Malicious Attack Defense in Human-to-Machine Applications Through Concept Drift Adaptation

Citation for the original published paper (version of record):
Yu, X., Mondal, S., Natalino Da Silva, C. et al (2025). Malicious Attack Defense in Human-to-Machine Applications Through Concept Drift Adaptation. IEEE Internet of Things Journal, 12(24): 53563-53573. http://dx.doi.org/10.1109/JIOT.2025.3616787

Malicious Attack Defense in Human-to-Machine Applications Through Concept Drift Adaptation

Xiangyu Yu, *Student Member, IEEE,* Sourav Mondal, *Member, IEEE,* Carlos Natalino, *Senior Member, IEEE,* Paolo Monti, *Senior Member, IEEE,* Lena Wosinska, *Senior Member, IEEE,* Yuxiao Wang, *Student Member, IEEE,* and Elaine Wong, *Senior Member, IEEE*

***Abstract*—The operational security of latency-sensitive networked applications is increasingly threatened by evolving malicious attacks that compromise operational integrity and network performance. Human-to-machine (H2M) applications, which rely on seamless bidirectional control signals and haptic feedback transmission, exemplify such latency-sensitive use cases. Existing learning-based malicious attack detection frameworks suffer from their reliance on pre-trained datasets, making machine learning models within them ineffective against previously unseen attack patterns. As attack profiles dynamically evolve, static models become obsolete, necessitating adaptive mechanisms to maintain detection accuracy. In this context, concept drift adaptation will serve as a critical tool for enabling models to continuously adjust to changing traffic distributions and emerging attack patterns. However, real-world H2M applications lack access to accurately labeled malicious traffic data, making real-time adaptation of defense mechanisms infeasible. To address these challenges, we propose a Concept Drift Adaptation-facilitated malicious attack Defense framework (CDAD). Firstly, CDAD employs Adaptive Random Forest as an incremental learning approach, integrating an error-rate-based concept drift detection mechanism to dynamically identify evolving attack patterns and trigger adaptive model updates. Secondly, a haptic behavior classifier is introduced to classify expected human operator interactions and compare them with real-time haptic feedback from remote machines. This enables automated traffic relabeling, allowing CDAD to adapt to previously unseen attacks without relying on pre-labeled datasets. The superior performance of CDAD over existing state-of-the-art methods is demonstrated across various malicious attack scenarios through extensive simulations. Results show that with CDAD, the attack success rate can be limited to 3%, while maintaining an inference time below $1ms$, thereby ensuring effective and efficient malicious attack defense in latency-sensitive H2M applications.**

***Index Terms*—Network security, human-to-machine, malicious attack, concept drift adaptation.**

I. Introduction

THE evolution of communication networks has shifted from traditional data-centric transmission to highly interactive and immersive human-to-machine (H2M) applications as part of the paradigm shift towards Tactile Internet [1]–[3]. It is envisioned that these applications bridge physical and virtual environments, allowing human operators to control remote machines with seamless feedback. Unlike conventional IoT systems that rely solely on autonomous machine-to-machine communication, H2M applications introduce dynamic human intervention, facilitating domains such as industrial automation, telesurgery, and immersive virtual reality interactions [4]. To ensure an immersive and real-time responsive user experience, H2M applications impose stringent latency constraints, typically from $1ms$ to $10ms$, and ultra-high reliability of 99.9999% [5]. Converged fiber and wireless access networks play a crucial role in supporting ultra-low-latency H2M applications. These networks, illustrated in Fig. 1, effectively accommodate diverse H2M interaction scenarios based on the relative connectivity and geographical distribution of human and machine entities by facilitating both optical and wireless segments [5], [6]. To overcome latency barriers caused by long distances and shared optical network unit (ONU) bandwidth, machine learning (ML)-driven predictive dynamic bandwidth allocation schemes [2], [7] have been proposed recently. These schemes assign bandwidth based on predicted H2M application traffic of control signals from human operators and haptic feedback from remote machines, thereby reducing queuing delays and ensuring efficient uplink transmission. Beyond network layer optimizations, ML-enhanced H2M servers are deployed near ONUs to further reduce transmission delays. A haptic classification mechanism is employed in those servers to preemptively predict and transmit haptic feedback based on the received control signals from human operators, effectively reducing the round-trip delay in H2M applications [5].

The deployment of security and protection measures to support traditional traffic [8], [9] in the access segment has been slow due to high cost-sensitivity. However, with the increased interest in deploying ML techniques to enhance efficiency and responsiveness of H2M applications, the threat of malicious traffic severely introduces vulnerabilities that will compromise the *operational security* of human-machine interactions within the network. For example, after getting access to an ONU, a malicious attacker can impersonate a legitimate H2M device by mimicking its IP or MAC address via spoofing [10]. Consequently, the adversary can inject manipulated control signals or replace legitimate control signals with malicious packets, deceiving the remote machine into executing an unintended action that could result in operational failure or safety hazards. Such adversarial interference could result in catastrophic failures, particularly in applications such as telesurgery or remote bomb disposal, where precise task execution is

(Corresponding author: Xiangyu Yu.)

Xiangyu Yu, Sourav Mondal, Yuxiao Wang, and Elaine Wong are with Department of Electrical and Electronic Engineering, University of Melbourne, Australia (e-mail: xiangyuy4@student.unimelb.edu.au; sourav.mondal@unimelb.edu.au; yuxiao.wang2@student.unimelb.edu.au; ewon@unimelb.edu.au).

Carlos Natalino, Paolo Monti, and Lena Wosinska are with Electrical Engineering Department, Chalmers University of Technology, Sweden (e-mail: carlos.natalino@chalmers.se; mpaolo@chalmers.se; wosinska@chalmers.se).

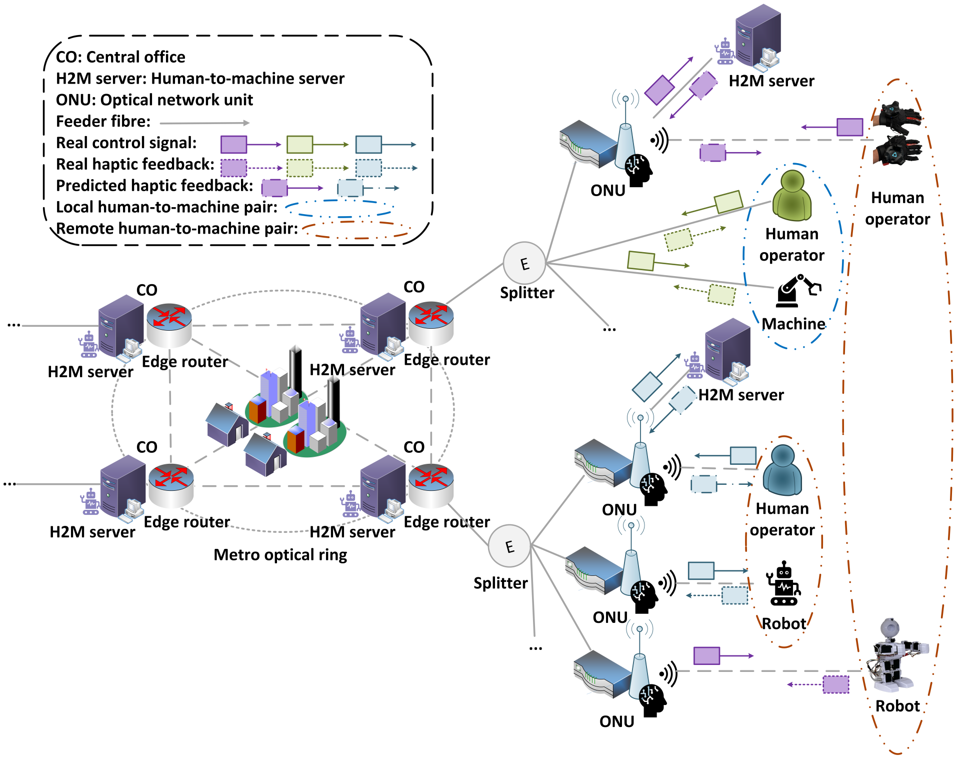


Fig. 1: ML-enhanced converged networks supporting H2M applications

crucial. Meanwhile, malicious traffic can significantly degrade network performance, as the continued injection of malicious packets increases the bandwidth utilization and the uplink latency, potentially leading to violations of the strict latency constraints typical of seamless H2M operations [11]. In this context, it is essential to guarantee the *operational security* and stability of H2M applications by adopting robust malicious traffic detection and mitigation strategies. In our previous work [11], we proposed a malicious traffic detection and mitigation framework for H2M applications against malicious attacks that employed a pre-trained XGBoost-based traffic classifier to identify and filter known malicious traffic patterns. While this framework effectively defended against attacks seen during training, it heavily relied on static datasets and lacked adaptability to unforeseen threats, which is remained fixed after deployment. This poses a vulnerability to concept drift [12], where evolving attack patterns lead to changes in the statistical properties of network traffic not observed during the training phase. As a result, it fails to detect and mitigate emerged attack variants, making it vulnerable to adversaries who continuously evolve their attack patterns. Furthermore, existing malicious attack defense frameworks operate under the assumption that accurate ground-truth labels for network traffic are readily available, allowing models to be retrained on newly observed attack instances. However, in real-world scenarios, labeling malicious traffic is inherently challenging and often infeasible, as determining whether an incoming packet is truly malicious requires deep packet inspection, forensic analysis, or domain expertise, which are time-consuming, resource-intensive, and impractical processes in real-time network environments. The lack of timely, accurately labeled attack data significantly impairs the ability of existing concept drift adaptation models to incrementally update effectively, leaving them unable to respond dynamically to evolving threats.

In light of the above, this paper builds on the foundation of [11], where a Concept Drift Adaptation-facilitated malicious attack Defense framework (CDAD) is proposed to defend H2M applications against evolving malicious attacks and to remove the malicious traffic at the ONU. In contrast to [11], which relies on a static pre-trained traffic classifier and does not account for dynamic changes from previously unseen malicious attack patterns during training, CDAD incorporates a concept drift adaptation model, Adaptive Random Forest (ARF) [13], at the ONU. To ensure reliable and accurate labels for model adaptation, a haptic behavior classifier is integrated to enable an automated relabeling mechanism. This enables ARF to adapt quickly without manual labeling or offline supervision. The key contributions of this paper are outlined below:

1) Firstly, we propose CDAD, which integrates ARF with a haptic behavior classifier. ARF is adopted for its incremental online learning strategy, featuring an error-rate-based concept drift detection method. This enables continuous detection of shifts in malicious attack patterns and supports dynamic model updates upon detecting a concept drift. As a result, ARF can adapt to evolving and previously unseen attack patterns without requiring full model retraining. Simultaneously, the haptic behavior classifier automatically relabels every control signal packet by comparing the predicted haptic behavior with the actual haptic feedback received from remote machines. This behavioral consistency check

provides relabeled packets, which are then fed back into ARF for incremental self-update, thereby maintaining robust defense performance against malicious attacks in dynamic H2M application scenarios.

2) Secondly, to evaluate the performance of CDAD against evolving malicious attacks in H2M applications, we conduct a packet-level simulation based on a 10 km 10G-passive optical network (PON) with 16 ONUs.
3) Finally, we compare CDAD with existing traffic classification-based defense mechanisms through comprehensive simulation and performance evaluation in terms of attack success rate, bandwidth utilization, inference time, and uplink latency. Results demonstrate that CDAD achieves superior attack detection and mitigation performance, keeping the attack success rate below 3% with an inference time of less than $1ms$, thus ensuring responsiveness and operational security in H2M applications.

The rest of the paper is organized as follows. Section II presents a comprehensive review of the work related to malicious attack detection and mitigation, Section III introduces the architecture and methodology of CDAD, Section IV presents the performance evaluation results, and Section V summarizes the paper.

II. Related Work

A. Malicious Attacks and Defense Strategies

Malicious attacks correspond to any adversarial action that can compromise system integrity, confidentiality, or availability by manipulating network traffic, injecting malicious data, or exploiting vulnerabilities in security mechanisms [14]. Two primary malicious attack patterns that significantly threaten the operational security of H2M applications are: packet injection as shown in Fig. 2 and payload manipulation as shown in Fig. 3. Packet injection attacks occur when adversaries inject malicious packets into a legitimate data stream, exploiting vulnerabilities in the packet-handling mechanisms to disrupt communication, consume network resources, or evade detection-based security measures. Packet injection attacks in software-defined networks have been studied in [15], where adversaries exploit the reactive mode of OpenFlow switches to manipulate topology views and overwhelm controllers. Similar threats emerge in H2M applications, where attackers can spoof ONUs to inject malicious packets alongside legitimate H2M application traffic, leading to severe network performance degradation. This malicious traffic consumes critical network resources, resulting in bandwidth over-utilization and inefficient allocation of available resources. More critically, excessive packet loads introduce transmission delays, increasing uplink latency and potentially violating the latency constraints of H2M applications [11]. Payload manipulation involves the modification of legitimate packets to alter their intended function while maintaining their structural validity, thus exploiting legitimate traffic streams to evade detection. Furthermore, attackers who gain control over routers can intercept traffic streams and execute malicious modifications, such as dropping, altering, or rerouting packets, to compromise data integrity and network functionality [16]. In the context of H2M applications, payload manipulation poses a significant threat, as minor perturbations to control signal payloads can result in degraded operational accuracy, unexpected machine behaviors, or compromised task execution [17].

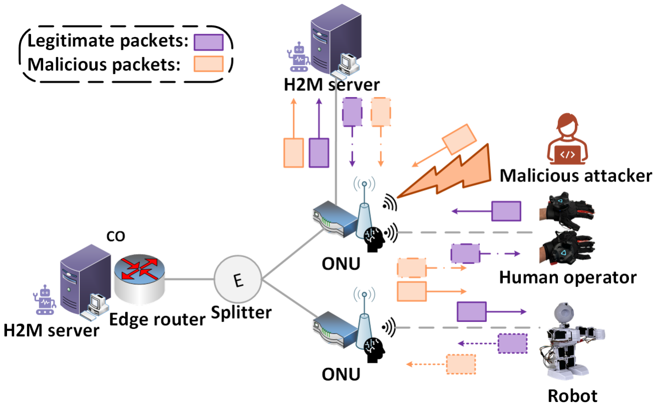


Fig. 2: Packet injection in H2M applications

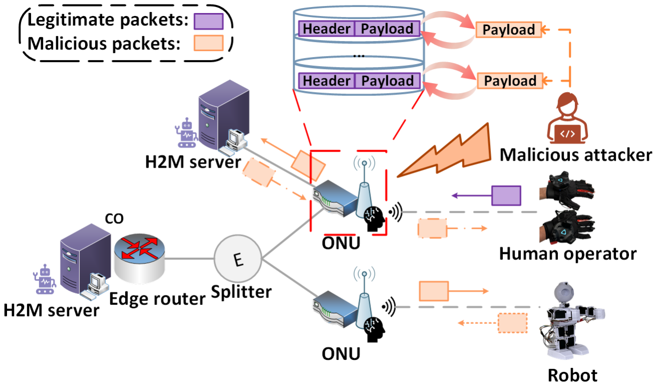


Fig. 3: Payload manipulation in H2M applications

With the increasing complexity of network systems and the growing reliance on artificial intelligence-driven security mechanisms, malicious attack defense has become a critical research area. Specifically, ML-enhanced network intrusion detection systems (NIDS) have demonstrated improved adaptability by learning traffic patterns and distinguishing between normal and malicious activities [23]. The authors of [18] and [22] report that deep learning models such as Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) have further advanced malicious traffic classification and anomaly detection by capturing both spatial and temporal dependencies in network behavior. However, despite these advantages, learning-based security mechanisms introduce new vulnerabilities that malicious attackers can exploit through adversarial attacks, where carefully crafted perturbations can cause malicious packets to be misclassified as legitimate traffic [24]. Examples of adversarial attacks include the Fast Gradient Sign Method (FGSM) [25], which generates adversarial examples by computing the gradient of the loss function and modifying input data in the direction of the gradient, effectively deceiving deep learning-based NIDS in IoT systems [26]. The

TABLE I: Comparison among different malicious attack defense frameworks

Paper	Attack type	Defense methodology	Suitability for H2M application
[18]	Denial of service (DoS) attacks, remote to local (R2L) attacks, user to root (U2R) attacks, and probing attacks	Proposes a multi-CNN fusion approach to improve binary and multiclass intrusion detection accuracy	Relies on the known attack patterns, leaving robustness under unknown malicious attacks uncertain; no consideration of evolving or adversarial attack patterns
[19]	FGSM, BIM	Uses adversarial training with both legitimate and adversarially manipulated samples to enhance model robustness against adversarial inputs	Relies on known attack patterns; requires extensive adversarial sample generation and labeling, limiting real-time applicability; no consideration of evolving attack patterns
[20]	Adversarial examples produced by the proposed GANs-based framework	Proposes the GANs-based framework to generate synthetic adversarial samples for adversarial training to enhance the robustness of ML/DL-based IDSs against adversarial ML attack	Computationally intensive when generating GANs-based adversarial samples; defense effectiveness is constrained by the representativeness of GAN-generated samples; no consideration of evolving attack patterns
[21]	DoS, R2L, U2R and probing attacks	Proposes a hybrid network anomaly detection framework that incorporates concept drift detection to capture traffic distribution shifts under evolving attack scenarios	Computationally intensive due to repeated distribution checks; Relies on pre-labeled attack data for model adaptation; no consideration of adversarial attack patterns
[22]	DoS attackmalicious scan, malicious control, malicious operation, spying, data probing, wrong setting	Proposes an improved LSTM with concept drift adaptive method to enhance anomaly detection and multi-class classification under streaming IoT data	Relies on pre-labeled attack data for model adaptation; no consideration of adversarial attack patterns
Our paper	FGSM, BIM, ZOO, malicious user-generated attack data in H2M applications	Proposes CDAD framework, which integrates ARF to rapidly detect and adapt to evolving malicious traffic in H2M applications. A haptic behavior classifier is incorporated into ARF to enable an automated relabeling mechanism without requiring manual labeling	Eliminates dependence on pre-labeled data via automated relabeling; enables online defense adaptation against evolving malicious attack patterns targeting real-time H2M applications

Basic Iterative Method (BIM) [27] extends FGSM by running the gradient update in multiple iterations to maximize the attack success rate, while the zeroth-order optimization attack (ZOO) [28], also known as a grey-box attack, utilizes zeroth-order stochastic coordinate descent to iteratively add perturbations and estimate the gradients of classifiers. To improve the resilience of learning-based classifiers against adversarial attacks, adversarial training [19] has emerged as a promising defense mechanism, where the classifier is trained on both legitimate and adversarially manipulated samples, allowing the model to learn how to recognize and mitigate adversarial attack samples. However, adversarial training is limited by its reliance on known attack patterns, making it ineffective against unknown malicious attack threats. Furthermore, generating and labeling sufficient adversarial samples requires significant effort, making adversarial training resource-intensive and less practical in real-time network environments. To address this challenge, generative adversarial networks (GANs)-based defenses [20] are proposed to supplement adversarial training. GANs-based defenses enhance the robustness of NIDS and the classifier by simulating adversarial attack scenarios and generating synthetic adversarial samples for training. However, GANs-based defenses introduce additional computational overhead, as generating realistic adversarial samples requires significant training time and computing power. Moreover, the effectiveness of GANs-generated adversarial samples depends on the quality and diversity of the generated attack traffic, which may not always accurately reflect real-world adversarial strategies.

Despite advancements in ML-based security mechanisms, existing frameworks remain inadequate against evolving malicious attack patterns, as they rely on static training data and struggle to adapt to previously unseen threats, making them vulnerable to concept drift where the statistical properties of malicious traffic change over time, rendering pre-trained models ineffective. The limitations of adversarial training and GANs-based defense further emphasize the need for adaptive malicious attack defense mechanisms that can evolve alongside emerging threats. In H2M applications, where low latency and operational security are paramount, an effective defense framework must dynamically identify, mitigate, and adapt to new attack patterns in real time without imposing excessive computational overhead. In this context, concept drift detection and adaptation techniques [29], [30] will play a crucial role by enabling continuous model refinement, allowing the defense framework to remain proactive and resilient against emerging attack patterns in H2M applications.

B. Concept Drift Detection and Adaptation

Concept drift in malicious traffic detection refers to changes in the statistical properties of malicious traffic caused by shifts in network environments and malicious attack patterns [31]. This dynamic nature makes it challenging to maintain accurate malicious traffic detection and ensure system responsiveness. Two primary approaches are commonly used to detect concept drift: distribution-shift-based and error-rate-based methods [32]. Distribution-shift-based approaches detect concept drift by analyzing differences in data distribution across two distinct time windows [33]. For instance, in [21], the authors proposed a hybrid network anomaly detection framework that incorporates concept drift detection by applying a sliding window approach with KL divergence to capture distribution shifts under evolving traffic conditions. However, distribution-shift-based methods often entail high computational complexity, making them less suitable for real-time defense. In contrast, error-rate-based drift detection methods monitor the performance degradation of predictive models using predefined thresholds to detect concept drift. Since attackers often introduce minimal modifications to malicious traffic to evade detection, error-rate-based approaches are more suitable for identifying subtle yet impactful changes in malicious traffic patterns. Once concept drift is detected, learning models must adapt to evolving data distributions to maintain detection accuracy and enhance overall model performance. Ensemble-based online learning models have emerged as effective drift-adaptive techniques by integrating multiple base learners to improve classification robustness [34]. Leverage bagging (LB) [35] is a basic ensemble approach that enhances model diversity by constructing multiple base learners, such as Hoeffding Trees [32], through bootstrap sampling, with final predictions determined via majority voting. Moreover, ARF [13] and Streaming Random Patches (SRP) [36] are two state-of-the-art ensemble learning models that train multiple Hoeffding Trees as base classifiers. Both methods include drift detection mechanisms for each Hoeffding Tree to identify and address changes in data distributions. However, their structural design differs: ARF employs local subspace randomization, where each Hoeffding Tree is trained on a randomly selected subset of features, balancing computational efficiency and adaptability to different types of drift. In contrast, SRP adopts a global subspace randomization approach, where different instances of the ensemble are trained on distinct feature subsets. Furthermore, SRP integrates online bagging techniques, enhancing its ability to handle non-stationary distributions. Although ensemble-based learning models provide an effective mechanism for handling concept drift in malicious attack defense, they inherently rely on access to labeled data for incremental model updates and drift adaptation. In real-world network environments, obtaining timely, accurately labeled malicious traffic is impractical. The dynamic and evolving nature of network traffic and the emergence of new attack patterns render the manual labeling of incoming data infeasible. Notably, for such attack defense mechanisms to function effectively, they would require frequent, high-speed labeling by domain experts, which is an unrealistic expectation in an online setting. This problem is common to other works in the intrusion detection and concept drift literature [22], [33], [37]. To address this critical limitation, this paper introduces a traffic relabeling mechanism into CDAD, leveraging a haptic behavior classifier to validate and relabel incoming traffic in the absence of ground-truth labels. This approach enables CDAD to maintain accurate model adaptation in evolving attack environments without requiring manual labeling or offline supervision. In Table I, we provide a consolidated comparison among some of the aforementioned works, positioning the contributions of our proposed CDAD framework relative to these existing defense frameworks.

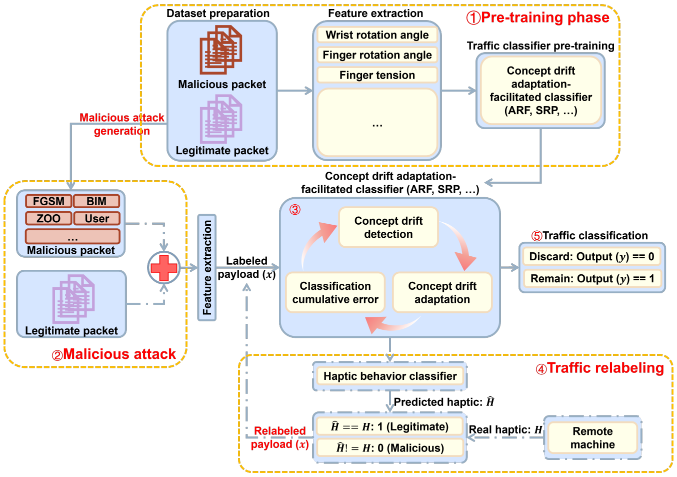


Fig. 4: Proposed CDAD framework

III. Malicious Attack Defense in H2M Applications

In H2M applications within network environments, instead of encrypting control signals, which is commonly used for haptic feedback prediction [5], to preserve the operational integrity, a learning-based attack defense mechanism is employed to detect malicious packets via control signal traffic payload analysis. Each packet contains a feature vector $x \in \mathbb{R}^d$, representing the control signal payload, where d is the dimensionality of the control signal feature space. A traffic classifier $f : \mathbb{R}^d \to \mathcal{Y}$, with $\mathcal{Y} = \{0, 1\}$, is employed to assign a label $y = f(x)$, indicating whether a packet is legitimate (labeled 1) or malicious (labeled 0). Fig. 4 illustrates the proposed CDAD against malicious attack in H2M applications. Specifically, the feature extraction module employed in CDAD inspects every incoming packet at the application layer of the ONU to extract crucial payload features for subsequent traffic classification. The concept drift adaptation-facilitated classifier (e.g., ARF and SRP) deployed at the ONU differentiates between legitimate and evolving malicious packets based on extracted payload features, enabling early malicious attack detection and mitigation before malicious traffic is forwarded upstream to the CO. To effectively classify diverse malicious packets generated by a malicious attacker, CDAD harnesses a dynamic traffic relabeling mechanism, which is uniquely designed for the concept drift adaptation-facilitated classifier. To enhance adaptability, CDAD is also equipped with a haptic behavior classifier to relabel every packet deployed at the H2M server near the ONU. Consequently, the traffic relabeling guarantees real-time correction at the ONU and enables the traffic classifier to continuously refine its understanding of evolving malicious attack patterns, ensuring classifiers remain robust and effective against emerging malicious attack threats in H2M applications.

A. *Malicious Attack Generation*

The goal of a malicious attacker is to mislead the deployed traffic classifier model, making it fail to distinguish between legitimate H2M application packets and malicious packets. To achieve this, an attacker can slightly modify the payload of the malicious packet with a feature vector x by introducing a crafted perturbation $\delta \in \mathbb{R}^d$ such that $\widetilde{x} = x + \delta$. Through these subtle adjustments, the attacker can bypass the traffic classification, enabling malicious packets to appear legitimate while still compromising the normal operation of the H2M application. Thus, the malicious attack process can be formulated as an optimization problem where the attacker aims to minimize the perturbation magnitude $||\delta||$ while ensuring that malicious packets $\widetilde{x}$ are misclassified (remain undetected) by the traffic classifier as shown in (1) [26].

$$\min \|\delta\|, \quad \text{s.t. } f(\tilde{x}) \neq f(x). \tag{1}$$

The assumptions regarding the capabilities and constraints of the adversary are defined following the black-box threat model detailed in [26] and [38], which align with realistic and practical H2M operational security challenges. We assume that the attacker only knows the defense mechanism, that is traffic classification, but lacks knowledge about the detailed configuration of the employed traffic classifier, such as the specific hyperparameters. This assumption is consistent with real-world scenarios where specific details of the defense system are public, but the internal configurations are kept confidential [28]. Importantly, the attacker must possess significant domain knowledge about H2M applications, such as network protocols, H2M packet inter-arrival time distributions, H2M traffic payload structures, *etc*., to carefully craft effective malicious packets, as a key constraint on generating malicious attack data in H2M applications is to preserve the functional behavior of generated malicious attack packets. The attacker must ensure that malicious packets retain the critical H2M application payload features required for the operation of the application. Any disruption to these features would prevent the application from correctly processing the packet, making the attack ineffective. Additionally, we assume the integrity of the attack defense framework is preserved, meaning the attacker cannot modify the deployed traffic classifier or tamper with its detection results. The target traffic classifier is also assumed to be well-trained on clean H2M traffic flows without poisoned samples.The malicious attacker is assumed to be able to passively monitor network traffic flows and actively replace legitimate packets to compromise the legitimate H2M application. This ability allows the attacker to craft and inject malicious packets while adhering to the constraints necessary to maintain the operation of the H2M application.

Two complementary approaches that align with the assumptions and the described threat model can be utilized to generate malicious attack packets in H2M applications. The first approach employs adversarial attack generation algorithms, such as FGSM, BIM, and ZOO. In this method, the attacker constructs a shadow model that mimics the behavior of the target traffic classifier using a subset of the legitimate H2M application packets dataset. By training the shadow model on this dataset, the attacker ensures that it behaves similarly to the target traffic classifier. Subsequently, adversarial algorithms are applied to the shadow model to introduce minimal perturbations (δ) to the legitimate payloads. These perturbations are carefully crafted to bypass the traffic classification mechanism while preserving the functional integrity of the H2M packets. The second approach involves generating malicious packets by directly operating the legitimate H2M application but performing alternative actions to collect corresponding control signals. These actions produce control signals structurally similar to legitimate traffic but serve as malicious data. For example, the malicious attacker could perform different yet legitimate-looking operations with different wrist rotation angles or finger tension signals to generate a dataset representing malicious behavior. This method adheres to the constraint of maintaining the functional features of the H2M application, ensuring that malicious packets do not disrupt its operational requirements while compromising its security. Both malicious attack generation methods preserve the functional behavior of H2M applications and operate within the constraints of a black-box threat model, ensuring the validity of the generated malicious packets in real-world H2M application scenarios.

B. Malicious Attack Detection and Mitigation

To guarantee robust defense performance against malicious attacks in H2M applications, pre-training the deployed traffic classifier is essential. This process begins with the preparation of pre-training data, which includes two components: legitimate H2M application packets and malicious packets from known attacks. Including known attack packets ensures that the pre-training phase effectively models scenarios encountered during deployment. Feature extraction is then conducted to distill distinct H2M application payload features from the traffic datasets, as detailed in [5], for training the traffic classifier to differentiate between benign and malicious traffic. Upon pre-training, the traffic classifier is deployed into the ONU that connects to the human operator, as shown in Fig. 1, to classify incoming traffic during real-time operation to ensure the secure and reliable operation of H2M applications. All incoming packets received at the ONU are subjected to a thorough application-layer packet inspection process, during which their payloads are extracted for analysis. Using the same feature extraction method employed during pre-training, the extracted payload is represented as a feature vector x consisting of control signal features, such as wrist rotation angles, finger rotation angles, *etc*. This feature vector is then fed into a binary traffic classifier (i.e., $f : \mathbb{R}^d \rightarrow \mathcal{Y}$) which determines whether an incoming packet is legitimate ($y = 1$) or malicious ($y = 0$). Following classification, the framework takes appropriate mitigation actions: legitimate H2M packets are queued for upstream transmission to the CO, maintaining seamless operation and adherence to strict latency requirements; malicious packets are discarded to protect the network from contamination and service disruptions.

To detect attack types not included in the training phase and ensure adaptability to evolving threats, CDAD is proposed to incorporate a concept drift adaptation-facilitated classifier, which integrates an error-rate-based concept drift detection method and a concept drift adaptation model, promptly detecting and responding to evolving attack patterns in real-time without requiring prior knowledge of the attack. The concept drift detection mechanism monitors the traffic classification performance degradation against user-defined confidence thresholds and triggers the concept drift adaptation process when significant deviations are detected [39]. Specifically, two user-defined confidence levels are used to quantify the model degradation: the warning and drift confidence levels. The classification error rate of the classifier is continuously monitored using the cumulative mean error rate and standard deviation computed at each time step t, as shown in (2) and (3), respectively.

$$\overline{e}_i = \frac{\sum_{t=0}^{t=i} e_t}{i}, \tag{2}$$

$$\overline{s}_i = \sqrt{\frac{\overline{e}_i(1-\overline{e}_i)}{i}}, \tag{3}$$

where $\overline{e}_i$ is the cumulative mean error rate, and $\overline{s}_i$ the standard deviation of the mean error rate. The drift status of the classifier is determined by whether the degree of the observed error rate change reaches a certain confidence level illustrated as follows:

$$\overline{e}_i + \overline{s}_i \geq \overline{e}_{min} + \lambda \cdot \overline{s}_{min}, \tag{4}$$

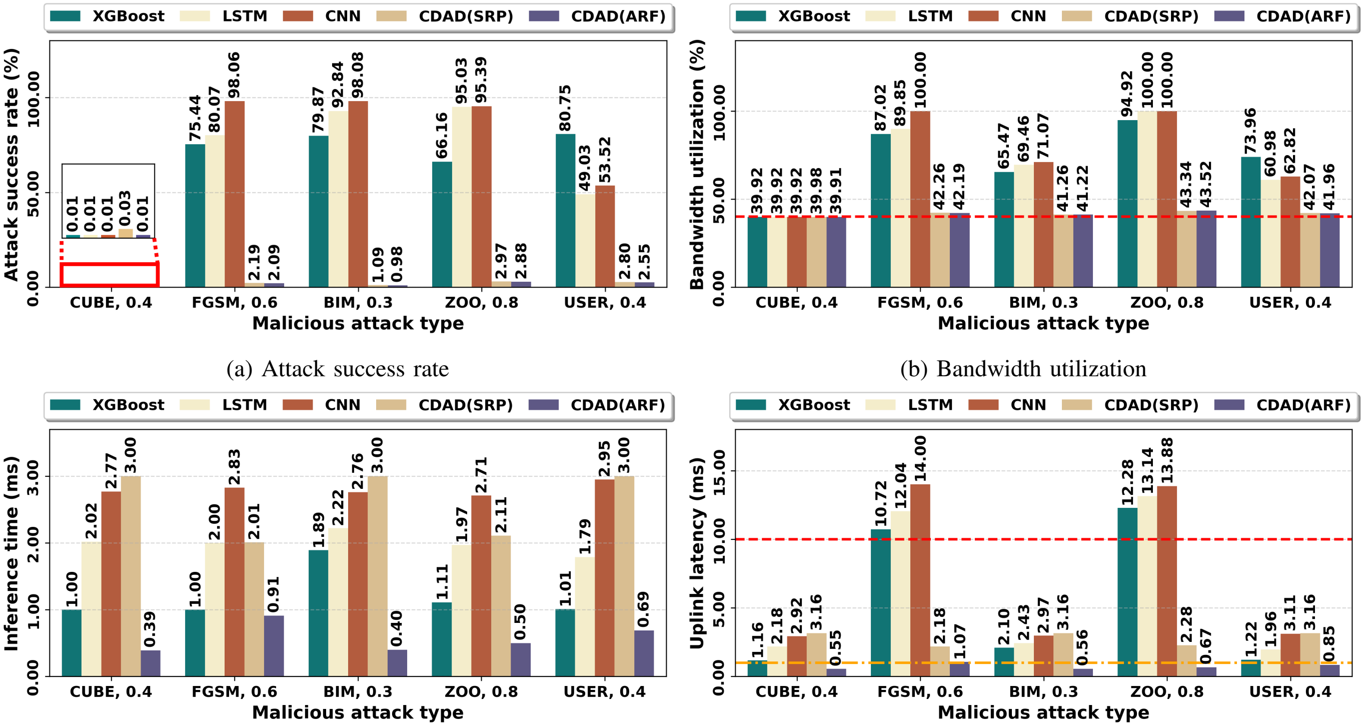
where $\overline{e}_{min}$ and $\overline{s}_{min}$ represent minimum recorded values of $\overline{e}_i$ and $\overline{s}_i$. Compared with the distribution-based concept drift detection method, the incorporated error-rate-based concept drift detection method enhances the capability of CDAD by statistically identifying significant concept drift in evolving malicious attack scenarios, triggering timely updates to the classification model without compromising latency constraints critical to real-time H2M operations. Meanwhile, to achieve efficient concept drift adaptation against evolving malicious attacks, ARF, an advanced ensemble-based online learning model, is employed in CDAD, overcoming the constraints of static, pre-trained traffic classifiers. Unlike the methods proposed in [21], which require complete model retraining when concept drift is detected, leading to high computational overhead and increased latency, ARF incrementally updates the background Hoeffding Trees. This incremental approach continuously adapts to evolving malicious traffic patterns while remaining computationally efficient. Notably, the warning confidence level serves as a preemptive alert for potential drifts in ARF, enabling early model adjustments and a better model adaptation efficiency before model replacement when a real drift is detected. Studies [13], [36], [40] have shown that ARF achieves superior classification accuracy and reduced inference time compared to alternative drift adaptation methods such as SRP and LB, making it a strong candidate for real-time H2M applications. Furthermore, the concept drift adaptation-facilitated classifier is designed to be model-agnostic, enabling integration with any concept drift detection method and concept drift adaptation model with similar or improved performance.

A fundamental strategy of CDAD involves continuous model refinement through incremental updates of the concept drift adaptation-facilitated classifier, enabling rapid adaptation to emerging malicious attack patterns. However, this incremental updating process critically relies on the availability of accurately labeled traffic data. This is particularly challenging when dealing with unknown malicious attacks that lack predefined labels in real-world scenarios. Without an automated and accurate traffic relabeling mechanism, the adaptation capabilities of the traffic classifier would be significantly limited, hindering its ability to detect and mitigate emerging threats in real time. To further enhance the adaptability of CDAD, a haptic behavior classifier is harnessed to facilitate traffic relabeling. Similarly to the traffic classifier, the haptic behavior classifier is trained using control signal features (see Section IV-A) from a human operating a legitimate H2M application before being deployed in the H2M application server near the ONU. These features capture the human operator's natural operational patterns, enabling the haptic behavior classifier to establish a baseline operational profile during legitimate H2M interactions. A traffic classifier functions as a binary classifier, distinguishing between legitimate and malicious packets based solely on packet payload. Conversely, the haptic behavior classifier evaluates whether the control signal payload in received packets exhibits coherence with legitimate human interaction profiles. Since remote machines execute operations based on the received traffic and generate real haptic feedback (denoted as H) accordingly, malicious attack packets can disrupt this process. If a malicious packet is successfully transmitted and received by the remote machine, the resulting haptic feedback will deviate significantly from the predicted haptic feedback (denoted as $\hat{H}$) determined by the haptic classifier. This mismatch indicates an inconsistency between the expected operational profile and the actual executed operation, signaling the presence of a malicious packet. When a discrepancy is detected, the traffic payload is relabeled as malicious, and both the modified payload and its new label are fed back into the drift adaptation model. This feedback loop allows CDAD to update the classification model, ensuring adaptation to evolving malicious attack patterns and enhancing long-term resilience. By integrating the haptic behavior classifier with the concept drift adaptation-facilitated classifier, CDAD achieves a more robust and adaptive defense approach against evolving malicious attacks, ultimately safeguarding the operational integrity of H2M applications.

IV. Performance Evaluation

A. Simulation Setup

To evaluate the performance of CDAD, we conducted packet-level simulations based on a 10 km 10G-PON comprising 16 ONUs on a Windows-based machine with an i5-13600KF CPU, Nvidia RTX4060 GPU, and 64 GB of memory. The evaluation utilizes one legitimate H2M traffic dataset and five malicious traffic datasets, namely Cube, FGSM, BIM, ZOO, and a malicious user dataset, to assess the performance of each traffic classifier. Specifically, the legitimate H2M traffic dataset represents traffic generated by a legitimate user operating an authorized H2M application. The Cube dataset reflects H2M application traffic generated when the same legitimate user operates a different H2M application, which is not authorized to operate at the given time. The FGSM, BIM, and ZOO datasets are malicious datasets derived from the legitimate H2M application traffic dataset using the respective FGSM, BIM, and ZOO adversarial attack generation methods. Finally, the malicious user dataset represents H2M application traffic generated by a malicious user operating the legitimate H2M application. All tested traffic classifiers are pre-trained using the legitimate traffic dataset (labeled as 1) and the cube



(a) Attack success rate

(b) Bandwidth utilization

(c) Inference time

(d) Uplink latency

Fig. 5: Network performance under different malicious attack and traffic loads

dataset (labeled as 0). All traffic data is injected into all ONUs following the Generalized Pareto distribution packet arrival time [5], [7]. The legitimate H2M traffic load is set to 0.4, while the malicious traffic load is randomly selected from 0.1 to 0.9 during the simulation for online testing. A predictive dynamic bandwidth allocation (DBA) scheme is adopted to assign bandwidth by predicting network traffic distribution based on observed traffic load to minimize queuing delays and maximize transmission efficiency within the network [2], [7], [30]. To support timely and accurate traffic relabeling, an XGBoost model is employed as the haptic behavior classifier. This classifier is pre-trained using control signal features collected from a legitimate H2M application running on a VR-based H2M platform [5]. These control signal features include Quaternion and Euler angles, which accurately track the orientations of thumb, wrist, and finger joints. Additionally, tension readings from two flex sensors per finger are included to comprehensively capture fine-grained variations in grip force and finger movement during interactions. Each instance of control signal data comprises 64 elements, providing a comprehensive representation of fine-grained human operator movements when interacting with the legitimate H2M application. The corresponding output from the haptic behavior classifier consists of five distinct haptic feedback categories, each reflecting a specific interaction pattern. These categories encapsulate key tactile responses associated with various operational behaviors, including single-finger touch, multi-finger grasp, *etc*. Moreover, to balance the sensitivity and robustness of the concept drift detection algorithm shown in (4), the coefficient λ is set to 2 for the warning level and 3 for the drift level.

TABLE II: Hyperparameters of the traffic classifiers.

Model	Components
XGBoost [11]	n_estimators(100), max_depth (50), learning_rate (0.05), subsample (0.8), colsample_bytree (0.8), eval_metric (logloss)
LSTM [22]	LSTM layer (128), Dropout (0.3), Dense layer (64), Output layer (1)
CNN [18]	Conv1D layer (64), MaxPooling1D (2), Dropout (0.3), Dense layer (64), Output layer (1)
CDAD (SRP)	n_models (5), max_depth (14), delta (0.07914), grace_period (95), leaf_prediction (nba), split_criterion (info_gain)
CDAD (ARF)	n_models (5), max_depth (25), max_features (0.2), grace_period (155), leaf_prediction (nba), split_criterion (info_gain)

Table II outlines the structure and hyperparameters of the proposed CDAD and three commonly used traffic classifiers from literature. Meanwhile, to assess the generality of CDAD in integrating different concept drift adaptation models, SRP is incorporated as an alternative adaptation model within the framework, serving as a benchmark to evaluate the effectiveness of ARF in malicious attack defense. Four performance metrics averaged across all 16 ONUs are considered to evaluate the malicious attack defense performance: attack success rate in %; bandwidth utilization in %; inference time per sample in ms; and uplink latency in ms. The attack success rate is calculated using:

$$Attack\ success\ rate = \frac{1}{N}\sum_{i=1}^{N}\frac{P_{COi}}{S_{ONUi}}, \tag{5}$$

where N, P_{COi}, S_{ONUi} represent the total numbers of the ONU, the total number of received malicious packets at the CO, and the total number of packets sent from the ONU, respectively. A lower attack success rate indicates a better attack defense performance as fewer malicious packets are received at the CO. Bandwidth utilization is calculated as the ratio of the total number of bytes from the packets received at the CO and the total available network bandwidth over a time period. In the absence of malicious attacks, the predictive DBA scheme allocates bandwidth in proportion to the traffic load of the legitimate H2M application, resulting in bandwidth utilization that accurately reflects real traffic demands without unnecessary overhead. Inference time, defined as the time the traffic classifier takes to process and classify a packet, is particularly critical in H2M applications. A lower inference time means a more efficient traffic classification process, essential for meeting the stringent latency requirements of 1 to $10ms$. As defined in (6), the uplink latency in H2M applications is composed of packet transmission delay (D_{trans}), packet queuing delay (D_{queue}), packet propagation delay (D_{prop}), and packet inference time (T_{infer}).

$$Uplink\ latency = D_{trans} + D_{prop} + D_{queue} + T_{infer}. \quad (6)$$

In particular, D_{trans} is a function of the transmitted packet's size and the link transmission bandwidth, while D_{prop} depends on the transmission distance. It is important to note that bandwidth utilization is critical in determining D_{queue} experienced in the uplink transmission. Excessive bandwidth utilization increases queuing delays as multiple ONUs compete for limited uplink bandwidth, causing significant transmission bottlenecks. Moreover, inference time further contributes to uplink latency, as illustrated in (6).

B. Simulation Results and Discussion

Fig. 5(a) presents the attack success rate achieved by different traffic classifiers under a packet injection attack scenario, where malicious packets generated using different attack patterns are actively injected into the network with varying traffic loads alongside legitimate H2M application packets. As can be observed, CDAD consistently outperforms all other learning-based traffic classifiers across all types of malicious attacks, regardless of the concept drift adaptation-facilitated classifier employed. This superior performance highlights the robustness and adaptability of CDAD, even without prior knowledge of malicious attack types. By leveraging malicious attack adaptation and traffic relabeling, CDAD addresses the limitations of both machine learning and deep learning-based traffic classifiers, ensuring a significantly lower attack success rate across diverse attack scenarios. While deep learning-based traffic classifiers show promising classification accuracy when trained on known malicious attack datasets, they exhibit significant drawbacks during unknown malicious attacks, such as under FGSM, BIM, and ZOO attacks. For instance, the CNN experiences attack success rates of 98.06% under FGSM attacks and 95.39% under BIM attacks. These attacks are highly effective because they exploit the gradient-based optimization methods used in deep learning models. Specifically, FGSM generates adversarial examples by adding perturbations in the direction of the model gradient to maximize classification error, while BIM extends this strategy by iteratively applying small perturbations. Similarly, ZOO attacks use zeroth-order optimization to create adversarial examples, further compromising the robustness and effectiveness of these deep learning models. In this context, the malicious concept drift adaptation mechanism is particularly critical in mitigating adversarial attack patterns such as those employed in FGSM, BIM, and ZOO attacks. In contrast, the machine learning-based traffic classifier, XGBoost, demonstrates relative robustness to gradient-based attacks but is highly susceptible to the malicious user dataset, showing an attack success rate of 79.87%. This vulnerability stems from its inability to adapt to subtle behavioral changes caused by malicious users, highlighting the limitations of traditional machine learning models in dynamic malicious attack environments.

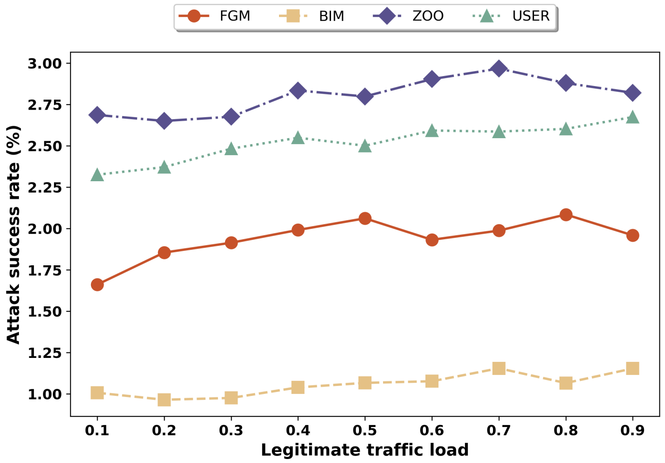


Fig. 6: Attack success rate of CDAD

In an optimal scenario with no malicious packets in the network, the bandwidth utilization should align with the legitimate traffic load, which is approximately 40% highlighted by the red dash line in Fig. 5(b). As can be observed, it is evident that lower attack success rates correlate with reduced bandwidth utilization, highlighting the effectiveness of CDAD in mitigating the impact of malicious traffic across all attack scenarios. Results demonstrate that CDAD, irrespective of the concept drift adaptation method employed, maintains a bandwidth utilization close to the expected threshold in both concept drift adaptation methods, whereas other traffic classifiers exhibit excessive bandwidth utilization, even reaching 100% under certain attack scenarios. Furthermore, the inference time significantly impacts real-time attack detection performance, as illustrated in Fig. 5(c). Results indicate that CNN and LSTM incur inference times exceeding $1ms$ among all malicious attack scenarios. Additionally, while CDAD (SRP) achieves comparable attack mitigation to CDAD (ARF) shown in Fig. 5(a), it demonstrates even higher inference times than other classifiers in some malicious attack scenarios, specifically, Cube, 0.4, BIM, 0.3, and USER, 0.4. Notably, CDAD (ARF) maintains the lowest inference time across all scenarios, ensuring minimal computational delays in traffic classification.

As can be observed from Fig. 5(d), the impact of bandwidth utilization and inference time on the uplink latency is evident, where excessive bandwidth consumption and higher inference times result in substantial increases in overall uplink latency. Specifically, $1ms$ is highlighted in an orange dashed line, and $10ms$ is highlighted in a red dashed line. The performance of CNN and LSTM is significantly impacted under high-traffic attack scenarios, with CNN and LSTM reaching an uplink latency of $14.00ms$ and $12.04ms$ under FGSM, respectively, due to excessive bandwidth utilization and prolonged inference times. Similarly, XGBoost, despite having lower inference times than deep learning models, experiences a degraded uplink latency of $10.72ms$ and $12.28ms$ under FGSM and ZOO attacks, respectively, due to network congestion from high malicious traffic loads. In contrast, CDAD (ARF) maintains the lowest uplink latency across all attack scenarios, with a peak of only $1.07ms$, effectively minimizing queuing delays and computational overhead. These results further validate CDAD (ARF) as an appealing solution, balancing security, computational efficiency, and ultra-low latency requirements for real-time H2M applications.

To further evaluate the robustness of CDAD under different traffic conditions, we conducted simulations in a packet manipulation scenario where all incoming legitimate H2M application packets are intercepted and replaced with malicious attack packets generated by the attacker at the application layer. For every intercepted legitimate packet, a corresponding adversarial packet is substituted in its place, effectively overwriting the original and simulating a one-to-one packet replacement attack. Consequently, the traffic load of malicious packets mirrors the original legitimate traffic load, which is varied between 0.1 and 0.9 to simulate different network conditions. Results, in Fig. 6, demonstrate the consistent effectiveness of CDAD across all considered attack types. The attack success rate remains below 3% for FGSM, BIM, ZOO, and USER attack types, irrespective of the traffic load. This performance highlights the robustness of CDAD to different traffic conditions, which maintains robust defense capabilities even as the proportion of malicious traffic increases. Notably, the overall performance of CDAD is further underscored by its minimal sensitivity to changes in network traffic load. Again, this resilience can be attributed to its dynamic traffic relabeling mechanism, which continuously adapts to incoming traffic patterns to counteract evolving malicious attack threats, further emphasizing its adaptability and stability under dynamic network conditions.

V. Conclusion

In this paper, we introduced a Concept Drift Adaptation-facilitated malicious attack Defense framework (CDAD) to enhance the robustness of H2M applications against evolving malicious attacks. The employed Adaptive Random Forest (ARF) integrates an error-rate-based concept drift detection mechanism to enable incremental learning, dynamically identifying emerging malicious attack patterns and updating its classification model accordingly in real-time H2M malicious attack defense. By learning the packet payload characteristics of both legitimate and malicious H2M traffic, CDAD ensures continuous adaptation without requiring extensive retraining. Additionally, a haptic behavior classifier enables automated traffic relabeling, addressing the critical challenge of unavailable ground-truth labels in real-world malicious traffic scenarios. Extensive packet-level simulations were used to evaluate the performance of CDAD across multiple malicious attack patterns in terms of the attack success rate, bandwidth utilization, inference time, and uplink latency. Results demonstrate that CDAD significantly outperforms other traffic classifiers, limiting the attack success rate to at most 3% with an inference time below $1ms$, ensuring real-time detection and mitigation of adversarial traffic. These findings collectively demonstrate that CDAD offers a robust balance among security, computational efficiency, and latency optimization, ensuring a secure, adaptive, and low-latency defense framework against evolving malicious attacks in H2M applications.

References

[1] G. P. Fettweis, "The tactile internet: Applications and challenges," *IEEE vehicular technology magazine*, vol. 9, no. 1, pp. 64–70, 2014.

[2] E. Wong, M. Pubudini Imali Dias, and L. Ruan, "Predictive resource allocation for tactile internet capable passive optical lans," *Journal of Lightwave Technology*, vol. 35, no. 13, pp. 2629–2641, 2017.

[3] K. Antonakoglou, X. Xu, E. Steinbach, T. Mahmoodi, and M. Dohler, "Toward haptic communications over the 5G tactile internet," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 3034–3059, 2018.

[4] I. Kardush, S. Kim, and E. Wong, "A techno-economic study of industry 5.0 enterprise deployments for human-to-machine communications," *IEEE Communications Magazine*, vol. 60, no. 12, pp. 74–80, 2022.

[5] S. Mondal, L. Ruan, M. Maier, D. Larrabeiti, G. Das, and E. Wong, "Enabling remote human-to-machine applications with AI-enhanced servers over access networks," *IEEE Open Journal of the Communications Society*, vol. 1, pp. 889–899, 2020.

[6] E. Agrell, M. Karlsson, A. Chraplyvy, D. J. Richardson, P. M. Krummrich, P. Winzer, K. Roberts, J. K. Fischer, S. J. Savory, B. J. Eggleton, *et al.*, "Roadmap of optical communications," *Journal of optics*, vol. 18, no. 6, p. 063002, 2016.

[7] L. Ruan, M. P. I. Dias, and E. Wong, "Achieving low-latency human-to-machine (H2M) applications: An understanding of H2M traffic for AI-facilitated bandwidth allocation," *IEEE Internet of Things Journal*, vol. 8, no. 1, pp. 626–635, 2020.

[8] N. Nadarajah, E. Wong, and A. Nirmalathas, "Implementation of multiple secure virtual private networks over passive optical networks using electronic CDMA," *IEEE Photonics Technology Letters*, vol. 18, no. 3, pp. 484–486, 2006.

[9] E. Wong, "Survivable architectures for time and wavelength division multiplexed passive optical networks," *Optics Communications*, vol. 325, pp. 152–159, 2014.

[10] H. Aldabbas and R. Amin, "A novel mechanism to handle address spoofing attacks in SDN based IoT," *Cluster Computing*, vol. 24, no. 4, pp. 3011–3026, 2021.

[11] X. Yu, C. Natalino, P. Monti, L. Wosinska, S. Mondal, Y. Wang, and E. Wong, "Enhancing operational security of human-to-machine applications through concept drift detection," in *Optical Fiber Communication Conference (OFC) 2025*, p. W3J.5, Optica Publishing Group, 2025.

[12] E. Wong, S. Mondal, and L. Ruan, "Machine learning enhanced next-generation optical access networks—challenges and emerging solutions [invited tutorial]," *Journal of Optical Communications and Networking*, vol. 15, no. 2, pp. A49–A62, 2023.

[13] H. M. Gomes, A. Bifet, J. Read, J. P. Barddal, F. Enembreck, B. Pfharinger, G. Holmes, and T. Abdessalem, "Adaptive random forests for evolving data stream classification," *Machine Learning*, vol. 106, pp. 1469–1495, 2017.

[14] B. D. Son, N. T. Hoa, T. Van Chien, W. Khalid, M. A. Ferrag, W. Choi, and M. Debbah, "Adversarial attacks and defenses in 6G network-assisted IoT systems," *IEEE Internet of Things Journal*, 2024.

[15] S. Deng, X. Gao, Z. Lu, and X. Gao, "Packet injection attack and its defense in software-defined networks," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 3, pp. 695–705, 2017.

[16] A. T. Mzrak, S. Savage, and K. Marzullo, "Detecting malicious packet losses," *IEEE Transactions on Parallel and distributed systems*, vol. 20, no. 2, pp. 191–206, 2008.

[17] N. Promwongsa, A. Ebrahimzadeh, D. Naboulsi, S. Kianpisheh, F. Belqasmi, R. Glitho, N. Crespi, and O. Alfandi, "A comprehensive survey of the tactile internet: State-of-the-art and research directions," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 1, pp. 472–523, 2020.

[18] Y. Li, Y. Xu, Z. Liu, H. Hou, Y. Zheng, Y. Xin, Y. Zhao, and L. Cui, "Robust detection for network intrusion of industrial iot based on multi-cnn fusion," *Measurement*, vol. 154, p. 107450, 2020.

[19] L. Yang, M. El Rajab, A. Shami, and S. Muhaidat, "Enabling AutoML for zero-touch network security: Use-case driven analysis," *IEEE Transactions on Network and Service Management*, 2024.

[20] M. Usama, M. Asim, S. Latif, J. Qadir, *et al.*, "Generative adversarial networks for launching and thwarting adversarial attacks on network intrusion detection systems," in *2019 15th international wireless communications & mobile computing conference (IWCMC)*, pp. 78–83, IEEE, 2019.

[21] M. Jain, G. Kaur, and V. Saxena, "A K-Means clustering and SVM based hybrid concept drift detection technique for network anomaly detection," *Expert Systems with Applications*, vol. 193, p. 116510, 2022.

[22] R. Xu, Y. Cheng, Z. Liu, Y. Xie, and Y. Yang, "Improved long short-term memory based anomaly detection with concept drift adaptive method for supporting IoT services," *Future Generation Computer Systems*, vol. 112, pp. 228–242, 2020.

[23] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A deep learning approach to network intrusion detection," *IEEE transactions on emerging topics in computational intelligence*, vol. 2, no. 1, pp. 41–50, 2018.

[24] C. Benzaid and T. Taleb, "AI-driven zero touch network and service management in 5G and beyond: Challenges and research directions," *IEEE Network*, vol. 34, no. 2, pp. 186–194, 2020.

[25] B. Kim, Y. Shi, Y. E. Sagduyu, T. Erpek, and S. Ulukus, "Adversarial attacks against deep learning based power control in wireless communications," in *2021 IEEE Globecom Workshops (GC Wkshps)*, pp. 1–6, IEEE, 2021.

[26] H. Qiu, T. Dong, T. Zhang, J. Lu, G. Memmi, and M. Qiu, "Adversarial attacks against network intrusion detection in IoT systems," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10327–10335, 2020.

[27] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial machine learning at scale," *arXiv preprint arXiv:1611.01236*, 2016.

[28] P.-Y. Chen, H. Zhang, Y. Sharma, J. Yi, and C.-J. Hsieh, "Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models," in *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pp. 15–26, 2017.

[29] X. Yu, L. Ruan, J. S. Evans, and E. Wong, "Novel concept drift detection and adaptation (cdda) framework for human-to-machine (h2m) applications over future communication networks," in *ICC 2024-IEEE International Conference on Communications*, pp. 5509–5514, IEEE, 2024.

[30] X. Yu, L. Ruan, J. S. Evans, and E. Wong, "Adaptive windowing-based concept drift detection and adaptation framework for human-to-machine applications over future communication networks," *Journal of Optical Communications and Networking*, vol. 17, no. 4, pp. 338–351, 2025.

[31] O. Abdel Wahab, "Intrusion detection in the iot under data and concept drifts: Online deep learning approach," *IEEE Internet of Things Journal*, vol. 9, no. 20, pp. 19706–19716, 2022.

[32] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, "Learning under concept drift: A review," *IEEE transactions on knowledge and data engineering*, vol. 31, no. 12, pp. 2346–2363, 2018.

[33] A. Liu, J. Lu, and G. Zhang, "Concept drift detection via equal intensity K-Means space partitioning," *IEEE Transactions on Cybernetics*, vol. 51, no. 6, pp. 3198–3211, 2021.

[34] L. Yang and A. Shami, "A multi-stage automated online network data stream analytics framework for IIoT systems," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 2, pp. 2107–2116, 2023.

[35] A. Bifet, G. Holmes, and B. Pfahringer, "Leveraging bagging for evolving data streams," in *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2010, Barcelona, Spain, September 20-24, 2010, Proceedings, Part I 21*, pp. 135–150, Springer, 2010.

[36] H. M. Gomes, J. Read, and A. Bifet, "Streaming random patches for evolving data stream classification," in *2019 IEEE international conference on data mining (ICDM)*, pp. 240–249, IEEE, 2019.

[37] O. A. Wahab, "Intrusion detection in the IoT under data and concept drifts: Online deep learning approach," *IEEE Internet of Things Journal*, vol. 9, no. 20, pp. 19706–19716, 2022.

[38] Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai, "Kitsune: An ensemble of autoencoders for online network intrusion detection. arxiv 2018," *arXiv preprint arXiv:1802.09089*, 2018.

[39] J. Gama, P. Medas, G. Castillo, and P. Rodrigues, "Learning with drift detection," in *Advances in Artificial Intelligence–SBIA 2004: 17th Brazilian Symposium on Artificial Intelligence, Sao Luis, Maranhao, Brazil, September 29-Ocotber 1, 2004. Proceedings 17*, pp. 286–295, Springer, 2004.

[40] L. Yang, D. M. Manias, and A. Shami, "PWPAE: An ensemble framework for concept drift adaptation in IoT data streams," in *2021 IEEE Global Communications Conference (GLOBECOM)*, pp. 01–06, IEEE, 2021.