IoT Networks, in: 2019 IEEE Global Communications Conference (GLOBECOM), 2019, pp. 1–6. doi:10.1109/GLOBECOM38437.2019.9014337.

[64] A. Chakraborty, M. Alam, V. Dey, A. Chattopadhyay, D. Mukhopadhyay, A survey on adversarial attacks and defences, CAAI Transactions on Intelligence Technology 6 (1) (2021) 25–45. doi:10.1049/cit2.12028.
URL https://onlinelibrary.wiley.com/doi/10.1049/cit2.12028

[65] A. Ilyas, L. Engstrom, A. Athalye, J. Lin, Black-box Adversarial Attacks with Limited Queries and Information, in: J. Dy, A. Krause (Eds.), Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Vol. 80 of Proceedings of Machine Learning Research, PMLR, 2018, pp. 2137–2146.
URL https://proceedings.mlr.press/v80/ilyas18a.html

[66] M. C. Belavagi, B. Muniyal, Performance Evaluation of Supervised Machine Learning Algorithms for Intrusion Detection, Procedia Computer Science 89 (2016) 117–123. doi:10.1016/j.procs.2016.06.016.
URL http://dx.doi.org/10.1016/j.procs.2016.06.016

[67] R. Primartha, B. A. Tama, Anomaly detection using random forest: A performance revisited, in: Proceedings of 2017 International Conference on Data and Software Engineering, 2018, pp. 1–6. doi:10.1109/ICODSE.2017.8285847.

[68] M. Pujari, Y. Pacheco, B. Cherukuri, W. Sun, A Comparative Study on the Impact of Adversarial Machine Learning Attacks on Contemporary Intrusion Detection Datasets, SN Computer Science 3 (5) (2022) 412. doi:10.1007/s42979-022-01321-8.

[69] N. Carlini, D. Wagner, Towards Evaluating the Robustness of Neural Networks, in: Proceedings - IEEE Symposium on Security and Privacy, IEEE, 2017, pp. 39–57. doi:10.1109/SP.2017.49.

[70] S. M. Moosavi-Dezfooli, A. Fawzi, P. Frossard, DeepFool: A Simple and Accurate Method to Fool Deep Neural Networks, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 2016-Decem, 2016, pp. 2574–2582. doi:10.1109/CVPR.2016.282.

[71] M. Cisse, Y. Adi, N. Neverova, J. Keshet, Houdini: Fooling Deep Structured Visual and Speech Recognition Models with Adversarial Examples, in: Advances in Neural Information Processing Systems, NIPS'17, Curran Associates Inc., Red Hook, NY, USA, 2017, pp. 6980–6990.

[72] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, A. Swami, The Limitations of Deep Learning in Adversarial Settings, in: 2016 IEEE European Symposium on Security and Privacy, 2016, pp. 372–387. doi:10.1109/EuroSP.2016.36.

[73] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, A. Vladu, Towards deep learning models resistant to adversarial attacks, in: Proceedings of the 6th International Conference on Learning Representations, ICLR 2018, 2018, pp. 1–28. doi:10.48550/ARXIV.1706.06083.

[74] K. Xu, S. Liu, P. Zhao, P. Y. Chen, H. Zhang, Q. Fan, D. Erdogmus, Y. Wang, X. Lin, Structured adversarial attack: Towards general implementation and better interpretability, Proceedings of the 7th International Conference on Learning Representations, ICLR 2019 (2019). doi:10.48550/ARXIV.1808.01664.

[75] Z. Wang, M. Gao, J. Li, J. Zhang, J. Zhong, Gray-Box Shilling Attack: An Adversarial Learning Approach, ACM Trans. Intell. Syst. Technol. 13 (5) (10 2022). doi:10.1145/3512352.
URL https://doi.org/10.1145/3512352

[76] S. Zhang, X. Xie, Y. Xu, A Brute-Force Black-Box Method to Attack Machine Learning-Based Systems in Cybersecurity, IEEE Access 8 (2020) 128250–128263. doi:10.1109/ACCESS.2020.3008433.

[77] J. Lin, L. Xu, Y. Liu, X. Zhang, Black-box adversarial sample generation based on differential evolution, Journal of Systems and Software 170 (2020). doi:10.1016/j.jss.2020.110767.

[78] J. Su, D. V. Vargas, K. Sakurai, One Pixel Attack for Fooling Deep Neural Networks, IEEE Transactions on Evolutionary Computation 23 (5) (2019) 828–841. doi:10.1109/TEVC.2019.2890858.

[79] H. Dai, H. Li, T. Tian, X. Huang, L. Wang, J. Zhu, L. Song, Adversarial Attack on Graph Structured Data, in: J. Dy, A. Krause (Eds.), Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Vol. 80 of Proceedings of Machine Learning Research, PMLR, 2018, pp. 1115–1124.
URL https://proceedings.mlr.press/v80/dai18b.html

[80] M. Arjovsky, S. Chintala, L. Bottou, Wasserstein Generative Adversarial Networks, in: D. Precup, Y. W. Teh (Eds.), Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Vol. 70 of Proceedings of Machine Learning Research, PMLR, 2017, pp. 214–223.
URL https://proceedings.mlr.press/v70/arjovsky17a.html

[81] P. Y. Chen, H. Zhang, Y. Sharma, J. Yi, C. J. Hsieh, ZOO: Zeroth order optimization based black-box atacks to deep neural networks without training substitute models, in: AISec 2017 - Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, co-located with CCS 2017, 2017, pp. 15–26. doi:10.1145/3128572.3140448.

[82] W. Brendel, J. Rauber, M. Bethge, Decision-based adversarial attacks: Reliable attacks against black-box machine learning models, in: Proceedings of the 6th International Conference on Learning Representations, ICLR 2018, 2018, pp. 1–12. doi:10.48550/ARXIV.1712.04248.
URL https://arxiv.org/abs/1712.04248

[83] M. Mirza, S. Osindero, Conditional Generative Adversarial Nets, arXiv (2014). doi:10.48550/ARXIV.1411.1784.

[84] K. Sohn, H. Lee, X. Yan, Learning Structured Output Representation using Deep Conditional Generative Models, in: C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, R. Garnett (Eds.), Advances in Neural Information Processing Systems, Vol. 28, Curran Associates, Inc., 2015.

[85] I. Rosenberg, A. Shabtai, L. Rokach, Y. Elovici, Generic Black-Box End-to-End Attack Against State of the Art API Call Based Malware Classifiers, in: Research in Attacks, Intrusions, and Defenses, Springer International Publishing, 2018, pp. 490–510. doi:10.1007/978-3-030-00470-5_23.
URL http://link.springer.com/10.1007/978-3-030-00470-5_23

[86] J. Chen, M. I. Jordan, M. J. Wainwright, HopSkipJumpAttack: A Query-Efficient Decision-Based Attack, in: 2020 IEEE Symposium on Security and Privacy (SP), 2020, pp. 1277–1294. doi:10.1109/SP40000.2020.00045.

[87] M. Cheng, H. Zhang, C. J. Hsieh, T. Le, P. Y. Chen, J. Yi, Query-efficient hard-label black-box attack: An optimization-based approach, in: Proceedings of the 7th International Conference on Learning Representations, ICLR 2019, 2019, pp. 1–12. doi:10.48550/ARXIV.1807.04457.

[88] MITRE ATLAS, Adversarial Threat Landscape for Artificial-Intelligence Systems (3 2023).
URL https://atlas.mitre.org/

[89] European Union Agency for Cybersecurity, A. Malatras, I. Agrafiotis, M. Adamczyk, Securing Machine Learning Algorithms, Tech. rep. (2022). doi:10.2824/874249.

[90] B. Biggio, G. Fumera, F. Roli, Security Evaluation of Pattern Classifiers under Attack, IEEE Transactions on Knowledge and Data Engineering 26 (4) (2014) 984–996. doi:10.1109/TKDE.2013.57.

[91] L. Smith, Y. Gal, Understanding Measures of Uncertainty for Adversarial Example Detection, in: 34th Conference on Uncertainty in Artificial Intelligence, UAI 2018 - Conference Track Proceedings, 2018. doi:10.48550/ARXIV.1803.08533.
URL https://arxiv.org/abs/1803.08533

[92] N. Papernot, P. McDaniel, X. Wu, S. Jha, A. Swami, Distillation as a Defense to Adversarial Perturbations Against Deep Neural Networks, in: 2016 IEEE Symposium on Security and Privacy (SP), 2016, pp. 582–597. doi:10.1109/SP.2016.41.

[93] T. Bai, J. Luo, J. Zhao, B. Wen, Q. Wang, Recent Advances in Adversarial Training for Adversarial Robustness, in: Z.-H. Zhou (Ed.), Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, International Joint Conferences on Artificial Intelligence Organization, 2021, pp. 4312–4321. doi:10.24963/ijcai.2021/591.
URL https://doi.org/10.24963/ijcai.2021/591

[94] L. Schmidt, K. Talwar, S. Santurkar, D. Tsipras, A. Madry, Adversarially robust generalization requires more data, Advances in Neural Information Processing Systems 2018-Decem (NeurIPS) (2018) 5014–5026. doi:10.48550/ARXIV.1804.11285.

[95] R. A. Khamis, M. O. Shafiq, A. Matrawy, Investigating resistance of deep learning-based ids against adversaries using min-max optimization, in: ICC 2020 - 2020 IEEE International Conference on Communications (ICC), 2020, pp. 1–7. doi:10.1109/ICC40277.2020.9149117.

[96] R. Feinman, R. R. Curtin, S. Shintre, A. B. Gardner, Detecting Adversarial Samples from Artifacts, in: Proceedings of the 34th International Conference on Machine Learning, ICML 2017, 2017. doi:10.48550/ARXIV.1703.00410.
URL https://arxiv.org/abs/1703.00410

[97] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, V. Lempitsky, Domain-adversarial training of neural networks, Advances in Computer Vision and Pattern Recognition (9783319583464) (2017) 189–209. doi:10.1007/978-3-319-58347-1_10.

[98] D. J. Miller, Z. Xiang, G. Kesidis, Adversarial Learning Targeting Deep Neural Network Classification: A Comprehensive Review of Defenses Against Attacks, Proceedings of the IEEE 108 (3) (2020) 402–433. doi:10.1109/JPROC.2020.2970615.

[99] F. Tramèr, A. Kurakin, N. Papernot, I. Goodfellow, D. Boneh, P. McDaniel, Ensemble adversarial training: Attacks and defenses, in: Proceedings of the 6th International Conference on Learning Representations, ICLR 2018, 2018, pp. 1–22. doi:10.48550/ARXIV.1705.07204.
URL https://arxiv.org/abs/1705.07204

[100] E. Anthi, L. Williams, M. Rhode, P. Burnap, A. Wedgbury, Adversarial attacks on machine learning cybersecurity defences in Industrial Control Systems, Journal of Information Security and Applications 58 (February) (2021) 102717. doi:10.1016/j.jisa.2020.102717.
URL https://doi.org/10.1016/j.jisa.2020.102717

[101] G. Apruzzese, M. Andreolini, M. Colajanni, M. Marchetti, Hardening Random Forest Cyber Detectors Against Adversarial Attacks, IEEE Transactions on Emerging Topics in Computational Intelligence 4 (4) (2020) 427–439. doi:10.1109/TETCI.2019.2961157.

[102] Y. Chen, S. Wang, W. Jiang, A. Cidon, S. Jana, Cost-aware robust tree ensembles for security applications, in: Proceedings of the 30th USENIX Security Symposium, 2021, pp. 2291–2308. doi:10.48550/ARXIV.1912.01149.

[103] A. Kantchelian, J. D. Tygar, A. D. Joseph, Evasion and hardening of tree ensemble classifiers, in: Proceedings of the 33rd International Conference on Machine Learning, ICML 2016, Vol. 5, 2016, pp. 3562–3573. doi:10.48550/ARXIV.1509.07892.

[104] H. Chen, H. Zhang, D. Boning, C. J. Hsieh, Robust decision trees against adversarial examples, in: Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 2019. doi:10.48550/ARXIV.1902.10660.
URL https://arxiv.org/abs/1902.10660

[105] D. Vos, S. Verwer, Efficient Training of Robust Decision Trees Against Adversarial Examples, in: M. Meila, T. Zhang (Eds.), Proceedings of the 38th International Conference on Machine Learning, ICML 2021, Vol. 139 of Proceedings of Machine Learning Research, PMLR, 2021, pp. 10586–10595.
URL https://proceedings.mlr.press/v139/vos21a.html

[106] M. Andriushchenko, N. Flammarion, Understanding and Improving Fast Adversarial Training, in: H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems, Vol. 33, Curran Associates, Inc., 2020, pp. 16048–16059.

[107] X. Fu, N. Zhou, L. Jiao, H. Li, J. Zhang, The robust deep learning–based schemes for intrusion detection in Internet of Things environments, Annals of Telecommunications 76 (5) (2021) 273–285. doi:10.1007/s12243-021-00854-y.
URL https://doi.org/10.1007/s12243-021-00854-y

[108] A. Shafahi, M. Najibi, M. A. Ghiasi, Z. Xu, J. Dickerson, C. Studer, L. S. Davis, G. Taylor, T. Goldstein, Adversarial training for free!, in: H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, R. Garnett (Eds.), Advances in Neural Information Processing Systems, Vol. 32, Curran Associates, Inc., 2019.

[109] A. Shafahi, M. Najibi, Z. Xu, J. Dickerson, L. S. Davis, T. Goldstein, Universal Adversarial Training, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 5636–5643. doi:10.1609/aaai.v34i04.6017.
URL https://ojs.aaai.org/index.php/AAAI/article/view/6017

[110] W. Zhao, S. Alwidian, Q. H. Mahmoud, Adversarial Training Methods for Deep Learning: A Systematic Review, Algorithms 15 (8) (2022). doi:10.3390/a15080283.
URL https://www.mdpi.com/1999-4893/15/8/283

Short Biographical Sketch

João Vitorino

João Vitorino is a researcher at GECAD, an R&D unit of the School of Engineering, Polytechnic of Porto (ISEP/IPP). He has collaborated with various companies and institutions in multiple international projects and has been responsible for the conceptualization and the development of AI solutions for several cybersecurity applications. His main interests are algorithm development, computer networking, and the use of AI in a secure way.

The focus of his work has been ML explainability and adversarial robustness. He has developed an adversarial attack method and training mechanisms focused on the constraints of communication networks, and has also analysed and implemented robust ML models for anomaly detection and cyber-attack classification in network intrusion detection systems.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐ The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: