NI-Diff: Zero-Day and Adversarial Network Intrusion Detection with Diffusion Models

Milin Zhang[×], Michael De Lucia[‡], Ananthram Swami[‡],
Jonathan Ashdown[*], Nathaniel D. Bastian[◦] and Francesco Restuccia[×]
[‡] DEVCOM Army Research Laboratory, United States [*] Air Force Research Laboratory, United States
[◦] United States Military Academy, United States [×] Northeastern University, United States

***Abstract*—While Deep Learning (DL) has achieved remarkable success in Network Intrusion Detection System (NIDS), its inherent data-driven nature makes it vulnerable to distribution shift. This limitation exposes DL-based NIDS to both adversarial attacks that are crafted by adding subtle change to original samples and zero-day attacks that are out-of-distribution (OOD) data unseen during training. However, existing work focusing on adversarial detection often fails to identify zero-day attacks and vice versa, leaving a security gap in DL-based NIDS. We propose `NI-Diff`, a novel detection approach that can effectively identify both adversarial network flow as well as zero-day intrusion by estimating their distribution with generative models. More specifically, we leverage a variational auto-encoder to map the network flow into a latent space and use a diffusion model to reconstruct the likely-hood from noise. Our key intuition is that the in-distribution data and the reconstructed data will have a similar likelyhood which results in similar inference output in the DL classifier. Extensive experiments on two large-scale NIDS datasets demonstrate that our approach can effectively identify 97% adversarial network flow and 92% zero-day threat with less than 2% false positive rate, outperforming state-of-the-art adversarial detection and OOD detection baselines.**

***Index Terms*—Network Intrusion Detection, Diffusion Models, Adversarial Detection, OOD Detection**

I. Introduction

Emerging technologies such as augmented/virtual reality (AR/VR), smart home, and intelligent healthcare create an increasingly sophisticate Internet-of-Things (IoT) network [1]. Network security has becomes a significant concern in the expanding interconnected system. Traditional Network Intrusion Detection System (NIDS) based on feature engineering provide insufficient protection against evolving intrusions [2]. While Deep Learning (DL) has emerged as a promising solution to improve the robustness of NIDS [3, 4], it relies on the large-scale network data to train the classifier. This data-driven nature makes the system vulnerable to zero-day attacks that are new network threats not encountered during the training phase [5]. In addition, Deep Neural Networks (DNNs) exhibit susceptibility to adversarial actions, where a negligible manipulation to input data can dramatically impact inference outcomes [6], potentially compromising the detection performance of NIDS .

To address the challenge, various out-of-distribution (OOD) detection and adversarial detection methods are proposed for NIDS [7–9]. However, one fundamental limitation is that existing work is often specifically designed for one threat but neglect the detection performance on the other threat. For example, research that can detect zero-day intrusions often fails to identify adversarial attacks and vice versa. In addition, most of existing work are based on small-scale dataset [7, 9–11], lacking of evaluation on modern IoT network.

We propose `NI-Diff`, a framework that is capable to jointly detection zero-day and adversarial network intrusions with diffusion models [12]. It consists of three key components: a multi-class DNN classifier that detects known benign traffic and network intrusions; a variational auto-encoder [13] that transforms network flows into estimated distributions; and a diffusion model [12] to reconstruct the estimated distribution from noise. The key idea is that the distribution shift introduced by zero-day and adversarial network intrusions can be captured and amplified during the diffusion-denoising process, hence resulting in noticeable differences in the DNN classifier's output. We evaluate the proposed framework on ACIIoT-2023 [14] and CICIoT-2023 [15] which are two most recent and large-scale IoT datasets using both OOD samples and two different adversarial attacks.

Summary of Novel Contributions

- We investigated how diffusion models can effectively capture the subtle distribution shifts of adversarial and OOD samples. A new defensive framework `NI-Diff` is introduced to jointly detect both zero-day and adversarial network intrusions. To the best of the authors' knowledge, this is the first work to leverage diffusion models for both OOD and adversarial detection in the cybersecurity domain.
- We conducted comprehensive evaluations using two recent large-scale IoT datasets [14, 15], providing greater relevance to next-generation IoT network. In addition, we designed two distinct types of adversarial network intrusions–gradient-based and generative model-based attacks–offering a more thorough evaluation than existing adversarial detection research [7, 11] that focuses exclusively on gradient-based adversarial samples.
- We compared `NI-Diff` with 3 state-of-the-art adversarial detection methods [7, 11, 16], and 3 commonly used OOD baselines [17–19]. Our experimental results demonstrate that `NI-Diff` can effectively detect up to 97% and 92% zero-day and adversarial network intrusions with less than 2% false positive rate, significantly outperform other six baselines.

II. Related Works

Network Intrusion Detection. Network traffic can be analyzed at either the packet or flow level. In packet-level intrusion

detection, researchers typically process packet payloads as images and apply computer vision techniques such as Convolutional Neural Network (CNN) to classify the payload as either benign or malicious. For example, De Lucia et al. [4] demonstrated that 1D-CNN can achieve superior performance compared to traditional methods by directly using raw network bytes. Zhang et al. [20] proposed a new encoding scheme to transform packets to two-dimensional gray-scale images and used CNN for classification. Ghadermazi et al. [21] proposed an algorithm to encode packets into RGB images. On the other hand, flow-level detection approaches first extract features of bidirectional network flows between source and destination devices (e.g. inter arrival time, average payload length, etc.), and then train a multi-class DNN classifier based on these aggregated flow-level features. For example, Xiao et al. [22] utilized CNN for network traffic classification. Sun et al. [23] proposed a hybrid architecture combining both CNN and Recurrent Neural Network (RNN) to improve accuracy. In this work, we consider adversarial and zero-day attacks at the flow level.

Adversarial Detection. Adversarial attacks on DNN were first introduced in [24] for computer vision where an invisible perturbation can result in significant difference in the output, hampering the performance of the DL system. Adversarial detection is proposed to identify attack samples at test time without compromising the DNN performance. For example, Drenkow et al. [16] proposed a random projection method to map latent representations to multiple low dimensional manifolds to magnify the difference between adversarial and clean data. However, as the network data presents a different structure compared to images, adversarial attacks to NIDS has physical-world constraints to create a realistic adversarial network traffic [25, 26]. This makes detection methods in computer vision less effective to identify adversarial samples in network intrusions. To this end, a vast amount of effort is made to detect adversarial network intrusions. Wang et al. [7] revealed that the adversarial network intrusion is often an outlier in the transformed manifold hence can be detected with manifold learning methods. Kumar et al. [11] used an auto-encoder to detect the adversarial intrusion samples.

OOD Detection. Real world input data can be open-set and unknown, making it challenging for DNN trained with close-set data to process the new information. Therefore, a secure and robust DL system needs the ability to identify if the input is valid in-distribution sample or OOD. Existing research aims to design advanced score function to differentiate the in-distribution and OOD samples. Fore example, Liu et al. [17] proposed a generalized entropy score that can assess the uncertainty of the DNN output to detect OOD samples. Liu et al. [18] proposed to identify the OOD using the energy function of output logits. Huang et al. [19] designed the score function in gradient space by measuring the Kullback–Leibler Divergence (KLD) between the gradient of output and the gradient of the uniform distribution. Research in cybersecurity also refers to OOD detection as zero-day attack detection or novelty detection. Matejek et al. [8] introduced normalizing flow to detect zero-day network intrusions while Bradley et al. [27] estimated the probability of zero-day attack using survival analysis. Baye et al. [9] proposed a multi-step approach including top-difference classification and energy-based detection for detecting unknown network activities.

Diffusion Models. Ho et al. [12] proposed a Denoising Diffusion Probabilistic Model (DDPM) that can generate realistic new image from noise. It involves a diffusion process that gradually adds Gaussian noise to the original data and a denoising process that can reconstruct meaningful information from the noise. Recent research has demonstrated the effectiveness of diffusion models in improving adversarial robustness [28]. In the cybersecurity domain, Cai et al. [29] leveraged diffusion models to address the data imbalance issue in NIDS while Zhang et al. [30] proposed a hierarchical diffusion model to generate new network flows. Alhussien and Aleroud [31] used diffusion models to denoise the adversarial perturbation hence improving the robustness of the NIDS. Although this method [31] can protect the system from adversarial attacks, it lacks the capability to handle unseen zero-day threats. In contrast, we propose a novel approach that can jointly detect adversarial and OOD network intrusions using diffusion models.

III. The NI-Diff Framework

Figure 1 demonstrates the system framework of NI-Diff, which consists of a multi-class DNN classifier, an auxiliary distribution modeling block and a zero-day/adversarial detection block. The DNN classifier is trained to identify benign and various known network intrusions. In the distribution modeling block, a Variational Auto-Encoder (VAE) is used to transform known network traffic into a likelihood function while a diffusion model is used to learn this distribution. During inference, detection is achieved by measuring the Kullback–Leibler Divergence (KLD) between the softmax score of the original data and the output of the reconstructed data using distribution modeling block.

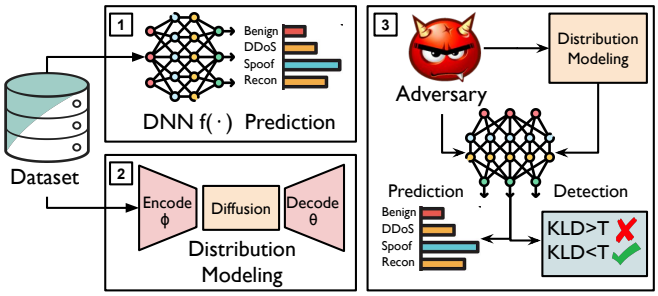

Fig. 1: Overview of NI-Diff framework.

DNN classifier. For multi-class network traffic classification, we implemented two distinct DNN architectures for the two different datasets. We first chose a vanilla 1D-CNN model comprising 5 convolutional layers, each followed by batch normalization, ReLU activation and maxpooling. We denote

this model as NI-Diff-base. However, to achieve a better classification performance on the more complex CICIoT-2023 dataset, we integrated residual connections and self attention with the basic model, denoted as NI-Diff-large. Table I summarizes these two model architectures.

TABLE I: Summary of DNN Classifiers

NI-Diff-base	NI-Diff-large
Conv 1×3, 16	Res 64, Self Attn
Conv 1×3, 16	Res 128, Self Attn
Conv 1×3, 32	Res 256, Self Attn
Conv 1×3, 32	Res 512, Self Attn
Conv 1×3, 64	Res 1024, Self Attn
Linear 64×n	Linear 1024×n

Variational Auto-Encoder. Prior research has demonstrated that diffusion models can enhance adversarial robustness in NIDS [31]. The basic idea is to use the diffusion model as a purifier to remove adversarial noise in network flow data. It first introduces Gaussian noise to network traffic during the diffusion process, which can successfully masks the adversarial noise. Subsequently, the diffusion model can purify the added noise during with denoising, hence enhancing the robustness. However, this approach has significant limitations when applied to real world network traffic data which contains symbolic features with specific physical meanings (such as ACK flag counts and number of packets). Introducing noise to these symbolic input can break their semantic integrity, resulting in unstable detection performance.

To address this challenge, we first leverage a VAE to transform the data into a probabilistic latent space and then apply the diffusion model to the probabilistic representation. Formally, the encoder can be denoted as $q_\phi(z|x)$ and the decoder is denoted as $p_\theta(x|z)$, where x and z are input and latent variable, respectively. The model is trained to maximize the evidence lower bound [13],

$$L(\theta,\phi,x) = \mathbb{E}_{q_\phi(z|x)}(\underbrace{\log(p_\theta(x|z))}_{Reconstruction} - \underbrace{D_{KL}(q_\phi(z|x)||p(z))}_{KL\ divergence}). \tag{1}$$

In practice, the reconstruction in equation 1 is measured with mean square error (MSE) between x and the reconstructed x' while the KLD is measured between the latent variable z and a normal distribution $p(z) \sim N(0, I)$. In addition, we applied perceptual loss [32] to ensure the semantic properties of the reconstructed network traffic data. The total loss becomes

$$L = \underbrace{MSE(x,x')}_{reconstruction} + \underbrace{\lambda_1 \cdot KLD(q_\phi(z|x)||p(z))}_{KL\ divergence} + \underbrace{\lambda_2 \cdot P_l}_{perceptual}. \tag{2}$$

In practice, the trade-off factors λ_1 and λ_2 in Equation 2 are set to 1e-6 and 1e-3 respectively. We use identical VAE architecture for the two DNN classifiers and datasets. It is composed of 6 layers of 1×3 1D-CNN for both encoder and decoder. A maxpooling and upsampling is used to rescale the dimension after every two convolutional layers in both encoder and decoder, except for the last layer pairs. All layers maintain a uniform channel dimension of 64, with two exceptions: the encoder's final layer employs 2 channels (for the reparameterization trick [13]), while the decoder's final layer utilizes a single channel.

Diffusion Models can be modeled with Markov chains involving two processes: i) a diffusion process that gradually add noise to the input, and ii) a denoising process that reconstructs input from the noise. Formally, for a input x_0, the diffusion process at step t can be described as

$$q(x_t|x_{t-1}) = N(\sqrt{1-\beta_t}x_{t-1}, \beta_t I), \tag{3}$$

where x_t, $q(x_t|x_{t-1})$ and $N(\cdot,\cdot)$ denotes the random variable, conditional distribution at step t, and Gaussian distribution respectively. β_t is a hyper parameter to control the noise factor added to the input at step t.

Similarly, the denoising process at step t can be described as

$$p_\theta(x_{t-1}|x_t) = N(\mu_\theta(x_t,t), \sigma_\theta(x_t,t)). \tag{4}$$

To train the denoising process to reverse the diffusion, the objective can be reduced to

$$L_\theta = \mathbb{E}_{\epsilon\sim N(0,I)}(||\epsilon - \epsilon_\theta(x_t,t)||^2) \tag{5}$$

where ϵ_θ is the estimated noise and ϵ is the standard Gaussian noise.

It has been shown in [28] that diffusion models can be used to enhance the adversarial robustness without retraining the DNN classifier. The key idea is that the diffusion-denoising process can generate samples that closely align with the original training distribution, causing adversarial samples and generated samples to produce distinct outputs from the DNN classifier. We extend this idea to detect adversarial and zero-day threats by applying diffusion models to a probabilistic latent space. In practice, a denoising U-Net [12] is trained to predict the noise using Equation 4. We developed a custom U-Net architecture that incorporates 1D-CNN, residual connection and self-attention blocks. However, unlike the DNN classifier which uses batch normalization and ReLU activation, the U-Net employs group normalization and SiLU activation. The details of the model are summarized in Table II.

TABLE II: Summary of denoising U-Net

Down	Mid	Up
Conv 1×3, 32	Res 128	Res 64
Res 32	Self Attn 128	Res 32
Res 64	Res 128	Conv 1×3, 1

Detection Algorithm. One advantage of NI-Diff is that the adversarial/zero-day detection is based on the auxiliary distribution modeling block without affecting the DNN classification. During testing, the system will classify the network traffic x using the DNN classifier $f(x) = y$. In parallel, the variational encoder $E(x) = z$ will map the input to a likelihood and the diffusion model $DM(z) = z'$ will sample a new likelihood. z' is then processed by the variational decoder $D(z')$ and the classifier $f(D(z')) = y'$. The difference between z and z' is measured with the KLD between the softmax

score y and y'. Note that Carlini et al. [28] suggested that one-step denoising can provide more stable results compared to a large timestep of denoising. As such, we use one-step diffusion-denoising for generating new z'. Algorithm 1 describes the joint classification and attack detection process of our NI-Diff framework.

Algorithm 1 Classification and Attack Detection

Initialize: DNN classifier $f(\cdot)$, variational encoder $E(\cdot)$, decoder $D(\cdot)$, diffusion model $DM(\cdot)$, threshold T, detection flag ρ.
Input: network traffic x
$y = f(x)$, $z' = DM(E(x))$, $y' = f(D(z'))$
If $D_{KL}(y; y') < T$ **return** y
Else return ρ

IV. Experimental Setup

Dataset. We assess NI-Diff with two recent IoT datasets: ACIIoT-2023 [14] and CICIoT-2023 [15]. For the ACIIoT dataset, which contains 1.2 million flow-level samples across 12 different classes, we employ the NI-Diff-base model and utilize the complete dataset for our evaluation. The CICIoT dataset is more extensive, comprising over 20 million flow-level samples distributed across 34 different classes which can be further categorized into 8 super classes. In this case, we implement the NI-Diff-large model and perform balanced resampling to extract 1.2 million samples from the original dataset.

Training Recipe. DNN classifiers for both dataset are trained using Adam optimizer with a learning rate of 1e-4 and a batchsize of 1024 for 50 epochs. VAEs are trained using the same optimizer and learning rate setting but for 100 epochs. Diffusion models are trained for 1000 epochs using Adam with a smaller learning rate of 1e-5 and exponential moving weight average. The β_t in Equation 3 is chosen from 1e-4 to 0.02 using linear increasing schedule and the maximum timestep is 1000. All experiments are performed on a Nvidia RTX A4000 GPU.

Attack Model. For ACIIoT dataset, we select *Vulnerability scan*, *UDP flood* and *ARP spoofing* as the zero-day attacks and the classifier only uses 9 classes of data for training. For CICIoT, we select all data from 4 super classes *Brute force*, *Spoofing*, *Recon* and *Web-based* as zero-day samples and the classifier leverages the remaining 20 classes for training.

For adversarial attack, we employ both gradient-based [33] and Generative Adversarial Network (GAN)-based [34] approaches. We consider target attack aiming to manipulate intrusion data to be misclassified as benign. To ensure the realism of generated data, we put constraints on the symbolic features and only modify those statistical features such as average length, inter arrival time, etc.

Baseline Methods. We compare our method to 3 OOD detection baselines: GEN [17], Energy [18], and GradNorm [19]. In addition, we evaluate NI-Diff in comparison with 3 adversarial detection baselines: MANDA [7], DeepRP [16] and NIDS-DA [11].

V. Performance Evaluation

We first assess the effectiveness of DNN classifiers and adversarial attacks. Table III shows the DNN classification accuracy and the attack success rate (ASR) of gradient-based and GAN-based attacks. NI-Diff can achieve up to 99% accuracy on in-distribution network intrusions while adversarial attacks successfully cause around 90% of malicious data to be misclassified as benign.

TABLE III: Accuracy and Attack Success Rate

	Acc	ASR (Grad)	ASR (GAN)
ACIIoT	99.19%	98.30%	99.97%
CICIoT	99.42%	89.29%	99.97%

We evaluate the adversarial/zero-day detection performance of NI-Diff in comparison to other baseline methods using various metrics. First we consider True Positive Rate (TPR) and False Positive Rate (FPR) that directly measure the correctness of detectors in malicious and benign cases. Tables IV and V summarize the TPR and FPR of three different attack scenarios for ACIIoT and CICIoT dataset respectively.

For the ACIIoT dataset, those OOD detection methods are less effective in identifying adversarial samples. For example, GradNorm detects 0.95% gradient-based and 0% GAN-based attacks. On the other hand, adversarial detection baselines demonstrate a significant performance loss when applied to zero-day attacks. Only DeepRP shows high detection rate on both zero-day and adversarial attacks but it comes at a cost of higher FPR. In contrast, NI-Diff demonstrates the best TPR on adversarial attacks and a comparable TPR on zero-day attacks with a lowest FPR on in-distribution data.

For CICIoT, OOD detection baselines achieve decent performance on adversarial attacks but are underperforming on zero-day samples. This is because the CICIoT dataset has a larger scale compared to ACIIoT, making the OOD samples more difficult to identify while increasing the complexity for adversarial attacks to succeed. As such, effective and realistic adversarial samples are more like OOD samples [10]. This also makes the adversarial detection baselines less effective in detecting the gradient-based attack. Compared to other baselines, NI-Diff demonstrates the best detection performance on zero-day and GAN-based attacks while a comparable performance on gradient-based and in-distribution samples.

TABLE IV: True and False Positive Rate for ACIIoT

	TPR (zero-day)	TPR (grad)	TPR (GAN)	FPR
GEN	**95.90%**	50.25%	0.63%	8.63%
Energy	93.73%	64.64%	5.33%	7.52%
GradNorm	89.27%	0.95%	0.00%	12.76%
MANDA	6.55%	85.17%	61.38%	8.54%
DeepRP	95.54%	84.77%	84.94%	11.62%
NIDS-DA	1.95%	1.49%	69.28%	6.05%
NI-Diff	91.86%	**97.05%**	**88.50%**	**1.59%**

TABLE V: True and False Positive Rate for CICIoT

	TPR (zero-day)	TPR (grad)	TPR (GAN)	FPR
GEN	38.38%	**99.96%**	99.97%	4.89%
Energy	39.64%	99.82%	99.97%	4.31%
GradNorm	30.11%	99.86%	99.94%	6.05%
MANDA	11.77%	56.77%	99.95%	13.87%
DeepRP	16.96%	3.96%	94.97%	**2.02%**
NIDS-DA	13.38%	32.73%	94.74%	3.70%
NI-Diff	**51.18%**	94.31%	**99.98%**	3.28%

TPR and FPR evaluate the detector's performance from limited perspectives focusing solely on either attack scenarios or in-distribution data. To have a more comprehensive evaluation, we compared NI-Diff with other baselines using precision and F1 score, which offer balanced evaluations that consider both attack and in-distribution data simultaneously. Tables VI and VII show the precision and F1 score for ACIIoT and CICIoT. As shown, NI-Diff constantly outperforms other baselines in both metrics on ACIIoT. For CICIoT, NI-Diff achieves the best performance on zero-day and GAN-based attacks and has a comparable performance on gradient-based attack.

TABLE VI: Precision and F1 Score for ACIIoT

	Zero-day		Gradient		GAN	
	P	F1	P	F1	P	F1
GEN	90.87%	93.32%	85.34%	63.25%	6.80%	1.15%
Energy	92.57%	93.15%	89.58%	75.09%	41.48%	9.45%
GradNorm	87.49%	88.37%	6.93%	1.67%	0.00%	0.00%
MANDA	43.41%	11.38%	90.88%	87.93%	87.78%	72.24%
DeepRP	89.16%	92.24%	87.94%	86.32%	87.97%	86.43%
NIDS-DA	24.37%	3.61%	19.76%	2.77%	91.97%	79.03%
NI-Diff	**98.30%**	**94.97%**	**98.39%**	**97.91%**	**98.23%**	**93.11%**

TABLE VII: Precision and F1 Score for CICIoT

	Zero-day		Gradient		GAN	
	P	F1	P	F1	P	F1
GEN	88.70%	53.58%	95.34%	97.59%	95.34%	97.60%
Energy	90.19%	55.07%	95.86%	**97.80%**	95.87%	97.88%
GradNorm	83.27%	44.23%	94.29%	96.99%	94.29%	97.03%
MANDA	45.90%	18.74%	80.37%	66.54%	87.81%	93.49%
DeepRP	89.36%	28.51%	**97.92%**	96.42%	66.22%	7.47%
NIDS-DA	78.34%	22.86%	89.84%	47.98%	96.24%	95.48%
NI-Diff	**93.98%**	**66.27%**	96.64%	95.46%	**96.82%**	**98.37%**

Precision and F1 scores are threshold-dependent metrics that evaluate the performance with a specific detection threshold. To this end, we further study the performance under different detection threshold settings using the area under the receiver operating characteristic (AUROC). Table VIII summarizes AUROC results for three attack types across both datasets. As demonstrated, OOD detectors demonstrate poor effectiveness against adversarial samples. For instance, GradNorm on the ACIIoT dataset achieves merely 33.55% and 7.58% AUROC for gradient and GAN-based attacks respectively–falling below the 50% threshold that indicates random guessing, essentially rendering it unable to distinguish between in-distribution and adversarial samples. Conversely, adversarial detection methods struggle with OOD sample identification. MANDA, for example, achieves only 26.17% AUROC on the CICIoT dataset. In contrast, NI-Diff achieves the best performance in detecting zero-day and GAN-based attacks across both datasets. For the gradient attack, NI-Diff achieves the best performance on the ACIIoT dataset while maintaining comparable effectiveness on CICIoT.

TABLE VIII: AUROC for ACIIoT and CICIoT

	Zero-day		Gradient		GAN	
	ACI	CIC	ACI	CIC	ACI	CIC
GEN	97.14%	82.05%	74.35%	99.89%	29.76%	99.33%
Energy	97.72%	84.01%	83.07%	**99.90%**	49.55%	99.42%
GradNorm	93.74%	76.50%	33.55%	99.71%	7.58%	99.39%
MANDA	84.99%	26.17%	90.79%	82.33%	89.01%	97.71%
DeepRP	98.16%	58.41%	92.04%	62.97%	92.62%	97.51%
NIDS-DA	87.69%	83.69%	86.45%	98.08%	96.61%	99.48%
NI-Diff	**98.85%**	**95.70%**	**99.61%**	99.46%	**98.55%**	**99.94%**

In Algorithm 1, the diffusion model uses one-step denoising for a stable detection performance. This is because the detection algorithm is based on the DNN classifier's capacity to distinguish anomaly and normal data. While a large timestep can help diffusion to generate high quality data from noise, these synthetic data may be semantically new to the classifier, hence resulting in an unstable detection performance. To demonstrate this point, we study the detection performance of NI-Diff as a function of denoising timesteps using ACIIoT dataset. As demonstrated in Figure 2, NI-Diff achieves the lowest FPR and highest TPR with one-step diffusion. With a growing timestep, the FPR and TPR converge to around 70%, which indicates that the DNN classifier cannot distinguish between the newly generated normal data and anomaly data. This one-step setting makes NI-Diff fundamentally different from other work using diffusion models as a data generation tool [29, 30].

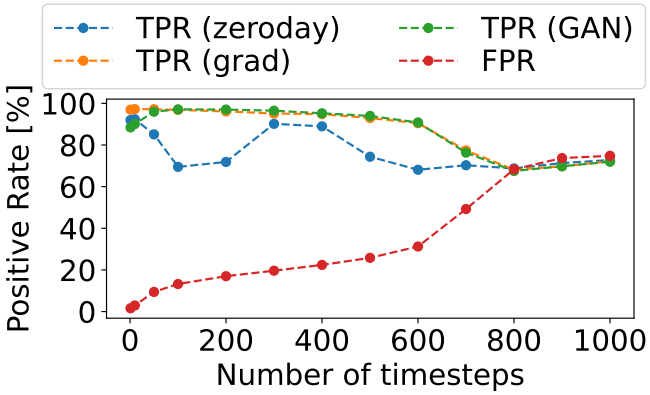

Fig. 2: NI-Diff performance on ACIIoT as a function of denoising timesteps.

VI. Conclusion

We proposed a novel framework to jointly detect zero-day and adversarial attacks against network intrusion detection systems using diffusion models. Experiments on two recent IoT datasets demonstrated that the proposed approach can effectively identify 97% and 92% adversarial and zero-day threats with less than 2% false positive rate, and uniformly

outperform six other state-of-the-art methods across various metrics. In future work, we aim to apply this framework to packet-level zero-day and adversarial intrusion detections.

ACKNOWLEDGMENTS

This work has been funded in part by the National Science Foundation under grants CNS-2312875 and OAC-2530896, by the Air Force Office of Scientific Research under contract number FA9550-23-1-0261, by the Office of Naval Research under award number N00014-23-1-2221, and by the Defense Advanced Research Projects Agency (DARPA) under the Young Faculty Award program.

REFERENCES

[1] R. Douglass, K. Gremban, A. Swami, and S. Gerali, *IoT for Defense and National Security*. John Wiley & Sons, 2023.

[2] S. Gamage and J. Samarabandu, "Deep learning methods in network intrusion detection: A survey and an objective comparison," *Journal of Network and Computer Applications*, vol. 169, p. 102767, 2020.

[3] M. J. De Lucia and C. Cotton, "Detection of encrypted malicious network traffic using machine learning," in *MILCOM 2019-2019 IEEE Military Communications Conference (MILCOM)*. IEEE, 2019, pp. 1–6.

[4] M. J. De Lucia, P. E. Maxwell, N. D. Bastian, A. Swami, B. Jalaian, and N. Leslie, "Machine learning raw network traffic detection," in *Artificial intelligence and machine learning for multi-domain operations applications III*, vol. 11746. SPIE, 2021, pp. 185–194.

[5] B. Matejek, A. Gehani, N. D. Bastian, D. J. Clouse, B. J. Kline, and S. Jha, "SAFE-NID: Self-attention with normalizing-flow encodings for network intrusion detection," *Transactions on Machine Learning Research*, 2025.

[6] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, "The limitations of deep learning in adversarial settings," in *2016 IEEE European symposium on security and privacy (EuroS&P)*. IEEE, 2016, pp. 372–387.

[7] N. Wang, Y. Chen, Y. Xiao, Y. Hu, W. Lou, and Y. T. Hou, "Manda: On adversarial example detection for network intrusion detection system," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 2, pp. 1139–1153, 2022.

[8] B. Matejek, A. Gehani, N. D. Bastian, D. Clouse, B. Kline, and S. Jha, "Safeguarding network intrusion detection models from zero-day attacks and concept drift," in *AAAI Workshop on Artificial Intelligence for Cyber Security (AICS)*, 2024.

[9] G. Baye, P. Silva, A. Broggi, N. D. Bastian, L. Fiondella, and G. Kul, "varmax: Towards confidence-based zero-day attack recognition," in *MILCOM 2024-2024 IEEE Military Communications Conference (MILCOM)*. IEEE, 2024, pp. 863–868.

[10] S. Hore, J. Ghadermazi, D. Paudel, A. Shah, T. Das, and N. Bastian, "Deep packgen: A deep reinforcement learning framework for adversarial network packet generation," *ACM Transactions on Privacy and Security*, vol. 28, no. 2, pp. 1–33, 2025.

[11] V. Kumar, K. Kumar, M. Singh, and N. Kumar, "Nids-da: Detecting functionally preserved adversarial examples for network intrusion detection system using deep autoencoders," *Expert Systems with Applications*, p. 126513, 2025.

[12] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[13] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *CoRR*, vol. abs/1312.6114, 2013.

[14] N. Bastian, D. Bierbrauer, M. McKenzie, and E. Nack, "Aci iot network traffic dataset 2023," 2023. [Online]. Available: https://dx.doi.org/10.21227/qacj-3x32

[15] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "Ciciot2023: A real-time dataset and benchmark for large-scale attacks in iot environment," *Sensors*, vol. 23, no. 13, p. 5941, 2023.

[16] N. Drenkow, N. Fendley, and P. Burlina, "Attack agnostic detection of adversarial examples via random subspace analysis," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022, pp. 472–482.

[17] X. Liu, Y. Lochman, and C. Zach, "Gen: Pushing the limits of softmax-based out-of-distribution detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 23 946–23 955.

[18] W. Liu, X. Wang, J. Owens, and Y. Li, "Energy-based out-of-distribution detection," *Advances in neural information processing systems*, vol. 33, pp. 21 464–21 475, 2020.

[19] R. Huang, A. Geng, and Y. Li, "On the importance of gradients for detecting distributional shifts in the wild," *Advances in Neural Information Processing Systems*, vol. 34, pp. 677–689, 2021.

[20] X. Zhang, J. Chen, Y. Zhou, L. Han, and J. Lin, "A multiple-layer representation learning model for network-based attack detection," *IEEE Access*, vol. 7, pp. 91 992–92 008, 2019.

[21] J. Ghadermazi, A. Shah, and N. D. Bastian, "Towards real-time network intrusion detection with image-based sequential packets representation," *IEEE Transactions on Big Data*, 2024.

[22] Y. Xiao, C. Xing, T. Zhang, and Z. Zhao, "An intrusion detection model based on feature reduction and convolutional neural networks," *IEEE Access*, vol. 7, pp. 42 210–42 219, 2019.

[23] P. Sun, P. Liu, Q. Li, C. Liu, X. Lu, R. Hao, and J. Chen, "Dl-ids: Extracting features using cnn-lstm hybrid network for intrusion detection system," *Security and communication networks*, vol. 2020, no. 1, p. 8890306, 2020.

[24] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *arXiv preprint arXiv:1312.6199*, 2013.

[25] R. Sheatsley, B. Hoak, E. Pauley, Y. Beugin, M. J. Weisman, and P. McDaniel, "On the robustness of domain constraints," in *Proceedings of the 2021 ACM SIGSAC conference on computer and communications security*, 2021, pp. 495–515.

[26] R. Sheatsley, N. Papernot, M. Weisman, G. Verma, and P. McDaniel, "Adversarial examples in constrained domains," *arXiv preprint arXiv:2011.01183*, 2020.

[27] T. Bradley, E. Alhajjar, and N. D. Bastian, "Novelty detection in network traffic: Using survival analysis for feature identification," in *2023 IEEE International Conference on Assured Autonomy (ICAA)*. IEEE, 2023, pp. 11–18.

[28] N. Carlini, F. Tramer, K. D. Dvijotham, L. Rice, M. Sun, and J. Z. Kolter, "(certified!!) adversarial robustness for free!" *arXiv preprint arXiv:2206.10550*, 2022.

[29] S. Cai, Y. Zhao, J. Lyu, S. Wang, Y. Hu, M. Cheng, and G. Zhang, "Ddpdar: Network intrusion detection based on denoising diffusion probabilistic model and dual-attention residual network," *Neural Networks*, vol. 184, p. 107064, 2025.

[30] S. Zhang, T. Li, D. Jin, and Y. Li, "Netdiff: A service-guided hierarchical diffusion model for network flow trace generation," *Proceedings of the ACM on Networking*, vol. 2, no. CoNEXT3, pp. 1–21, 2024.

[31] N. Alhussien and A. Aleroud, "Advpurrec: Strengthening network intrusion detection with diffusion model reconstruction against adversarial attacks," in *2024 IEEE 23rd International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. IEEE, 2024, pp. 1638–1646.

[32] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*. Springer, 2016, pp. 694–711.

[33] A. Chernikova and A. Oprea, "Fence: Feasible evasion attacks on neural networks in constrained environments," *ACM Transactions on Privacy and Security*, vol. 25, no. 4, pp. 1–34, 2022.

[34] E. Alhajjar, P. Maxwell, and N. Bastian, "Adversarial machine learning in network intrusion detection systems," *Expert Systems with Applications*, vol. 186, p. 115782, 2021.