Perturbation-Resilient for Temporal-Camouflaged IoT Attacks

Xiaohui Li, *Member, IEEE,* Yuanyuan Li, Zhentian Zhong, Linfeng Tan, Junfeng Wang*, Jiayong Liu

***Abstract*—The growing adoption of Internet of Things (IoT) devices has introduced significant challenges to network security due to their heterogeneous nature and temporal pattern vulnerabilities. Among the emerging threats, adversarial attacks targeting IoT network traffic have gained attention for their ability to evade traditional and machine learning-based Network Intrusion Detection Systems (NIDS). While prior work has focused on static adversarial perturbations, these approaches fail to account for the temporal dynamics inherent in IoT traffic. IoT networks exhibit time-dependent patterns driven by device behavior, environmental factors and user interactions, creating an opportunity for more sophisticated adversarial strategies. This paper introduces a co-evolutionary adversarial framework termed dynamic Adversarial Temporal-Camouflaged Perturbation (ATCP) for IoT network attack traffic. ATCP dynamically segments network traffic into temporal intervals and applies targeted adversarial perturbations to each segment. By leveraging the temporal characteristics of IoT traffic, the proposed method generates subtle yet effective adversarial modifications that confuse NIDS by disrupting their ability to model time-dependent traffic patterns. Unlike static perturbation methods, ATCP provides valuable insights into adversarial attack methodologies, lays the foundation for developing more robust IoT security frameworks, and adapts to evolving traffic dynamics, making it a more effective and robust NIDS in real-world scenarios. Extensive experiments conducted on real-world IoT network datasets demonstrate that the proposed method achieves high evasion rates against Machine Learning (ML) based NIDS while preserving the functional integrity of IoT communications. Notably, among the four NIDS evaluated, KitNET experiences the most significant degradation, with its detection rate dropping from 93.07% to 18.55% after applying ATCP. Furthermore, ATCP exhibits strong adaptability across diverse IoT device types and network configurations, highlighting its generalizability.**

***Index Terms*—IoT attacks, NIDS vulnerability, temporal adversarial, attack evasion, defense resilience.**

I. Introduction

INTERNET of things (IoT) has revolutionized modern computing by enabling a wide array of interconnected devices to seamlessly exchange information. From smart homes to industrial automation, IoT networks have become the cornerstone of critical infrastructures [1], [2]. However, this unprecedented connectivity also introduces significant security vulnerabilities. IoT devices, often resource-constrained and deployed without robust security protocols, are attractive targets for attackers seeking to exploit network traffic patterns [3], [4], [5], [6], [7], [8]. As a result, the security of IoT networks has become a pressing challenge, particularly in the context of advanced adversarial threats [9].

Among these emerging threats, adversarial attacks on network traffic have gained prominence [3], [4]. These attacks involve crafting perturbations to network traffic that evade detection by traditional and ML based NIDS, while maintaining the functionality of the IoT network. Existing research has primarily focused on static adversarial techniques, which generate perturbations without considering the temporal dynamics of IoT traffic. However, IoT networks are inherently temporal systems, where traffic patterns evolve over time due to device behavior, environmental changes and user interactions [10], [11], [12], [13]. This temporal nature opens a new frontier for adversarial strategies that exploit time-dependent vulnerabilities in NIDS [14], [15], [16]. There are three core problems:

- **Weak temporal modeling in NIDS.** Existing NIDS primarily rely on static features, e.g., traffic volume, packet size, or global statistical patterns, neglecting the significant temporal dynamics in IoT network traffic. IoT devices often exhibit time-dependent communication behaviors, such as periodic or event-driven patterns, which introduce vulnerabilities exploitable by adversarial traffic. The inability of current NIDS to effectively model these temporal dependencies makes them susceptible to dynamic, time-based adversarial attacks.
- **Limitations of static perturbation methods.** Static adversarial perturbation techniques are designed based on global optimization strategies, ignoring the highly dynamic nature of IoT traffic that changes over time. IoT traffic requires adversarial methods to dynamically adapt perturbations to unique characteristics of traffic at different time intervals. The lack of adaptability in static methods restricts their effectiveness and applicability in real-world IoT scenarios.
- **Conflict between stealthiness and attack effectiveness.** Adversarial perturbation must balance three critical factors: stealthiness (avoiding detection by NIDS), attack effectiveness (successfully evading detection), and maintaining communication functionality (preserving normal IoT device operations). Existing methods often prioritize attack effectiveness, neglecting the potential impact on network performance, leading to communication disruptions or device malfunctions, which limits their usability in practical environments.

In this paper, we propose a novel adversarial methodology termed ATCP for IoT network traffic reshaping. The core idea

Manuscript received 17 April 2025; revised X X 2025; accepted X X 2025. Date of publication X X 2025; date of current version X X 2025.

This work was supported in part by the National Natural Science Foundation of China under Grant U24B20147 and Grant U2133208; and in part by the Major Science and Technology Special Project of Sichuan Province under Grant 2024ZDZX0044, 2024ZHCG0195 and 2024ZYD0269

Xiaohui Li, Yuanyuan Li, Lingfeng Tan and Jiayong Liu are from the School of Cyber Science and Engineering, Sichuan University, Chengdu, China, 610065.

Zhentian Zhong is with the Pittsburgh Institute, Sichuan University, Chengdu, China, 610065.

Junfeng Wang is from the College of Computer Science, Sichuan University, Chengdu, China, 610065. (*Corresponding author: wangjf@scu.edu.cn*).

Manuscript received April 17, 2025.

of ATCP is to segment IoT network traffic into time-dependent fragments and apply dynamic adversarial perturbations tailored to each segment. By leveraging the temporal characteristics of IoT traffic, the proposed approach introduces subtle yet effective modifications that evade NIDS by disrupting their ability to model temporal dependencies. Unlike static approaches, ATCP dynamically adapts to evolving traffic patterns, making it more robust and effective in real-world IoT scenarios.

The primary work of this paper are as follows. First, we introduce a dynamic time-segmentation framework that identifies key temporal intervals in IoT traffic and applies targeted adversarial perturbations. Second, we demonstrate the efficacy of ATCP in evading ML based NIDS through extensive experiments in realistic IoT environments. Third, we propose a co-evolutionary mechanism for NIDS that enhances detection robustness against temporal adversarial perturbations.

Ultimately, this study highlights the critical need for incorporating temporal dynamics into both adversarial attack methodologies and NIDS defense mechanisms. By exposing the vulnerabilities of time-sensitive traffic analysis, we aim to advance the understanding of adversarial threats in IoT networks and provide insights into developing more resilient security frameworks. This paper makes the following main contributions to the field of IoT network security and adversarial attack methodologies:

- We introduce dynamic ATCP, a novel method that leverages IoT traffic's temporal dynamics. ATCP dynamically segments traffic into time intervals and generates tailored adversarial perturbations, effectively exploiting the temporal dependencies in IoT traffic to evade detection by NIDS.
- A dynamic segmentation framework is developed to identify critical temporal intervals in IoT network traffic. This framework adapts to the evolving traffic patterns, enabling more precise and effective perturbations compared to traditional static adversarial strategies.
- The ATCP framework achieves dual objectives through an integrated coevolution mechanism to optimal balance between attack stealthiness and evasion success. Moreover, it systematically hardens NIDS defenses via adversarial sample recycling. This automated process curates successful attack patterns to generate dynamic retraining datasets, creating a continuous learning loop where detectors progressively adapt to emerging temporal attack strategies.

The rest of the paper is organized as follows. Section II provides the background and related work on IoT NIDS and adversarial attack techniques. Section III formalizes the problem, outlines the adversary's objectives, and details the challenges posed by IoT traffic characteristics. Section IV presents the ATCP framework, including its dynamic segmentation strategy, perturbation generation process and defense mechanisms. Section V discusses the comparison results of different dimensions on public and real datasets and the effectiveness and adaptability of ATCP. Section VI concludes the paper.

II. Related Work

This section will discuss related work, including NIDS vulnerabilities in IoT environments, adversarial attacks on NIDS, and adversarial attacks on time series prediction.

A. NIDS for IoT environment

IDS is an essential component of network security, widely utilized to identify and defend against malicious activities within networks [17]. Early research on IDS predominantly focused on two approaches: signature-based IDS, which relies on pattern matching to detect known attacks, and anomaly detection-based IDS, which identifies unusual activities by modeling what is considered normal behavior. NIDS serves as a critical line of defense, responsible for detecting malicious traffic and maintaining the stable operation of systems in IoT environments. However, recent studies have identified significant vulnerabilities in mainstream NIDS when confronted with adversarial attacks, as attackers can circumvent detection by subtly altering network traffic [18], [19], [20], [21], [22], [23], [24], [25]. The diverse array of devices and the resource constraints commonly found in IoT environments heighten the sensitivity of IoT-IDS to adversarial samples, particularly at the time series level, where robustness tends to be inadequate. While existing research has primarily focused on adversarial perturbations in feature space, the exploration of attacks targeting the traffic level, particularly within the time series dimension, remains limited. This presents a valuable opportunity for the development of new attack methodologies.

B. Adversarial Attacks on NIDS

Adversarial attacks aimed at NIDS typically involve manipulating network traffic through minor perturbations, compromising the ability to accurately identify malicious activities. Recent studies have classified these attacks into two principal categories: feature-space attacks and traffic-space attacks.

1) Feature-Space Attacks: Feature-space attacks are the most mature path in the current IDS adversarial attack research. The core of feature-space attacks is to impose small perturbations on the feature vectors extracted from network traffic, thereby inducing NIDS to make misclassifications [26], [27], [28], [29], [30], [31], [32]. This type of attack does not directly modify the original traffic data but attacks after the traffic is converted into structured features. It has a low implementation threshold and a wide application range. Representative research includes IDSGAN [27], which uses GAN to generate adversarial samples with original attack semantics in a black-box setting and limits the perturbation range through a function preservation mechanism to avoid destroying the attack behavior. DIGFuPAS [28] is extended to the SDN environment, optimizing the distribution of generated samples through WGAN and maintaining the protocol consistency of the attack traffic. NIDSFM [29] reconstructs the latent space distribution based on the flow model, isolates the discriminant features to maintain malicious behavior, and effectively deceives multiple NIDS. AIDAE [30] uses adversarial autoencoders to reconstruct the normal feature distribution, sample and generate

misleading features, significantly reducing the detection performance. GPMT [31] emphasizes practicality. With very little prior knowledge, it attacks multiple models through WGAN to generate deceptive and feasible feature perturbations. In addition, Roshan *et al.* [32] showed that even in white-box attacks, NIDS still showed significant vulnerability to common attacks such as FGSM, JSMA, PGD, and C&W using heuristic defense methods such as adversarial training, which further proves the challenge of feature-space attacks to NIDS. Even if the target model has been trained based on adversarial samples, carefully designed feature perturbations can still significantly improve the escape rate, reflecting the widespread threat of such attacks to the robustness of NIDS at the feature-space level.

Although Feature-Space Attacks have good evasion capabilities at the numerical level, one of their major limitations is that they cannot guarantee that the perturbed features can be mapped to legitimate traffic. Therefore, they are often considered to have a problem of insufficient practical feasibility.

2) Traffic-Space Attacks: Compared with Feature-Space Attacks, Traffic-Space Attacks directly act on the original network data layer. By directly modifying the original data, such as timestamps, packet order, or packet frequency, they avoid the irreversible problem of feature extraction and are more in line with the actual deployment environment [33], [34], [35], [36], [37], [38]. Apruzzese *et al.* [33] proposed a method based on Deep Reinforcement Learning (DRL) to generate adversarial samples by directly modifying the timing characteristics of network traffic, such as packet interval and duration, to evade detection. ANT [34] designed three attack methods for different input spaces: AdvPad modifies the packet payload, AdvPay inserts dummy packets, and AdvBurst perturbs the packet mode to achieve universal adversarial traffic injection for packet level, flow level and timing model. Han *et al.* [35] constructed a traffic perturbation framework in an actual deployment environment. They used a heuristic transformation strategy to modify statistical features such as delay and rate between packets to evade detection without changing the function. TANTRA [36] uses LSTM to model the time series of regular traffic. In the attack phase, it adjusts the inter-packet delays of malicious traffic to achieve timing camouflage and conducts end-to-end attacks based on time disturbance. FENCE [37] modifies traffic fields such as the number of network connections and timing structure under the constraints of syntax/semantics through iterative gradient optimization to generate real and deliverable adversarial flows, emphasizing feature dependency consistency. Adv-Bot [38] proposes a black-box adversarial method that modifies the original data packet while retaining the attack function and protocol semantics to construct executable adversarial Botnet traffic and bypass NIDS.

Traffic-space attacks directly manipulate the time structure or packet content of the original network traffic while maintaining the attack semantics and protocol compliance to evade NIDS detection. This avoids the traffic irreversibility problem caused by directly modifying features in Feature-Space Attacks.

C. Adversarial Attacks on Time Series

In recent years, researchers have begun to extend adversarial attacks from the feature domain to the time domain and mislead the output of prediction models by manipulating the local or global structure of time series, further exposing the vulnerability of deep time series models [39], [40], [41], [42], [43], [44]. Compared with static data such as images, time series attacks need to consider the persistence and interpretability of disturbances. Its unique dynamic dependency structure makes it more sensitive to abnormal changes [39]. TANTRA proved for the first time that NIDS detection based on time series modeling can be effectively evaded by adjusting the time series structure of malicious traffic, which triggered widespread attention to the "perturbation time series structure" attack method in the field of IDS adversarial. Wu *et al.* [40] designed and combined a global gradient perturbation algorithm with importance scoring to select the minimum perturbation point. Yang *et al.* [41] proposed a black-box adversarial attack method, TSadv, which generates adversarial samples through local perturbation and differential evolution algorithm and successfully evades the detection of time series classification models. Zhao *et al.* [42] introduced a query-free perturbation localization and learning attack framework, significantly reducing the number of model accesses through prior learning. Zhang *et al.* [43] used a local perturbation strategy that combined sliding windows and differential evolution to reduce perception while maintaining the ability to predict offsets. Shen *et al.* [44] explored a TCA method that introduced time consistency constraints to achieve controllable time perturbation attacks on prediction models such as LSTM.

These studies together constitute a new paradigm for temporal adversarial attacks, emphasizing targeted, executable, and low-perception perturbation generation in the temporal dimension.

III. Problem Scope

A. Threat Model

ATCP establishes a realistic and challenging black-box attack model targeting NIDS. In this model, the attacker has no access to the internal structure, parameters, or feature extraction processes of the NIDS and cannot modify the network environment or detection logic. The attack is constrained to operate within the problem space, where adjustments are made to raw malicious traffic before it enters the NIDS. The attacker passively observes normal network traffic to extract temporal behavioral features such as packet inter-arrival times, periodicity and transmission regularity. These features are then used to transform malicious traffic, adjusting packet timing to mimic normal traffic patterns and effectively disguise the attack.

This attack strategy bypasses the need for reverse engineering or gradient-based optimization, making it computationally efficient and highly effective. The transformed malicious traffic is designed to evade detection by resembling normal traffic in its temporal behavior, reducing the likelihood of being flagged by the NIDS. The proposed model is particularly well-suited for resource-constrained IOT environments, where attackers

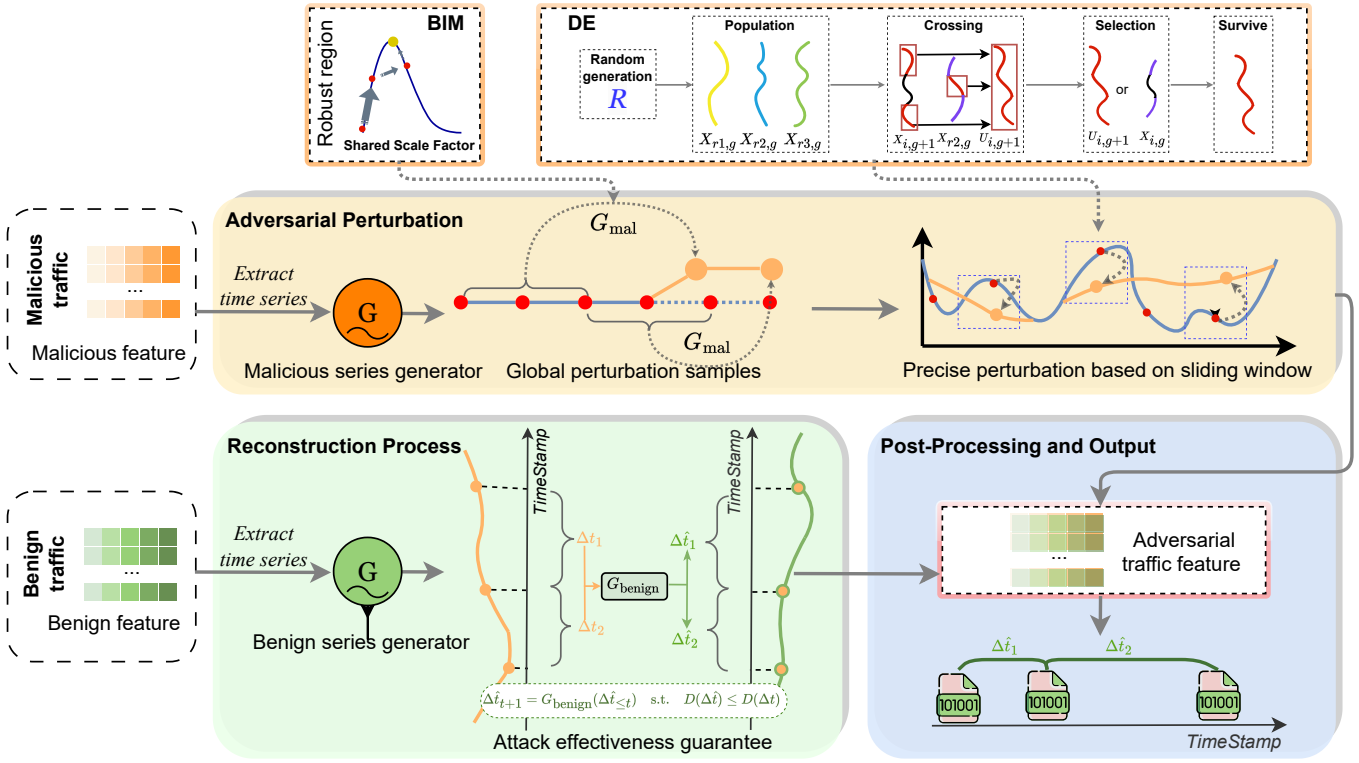


Fig. 1. ATCP Framework Architecture.

often operate under limited resources but have a strong intent to evade detection. This approach highlights the vulnerabilities of NIDS systems to temporal behavior-based evasion techniques and provides a foundation for understanding and mitigating such attacks.

B. Problem Formulation

Let the original malicious traffic be represented by a sequence of packets $\{p_0, p_1, \ldots, p_n\}$, where each packet p_i has a corresponding timestamp t_i. The temporal behavior of the traffic can be described by the Inter-Packet Delay (IPD) sequence:

$$\Delta T = \{\Delta t_0, \Delta t_1, \ldots, \Delta t_{n-1}\}, \quad \Delta t_i = t_{i+1} - t_i \tag{1}$$

where t_i represents the timestamp of the i-th packet. Δt_i is the time interval between consecutive packets.

To achieve robust traffic camouflage, we propose a dual-phase optimization framework. In the first phase, the temporal patterns of malicious traffic are deliberately perturbed to obscure distinctive timing signatures that are easily recognized by detection systems. In the second phase, the modified sequence is reshaped to resemble the timing characteristics of benign traffic. This progressive transformation of the traffic's temporal profile ensures that it no longer retains its original malicious signature while exhibiting realistic timing behavior, effectively reducing the risk of detection.

1) Temporal Perturbation: The ATCP method introduces temporal perturbations $\delta = \{\delta_0, \delta_1, \ldots, \delta_{n-1}\}$ to the original IPD sequence ΔT, generating a perturbed sequence ΔT^{adv}:

$$\Delta t_i^{adv} = \Delta t_i + \delta_i \tag{2}$$

where $|\delta_i| \leq \epsilon$ ensures that the perturbation is bounded to avoid introducing obvious anomalies.

The first goal is to maximize the discrepancy between the perturbed IPD distribution $P_{adv}(\Delta t)$ and the original malicious IPD distribution $P_{mal}(\Delta t)$. This is achieved by maximizing the Kullback-Leibler (KL) divergence:

$$\mathcal{L}_{adv} = D_{\mathrm{KL}}\left(P_{mal}(\Delta t) \| P_{adv}(\Delta t)\right) \tag{3}$$

where Δt denotes the IPD between consecutive packets in a traffic sequence. In IoT traffic, such timing patterns often reflect application-layer behavior and are commonly exploited for detecting malicious activity. Maximizing the KL divergence increases the statistical distance between the original and perturbed timing distributions, thereby reducing the recognizability of malicious temporal signatures.

The KL divergence is defined as:

$$D_{\mathrm{KL}}(P\|Q) = \int P(\Delta t) \log \frac{P(\Delta t)}{Q(\Delta t)}\, d\Delta t \tag{4}$$

where it quantifies the divergence of the perturbed IPD distribution $Q(\Delta t)$ from the original $P(\Delta t)$. A larger value indicates a greater distortion of the original temporal characteristics.

After obtaining $P_{adv}(t)$, the traffic is reshaped using G_{benign} to generate the target traffic $P_T(\Delta t)$:

$$P_T(\Delta t) = G_{benign}(P_{adv}(\Delta t)) \tag{5}$$

This reshaping process operates in the IPD space, ensuring that the resulting traffic closely mimics benign traffic while preserving the essential attack logic.

2) Optimization Objective: The optimal perturbation δ^* is obtained by solving the following optimization problem:

$$\delta^* = \arg\min_{\delta} \mathcal{L}(\delta) \tag{6}$$

subject to $|\delta_i| \leq \epsilon, \forall i$, where ϵ is a predefined perturbation bound.

IV. ATCP Method

To achieve evasion against NIDS under black-box conditions, ATCP based on a dual-generator architecture, which is divided into three main phases shown in Figure 1.

A. Preparation

Under the black-box assumption, attackers cannot access the internal structure or parameters of the NIDS and must rely solely on collecting network traffic for analysis. To mimic the temporal behavior of benign traffic, the attacker uses a Long Short-Term Memory (LSTM) model to train a benign time-series generator based on the observed inter-packet timing sequence of normal traffic.

The LSTM-based generator G_{benign} is trained to minimize the Mean Squared Error (MSE) between the generated sequence $\Delta\hat{T}$ and the real sequence ΔT:

$$\mathcal{L}_{benign} = \frac{1}{n}\sum_{i=0}^{n-1}\left(\Delta t_i - \Delta\hat{t}_i\right)^2 \tag{7}$$

After training, the generator learns to approximate the temporal behavior of normal traffic, allowing it to produce sequences that are statistically similar to those of benign traffic.

B. Adversarial Perturbation Phase

To perform adversarial evasion, ATCP introduces carefully crafted perturbations into the temporal sequence of malicious traffic. The goal is to generate adversarial traffic that maximizes the difference from the temporal characteristics of malicious traffic while preserving the attack logic.

Let the temporal sequence of malicious traffic be denoted as $\Delta T_{\text{mal}} = \{\Delta t_0, \Delta t_1, \ldots, \Delta t_{n-1}\}$. An adversarial generator G_{adv} is trained to perturb the sequence ΔT_{mal} into an adversarial sequence ΔT_{adv}, such that ΔT_{adv} maintains the attack logic but introduces sufficient perturbations to make it distinct from ΔT_{mal}.

1) Global Perturbation: The Basic Iterative Method (BIM) [45] is used to generate global adversarial perturbations:

$$\delta_{global} = \epsilon \cdot \text{sign}(\nabla_{\Delta T_{mal}} \mathcal{L}_{NIDS}) \tag{8}$$

where ϵ is the perturbation magnitude, and $\mathcal{L}_{NIDS}$ is the loss function of the NIDS.

To generate perturbations effectively, the BIM is employed. The attacker initializes the perturbed sequence ΔT_{adv} as the original malicious sequence ΔT_{mal}. Perturbations are then iteratively applied to maximize the MSE between the adversarial sequence and the malicious target sequence:

$$\mathcal{L}_{adv} = \frac{1}{n}\sum_{i=0}^{n-1}\left(M_{mal}(\Delta t_i) - M_{adv}(\Delta t_i)\right)^2 \tag{9}$$

Algorithm 1: BIM-based Global Perturbation Attack

Input: Malicious sequence ΔT_{mal}, malicious LSTM model M_{adv}, iterations K, step α, max perturb ϵ
Output: Adversarial sample ΔT^{adv}

Function BIM(ΔT_{mal}, M_{adv}, K, α, ϵ):
  $\Delta T_{\text{adv}}^{(0)} \leftarrow \Delta T_{\text{mal}}$
  for $k = 1$ **to** K **do**
    // Compute loss:
    $\mathcal{L}_{\text{adv}} = \frac{1}{n}\sum_{i=0}^{n-1}\left(M_{\text{adv}}(\Delta t_i) - M_{\text{adv}}(\Delta t_i)\right)^2$
    // Compute gradient:
    $g^{(k)} = \nabla_{\Delta T_{\text{adv}}^{(k)}} \mathcal{L}_{\text{adv}}$
    // Update adversarial sequence:
    $\Delta T_{\text{adv}}^{(k+1)} = \text{Clip}_{\epsilon}\left(\Delta T_{\text{adv}}^{(k)} + \alpha \cdot \text{sign}(g^{(k)})\right)$
  return ΔT^{adv}

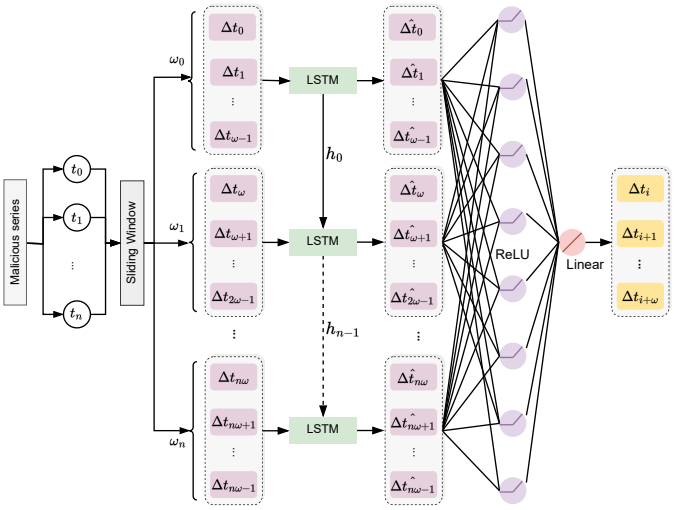


Fig. 2. LSTM Structure Diagram.

where $M_{mal}(\cdot)$ and $M_{adv}(\cdot)$ represent the temporal models of malicious and adversarial traffic, respectively.

At each iteration, the perturbation δ_i is updated as:

$$\Delta T_{adv}^{(k+1)} = \text{Clip}_{\epsilon}\left(\Delta T_{adv}^{(k)} + \alpha \cdot \text{sign}\left(\nabla_{\Delta T_{adv}^{(k)}} \mathcal{L}_{adv}\right)\right) \tag{10}$$

where α is the step size, ϵ is the perturbation bound, ensuring that $|\delta_i| \leq \epsilon$, $\nabla_{\Delta T_{adv}^{(k)}} \mathcal{L}_{adv}$ is the gradient of the loss function with respect to the current adversarial sequence.

The iterative process terminates once the adversarial sequence meets the predefined similarity threshold or a maximum number of iterations is reached.

2) Local Perturbation Optimization: To further enhance the effectiveness of adversarial perturbations at the local level, this subsection introduces Differential Evolution (DE) [46] as a strategy for optimizing within a sliding window. DE ensures that locally crafted perturbations can better approximate the temporal characteristics of benign traffic, while also maintaining the overall integrity of the malicious traffic.

- **Initialization.** Let the sliding window size be W, and the population size for DE be N. Each individual in the population represents a candidate perturbation vector x_i for the adversarial sequence ΔT_{adv} within the window. The elements of x_i are initialized as small random values

within the range $[-\epsilon, \epsilon]$, where ϵ is the perturbation bound.

- **Mutation.** For each individual x_i, a mutation vector v is created by combining three randomly selected distinct individuals x_{r1}, x_{r2}, and x_{r3} (where $r1, r2, r3 \neq i$) from the population:

$$v = x_{r1} + F \cdot (x_{r2} - x_{r3}) \tag{11}$$

where $F \in (0, 2]$ is the scaling factor that controls the amplification of the differential variation.

- **Crossover.** To enhance population diversity and combine traits from both the mutation vector v and the original individual x_i, a crossover operation generates the trial vector u as follows:

$$u_j = \begin{cases} v_j, & \text{if rand}(0,1) < CR \text{ or } j = j_{\text{rand}} \\ x_{ij}, & \text{otherwise} \end{cases} \tag{12}$$

where $CR \in [0, 1]$ is the crossover probability, $\text{rand}(0, 1)$ generates a random value in the range $[0, 1]$, and j_{rand} is a randomly chosen index that ensures at least one element from the mutation vector v is copied into u.

- **Selection.** The quality of the trial vector u is assessed using a local loss function that measures the deviation from the original malicious timing pattern:

$$\mathcal{L}_{adv} = \frac{1}{W} \sum_{j=0}^{W-1} \left(M_{mal}(\Delta t_j) - M_{adv}(\Delta t_j)\right)^2 \tag{13}$$

where $M_{mal}(\cdot)$ and $M_{adv}(\cdot)$ represent the temporal models for malicious and adversarial traffic, respectively.
The trial vector u replaces x_i in the population if it results in a lower loss:

$$x_i = \begin{cases} u, & \text{if } \mathcal{L}_{adv}(u) < \mathcal{L}_{adv}(x_i) \\ x_i, & \text{otherwise} \end{cases} \tag{14}$$

- **Iterative Process.** Within a sliding window of size w, DE is applied to refine the perturbations:

$$\delta^*_{local} = \arg\max_{\delta_{local}} D_{\text{KL}} \left(P_{mal}(\Delta t) \| P_{adv}(\Delta t; \delta_{global} + \delta_{local})\right) \tag{15}$$

The DE algorithm iteratively refines the perturbations within the sliding window until one of the following termination criteria is met:

1) The loss $\mathcal{L}_{adv}$ falls below a predefined threshold.
2) A maximum number of iterations is reached.

C. Reconstruction Phase

To further improve the naturalness of the temporal behavior of the adversarial sequence and ensure it closely mimics benign traffic, the ATCP method introduces a traffic reconstruction phase. This phase reconstructs the perturbed sequence ΔT_{adv} using an autoregressive benign generator G_{benign}, trained on normal traffic data. The goal is to produce a refined IPD sequence $\Delta \hat{T}_{adv}$ that appears more natural.

- **Reconstruction Process.** Let $\Delta T_{adv} = \{\Delta t_0^{adv}, \Delta t_1^{adv}, \ldots, \Delta t_{n-1}^{adv}\}$ represent the adversarial IPD sequence. The benign generator G_{benign}, trained on normal traffic, reconstructs the sequence iteratively:

$$\Delta \hat{t}_i = G_{benign}(\Delta \hat{t}_{i-1}) \tag{16}$$

where $\Delta \hat{t}_i$ is the reconstructed IPD at step i, and $\Delta \hat{t}_{i-1}$ is the output from the previous step.
The refined adversarial sequence is then converted back into a timestamp sequence:

$$\hat{t}_i = t_0 + \sum_{j=0}^{i} \Delta \hat{t}_j \tag{17}$$

where t_0 is the starting timestamp of the traffic.

- **Post-Processing and Output.** Once the refined sequence $\Delta \hat{T}_{adv}$ is generated, it is mapped onto the malicious payload packets to produce a complete PCAP file. This reconstructed traffic retains the attack logic while presenting temporal features that are statistically indistinguishable from benign traffic. The reshaped traffic improves the evasion success rate against NIDS without compromising the functional integrity of the attack.

D. Defense Mechanism

To counteract the ATCP attack, this paper proposes a defense mechanism based on adversarial training. The goal is to enhance the robustness of the NIDS by exposing it to adversarial examples during training, allowing it to better detect disguised malicious traffic.

- **Adversarial Training Process.** Let $\Delta \hat{T}_{adv}$ represent adversarial traffic generated by the ATCP method, and let ΔT_{benign} represent benign traffic. The NIDS classifier f_θ with parameters θ is trained to minimize the loss function:

$$\mathcal{L}_{NIDS}(\theta, v) = \mathbb{E}_{x \sim \mathcal{D}} \left[\ell\left(f_\theta(x), y\right)\right] \tag{18}$$

where $\mathcal{D} = \Delta T_{benign}, (\Delta \hat{T}_{adv})$ is the dataset consisting of both benign and adversarial traffic, y is the true label of the traffic ($y = 0$ for benign, $y = 1$ for adversarial), and $\ell(\cdot)$ is the classification loss function, such as cross-entropy.

- **Robustness Objective.** The adversarially trained NIDS seeks to minimize the worst-case loss over potential adversarial examples:

$$\min_{\theta} \max_{\Delta T_{adv} \in \mathcal{D}_{adv}} \mathcal{L}_{NIDS}(\theta, \Delta \hat{T}_{adv}) \tag{19}$$

where $\mathcal{D}_{adv}$ represents the set of adversarial traffic generated by the ATCP method.

- **Defense Implementation.** The defense mechanism consists of three key steps:

1) **Adversarial Example Generation.** Generate adversarial traffic $\Delta \hat{T}_{adv}$ using ATCP.
2) **Adversarial Training.** Incorporate $\Delta \hat{T}_{adv}$ into the training dataset to improve the classifier's robustness.
3) **Evaluation.** Continuously evaluate the NIDS against new adversarial traffic to ensure sustained performance.

TABLE I
DATASET.

Attack Type	Attack Vector	Tool	Description	# Packets	Train [min.]	Execute [min.]
Botnet Malware	Mirai	Telnet	The attacker infects IoT with the Mirai malware by exploiting default credentials, and then scans for new vulnerable victims network.	764,137	52.0	66.9
Recon.	Fuzzing	SFuzz	The attacker searches for vulnerabilities in the camera's web servers bysending random commands to their cgis.	2,244,139	33.3	52.2
Man in the Middle	ARP MitM	Ettercap	The attacker intercepts all LAN trafic via an ARP poisoning attack.	2,504,267	8.05	20.1
	Active Wiretap	Raspberry PI 3B	The attacker intercepts all LAN traffic via active wiretap (network bridge) covertly installed on an exposed cable.	4,554,925	20.8	74.8
Denial of Service	SYN DoS	Hping3	The attacker disables a camera's video stream by overloading its web server.	2,771,276	18.7	34.1
	SSDP Flood	Saddam	The attacker overloads the DVR by causing cameras to spam the serverwith UPnP advertisements.	4,077,266	14.4	26.4
	SSL Renegotiation	THC	The attacker disables a camera's video stream by sending manySSL renegotiation packets to the camera.	6,084,492	10.7	54.9

This approach effectively enhances the NIDS's capability to detect temporal evasion attacks, reducing the success rate of ATCP-generated adversarial traffic.

V. EVALUATION

To rigorously evaluate the performance and impact of the ATCP in evading detection of malicious IoT traffic, we designed a series of systematic experiments. This chapter provides a detailed overview of the experimental setup and results. Specifically, we analyze the effectiveness of ATCP across different NIDS models and attack scenarios, assess its evasion capabilities under various perturbation strategies, and explore potential defense mechanisms, such as adversarial training, to enhance the robustness of NIDS. The experiments are structured to provide a holistic understanding of ATCP's strengths, limitations, and implications for IoT security.

A. Experimental Setup

1) Datasets: The experiments utilized the Kitsune network traffic dataset[1] shown in Table I, developed by Czech Technical University, specifically designed to support the evaluation of lightweight network intrusion detection systems. The dataset was collected from real-world network environments and contains various attack samples alongside their corresponding normal traffic. Notably, some attack samples originate from IoT devices, endowing the dataset with high realism and complexity. To ensure the comprehensiveness of the experiments, seven representative attack types were selected, accompanied by detailed information on attack vectors, tools, packet counts, training times, execution times, and whether the targets were IoT devices. These attack types include Botnet Malware, Reconnaissance, Man-in-the-Middle (MitM), and Denial of Service (DoS), specifically Mirai, SYN DoS, SSDP Flood, and SSL Renegotiation. These attacks are characterized by distinctive IoT-specific features, highly realistic and complex temporal behaviors, and regular patterns, making them particularly suitable for assessing the effectiveness of temporal adversarial methods.

[1]https://www.kaggle.com/datasets/ymirsky/network-attack-dataset-kitsune

2) Baseline NIDS: To evaluate the evasion performance of ATCP across different detection paradigms, we selected four representative NIDS: KitNET, Autoencoder (AE), Logistic Regression (LR), and Support Vector Machine (SVM). These models span unsupervised and supervised learning approaches, as well as deep and shallow architectures. KitNET and AE, as unsupervised anomaly detection systems, excel at identifying unknown attacks through temporal behavior modeling. In contrast, LR and SVM, as supervised classifiers, are effective in distinguishing known attack patterns. This diverse selection enables a comprehensive assessment of ATCP's generalizability and impact across varied detection mechanisms.

- **KitNET** is a lightweight unsupervised autoencoder network designed to learn the nonlinear distribution of high-dimensional features. In our case, the maximum autoencoder size was set to 10, with 5000 steps for feature mapping and 50,000 steps for anomaly detection. The learning rate was 0.1, and the hidden layer ratio was 0.75, balancing complexity and training efficiency to enhance anomaly detection.
- **AE** is a deep, symmetric autoencoder commonly used in anomaly detection tasks with three ReLU-activated encoder layers (100, 64, 32 units) and a mirrored decoder. Trained with the Adam optimizer (learning rate: 0.001, batch size: 32) for one epoch using a 70:30 train-validation split. Anomalies were identified by RMSE.
- **LR** is a linear classifier suitable for detecting attacks in separable feature spaces. Hyperparameters, where $C = \{0.1, 1, 10\}$ and $iterations_{max} = \{100, 500, 1000\}$, were tuned via five-fold GridSearchCV. The best model was used for anomaly detection.
- **SVM** is a supervised classifier well-suited for handling nonlinear feature distributions. Hyperparameters, where ($C = 0.1, 1, 10$ and $gamma = $ '$scale$', '$auto$', $0.01, 0.1, 1$), were optimized using five-fold GridSearchCV. The best model, trained with a maximum of 50 iterations, was used for classification.

3) Evaluation Metrics: In order to measure the attack effectiveness of ATCP and its resistance to temporal perturbations

TABLE II
EVALUATION METRICS.

Metrics	Formula
Detection Rate (DR)	$DR = \frac{TP}{TP+FN}$
Detection Evasion Rate (DER)	$DER = 1 - DR = \frac{FN}{TP+FN}$
Root Mean Square Error (RMSE)	$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2}$
Pearson Correlation Coefficient (R)	$R = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2 \sum_{i=1}^{n}(y_i-\bar{y})^2}}$

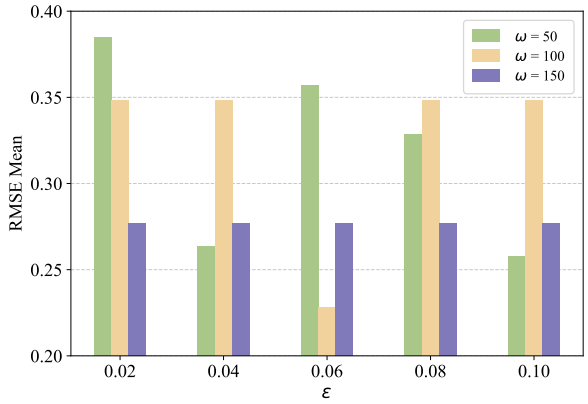


Fig. 3. Mean RMSE Values of Mirai Attack Detection using KitNET with Different ϵ and ω Values.

comprehensively, this paper use the following metrics, also summarized in Table Table II.

- **Detection Rate (DR).** Measures the model's ability to detect malicious traffic. True Positive (TP) is the number of malicious traffic samples correctly identified, and False Negative (FN) is the number of malicious traffic samples misclassified as normal traffic. A lower detection rate indicates stronger evasion effects against the model.
- **Detection Evasion Rate (DER).** A complementary metric to the detection rate measures the proportion of adversarial samples that successfully evade the detection system. A higher DER indicates a higher probability of successful evasion, making it a key metric for evaluating the effectiveness of adversarial attacks.
- **Root Mean Square Error (RMSE).** Measures the reconstruction error of the input by an autoencoder. A more significant RMSE value indicates that the input is more anomalous, x_i is the original input, $\hat{x}_i$ is the reconstructed output, and n is the feature dimension.
- **Pearson Correlation Coefficient (R).** Evaluate the linear relationship between the original malicious and adversarial traffic. Given two sequences, $X = \{x_1, x_2, \ldots, x_n\}$ and $Y = \{y_1, y_2, \ldots, y_n\}$, and $\bar{x}$ and $\bar{y}$ are the means of sequences X and Y, respectively. A smaller R value indicates weaker linear similarity between the adversarial and original traffic, highlighting a more significant effect of the temporal perturbations.

TABLE III
PARAMETERS.

Variable	Value	Description
ϵ	$\{0.02, 0.04, 0.06, 0.08, 0.10\}$	Perturbation amplitude
ω	$\{50, 100, 150\}$	Sliding window size
α	0.02	Sliding window step size
K	10	Maximum iterations of BIM
G	60	Maximum iterations of DE
P	600	Population size of DE
F	0.5	Scaling factor of DE
CR	0.1	Crossover probability of DE

B. Parameter Analysis

Configuring the LSTM generator is crucial for effective detection evasion during ATCP execution. The LSTM layer utilizes 64 hidden units with the ReLU activation function to capture long-term dependencies in network traffic. It comprises a single output neuron predicting the next time interval, with MSE as the loss function. The model is trained for 10 epochs with a batch size of 32 using the Adam optimization algorithm. Through adversarial perturbation of malicious timing, the dual generator enhances the evasion efficacy of the ATCP attack.

Key factors influencing ATCP effectiveness include the perturbation amplitude (ϵ) and the sliding window size (ω). As shown in Figure 3, with the NIDS KitNET detecting the Mirai attack, the optimal settings of $\epsilon = 0.06$ and $\omega = 100$ yield the lowest mean RMSE of 0.2283, indicating enhanced evasion performance. In contrast, more minor perturbations and larger window sizes result in poorer outcomes.

Other parameters is summarized in Table III, such as sliding window step size (α), number of BIM iterations (K), and differential evolution settings (G, P, F, CR), also impact attack efficacy. An α of 0.02 smooths perturbations, while $K = 10$ ensures efficiency without a heavy computational load. $G = 60, P = 600, F = 0.5, CR = 0.1$ stabilize the optimization process. Although their contributions are modest, these parameters are essential for reliable attack effectiveness.

C. Detection Evasion Analysis

This study comprehensively evaluated the effectiveness of three evasion strategies predicated on time-series mutation for circumventing various NIDS. The experiment employed a standardized flow feature input dimension and consistent detection parameters to facilitate a robust comparison of the mutation strategies, ensuring that all methods were assessed under equivalent conditions. Significantly, to validate the efficacy of the time-series adversarial perturbation methodology, we enhanced the existing framework known as TANTRA. The resulting method, referred to as TANTRA+, introduced adversarial perturbations to malicious time series based on the principles established in TANTRA.

The experimental findings are presented in Table IV. The original malicious traffic manifested a high detection rate among all examined NIDS, particularly within the two autoencoder architectures, KitNET and AE, which recorded average detection rates of 93.07% and 84.79% across seven attack

TABLE IV
THE EVALUATION OF **DR** & **DER** ACROSS DIFFERENT NIDS AND ATTACK TYPES.

NIDS	Attack Type	Original (DR)	TANTRA (DR)	TANTRA+ (DR)	ATCP (DR)	DER Disparity
KitNET	Mirai	0.8811	0.8372	0.0005	**0.0005**	**99.94%** ↑
	SSDP Flood	**0.9999**	0.0000	0.9938	**0.0000**	0.00% –
	Fuzzing	**0.9929**	0.7768	0.7768	0.0049	**99.37%** ↑
	SSL Renegotiation	0.9400	0.4001	0.4001	0.1705	57.39% ↑
	Active Wiretap	**0.9996**	0.0026	0.0026	0.0803	2988.46% ↓
	ARP MitM	**0.9995**	0.0004	0.0004	0.6231	155675% ↓
	SYN DoS	0.7019	0.4361	0.4361	0.4194	3.83% ↑
	Average	0.9307	0.3505	0.3729	**0.1855**	47.08% ↑
AE	Mirai	0.8784	0.0005	0.0005	**0.0005**	0.00% –
	SSDP Flood	0.5421	0.9978	0.9978	0.9980	0.02% ↓
	Fuzzing	0.8253	0.9747	0.9747	0.9741	0.06% ↑
	SSL Renegotiation	0.9853	0.4239	0.4239	0.0255	**93.98%** ↑
	Active Wiretap	**0.9998**	0.0000	0.0000	**0.0000**	0.00% –
	ARP MitM	**0.9999**	0.0001	0.0001	**0.0001**	0.00% –
	SYN DoS	0.7047	0.5408	0.5408	0.5382	0.48% ↑
	Average	0.8479	0.4197	0.4197	**0.3623**	13.68% ↑
LR	Mirai	0.8965	0.8838	0.8838	0.8832	0.07% ↑
	SSDP Flood	0.8332	0.0000	0.0000	**0.0000**	0.00% –
	Fuzzing	0.0191	0.0105	0.0105	0.0100	4.76% ↑
	SSL Renegotiation	0.0344	0.0203	0.0203	0.0187	7.88% ↑
	Active Wiretap	0.4993	0.2478	0.2478	0.2427	2.06% ↑
	ARP MitM	0.0070	0.1564	0.1564	0.1598	2.17% ↓
	SYN DoS	0.0048	0.0125	0.0125	0.0136	8.80% ↓
	Average	0.3278	0.1902	0.1902	**0.1897**	0.26% ↑
SVM	Mirai	0.8567	0.0000	0.0000	**0.0000**	0.00% –
	SSDP Flood	0.4975	0.0007	0.0007	**0.0004**	42.86% ↑
	Fuzzing	0.0208	0.0044	0.0044	**0.0045**	2.27% ↓
	SSL Renegotiation	0.8392	0.0202	0.0202	0.0190	5.94% ↑
	Active Wiretap	0.8409	0.0000	0.0000	**0.0000**	0.00% –
	ARP MitM	0.3453	0.0000	0.0000	**0.0000**	0.00% –
	SYN DoS	0.4045	0.7697	0.7697	0.7691	0.08% ↑
	Average	0.5436	0.1136	0.1136	**0.1133**	0.26% ↑

types. These results suggest significant competence in modeling anomalous temporal behavior. However, following the implementation of ATCP, the detection rates for both KitNET and AE experienced a marked decline, dropping to 18.55% and 36.23% on average, respectively, across various attack types. This significant reduction indicates that these unsupervised models exhibit substantial robustness deficiencies in the face of time-series perturbations.

In contrast, the traditional shallow supervised models, specifically LR and SVM, demonstrated weaker performance, with average detection rates of 32.78% and 54.36%, respectively. Upon application of the ATCP perturbation, their detection rates further diminished to 18.97% and 11.33%, indicating low feature dependence and limited generalization ability, which complicates the identification of disguised attacks after time reconstruction.

A detailed analysis of attack types revealed that ATCP displays remarkable camouflage capabilities against various attacks, notably Mirai, Active Wiretap, and ARP MitM, which were able to evade detection with DERs nearing 100%. Conversely, SSL Renegotiation demonstrated heightened sensitivity across all NIDS. It successfully evaded all three mutation strategies, implying that its anomalous behavior can be effectively disguised through precise modifications of timing structure.

In comparison, SYN DoS attacks exhibited consistently low DERs across all tested NIDS, with only a modest +3.8% increase observed in KitNET. This indicates that the primary detection features of SYN DoS attacks are not reliant on timing, but rather on factors such as traffic volume, packet rates, and protocol-level anomalies, including repetitive SYN flags and rapid bursts of connections. These characteristics remain evident even when inter-packet delays are altered. Given that ATCP is designed to reshape the temporal structure of traffic, it is less effective against attacks dominated by non-temporal features.

Furthermore, the timing of perturbation strategies between TANTRA+ and ATCP also markedly influences evasion effectiveness. TANTRA+ applies disturbances post-time reconstruction completion, potentially disrupting the quasi-benign timing pattern established by the generator, which may exacerbate abnormalities following the disturbance. As illustrated in Table V, we quantified R as the timing similarity between each mutation strategy and the original malicious traffic to substantiate this. For instance, in the SSDP Flood detected by KitNET, the correlation between TANTRA and TANTRA+ was 0.2112, indicating that even with disturbances, their timing behavior remains largely intact. In contrast, ATCP reduced R to 0.2040 under identical conditions. Although this reduction appears limited, it demonstrates that ATCP effectively modifies the

TABLE V
THE EVALUATION OF **R** ACROSS DIFFERENT NIDS AND ATTACK TYPES.

NIDS	Attack Type	TANTRA	TANTRA+	ATCP
KitNET	Mirai	0.7188	0.2661	**0.1818**
	SSDP Flood	0.2112	0.2112	0.2040
	Fuzzing	0.7423	0.7423	0.7357
	SSL Renegotiation	0.8982	0.8982	0.8966
	Active Wiretap	0.6379	0.6379	0.6658
	ARP MitM	0.8598	0.8598	0.8628
	SYN DoS	0.9591	0.9592	0.9597
	Average	0.7182	**0.6535**	0.6438
AE	Mirai	0.4237	0.4237	0.4218
	SSDP Flood	0.1500	0.1500	0.1446
	Fuzzing	0.3858	0.3858	0.3444
	SSL Renegotiation	0.4872	0.4872	0.4246
	Active Wiretap	-0.0013	-0.0013	-0.0014
	ARP MitM	-0.0109	-0.0109	**-0.0109**
	SYN DoS	0.9177	0.9177	0.9166
	Average	0.3360	0.3360	**0.3200**
LR	Mirai	0.9308	0.9308	0.9274
	SSDP Flood	0	0	0
	Fuzzing	0.7397	0.7397	0.7196
	SSL Renegotiation	0.6625	0.6625	0.6324
	Active Wiretap	0.1956	0.1956	0.2010
	ARP MitM	-0.0215	-0.0215	**-0.0213**
	SYN DoS	-0.0083	-0.0083	-0.0082
	Average	0.3570	0.3570	**0.3501**
SVM	Mirai	0.0467	0.0467	0.0368
	SSDP Flood	0.0649	0.0649	0.0618
	Fuzzing	0.4254	0.4254	0.4099
	SSL Renegotiation	0.2535	0.2535	0.2352
	Active Wiretap	-0.0052	-0.0052	**-0.0050**
	ARP MitM	0.1932	0.1932	0.2280
	SYN DoS	0.7620	0.7620	0.7605
	Average	0.2486	0.2486	**0.2467**

timing structure from the model's perspective through reconstruction post-disturbance. Moreover, ATCP did not experience the detection rate rebound observed in TANTRA+, indicating that it enhances evasion capabilities and effectively mitigates the abnormal amplification associated with disturbances. This corroborates the superiority of the "disturb first, then reshape" strategy in countermeasure evasion.

In conclusion, compared to TANTRA and its enhanced iteration TANTRA+, ATCP achieves a more substantial reduction in detection rates while facilitating universal evasion across various detectors and attack types, all without compromising the traffic's executability. This underscores its effectiveness and versatility as a robust black-box strategy for counteracting attacks.

D. Case Analysis of Evasion Performance

To accurately illustrate the evasion capabilities of ATCP in detection tasks, we visualize the probability of anomaly $(\mathcal{P}_a)$ for seven distinct types of cyber-attacks across four NIDS. Among these, SVM and LR are binary classifiers with outputs ranging from $[0, 1]$, allowing for direct interpretation as probabilities of anomaly. In contrast, KitNET and AE produce RMSE outputs, which are unbounded and specific to the model. To ensure consistent visualization, we normalize their outputs based on each model's anomaly detection threshold: values exceeding the threshold are mapped to the interval $[0.5, 1]$, indicating anomalies, while those below the threshold are mapped to $[0, 0.5)$, representing normal instances.

In Figure 4, the red and blue dots represent $\mathcal{P}_a$ of each packet before and after the execution of ATCP, respectively. The black dashed line indicates the unified classification threshold $(\mathcal{P}_a = 0.5)$, above which a packet is deemed anomalous by the NIDS.

The experimental results indicate that ATCP markedly reduces the probability of anomaly judgments and compresses many anomalous data packets into the acceptable range across most attack scenarios. Before ATCP execution, malicious traffic typically manifests dense anomaly warnings within multiple NIDS. Post-processing with ATCP reveals a significant downward shift in the overall detection curve, with anomaly points dispersing more widely, thereby exhibiting a "de-anomaly" effect following the disturbance of the time series structure. These findings suggest that ATCP not only disrupts the inherent behavioral patterns characteristic of the attacks but also mitigates the associated detection risks through a benign reconstruction mechanism, effectively eliminating potential threats to detection.

A further analysis comparing the various models demonstrates that KitNET and AE exhibit a heightened sensitivity to the time series structure. In the context of attacks such as Mirai and SSDP Flood, the abnormal probability of the original traffic is highly concentrated, resulting in most judgment outcomes exceeding the established threshold. Following ATCP processing, the density of anomalous data decreases significantly, and the traffic behavior stabilizes, indicating that the dual generator mode inherent to ATCP can effectively smooth the abnormal patterns associated with malicious traffic—conversely, models utilizing LR and SVM display comparatively subdued reactions before and after perturbation. Although the overall changes observed in these supervised models are less pronounced, ATCP effectively compresses boundary samples across various attack scenarios, thereby enhancing evasion capabilities.

High-frequency and high-amplitude abnormal peaks within NIDS detection characterize attacks such as Mirai and SSL Renegotiation. After the execution of ATCP, the persistence of these abnormalities is significantly attenuated, resulting in a smoother distribution of abnormal probabilities. Furthermore, the distribution of abnormal points associated with Active Wiretap and ARP MitM becomes markedly sparse following perturbation, indicating a disruption of timing consistency. However, it is essential to acknowledge that the abnormal points corresponding to SYN DoS remain unchanged, suggesting that the inherent characteristics of this attack are particularly resistant to concealment.

In conclusion, ATCP disrupts and reconstructs the timing characteristics of malicious traffic through a combination of global perturbation and local optimization. It achieves robust evasion across various NIDS and attack types without relying on internal model access, demonstrating strong adaptability for black-box scenarios and practical deployment.

TABLE VI
THE EVALUATION OF **DR** ACROSS DIFFERENT NIDS AND ATTACK TYPES.

NIDS	Attack Type	DR
LR	Mirai	**1.0000**
	SSDP Flood	**1.0000**
	Fuzzing	0.9983
	SSL Renegotiation	0.9969
	Active Wiretap	0.0010
	ARP MitM	0.9999
	SYN DoS	0.0227
	Average	**0.7170**
SVM	Mirai	0.9996
	SSDP Flood	0.2447
	Fuzzing	0.7652
	SSL Renegotiation	0.9999
	Active Wiretap	**1.0000**
	ARP MitM	0.9972
	SYN DoS	0.9883
	Average	**0.8564**
	Total Average	**0.7867**

E. Defensive Strategies Against ATCP

Although the ATCP method has substantially enhanced the success rate of evasion strategies against various NIDS, an essential aspect from a defensive standpoint is evaluating the model's ability to recover after exposure to disguised samples. In this context, we propose incorporating an adversarial training mechanism, which integrates traffic samples generated by ATCP into the model training process, thereby augmenting its capacity to identify timing evasion behaviors.

Table VI presents the improvements in detection rates for LR and SVM models post-adversarial training. The average detection rate for the SVM increased markedly from 11.33% to 85.64%, while the LR model showed an increase from 18.97% to 71.70%. These findings indicate that, despite the high concealment of disguised traffic, the model can still deciphered underlying structural characteristics. The SVM exhibited particularly strong recovery capabilities across most attack vectors, notably in the cases of SSL Renegotiation and ARP MitM attacks, where recovery rates approached 100%. This trend suggests that, while ATCP effectively reconstructs temporal behaviors to evade detection during the perturbation phase, its perturbative characteristics are not entirely resistant to learning. The model can establish a new decision boundary when incorporated into the training distribution, thereby recovering identifiable information embedded within the disguised structures.

Although adversarial training is not a new concept, our approach distinguishes itself from traditional defenses in two significant ways. First, while conventional methods emphasize small, imperceptible perturbations within feature space and are primarily tailored for static classifiers, our method, ATCP, operates within the temporal domain. It introduces structured, semantically meaningful alterations to inter-packet delay sequences. Second, we demonstrate that, despite their structural complexity, these timing-based perturbations can be partially learned through retraining, extending adversarial training to dynamic, sequence-based intrusion scenarios.

These findings highlight the learnability of temporal disguises and provide fresh insights into defending against structured, sequential perturbations, a mounting and often overlooked challenge in modern NIDS.

VI. CONCLUSION

In this paper, we propose ATCP, a novel and effective approach designed to evade NIDS in IoT environments. By leveraging fine-grained temporal perturbations, ATCP disrupts the temporal behavioral patterns that NIDS rely on for anomaly detection, while preserving the naturalness and functionality of the traffic. Through extensive experiments on multiple NIDS models and attack types, we draw that ATCP demonstrates a significant ability to evade detection across various NIDS, including KitNET, AE, SVM, and LR. In particular, it achieves an evasion success rate exceeding 75% for attacks manipulating traffic timing, disrupting packet order and delays, such as SSL Renegotiation and Active Wiretap. This highlights the effectiveness of ATCP in transforming detectable malicious behaviors into patterns resembling normal traffic. Adversarial training, which incorporates ATCP-mutated samples into the training process, has proven effective in restoring detection capabilities. For example, adversarial training improves the average detection rate of SVM from 11.33% to 88.67% and LR from 18.97% to 81.03%. However, the effectiveness of this defense depends on the model's capacity and its ability to generalize to complex temporal behaviors, with linear models like LR showing limited adaptability for certain attack types.

REFERENCES

[1] P. Sun, S. Shen, Y. Wan, Z. Wu, Z. Fang, and X.-z. Gao, "A survey of iot privacy security: Architecture, technology, challenges, and trends," *IEEE Internet of Things Journal*, 2024.

[2] R. Kalakoti, H. Bahsi, and S. Nõmm, "Improving iot security with explainable ai: Quantitative evaluation of explainability for iot botnet detection," *IEEE Internet of Things Journal*, vol. 11, no. 10, pp. 18 237–18 254, 2024.

[3] Y. Wan, K. Xu, F. Wang, and G. Xue, "Characterizing and mining traffic patterns of iot devices in edge networks," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 1, pp. 89–101, 2020.

[4] A. Sivanathan, H. H. Gharakheili, F. Loi, A. Radford, C. Wijenayake, A. Vishwanath, and V. Sivaraman, "Classifying iot devices in smart environments using network traffic characteristics," *IEEE Transactions on Mobile Computing*, vol. 18, no. 8, pp. 1745–1759, 2018.

[5] I. Hafeez, M. Antikainen, A. Y. Ding, and S. Tarkoma, "Iot-keeper: Detecting malicious iot network activity using online traffic analysis at the edge," *IEEE Transactions on Network and Service Management*, vol. 17, no. 1, pp. 45–59, 2020.

[6] N. Moustafa, B. Turnbull, and K.-K. R. Choo, "An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of internet of things," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4815–4830, 2018.

[7] Y. Wu, G. Lin, L. Liu, Z. Hong, Y. Wang, X. Yang, Z. L. Jiang, S. Ji, and Z. Wen, "Masinet: Network intrusion detection for iot security based on meta-learning framework," *IEEE Internet of Things Journal*, 2024.

[8] Z. Wang, J. Li, S. Yang, X. Luo, D. Li, and S. Mahmoodi, "A lightweight iot intrusion detection model based on improved bert-of-theseus," *Expert Systems with Applications*, vol. 238, p. 122045, 2024.

[9] M. Shahin, M. Maghanaki, A. Hosseinzadeh, and F. F. Chen, "Advancing network security in industrial iot: a deep dive into ai-enabled intrusion detection systems," *Advanced Engineering Informatics*, vol. 62, p. 102685, 2024.

[10] S. Zhang, Y. Xu, and X. Xie, "Universal adversarial perturbations against machine learning-based intrusion detection systems in industrial internet of things," *IEEE Internet of Things Journal*, 2024.

[11] J. Kotak and Y. Elovici, "Adversarial attacks against iot identification systems," *IEEE Internet of Things Journal*, vol. 10, no. 9, pp. 7868–7883, 2022.

[12] L. Zhu, K. Feng, Z. Pu, and W. Ma, "Adversarial diffusion attacks on graph-based traffic prediction models," *IEEE Internet of Things Journal*, vol. 11, no. 1, pp. 1481–1495, 2023.

[13] G. C. Moura, S. Castro, J. Heidemann, and W. Hardaker, "Tsuname: exploiting misconfiguration and vulnerability to ddos dns," in *Proceedings of the 21st ACM Internet Measurement Conference*, 2021, pp. 398–418.

[14] K. Zhu, L. Huang, J. Nie, Y. Zhang, Z. Xiong, H.-N. Dai, and J. Jin, "Privacy-aware double auction with time-dependent valuation for blockchain-based dynamic spectrum sharing in iot systems," *IEEE Internet of Things Journal*, vol. 10, no. 8, pp. 6756–6768, 2022.

[15] P. Borkar, C. Chen, M. Rostami, N. Singh, R. Kande, A.-R. Sadeghi, C. Rebeiro, and J. Rajendran, "{WhisperFuzz}:{White-Box} fuzzing for detecting and locating timing vulnerabilities in processors," in *33rd USENIX Security Symposium (USENIX Security 24)*, 2024, pp. 5377–5394.

[16] O. Suciu, C. Nelson, Z. Lyu, T. Bao, and T. Dumitraș, "Expected exploitability: Predicting the development of functional vulnerability exploits," in *31st USENIX Security Symposium (USENIX Security 22)*, 2022, pp. 377–394.

[17] K. He, D. D. Kim, and M. R. Asghar, "Adversarial machine learning for network intrusion detection systems: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 1, pp. 538–566, 2023.

[18] S. Alkadi, S. Al-Ahmadi, and M. M. B. Ismail, "Better safe than never: A survey on adversarial machine learning applications towards iot environment," *Applied Sciences*, vol. 13, no. 10, p. 6001, 2023.

[19] N. Chaabouni, M. Mosbah, A. Zemmari, C. Sauvignac, and P. Faruki, "Network intrusion detection for iot security based on learning techniques," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2671–2701, 2019.

[20] A. Wani, R. S, and R. Khaliq, "Sdn-based intrusion detection system for iot using deep learning classifier (idsiot-sdl)," *CAAI Transactions on Intelligence Technology*, vol. 6, no. 3, pp. 281–290, 2021.

[21] O. A. Alghanam, W. Almobaideen, M. Saadeh, and O. Adwan, "An improved pio feature selection algorithm for iot network intrusion detection system based on ensemble learning," *Expert Systems with Applications*, vol. 213, p. 118745, 2023.

[22] M. Eskandari, Z. H. Janjua, M. Vecchio, and F. Antonelli, "Passban ids: An intelligent anomaly-based intrusion detection system for iot edge devices," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 6882–6897, 2020.

[23] E. Anthi, L. Williams, M. Słowińska, G. Theodorakopoulos, and P. Burnap, "A supervised intrusion detection system for smart home iot devices," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 9042–9053, 2019.

[24] H. Nandanwar and R. Katarya, "Deep learning enabled intrusion detection system for industrial iot environment," *Expert Systems with Applications*, vol. 249, p. 123808, 2024.

[25] X. Zhou, W. Liang, W. Li, K. Yan, S. Shimizu, and K. I.-K. Wang, "Hierarchical adversarial attacks against graph-neural-network-based iot network intrusion detection system," *IEEE Internet of Things Journal*, vol. 9, no. 12, pp. 9310–9319, 2021.

[26] T. Yi, X. Chen, Y. Zhu, W. Ge, and Z. Han, "Review on the application of deep learning in network attack detection," *Journal of Network and Computer Applications*, vol. 212, p. 103580, 2023.

[27] Z. Lin, Y. Shi, and Z. Xue, "Idsgan: Generative adversarial networks for attack generation against intrusion detection," in *Pacific-asia conference on knowledge discovery and data mining*. Springer, 2022, pp. 79–91.

[28] P. T. Duy, N. H. Khoa, A. G.-T. Nguyen, V.-H. Pham *et al.*, "Digfupas: Deceive ids with gan and function-preserving on adversarial samples in sdn-enabled networks," *Computers & Security*, vol. 109, p. 102367, 2021.

[29] R. Zhang, S. Luo, L. Pan, J. Hao, and J. Zhang, "Generating adversarial examples via enhancing latent spatial features of benign traffic and preserving malicious functions," *Neurocomputing*, vol. 490, pp. 413–430, 2022.

[30] J. Chen, D. Wu, Y. Zhao, N. Sharma, M. Blumenstein, and S. Yu, "Fooling intrusion detection systems using adversarially autoencoder," *Digital Communications and Networks*, vol. 7, no. 3, pp. 453–460, 2021.

[31] P. Sun, S. Li, J. Xie, H. Xu, Z. Cheng, and R. Yang, "Gpmt: Generating practical malicious traffic based on adversarial attacks with little prior knowledge," *Computers & Security*, vol. 130, p. 103257, 2023.

[32] K. Roshan, A. Zafar, and S. B. U. Haque, "Untargeted white-box adversarial attack with heuristic defence methods in real-time deep learning based network intrusion detection system," *Computer Communications*, vol. 218, pp. 97–113, 2024.

[33] G. Apruzzese, M. Andreolini, M. Marchetti, A. Venturi, and M. Colajanni, "Deep reinforcement adversarial learning against botnet evasion attacks," *IEEE Transactions on Network and Service Management*, vol. 17, no. 4, pp. 1975–1987, 2020.

[34] A. M. Sadeghzadeh, S. Shiravi, and R. Jalili, "Adversarial network traffic: Towards evaluating the robustness of deep-learning-based network traffic classification," *IEEE Transactions on Network and Service Management*, vol. 18, no. 2, pp. 1962–1976, 2021.

[35] D. Han, Z. Wang, Y. Zhong, W. Chen, J. Yang, S. Lu, X. Shi, and X. Yin, "Evaluating and improving adversarial robustness of machine learning-based network intrusion detectors," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 8, pp. 2632–2647, 2021.

[36] Y. Sharon, D. Berend, Y. Liu, A. Shabtai, and Y. Elovici, "Tantra: Timing-based adversarial network traffic reshaping attack," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 3225–3237, 2022.

[37] 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.

[38] I. Debicha, B. Cochez, T. Kenaza, T. Debatty, J.-M. Dricot, and W. Mees, "Adv-bot: Realistic adversarial botnet attacks against network intrusion detection systems," *Computers & Security*, vol. 129, p. 103176, 2023.

[39] Q. Ma, Z. Liu, Z. Zheng, Z. Huang, S. Zhu, Z. Yu, and J. T. Kwok, "A survey on time-series pre-trained models," *IEEE Transactions on Knowledge and Data Engineering*, 2024.

[40] T. Wu, X. Wang, S. Qiao, X. Xian, Y. Liu, and L. Zhang, "Small perturbations are enough: Adversarial attacks on time series prediction," *Information Sciences*, vol. 587, pp. 794–812, 2022.

[41] W. Yang, J. Yuan, X. Wang, and P. Zhao, "Tsadv: Black-box adversarial attack on time series with local perturbations," *Engineering Applications of Artificial Intelligence*, vol. 114, p. 105218, 2022.

[42] L. Zhao, H. Cai, G. Fan, and Y. Hu, "A query-less adversarial attack method against time series forecasting," in *2024 5th International Conference on Big Data & Artificial Intelligence & Software Engineering (ICBASE)*. IEEE, 2024, pp. 599–602.

[43] L. H.-Y. W. Z.-H. Z. P.-X. ZHANG Yao-Yuan, YUAN Ji-Dong, "Adversarial attack of time series forecasting based on local perturbations," *Journal of Software*, vol. 35, no. 11, p. 5210, 11 2024.

[44] Z. Shen and Y. Li, "Temporal characteristics-based adversarial attacks on time series forecasting," *Expert Systems with Applications*, vol. 264, p. 125950, 2025.

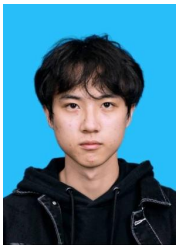
[45] M. M. Rashid, J. Kamruzzaman, M. M. Hassan, T. Imam, S. Wibowo, S. Gordon, and G. Fortino, "Adversarial training for deep learning-based cyberattack detection in iot-based smart city applications," *Computers & Security*, vol. 120, p. 102783, 2022.

[46] C. Li, H. Wang, J. Zhang, W. Yao, and T. Jiang, "An approximated gradient sign method using differential evolution for black-box adversarial attack," *IEEE Transactions on Evolutionary Computation*, vol. 26, no. 5, pp. 976–990, 2022.

Xiaohui Li (Member, IEEE) received the B.S. and Ph.D. degrees in computer science from the College of Computer Science, Sichuan University, Chengdu, China, in 2012 and 2017, respectively. She is currently an Associate Professor with the School of Cyber Science and Engineering, Sichuan University. Her research interests cover several areas with emphasis on spatial information networks, network and information security.

Yuanyuan Li is currently pursuing her B.S. degree at the School of Cyber Science and Engineering, Sichuan University, Chengdu, China. Her research interests span a wide range of fields, including network security, IDS and cyber threat detection.

Zhentian Zhong is currently pursuing his B.S. degree at the Pittsburgh Institute, Sichuan University, Chengdu, China. His research focuses on AI-driven security solutions and network security protocols, with a particular emphasis on time series analysis for anomaly detection and the development of intelligent systems to enhance cyber threat detection.



Lingfeng Tan is currently pursuing his M.S. degree in Computer Science at Sichuan University, Chengdu, China. His research focuses on AI-driven security solutions, network security protocols, and program analysis techniques, with an emphasis on developing intelligent systems to enhance cyber threat detection and software vulnerability assessment.

Junfeng Wang received the M.S. degree in computer application technology from Chongqing University of Posts and Telecommunications, Chongqing, in 2001, and the Ph.D. degree in computer science from the University of Electronic Science and Technology of China, Chengdu, in 2004. From 2004 to 2006, he was a Post-Doctoral Researcher with the Institute of Software, Chinese Academy of Sciences. He is currently a Professor with the College of Computer Science, Sichuan University. His recent research interests include spatial information networks, network and information security.

Jiayong Liu received his doctoral degree in Applied Mathematics from Sichuan University in 2008. He is currently a Professor at Sichuan University. The research field is Cyberspace Security. His research interests include Network Information Processing & Threat Intelligence Analysis, Data Mining, Covert Communication Construction & Analysis, Automated Analysis and Detection of Virtual Communities and Social Robots.

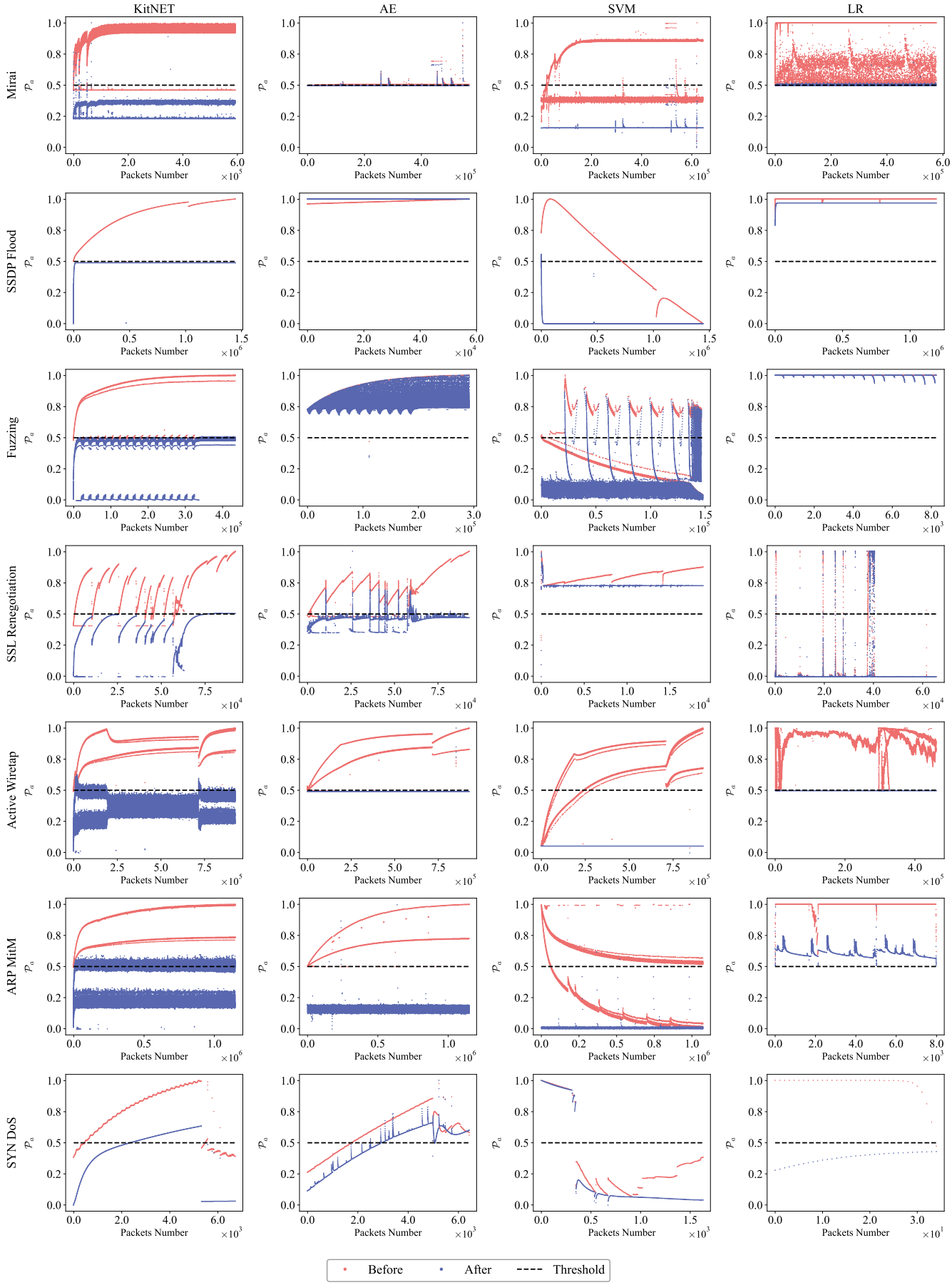


Fig. 4. Probability of Anomaly ($\mathcal{P}_a$) of ATCP in Various Detection Tasks.