2024 54th Annual IEEE/IFIP International Conference on Dependable Systems and Networks - Supplemental Volume (DSN-S)

PANDA: Practical Adversarial Attack Against Network Intrusion Detection

Subrat Kumar Swain
UQ-IITD Research Academy

Vireshwar Kumar
Indian Institute of Technology Delhi

Guangdong Bai
The University of Queensland

Dan Dongseong Kim
The University of Queensland

***Abstract*—While adversarial machine learning (AML) attacks have become prevalent in the computer vision (CV) domain, their applications in other domains, such as network intrusion detection systems (NIDS), remain limited. This gap stems from the lack of a well-defined input space in non-image domains, hindering the generation of adversarial examples. Unlike CV problems, where the input space is the feature space, other domains generally lack a precise inverse mapping from the feature space to the problem space. In this work, we propose PANDA, a novel approach that bridges this gap and enables AML attacks against NIDS. PANDA represents a series of packets as images for training a surrogate NIDS model. Benefiting from the invertibility of this representation, PANDA leverages well-evolved image-based AML attacks to generate adversarial examples against the surrogate model. It then repurposes the adversarial examples from the surrogate model to evade the target NIDS model. We demonstrate the effectiveness of PANDA by successfully crafting adversarial network intrusions with the UQ-IoT dataset. This work establishes a framework for transferring AML attacks from the CV domain to the network domain, opening new avenues for attack modelling and defence strategies in NIDS.**

***Index Terms*—Adversarial Attacks, Robustness, NIDS, Network Security**

I. Introduction

The term *adversarial attack* was initially coined within the context of image classifiers [1]–[3], which has gained tremendous popularity in the machine learning (ML) and computer vision (CV) domain. To launch an adversarial machine learning (AML) attack in the CV domain, you need to solve the following problem: "Add an adversarial perturbation δ to the original image x of class c, creating an adversarial image x_{adv} that looks the same to humans but is mistakenly classified by the target image classifier f as $c' \neq c$." Hence, the success of an adversarial attack hinges on accomplishing two primary objectives: (1) the resulting adversarial image maintains a perceptual resemblance to the original image, and (2) the target classifier misclassifies the adversarial image to a class different from the class of the original image.

Modern networks utilize ML-based network intrusion detection systems (NIDS) to detect malicious attacks against the connected systems. The commonly used NIDS utilizes an anomaly detection (AD) model that analyzes the streaming network data and raises the alarm for any activity that deviates from the benign activity. Such a detection pipeline uses a packet parsing layer to parse the packet fields and then uses a feature extractor to generate extracted features as vectors. The feature vectors are passed to an AD model to obtain the anomaly score. The AD model gets trained initially with the benign network data, which captures the benign data distribution. Hence, any network data other than benign during the operation leads to a higher anomaly score.

While AML attacks have proven effective in CV [4], their applicability to NIDS remains limited. This gap mainly stems from the lack of a well-defined input space in the NIDS domain, hindering the generation of adversarial examples. Unlike CV problems, where the input space is the feature space, network packets generally lack a precise inverse mapping from feature space to problem space [5]. Moreover, the non-differentiability of the feature extractor does not allow the addition of adversarial perturbations in the packet space.

Historically, there have been limited successful packet-level adversarial attacks. At first, attacks were based on random packet changes and relied on trial and error to find a workable solution [6], [7] without any theoretical directions. Newer attacks have used a two-level approach to solve the problem. First, they use standard AML attacks to find a packet's adversarial features. Then, they get the adversarial packets by changing them to be similar to the adversarial features [8], [9]. However, this overly intricate approach may introduce numerical instabilities and sub-optimal solutions.

In this work, we propose PANDA, a novel approach that bridges this gap and enables conventional AML attacks on NIDS. PANDA proposes to overcome this challenge by perturbing the network traffic using gradient-based techniques. Given that network traffic cannot be directly perturbed using gradient-based methods due to the non-differentiable nature of the feature extractor, PANDA creates a representation of the network traffic to enable this. The model trained with the proposed representation facilitates easy gradient propagation to the input layer by converting the space from discrete to continuous. In addition, we can adapt most conventional AML attacks proposed in the CV domain into the network domain. Finally, we can repurpose the crafted adversarial traffic for attacking the target NIDS by leveraging the transferability of adversarial attacks [10].

PANDA creates a new representation of traffic by converting a sequence of raw packets into images. Specifically, it

2833-292X/24/$31.00 ©2024 IEEE
DOI 10.1109/DSN-S60304.2024.00017

converts some of the essential fields in the packets into bits and concatenates them to form a bit vector. Several bit vectors are then stacked to form an image. The transformed images are used to train a surrogate model. Since the representation is invertible, we leverage existing image-based AML attacks to generate adversarial network packets for the surrogate model. Later, we repurposed the adversarial packets to attack target models. In summary, this work made the following contributions:

1) We proposed an invertible representation of network packets by representing network packets as grayscale images.
2) We applied conventional AML attacks to the NIDS by using the proposed PANDA framework.

II. PANDA

A. Notations

Let $\mathbb{T}$ denote the set of all valid network traffic between any two endpoints. A network traffic $t \in \mathbb{T}$ is an ordered set of n network packets $\{p_i\}_{i=1}^{n}$, where $p_i \in \mathbb{P}$ and $\mathbb{P}$ represents the set of all valid network packets. Further, the set of benign packets can be denoted as $\mathbb{P}_b$ and malicious as $\mathbb{P}_m$, and $\mathbb{P} = \mathbb{P}_b \cup \mathbb{P}_m$. Here, i represents the sequence in which a packet arrives at a destination router.

A packet $p_i \in \mathbb{P}$ consists of fields such as Source IP, Destination IP, ports, TTL, etc. We use a transformation function $T : \mathbb{P} \rightarrow \mathbb{X} \in \mathbb{R}$ to make a representation of the packet p. This function maps packet space (also called problem space) to feature space, where $\mathbb{X} \in \mathbb{R}^n$. We ensure that the function T is invertible. In our representation, we leveraged the timestamp and frame length of the packet along with a few essential fields from the packet header.

An anomaly-based IDS, $f_a : \mathbb{X} \rightarrow \mathbb{A} \in \mathbb{R}$, is an ML model that produces an anomaly score $a \in \mathbb{A}$ for a given feature vector x. Let λ represent a pre-defined threshold: if $a \geq \lambda$, then the corresponding packet is malicious, else benign.

$$p \in \begin{cases} \mathbb{P}_m, & \text{if } f_a(T(p)) \geq \lambda \\ \mathbb{P}_b, & \text{otherwise} \end{cases}$$

B. Threat Model

Our focus is specifically on ML-based IDS, especially anomaly-based IDS. We categorise the attacker's goal, knowledge, and capabilities as follows:

Goal: The attacker's goal hinges on two primary objectives: to preserve, either fully or partially, the security breaches caused by the malicious attack while simultaneously ensuring the target NIDS misclassifies the malicious attack as benign.

Knowledge: The attacker operates in a grey-box setting, possessing partial knowledge of the target system. It knows the training data distribution and the target model type, anomaly-based IDS or classification-based IDS. However, the adversary does not have knowledge of the architecture and parameters.

Capability: We operate under the assumption that the attacker is situated within the network, granting them the capability to intercept benign and malicious traffic. We also assume that devices communicate with limited or no encryption measures. Furthermore, the attacker can manipulate and introduce carefully constructed packets into the malicious traffic, retransmitting the altered data within the network.

C. Network Packet Representation

For an ML problem, data representation equally contributes to the quality of model performance, as does the choice of the model's architecture. Standard datasets (e.g., images, texts, and video) have a well-tuned data representation scheme and corresponding model architectures (e.g., texts are represented as embedding and passed on to sequential models such as LSTM or transformers). However, network packets do not conform to these representations, making it challenging to represent and fixate on a model architecture.

Prior works focused on creating a representation of network traffic aimed at a single goal: *Learning a representation that will help to learn an excellent consequent ML model.* Some of the prior works represented network packets in diverse ways. Some works represent network packets as individual images [11], where each field p_i is converted to corresponding bit streams. Since network packets are sequential data, some works consider individual packets as a single word and learn embedding corresponding to each packet [12].

In addition to the goal mentioned above, PANDA prioritizes an additional goal: the representation space (feature space) should be invertible, i.e., given a feature representation, we can obtain the corresponding packets (problem space). Singling on a representation of the network packet is challenging and requires adequate expertise. Hence, we concluded that converting a series of network packets to images would cater to both our objectives. Hence, we chose to represent a series of network packets as images and train the Convolutional Neural Network Autoencoder (CNN-AE) as its corresponding anomaly detection model.

We extracted crucial features from each packet p_i to effectively represent network packets as images, including the inter-arrival times, source and destination IP addresses, source and destination port numbers, source and destination MAC addresses, and frame length. These features were directly converted into binary values, resulting in a 235-bit vector representation for each corresponding packet. Subsequently, we aggregated a series of 235 packets to form a $235 * 235$ square grayscale image. The square output is the primary reason for choosing 235 network packets to create the image. This approach assumes that the resulting image adequately captures the underlying distributional differences between benign and malicious network traffic.

D. Surrogate Model

To model the characteristics of benign network traffic, we trained a CNN-AE using images generated from network packets as the surrogate model. The trained autoencoder effectively captures the patterns and behaviours of benign traffic. We employed the negative of the reconstruction loss as an anomaly measure, utilising a calculated threshold to distinguish between

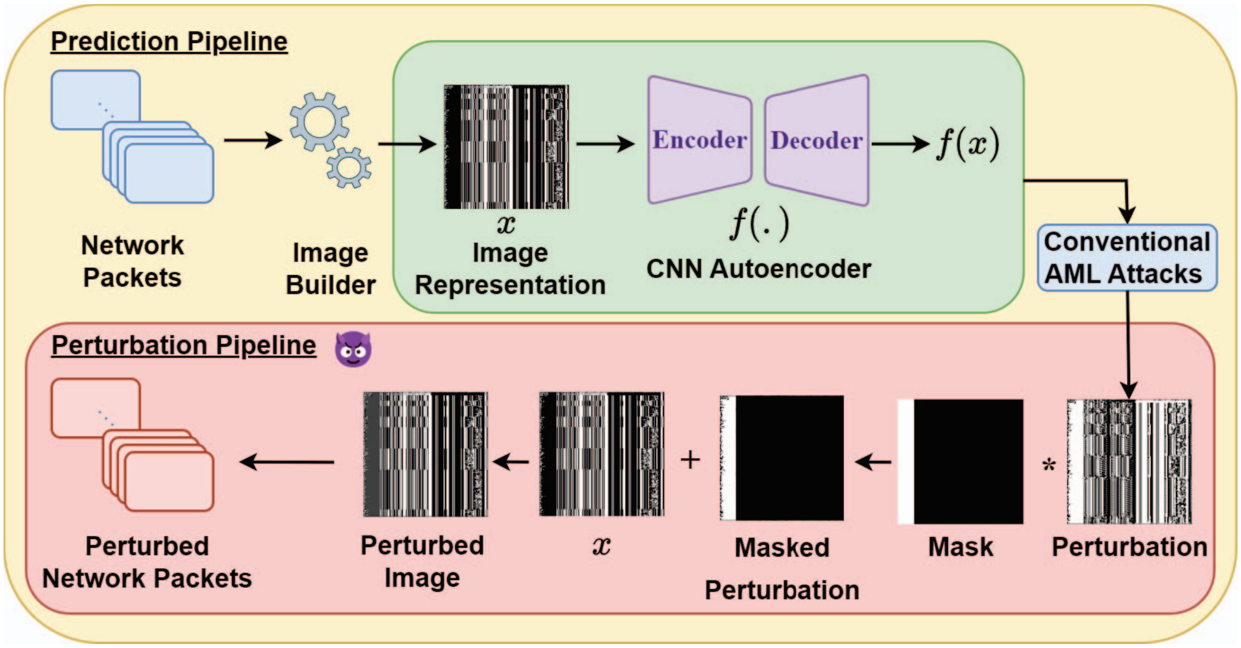

Fig. 1: PANDA Framework: The prediction pipeline and the AML attack pipeline

benign and malicious traffic. If the anomaly score exceeds the threshold, we classify the series of network packets as malicious, while a lower anomaly score indicates benign behaviour. The rationale behind this approach is that benign traffic typically comprises diverse packets from various connections, making it more complex to reconstruct and leading to higher reconstruction errors. In contrast, malicious packets exhibit more significant similarity, resulting in lower reconstruction errors.

E. Adversarial Attack on the Surrogate Model

Let's define δ_{adv} as an adversarial perturbation to manipulate an anomaly detection model's output. A successful adversarial attack on an anomaly detection model flips an input's classification, turning a benign input malicious and vice versa. Taking an *malicious input* x and adding the adversarial perturbation δ_{adv} makes its adversarial counterpart x_{adv}. This means that $x_{adv} = x + \delta_{adv}$.

Consequently, $T^{-1}(x) = p$ and $T^{-1}(x_{adv}) = p_{adv}$. Based on purity, we denote the set of adversarial traffic as $\mathbb{P}_{adv}$ and the set of clean traffic as $\mathbb{P}_c$. Also, we denote the traffic that has the packet p_{adv} as $t_{p_{adv}}$, irrespective of the presence of other adversarial packets.

We obtain the adversarial perturbation δ_{adv} by solving this optimization problem:

$$\begin{aligned} \underset{\delta_{adv}}{\text{argmin}} \quad & f_a(x+\delta_{adv}) \\ \text{s.t.} \quad & f_a(x+\delta_{adv}) < \lambda \\ \text{and} \quad & T^{-1}(x+\delta_{adv}) \in \mathbb{P} \\ \text{and} \quad & t_{p_{adv}} \in \mathbb{T} \end{aligned} \tag{1}$$

This optimization problem aims to minimize the adversarial anomaly score $f_a(x + \delta_{\text{adv}})$ while satisfying the following constraints:

1) **Evasion Constraint:** The adversarial anomaly score should be less than a threshold λ to ensure the adversarial input is classified as benign.
2) **Packet Validation Constraint:** The allowed input space $\mathbb{P}$ should contain the adversarial input $x + \delta_{adv}$ so that the anomaly detection model can still use it.
3) **Traffic Validation Constraint:** The traffic $t_{p_{adv}}$ containing the adversarial packet p_{adv} should belong to the valid traffic set $\mathbb{T}$ to ensure it represents realistic network traffic.

Our initial approach employed a modified version of the Fast Gradient Sign Method (FGSM) [1] to attack the target model. Given that the model's input is an image, we could directly apply this attack to generate perturbed images. Initially, we restricted perturbations to the inter-arrival time field, leaving other fields unmodified. This approach caters to two primary considerations. First, most NIDS use the temporal relationship between packets as a feature while training the underlying machine-learning model. Second, perturbing only the inter-arrival time will not violate packet or traffic validation constraints. We used a mask to make this selective perturbation work. The mask limited changes to the first 32 bits, which are the inter-arrival time, leaving all other fields unchanged. We call the modified FGSM as Masked-FGSM. To get the adversarial inter-arrival time, we applied the following adversarial perturbation:

$$\tau_i^{adv} = Clip_{(0.001,\epsilon)}\{\tau_i + \epsilon * sign(\nabla_{\tau_i} f_a(x_i) * M\}$$

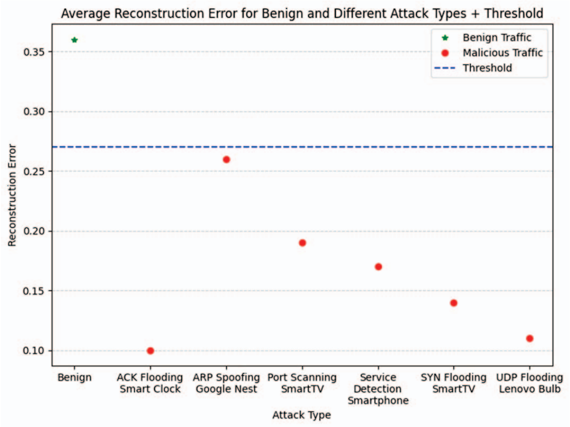

Fig. 2: Preliminary evaluation of surrogate model against UQ-IoT dataset

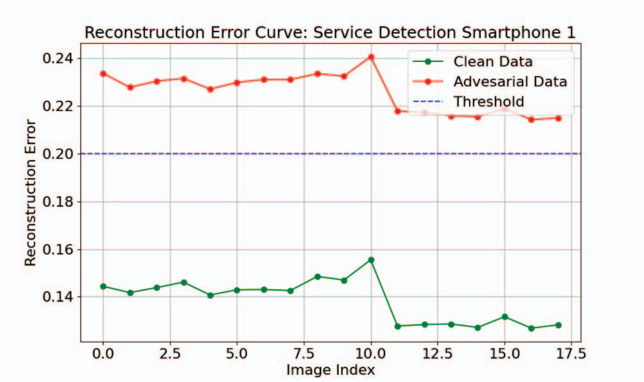

Fig. 3: Evaluation of CNN-AE model against *Adversarial Service Detection* data generated using Masked-FGSM

The complete pipeline is shown in Figure 1

III. Evaluation

To evaluate PANDA, we used the UQ-IoT Dateset. We chose a CNN-AE for the NIDS and trained it with benign network traffic to capture benign traffic behaviour. We derived the model's threshold to differentiate between benign and malicious network packets. We derived this by taking the 95^{th} percentile of the reconstruction errors while training. To measure the performance of the surrogate model, we used the standard metrics used to evaluate an anomaly detection model. The performance of the surrogate model is shown in Figure 2 as it is evident that the threshold differentiates between benign and malicious traffic. Note that the anomaly score is the negative of the reconstruction error; hence, the points lying below the threshold line are considered malicious, and the points lying above the threshold are considered benign.

We evaluated our attack by passing adversarial traffic to the surrogate model, which successfully evaded it. In Figure 3, Figure 4, and Figure 5, we can see the adversarial network packets bypass the surrogate NIDS by evading the threshold value.

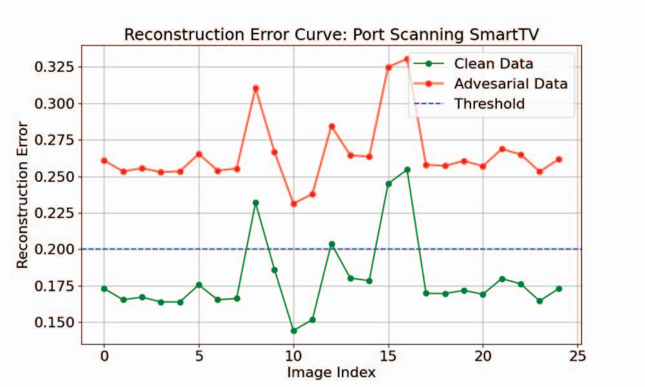

Fig. 4: Evaluation of CNN-AE model against *Adversarial Port Scanning* data generated using Masked-FGSM

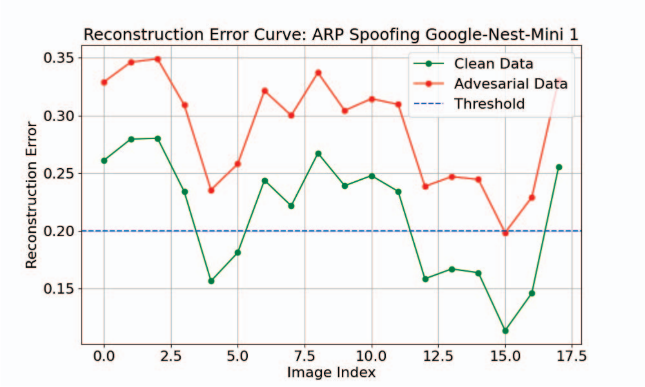

Fig. 5: Evaluation of CNN-AE model against *Adversarial ARP Spoofing* data generated using Masked-FGSM

Since the chosen CNN autoencoder is a surrogate model, we evaluate PANDA based on its performance on benign and malicious traffic compared to the target NIDS. We will detail the evaluations of the target models in our future work.

IV. Conclusion

AML attacks are a real threat to NIDS, making the study of AML attacks against NIDS indispensable. Our work PANDA tries to improve the pre-existing AML attacks proposed for NIDS. PANDA addresses the problem of inverse mapping in feature extraction by proposing an invertible feature representation method. The proposed representation method converts a series of network packets into images that achieve fast gradient-based perturbation on the network packets. Moreover, PANDA establishes a framework for translating AML attacks from the CV domain to the NIDS domain. Finally, the evaluation confirms the framework's validation by successfully evading the CNN-AE model. PANDA opens new avenues for attack modelling and network defence strategies.

Acknowledgements

This work was supported by the Ministry of Electronics and Information Technology (MeitY), Government of India.

References

[1] C. Szegedy, W. Zaremba, I. Sutskever, *et al.*, "Intriguing properties of neural networks," in *2nd International Conference on Learning Representations, ICLR*, 2014.

[2] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: A simple and accurate method to fool deep neural networks," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pp. 2574–2582, IEEE Computer Society, 2016.

[3] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, OpenReview.net, 2018.

[4] S. K. Swain, V. Kumar, D. D. Kim, and G. Bai, "SPAT: semantic-preserving adversarial transformation for perceptually similar adversarial examples," in *ECAI 2023 - 26th European Conference on Artificial Intelligence*, vol. 372 of *Frontiers in Artificial Intelligence and Applications*, pp. 2266–2273, IOS Press, 2023.

[5] F. Pierazzi, F. Pendlebury, J. Cortellazzi, and L. Cavallaro, "Intriguing properties of adversarial ml attacks in the problem space," in *2020 IEEE symposium on security and privacy (SP)*, pp. 1332–1349, 2020.

[6] M. J. Hashemi, G. Cusack, and 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*, pp. 14–21, 2019.

[7] I. Homoliak, M. Teknos, M. Ochoa, D. Breitenbacher, S. Hosseini, and P. Hanacek, "Improving network intrusion detection classifiers by non-payload-based exploit-independent obfuscations: An adversarial approach," *arXiv preprint arXiv:1805.02684*, 2018.

[8] D. Han, Z. Wang, Y. Zhong, W. Chen, J. Yang, S. Lu, X. Shi, and X. Yin, "Practical traffic-space adversarial attacks on learning-based nidss," *arXiv preprint arXiv:2005.07519*, 2020.

[9] A. Kuppa, S. Grzonkowski, M. R. Asghar, and N.-A. Le-Khac, "Black box attacks on deep anomaly detectors," in *Proceedings of the 14th international conference on availability, reliability and security*, pp. 1–10, 2019.

[10] A. Demontis, M. Melis, M. Pintor, M. Jagielski, B. Biggio, A. Oprea, C. Nita-Rotaru, and F. Roli, "Why do adversarial attacks transfer? explaining transferability of evasion and poisoning attacks," 2019.

[11] J. Holland, P. Schmitt, N. Feamster, and P. Mittal, "New directions in automated traffic analysis," in *CCS '21: 2021 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pp. 3366–3383, ACM, 2021.

[12] Z. Wu, H. Zhang, P. Wang, and Z. Sun, "Rtids: A robust transformer-based approach for intrusion detection system," *IEEE Access*, vol. 10, pp. 64375–64387, 2022.