Journal of Computer Science and Cybernetics, V.41, N.4 (2025), 387–402
DOI no. 10.15625/1813-9663/22884

ADVERSARIAL ATTACK AND DEFENSE IN AI-POWERED INTRUSION DETECTION

TUYEN T. NGUYEN[1,2], UYEN H. TRAN[1], HOA N. NGUYEN[1,*]

[1] *VNU University of Engineering and Technology, 144 Xuan Thuy, Cau Giay Ward, Ha Noi, Viet Nam*
[2] *Ministry of Public Security, 47 Pham Van Dong, Cau Giay Ward, Ha Noi, Viet Nam*

Abstract. The increasing sophistication of cyberattacks, causing global damages estimated at $9.22 trillion in 2024, highlights the critical importance of robust intrusion detection systems (IDS). AI-driven IDSs like APELID achieve high detection accuracy using advanced ML but remain vulnerable to adversarial ML attacks that craft inputs to evade detection. In this paper, we propose APELID+, an enhanced IDS framework integrating adversarial training and feature squeezing techniques to effectively counter AML threats. We systematically evaluate APELID's vulnerabilities using comprehensive adversarial attack strategies, including both white-box (FGSM, JSMA, PGD, DeepFool, and CW) and black-box attacks (ZOO, HSJA). Experimental results on the CSE-CIC-IDS2018 dataset reveal a significant reduction in APELID's accuracy (from 99.7% to as low as 1.14% under FGSM attacks). The enhanced APELID+ achieves robust performance, maintaining 98.73% accuracy under combined adversarial conditions, surpassing state-of-the-art methods such as Apollon and RAIDS.

Keywords. Adversarial machine learning, intrusion detection systems, adversarial attack, adversarial defense.

1. INTRODUCTION

In recent years, the integration of artificial intelligence (AI), especially machine learning (ML) and deep learning (DL), into IDS has significantly enhanced its capabilities. AI-powered IDS frameworks like APELID (augmented parallel ensemble learning-based intrusion detection), proposed by Vo et al. [1], have achieved remarkable accuracy and effectiveness, surpassing traditional signature-based methods by detecting complex. However, even advanced AI-based intrusion detection systems (IDSs) remain susceptible to sophisticated adversarial machine learning (AML) attacks, wherein inputs are deliberately crafted to mislead models and evade detection. Despite extensive research, existing AI-driven IDSs continue to exhibit vulnerabilities to adversarial examples [2]. Recent studies indicate that both ensemble-based and deep neural architectures are exposed to white-box and black-box attack scenarios, underscoring the need for more robust defensive mechanisms. Nevertheless, the unified integration of adversarial training and feature squeezing within ensemble IDS frameworks remains a relatively underexplored area of research.

*Corresponding author.
E-mail addresses: tuyennt@cybersecurity.vn (T.T. Nguyen); 21020672@vnu.edu.vn (U.H. Tran); hoa.nguyen@vnu.edu.vn (H.N. Nguyen).

To bridge this research gap, we propose an enhanced IDS framework, APELID+, that systematically integrates adversarial training and feature squeezing to enhance resilience against adversarial threats. This paper aims to:

- Systematically assess the vulnerabilities of APELID to a comprehensive set of adversarial attacks.
- Analyze the impact of adversarial attacks specifically targeting both deep learning and ensemble learning components of APELID.
- Introduce an improved adversarial defense model (APELID+) integrating adversarial training and feature squeezing to robustly counter AML threats.
- Evaluate and demonstrate the effectiveness of APELID+ using extensive experiments conducted on the CSE-CIC-IDS2018 dataset.

Our contributions can be summarized as follows:

1. We develop a comprehensive adversarial attack model utilizing diverse white-box and black-box techniques to rigorously evaluate vulnerabilities in APELID.
2. We propose APELID+, an enhanced IDS framework integrating adversarial training and feature squeezing to significantly bolster resilience against adversarial threats.
3. Extensive experimental results show that APELID+ maintains superior performance under adversarial conditions, achieving an accuracy of 98.73% and substantially outperforming current state-of-the-art methods such as Apollon and RAIDS.

The remainder of the paper is organized as follows: Section 2 reviews related work in adversarial attacks and defenses within IDS. Section 3 presents the architecture and methodologies underlying the APELID framework. Section 4 describes our proposed adversarial attack and defense approaches. Section 5 details experimental setups and evaluates the effectiveness of APELID+. Finally, Section 6 concludes the paper and discusses future research directions.

2. RELATED WORK

The field of AML has gained significant traction in recent years, with research focusing on both developing novel adversarial attack techniques and designing effective defense mechanisms to enhance the robustness of AI models. This section provides an overview of relevant works in these areas, particularly in the context of IDS.

2.1. AI-powered intrusion detection systems

The escalating severity of cyberattacks demands a robust IDS with high accuracy. ML and DL enhance IDS by processing large datasets, detecting complex patterns, and adapting to new threats without manual feature design [3, 4]. ML-based IDS, using algorithms like decision trees (DT), support vector machines (SVM), and K-nearest neighbors (KNN), excel in classifying novel threats. For instance, DT-based IDS achieved a 97.95% detection rate on UNSW-NB15 [3], while SVM with Naïve Bayes preprocessing reached 98.92–99.35% accuracy on CICIDS2017 and NSL-KDD [5]. Multi-layer SVM with KPCA and GA optimization hits 96.38% detection on KDDCUP'99 with low false alarms [6].

DL-based IDS, leveraging convolutional neural networks (CNN), recurrent neural networks (RNN), autoencoders (AE), and generative adversarial networks (GAN), detects sophisticated attacks. CNNs extract key features, improving minority class detection [7, 8]. RNNs, particularly LSTM, identify anomalies in network traffic [9]. AEs outperformed other methods [10], while GANs addressed data imbalance, enhancing robustness on NSL-KDD and UNSW-NB15 [11].

Ensemble methods like XGBoost, CatBoost, Gradient Boosting, and Bagging Meta-Estimator boost IDS resilience. Bagging with tree-based classifiers achieved 84.93% accuracy on NSL-KDD [12], while parallel ensemble with PSO and CFS hit 99.80% [13]. Combining GBDT and GRU improved spatial-temporal analysis [14], and multi-metric feature selection enhanced detection rates across datasets [4, 15].

2.2. Adversarial attacks on IDS

Adversarial attacks operate by creating adversarial samples, which are specialized inputs designed to appear legitimate to humans but negatively impact AI models' decision-making processes. These attacks are typically categorized as white-box, where the attacker has full knowledge of the model, and black-box, where only limited information about the model is available. Popular white-box attacks include fast gradient sign method (FGSM) [16], Jacobian-based saliency map attack (JSMA) [17], projected gradient descent (PGD) [18], Carlini and Wagner attack (CW) [19], and DeepFool [20]. In terms of black-box attacks, Zeroth-order optimization (ZOO) [21] and HopSkipJump (HSJA) [22] are demonstrated as the two most effective and widely used methods.

Recent research has explored various adversarial attack strategies targeting AI-powered IDS. Zhang et al. [23] introduced ETA, a black-box attack framework emphasizing transferability and explainability, achieving high success rates against various ML models on the CIC-IDS2017 and Kitsune datasets. Ennaji et al. [24] proposed a Transferability Feasibility Score (TFS) to assess the effectiveness of transferable adversarial attacks, providing insights for crafting more realistic attacks and enhancing defense strategies.

2.3. Adversarial defenses for IDS

In response to the growing threat of adversarial attacks, researchers have proposed several defense strategies for IDS. Apollon, presented by Paya et al. [25], is a defense system using a Multi-Armed Bandits (MAB) model to dynamically select the most suitable classifier for each network traffic input, showing improved detection rates against various attacks on the CIC-IDS2017 dataset. Zhao et al. [26] explored the use of generative adversarial networks (GANs) for data augmentation to enhance the performance of IDS with limited training data.

Roshan et al. [27] applied Gaussian data augmentation (GDA) and feature squeezing (FS) to improve the resilience of DNN-based NIDS against the Carlini and Wagner (CW) attack, achieving significant improvements in accuracy. Sarıkaya et al. [28] introduced RAIDS, a defense system combining autoencoders and an ensemble of ML classifiers with a LightGBM classifier, demonstrating enhanced accuracy against GAN-generated adversarial examples.

3. APELID FRAMEWORK

APELID [1] is a state-of-the-art intrusion detection framework designed for real-time malicious network traffic detection. It comprises two main components: Augmented wasser-

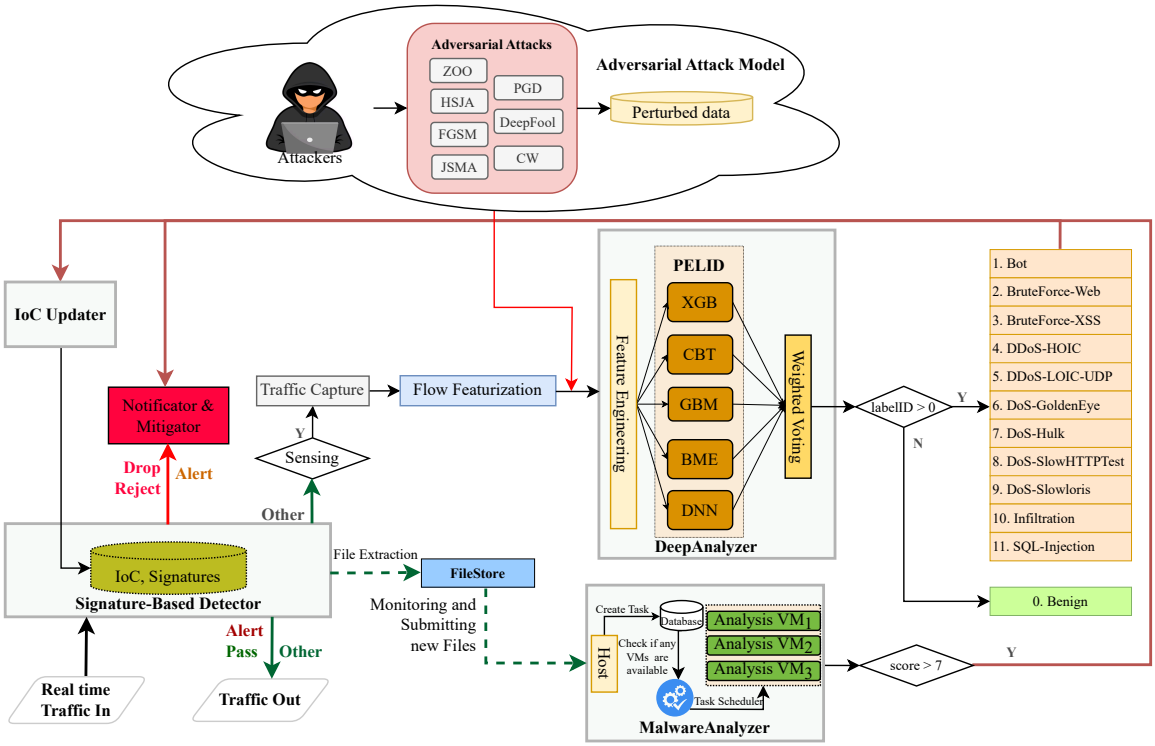

Figure 1: Adversarial attack model against APELID

stein generative adversarial networks (AWGAN) for data augmentation and parallel ensemble learning-based intrusion detection (PELID) for traffic analysis.

3.1. Augmented wasserstein generative adversarial networks

AWGAN addresses the issue of class imbalance in network traffic datasets through a two-pronged approach: handling majority classes and augmenting minority classes. The data preparation pipeline involves data cleaning, normalization using MinMaxScaler, and random splitting into training (70%) and testing (30%) sets.

For majority classes, AWGAN employs the edited nearest neighbor (ENN) algorithm to identify and potentially remove noisy or borderline instances, followed by a clustering algorithm to compress the remaining samples while preserving representative characteristics. For minority classes, AWGAN utilizes Wasserstein GAN (WGAN) [29] to generate synthetic samples, effectively increasing the representation of these underrepresented classes in the training dataset.

3.2. Parallel ensemble learning-based intrusion detection

PELID employs a parallel ensemble learning approach, integrating five distinct AI models: XGBoost (XGB), CatBoost (CBT), GradientBoostingMachine (GBM), BaggingMetaEstimator (BME), and a deep neural network (DNN). Each model is trained independently on the augmented training data generated by AWGAN.

During the detection phase, the predictions from each of the five models are combined using a soft-voting mechanism, where each model's prediction is weighted by a factor $\omega_i \in$

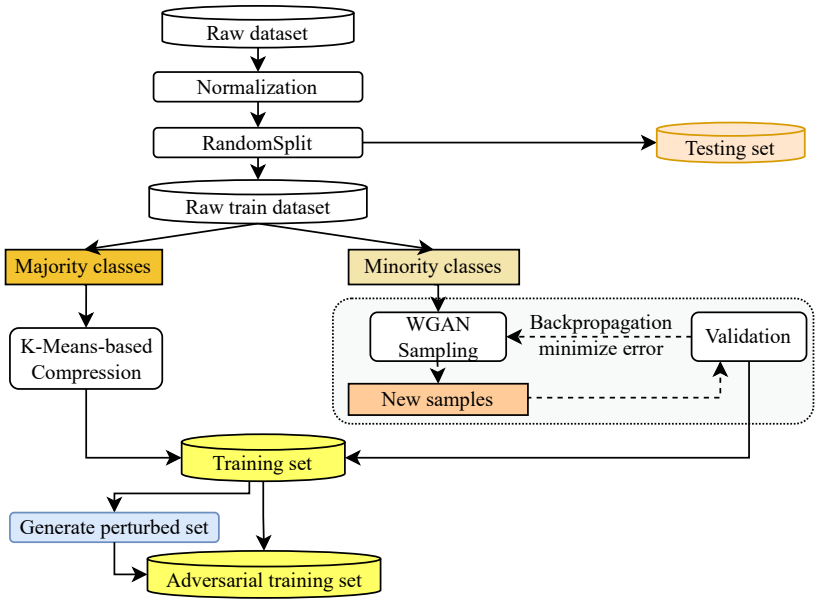

Figure 2: Augmented training dataset with adversarial samples

$(0, 1)$, with the constraint $\sum_{i=1}^{5} \omega_i = 1$. These weights are experimentally determined to optimize the ensemble's performance. The final prediction is based on the weighted average of the individual model outputs.

4. PROPOSED METHOD: APELID+

To enhance the robustness of the APELID framework against adversarial attacks, we propose APELID+, an enhanced version incorporating both adversarial attack and defense models.

4.1. Adversarial attack model

Our adversarial attack model, illustrated in Figure 1, is designed to evaluate the vulnerability of APELID's constituent models to evasion attacks. We target each model with adversarial techniques suitable for its architecture. For the DNN, we employ white-box gradient-based attacks: FGSM, JSMA, PGD, and CW. For the tree-based models (XGB, CBT, GBM, and BME), we utilize black-box attacks: ZOO and HSJA.

The attack model generates adversarial examples from the test dataset for each individual model in APELID. These perturbed datasets are then used to evaluate the performance degradation of each model and the overall ensemble. The complete procedure is outlined in Algorithm 1.

4.2. Adversarial defense model

Our adversarial defense model for APELID+, illustrated in Fig. 2 and Fig. 3, employs two complementary strategies: adversarial training and feature squeezing, chosen for their proven effectiveness and suitability for APELID's heterogeneous architecture. Adversarial training improves the robustness of tree-based models (XGB, GBM, CBT, and BME) by exposing them to adversarial examples, mitigating vulnerabilities to black-box attacks such as ZOO

Algorithm 1 Adversarial attacks against APELID

Input: V - Testing dataset.
XGB, GBM, BME, CBT, DNN - Trained models.
ω_i - Model weights; $\sum_{i=1}^{5} \omega_i = 1$.
Output: P - Predictions.

$XGBClassifier \leftarrow ARTClassifier(XGB)$ ▷ Create XGB classifier.
$GBMClassifier \leftarrow ARTClassifier(GBM)$ ▷ Create GBM classifier.
$BMEClassifier \leftarrow ARTClassifier(BME)$ ▷ Create BME classifier.
$CBTClassifier \leftarrow ARTClassifier(CBT)$ ▷ Create CBT classifier.
$DNNClassifier \leftarrow ARTClassifier(DNN)$ ▷ Create DNN classifier.
$XGB\_AEs \leftarrow Attack(XGBClassifier, V)$ ▷ Generate adversarial samples for XGB.
$GBM\_AEs \leftarrow Attack(GBMClassifier, V)$ ▷ Generate adversarial samples for GBM.
$BME\_AEs \leftarrow Attack(BMEClassifier, V)$ ▷ Generate adversarial samples for BME.
$CBT\_AEs \leftarrow Attack(CBTClassifier, V)$ ▷ Generate adversarial samples for CBT.
$DNN\_AEs \leftarrow Attack(DNNClassifier, V)$ ▷ Generate adversarial samples for DNN.
Perform in parallel five processes P1, P2, P3, P4, P5:
P1: $pXGB \leftarrow XGBClassifier.predict(XGB\_AEs)$ ▷ Perform predictions using XGB.
P2: $pGBM \leftarrow GBMClassifier.predict(GBM\_AEs)$ ▷ Perform predictions using GBM.
P3: $pBME \leftarrow BMEClassifier.predict(BME\_AEs)$ ▷ Perform predictions using BME.
P4: $pCBT \leftarrow CBTClassifier.predict(CBT\_AEs)$ ▷ Perform predictions using CBT.
P5: $pDNN \leftarrow DNNClassifier.predict(DNN\_AEs)$ ▷ Perform predictions using DNN.
Wait until P1, P2, P3, P4, P5 finished.
$P \leftarrow \omega_1 * pXGB + \omega_2 * pGBM + \omega_3 * pBME + \omega_4 * pCBT + \omega_5 * pDNN$ ▷ Ensemble predictions with weighted voting.
return P.

and HSJA. Feature squeezing, applied as a preprocessing step, counters white-box attacks (e.g., FGSM, JSMA) on the DNN by reducing input complexity and weakening crafted perturbations. The synergy between adversarial training and feature squeezing—where the former strengthens decision boundaries and the latter filters malicious perturbations—provides comprehensive protection across attack types. Key parameters, including the number of adversarial samples (7,000 per attack type in AWGAN+) and the squeezing threshold (bit-depth reduction to 5 bits), determine the balance between defense strength and computational cost.

Adversarial training [16] operates by generating adversarial examples during the training phase and incorporating them into the training dataset, effectively expanding the model's decision boundaries to include perturbed inputs. This technique is particularly suitable for the tree-based models (XGB, GBM, CBT, and BME) in APELID, as these models are vulnerable to black-box attacks like ZOO and HSJA due to their reliance on feature splits. By retraining with adversarial samples generated from ZOO and HSJA, the models learn to generalize better against such perturbations. For each model, we generate 7,000 adversarial examples per attack type from the training data and add them to the original dataset, enhancing robustness without significantly increasing training time.

Feature squeezing [30] is a preprocessing defense that reduces the complexity of input features to eliminate adversarial noise. It works by applying transformations such as bit-

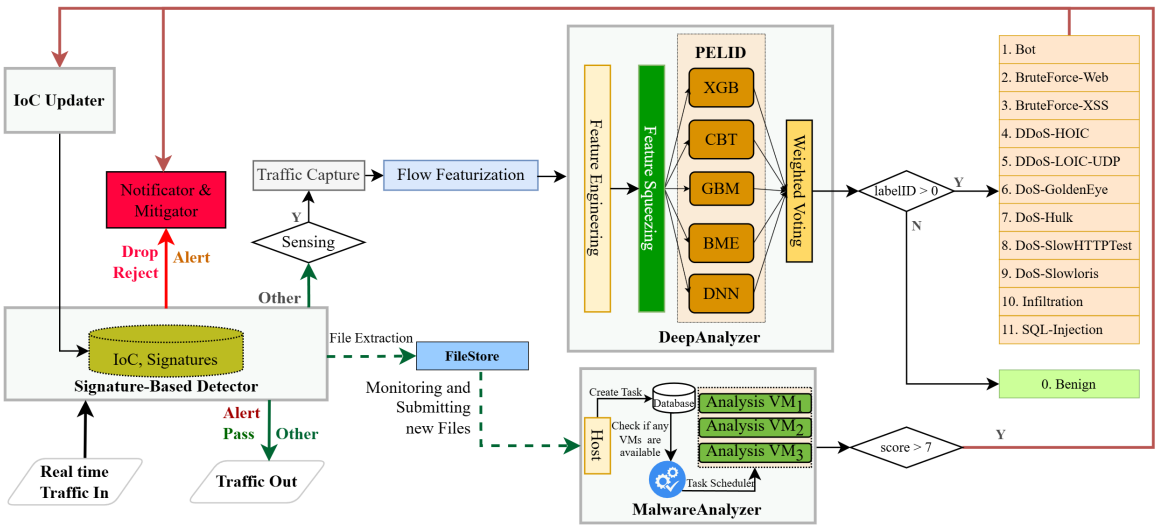

Figure 3: Adversarial defense using feature squeezing

depth reduction (e.g., from 8 bits to 5 bits per feature) or median smoothing (e.g., 2×2 kernel) to "squeeze" the input, then comparing the model's predictions on the original and squeezed inputs. If the predictions differ beyond a threshold (e.g., 0.1 in L1 distance), the input is flagged as adversarial and discarded or reclassified. This method is applied during the validation phase and is ideal for the DNN component of APELID, which is sensitive to white-box gradient-based attacks like FGSM and PGD, as it disrupts the precise perturbations crafted by these attacks. The integration of feature squeezing with adversarial training creates a layered defense: training fortifies the models internally, while squeezing acts as an external filter, making APELID+ resilient to a wide range of AML threats.

We refer to the enhanced data augmentation with adversarial examples as AWGAN+. The complete procedure for the adversarial defense model is summarized in Algorithm 2.

5. EXPERIMENTS AND EVALUATION

We conducted a series of experiments to evaluate the effectiveness of our proposed adversarial attack and defense models, aiming to answer the following research questions:

1. RQ1: To what extent are AI models, such as APELID, susceptible to adversarial attacks?
2. RQ2: What is the impact of different adversarial attacks on deep learning models and ensemble learning models?
3. RQ3: Does generating additional adversarial samples for model training help strengthen the resistance of ensemble-based IDS against adversarial attacks?
4. RQ4: Does the feature-squeezing technique enhance the resilience of deep learning models against adversarial attacks?
5. RQ5: Does applying ensemble learning enhance the performance of IDS against adversarial attacks?

Algorithm 2 Adversarial defenses against APELID

Input: T - Training dataset.
V - Testing dataset.
XGB, GBM, BME, CBT, DNN - Trained models.
Output: P - Predictions.

$XGBClassifier \leftarrow ARTClassifier(XGB)$ ▷ Create XGB classifier.
$GBMClassifier \leftarrow ARTClassifier(GBM)$ ▷ Create GBM classifier.
$BMEClassifier \leftarrow ARTClassifier(BME)$ ▷ Create BME classifier.
$CBTClassifier \leftarrow ARTClassifier(CBT)$ ▷ Create CBT classifier.
$DNNClassifier \leftarrow ARTClassifier(DNN)$ ▷ Create DNN classifier.
$XGB\_AEs \leftarrow Attack(XGBClassifier, T)$ ▷ Generate adversarial samples for XGB.
$GBM\_AEs \leftarrow Attack(GBMClassifier, T)$ ▷ Generate adversarial samples for GBM.
$BME\_AEs \leftarrow Attack(BMEClassifier, T)$ ▷ Generate adversarial samples for BME.
$CBT\_AEs \leftarrow Attack(CBTClassifier, T)$ ▷ Generate adversarial samples for CBT.
$tXGB \leftarrow T \cup XGB\_AEs$ ▷ Create new training dataset for XGB.
$tGBM \leftarrow T \cup GBM\_AEs$ ▷ Create new training dataset for GBM.
$tBME \leftarrow T \cup BME\_AEs$ ▷ Create new training dataset for BME.
$tCBT \leftarrow T \cup CBT\_AEs$ ▷ Create new training dataset for CBT.
$adXGB \leftarrow fit(tXGB)$ ▷ Retrain XGB with training dataset $tXGB$.
$adGBM \leftarrow fit(tGBM)$ ▷ Retrain GBM with training dataset $tGBM$.
$adBME \leftarrow fit(tBME)$ ▷ Retrain BME with training dataset $tBME$.
$adCBT \leftarrow fit(tCBT)$ ▷ Retrain CBT with training dataset $tCBT$.
$V \leftarrow FS(V)$ ▷ Perform feature squeezing on V
$pXGB \leftarrow adXGB.predict(V)$ ▷ Perform predictions using XGB.
$pGBM \leftarrow adGBM.predict(V)$ ▷ Perform predictions using GBM.
$pBME \leftarrow adBME.predict(V)$ ▷ Perform predictions using BME.
$pCBT \leftarrow adCBT.predict(V)$ ▷ Perform predictions using CBT.
$pDNN \leftarrow adDNN.predict(V)$ ▷ Perform predictions using DNN.
$P \leftarrow \omega_1 * pXGB + \omega_2 * pGBM + \omega_3 * pBME + \omega_4 * pCBT + \omega_5 * pDNN$ ▷ Ensemble predictions with weighted voting.
return P.

5.1. Experiment environment

All experiments were performed on a system with an Intel Core Ultra 5 125H CPU and 16GB RAM, running Python 3.10.16. We utilized the adversarial-robustness-toolbox v1.19.1, fastai v2.7.10, numpy v2.2.4, pandas v2.2.3, scikit-learn v1.6.1, torch v2.6.0, and xgboost v1.6.1 libraries.

5.2. Dataset preparation

We used the CSE-CIC-IDS2018 dataset [31] for our experiments. The data was preprocessed using the AWGAN algorithm as described in Section 3.1.. Six features were removed due to their insignificant impact on prediction accuracy, resulting in a final set of 74 features (including the label). The dataset was split into categorical (Dst Port, Protocol) and continuous variables. To address class imbalance, a threshold of 20,000 samples per label class

Table 1: Label distribution in augmented dataset

Label	Original	AWGAN	AWGAN+	Test
Benign	4,360,029	14,000	28,000	6,000
DDoS attacks-HOIC	668,461	14,000	28,000	6,000
DoS attacks-Hulk	434,873	14,000	28,000	6,000
Bot	282,310	14,000	28,000	6,000
Infiltration	160,604	14,000	28,000	6,000
DoS attacks-GoldenEye	41,455	14,000	28,000	6,000
DoS-SlowHTTPTest	13,067	14,000	28,000	4,082
DoS attacks-Slowloris	6,977	14,000	28,000	2,093
DDoS attack-LOIC-UDP	1,120	14,000	28,000	336
Brute Force-Web	261	14,000	28,000	78
Brute Force-XSS	97	14,000	28,000	29
SQL Injection	53	14,000	28,000	16
Total	**5,982,374**	**168,000**	**336,000**	**42,634**

Table 2: Model hyperparameters and weights

Model	Weight	Hyperparameter	Value
XGB	0.3	n_estimators max_depth learning_rate	3 6 0.3
GBM	0.2	n_estimators learning_rate	10 0.1
CBT	0.2	iterations	50
BME	0.2	n_estimators	30
DNN	0.1	epochs learning_rate	5 0.01

was applied, resulting in a training set with a maximum of 14,000 samples per class and a testing set with up to 6,000 samples per class. Then, we applied AWGAN+ to generate more training samples using ZOO and HSJA, with each method contributing 7,000 samples per label. The label distribution in the augmented dataset is presented in Table 1.

5.3. Hyperparameter optimization

The hyperparameters of each model in APELID were optimized using techniques described in [1], including Hyperparameter Optimization (via grid search for exhaustive exploration), Bayesian Optimization (for efficient sampling in high-dimensional spaces), and self-determination (manual tuning based on domain knowledge). This optimization is crucial for APELID+ because it ensures that each base model (e.g., XGB with n_estimators=3, max_depth=6) balances accuracy and robustness against adversarial perturbations, preventing over-fitting on augmented data from AWGAN+. The ensemble weights (ω_i) were tuned to prioritize robust models (e.g., higher weight for XGB at 0.3), enhancing overall defense efficacy. The final hyperparameters and ensemble weights are presented in Table 2.

Table 3: Original models performance against adversarial attacks (%)

Metrics	XGB			CBT			GBM		
	Original	ZOO	HSJA	Original	ZOO	HSJA	Original	ZOO	HSJA
F1	99.79	35.76	12.96	99.92	92.72	10.69	99.95	60.83	1.72
Acc	99.76	33.99	13.86	99.92	92.85	13.34	99.96	55.45	1.19
Prec	99.83	60.77	13.95	99.93	96.58	9.85	99.96	79.20	3.71
DR	99.76	33.99	13.86	99.92	92.85	13.34	99.96	55.45	1.19
ASR	-	65.93	86.12	-	7.10	86.70	-	44.30	98.81
Metrics	BME			DNN					
	Original	ZOO	HSJA	Original	FGSM	JSMA	PGD	DeepFool	CW
F1	99.77	94.73	7.02	97.95	1.20	22.64	42.50	0.46	21.57
Acc	99.98	95.12	13.92	97.54	1.14	19.78	43.47	4.91	22.74
Prec	99.98	96.25	4.70	98.20	22.89	49.24	44.98	14.31	49.39
DR	99.98	95.12	13.92	97.54	1.14	19.78	43.47	4.91	22.74
ASR	-	4.87	86.10	-	99.00	80.03	55.80	95.00	77.99

5.4. Evaluation metrics

We evaluated the performance of our models using standard classification metrics: Accuracy, F1 Score, Precision, and Recall (Detection Rate). For multi-class classification, these metrics were calculated as weighted averages across all classes. Additionally, we used Attack Success Rate (ASR) to measure the effectiveness of the adversarial attacks. It is defined as the percentage of correctly classified by the model on the original dataset but, were misclassified after adversarial perturbation [32].

5.5. Ablation studies

We conducted a series of ablation studies through three main experimental scenarios to comprehensively evaluate the performance of the original APELID and our enhanced APELID+ under various adversarial attack conditions. Specifically, the experiments were designed as follows: (S1) We evaluated the adversarial robustness of both individual base models and the original ensemble models within the APELID methodology; (S2) We analyzed the effectiveness and resilience of single models after augmentation with adversarial samples, which were generated using the feature squeezing technique; and (S3) We compared the performance of the improved ensemble model, APELID+, following its augmentation with adversarial samples incorporating feature squeezing, against that of the original APELID ensemble.

5.5.1. S1 evaluation

In this experiment, we evaluated the performance of the individual models (XGB, GBM, BME, CBT, and DNN) and the original APELID ensemble on both the original test dataset and adversarially perturbed datasets. As shown in Table 3, the performance significantly degrades under adversarial attacks. For instance, the accuracy of XGB and GBM drops considerably under ZOO and HSJA attacks, while the DNN model's accuracy is severely impacted by all gradient-based attacks. This confirms the vulnerability of APELID to adversarial attacks (RQ1). The impact of different attacks varies across model types (RQ2), with tree-based models being more susceptible to black-box attacks and the DNN to white-box attacks.

Table 4: Models with adversarial defenses performance against adversarial attacks (%)

Metrics	XGB			CBT			GBM		
	Original	ZOO	HSJA	Original	ZOO	HSJA	Original	ZOO	HSJA
F1	99.90	99.90	99.63	99.92	99.92	97.70	99.35	99.38	98.65
Acc	99.91	99.91	99.65	99.93	99.92	97.70	99.50	99.52	98.76
Prec	99.93	99.93	99.66	99.93	99.92	97.81	99.21	99.24	98.56
DR	99.91	99.91	99.65	99.93	99.92	97.70	99.50	99.52	98.76
Metrics	BME			DNN					
	Original	ZOO	HSJA	Original	FGSM	JSMA	PGD		
F1	99.96	99.96	99.58	97.75	97.75	97.75	82.42		
Acc	99.96	99.96	99.58	97.69	97.69	97.69	82.24		
Prec	99.97	99.97	99.61	98.05	98.05	98.05	89.84		
DR	99.96	99.96	99.58	97.69	97.69	97.69	82.24		

5.5.2. S2 evaluation

This experiment evaluated the performance of the individual models in APELID+ (after applying adversarial training and feature squeezing) against the same adversarial attacks. The results in Table 4 show a significant improvement in resilience. The accuracy of the DNN model against FGSM, JSMA, and PGD attacks increased dramatically, indicating the effectiveness of feature squeezing (RQ4). The tree-based models also showed improved performance against ZOO and HSJA attacks due to adversarial training (RQ3).

Table 5: APELID+ performance against adversarial attacks (%)

Metrics	Original		ZOO		HSJA	
	APELID	APELID+	APELID+	APELID+	APELID	APELID+
F1	99.69	99.97	99.62	99.96	52.24	99.91
Acc	99.70	99.96	99.62	99.96	52.75	99.92
Prec	99.71	99.97	99.64	99.97	68.50	99.92
DR	99.70	99.96	99.62	99.96	52.75	99.92
ASR	-	-	0.37	0.00	47.24	0.06
Metrics	FGSM		JSMA		PGD	
	APELID	APELID+	APELID+	APELID+	APELID	APELID+
F1	99.69	99.96	99.69	99.96	99.69	99.96
Acc	99.70	99.96	99.70	99.96	99.70	99.96
Prec	99.71	99.97	99.71	99.97	99.71	99.96
DR	99.70	99.96	99.70	99.96	99.70	99.96
ASR	0.00	0.00	0.00	0.00	0.00	0.00
Metrics	DeepFool		CW		All	
	APELID	APELID+	APELID+	APELID+	APELID	APELID+
F1	99.69	99.97	99.69	99.98	**66.76**	**98.73**
Acc	99.70	99.97	99.70	99.98	**63.26**	**98.73**
Prec	99.71	99.97	99.71	99.98	**79.91**	**98.73**
DR	99.70	99.97	99.70	99.98	**63.26**	**98.73**
ASR	0.00	0.00	0.02	0.00	**36.74**	**1.24**

5.5.3. S3 evaluation

Table 5 compares the performance of the original APELID and APELID+ against various adversarial attacks and a combined attack dataset. The results clearly demonstrate the significant improvement in robustness achieved by APELID+. For the combined attack dataset, the F1 score and accuracy of APELID+ increased from 66.76% and 63.26% to 98.73% and 98.73%, respectively. This highlights the effectiveness of our proposed defense

strategies. Furthermore, comparing the ensemble performance with individual models (as seen in Tables 3 and 4), the ensemble learning approach in APELID consistently shows better resilience against adversarial attacks compared to the individual base models (RQ5).

5.6. Comparison with state-of-the-art methods

To further validate the effectiveness of APELID+, we compared its performance against several state-of-the-art adversarial defense methods for IDS. Table 6 shows the accuracy scores of APELID+ and other methods under different adversarial attacks. APELID+ consistently outperforms the other methods across various attack types, demonstrating its superior robustness.

Table 6: Comparison with SOTA methods on accuracy with different adversarial attacks (%). Note: Comparisons are relative across datasets

Method	ZOO	HSJA	FGSM	JSMA	PGD	DeepFool	CW	All
APELID+	**99.96**	**99.92**	**99.96**	**99.96**	**99.96**	**99.97**	**99.98**	**98.73**
APELID	99.62	52.75	99.70	99.70	99.70	99.70	99.70	63.26
Def-IDS [33]	-	-	97.40	97.90	-	98.30	-	-
Apollon [25]	93.04	75.50	-	-	-	-	-	-
Omni [34]	-	-	86.68	77.87	-	87.12	75.23	-
RAIDS [28]	-	-	-	-	59.00	-	-	-
DLL-IDS [35]	-	-	95.00	-	-	71.50	63.70	-

APELID+'s hybrid defense, combining adversarial training (AWGAN+) and feature squeezing, effectively counters white-box (FGSM, JSMA, PGD, DeepFool, and CW) and black-box (ZOO, HSJA) attacks, achieving 98.73–99.98% accuracy on CSE-CIC-IDS2018. Unlike APELID, which drops to 52.75% under HSJA and 63.26% for combined attacks, APELID+ ensures robust performance. It outperforms Def-IDS [33] (97.40–98.30%, limited attack coverage), Apollon [25] (93.04% ZOO, 75.50% HSJA), Omni [34], and DLL-IDS [35] (63.70–87.12% for gradient-based attacks), and RAIDS [28] (59.00% PGD). APELID+'s ensemble of diverse models (XGBoost, CatBoost, GBM, BME, and DNN) and feature squeezing enhances stability, mitigates perturbations, making it a leading solution for AI-powered IDS.

6. CONCLUSION

This paper investigated adversarial threats to AI-powered IDS, focusing on the APELID framework. We introduced an adversarial attack model that significantly degraded APELID's performance using white-box and black-box techniques. To mitigate these vulnerabilities, we proposed APELID+, integrating adversarial training and feature squeezing. Experiments on the CSE-CIC-IDS2018 dataset revealed APELID's weaknesses and APELID+'s enhanced robustness, outperforming state-of-the-art defenses. APELID+'s efficient ensemble architecture and real-time adaptability make it suitable for deployment in operational IDS, ensuring robust protection against evolving cyberattacks with manageable computational overhead. Regarding practical applicability, while APELID+ introduces additional complexity through adversarial training and feature squeezing, its parallel ensemble design allows for efficient real-time monitoring. In our experiments, training time increased by approximately 20-30% due to added adversarial samples (7,000 per attack type), but inference time remains low (un-

der 1ms per sample on the tested hardware), making it feasible for network traffic analysis. Future optimizations, such as selective feature perturbation, could further reduce overhead. However, limitations persist. The attack model's reliance on perturbing all features reduces efficiency in resource-constrained environments. The defense model lacks tailored adversarial training for the DNN, limiting its effectiveness against gradient-based attacks. Validation is confined to CSE-CIC-IDS2018, restricting generalizability. Future work will optimize feature selection for attacks, develop DNN-specific defenses, and validate APELID+ on datasets like NSL-KDD, UNSW-NB15, and CIC-DDoS-2019 to enhance real-world applicability.

ACKNOWLEDGMENT

This research was funded by the Ministry of Science and Technology of Vietnam under the National Science and Technology Program with project code KC.01.06/21-30.

REFERENCES

[1] H. V. Vo, H. P. Du, and H. N. Nguyen, "APELID: Enhancing real-time intrusion detection with augmented WGAN and parallel ensemble learning," *Computers and Security*, vol. 136, p. 103567, 2024. [Online]. Available: https://doi.org/10.1016/j.cose.2023.103567

[2] A. Alotaibi and M. A. Rassam, "adversarial machine learning attacks against intrusion setection Systems: A survey on strategies and defense," *Future Internet*, vol. 15, no. 2, p. 62, 2023. [Online]. Available: https://doi.org/10.3390/fi15020062

[3] M. A. Umar, C. Zhanfang, and Y. Liu, "A hybrid intrusion detection with decision tree for feature selection," *Information amd Security: An International Journal*, vol. 49, 2021. [Online]. Available: https://doi.org/10.11610/isij.4901

[4] H. V. Vo, D. H. Nguyen, T. T. Nguyen, H. N. Nguyen, and D. V. Nguyen, "Leveraging ai-driven realtime intrusion detection by using wgan and xgboost," in *Proceedings of the 11th International Symposium on Information and Communication Technology*, ser. SoICT '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 208–215. [Online]. Available: https://doi.org/10.1145/3568562.3568660

[5] J. Gu and S. Lu, "An effective intrusion detection approach using SVM with Naïve Bayes feature embedding," *Computers and Security*, vol. 103, p. 102158, 2021. [Online]. Available: https://doi.org/10.1016/j.cose.2020.102158

[6] F. Kuang, W. Xu, and S. Zhang, "A novel hybrid KPCA and SVM with GA model for intrusion detection," *Applied Soft Computing*, vol. 18, pp. 178–184, 2014. [Online]. Available: https://doi.org/10.1016/j.asoc.2014.01.028

[7] V. V. Hoang, D. P. Hanh, and H. N. Nguyen, "Awdlid: Augmented wgan and deep learning for improved intrusion detection," in *2024 1st International Conference on Cryptography and Information Security (VCRIS)*, 2024, pp. 1–6. [Online]. Available: https://doi.org/10.1109/VCRIS63677.2024.10813392

[8] V. V. Hoang, D. P. Hanh, and N. N. Hoa, "Ai-powered intrusion detection in large-scale traffic networks based on flow sensing strategy and parallel deep analysis," *Journal of*

Network and Computer Applications, vol. 220, p. 103735, 2023. [Online]. Available: https://doi.org/10.1016/j.jnca.2023.103735

[9] B. J. Radforda, L. M. Apolonio, A. J. Trias, and J. A. Simpson, "Network traffic anomaly detection using recurrent neural networks," *arXiv*, vol. abs/1803.10769, 2018. [Online]. Available: https://doi.org/10.48550/arXiv.1803.10769

[10] F. Kamalov, R. Zgheib, H. H. Leung, A. Al-Gindy, and S. Moussa, "Autoencoder-based intrusion detection system," in *2021 International Conference on Engineering and Emerging Technologies (ICEET)*, 2021, pp. 1–5. [Online]. Available: https://doi.org/10.1109/ICEET53442.2021.9659562

[11] C. Park, J. Lee, Y. Kim, J.-G. Park, H. Kim, and D. Hong, "An enhanced AI-based network intrusion detection system using generative adversarial networks," *IEEE Internet of Things Journal*, vol. 10, no. 3, pp. 2330–2345, 2023. [Online]. Available: https://doi.org/10.1109/JIOT.2022.3211346

[12] M. M. Rashid, J. Kamruzzaman, M. Ahmed, N. Islam, S. Wibowo, and S. Gordon, "Performance enhancement of intrusion detection system using bagging ensemble technique with feature selection," in *2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, 2020, pp. 1–5. [Online]. Available: https://doi.org/10.1109/CSDE50874.2020.9411608

[13] B. A. Tama and K. H. Rhee, "A combination of PSO-based feature selection and tree-based classifiers ensemble for intrusion detection systems," in *Advances in Computer Science and Ubiquitous Computing*, 2015, pp. 489–495. [Online]. Available: https://doi.org/10.1007/978-981-10-0281-6_71

[14] J. Yang, Y. Sheng, and J. Wang, "A GBDT-paralleled quadratic ensemble learning for intrusion detection system," *IEEE Access*, vol. 8, pp. 175 467–175 482, 2020. [Online]. Available: https://doi.org/10.1109/ACCESS.2020.3026044

[15] H. V. Vo, H. N. Nguyen, T. N. Nguyen, and H. P. Du, "Sdaid: Towards a hybrid signature and deep analysis-based intrusion detection method," in *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*, 2022, pp. 2615–2620. [Online]. Available: https://doi.org/10.1109/GLOBECOM48099.2022.10001582

[16] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *CoRR*, vol. abs/1412.6572, 2014. [Online]. Available: https://doi.org/10.48550/arXiv.1412.6572

[17] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, "The limitations of deep learning in adversarial settings," in *2016 IEEE European Symposium on Security and Privacy (EuroS&P)*, 2016, pp. 372–387. [Online]. Available: https://doi.org/10.1109/EuroSP.2016.36

[18] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," *arXiv*, vol. abs/1706.06083, 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1706.06083

[19] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 IEEE Symposium on Security and Privacy (SP)*, 2017, pp. 39–57. [Online]. Available: https://doi.org/10.1109/SP.2017.49

[20] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "DeepFool: A simple and accurate method to fool deep neural networks," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2574–2582. [Online]. Available: https://doi.org/10.1109/CVPR.2016.282

[21] 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*, 2017, p. 15–26. [Online]. Available: https://doi.org/10.1145/3128572.3140448

[22] J. Chen, M. I. Jordan, and M. J. Wainwright, "HopSkipJumpAttack: A query-efficient decision-based attack," in *2020 IEEE Symposium on Security and Privacy (SP)*, 2020, pp. 1277–1294. [Online]. Available: https://doi.org/10.1109/SP40000.2020.00045

[23] H. Zhang, D. Han, S. Zhuang, Z. Wang, J. Sun, Y. Liu, J. Liu, and J. Dong, "Explainable and transferable adversarial attack for ML-based network intrusion detectors," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–18, 2025. [Online]. Available: https://doi.org/10.1109/TDSC.2025.3560486

[24] S. Ennaji, E. Benkhelifa, and L. V. Mancini, "Toward realistic adversarial attacks in IDS: A novel feasibility metric for transferability," in *arXiv*, 2025. [Online]. Available: https://doi.org/10.48550/arXiv.2504.08480

[25] A. Paya, S. Arroni, V. García-Díaz, and A. Gómez, "Apollon: A robust defense system against adversarial machine learning attacks in intrusion detection systems," *Applied Soft Computing*, vol. 136, p. 103546, 2024. [Online]. Available: https://doi.org/10.1016/j.cose.2023.103546

[26] X. Zhao, K. W. Fok, and V. L. Thing, "Enhancing network intrusion detection performance using generative adversarial networks," *Computers and Security*, vol. 145, p. 104005, 2024. [Online]. Available: https://doi.org/10.1016/j.cose.2024.104005

[27] M. K. Roshan and A. Zafar, "Boosting robustness of network intrusion detection systems: A novel two phase defense strategy against untargeted white-box optimization adversarial attack," *Expert Systems with Applications*, vol. 249, p. 123567, 2024. [Online]. Available: https://doi.org/10.1016/j.eswa.2024.123567

[28] A. Sarıkaya, B. G. Kılıç, and M. Demirci, "RAIDS: Robust autoencoder-based intrusion detection system model against adversarial attacks," *Computers and Security*, vol. 135, p. 103483, 2023. [Online]. Available: https://doi.org/10.1016/j.cose.2023.103483

[29] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," *arXiv*, vol. abs/1701.07875, 2017. [Online]. Available: https://doi.org/10.48550/arXiv.1701.07875

[30] W. Xu, D. Evans, and Y. Qi, "Feature squeezing: Detecting adversarial examples in deep neural networks," in *Proceedings 2018 Network and Distributed System Security Symposium*, 2018. [Online]. Available: https://doi.org/10.14722/ndss.2018.23198

[31] I. Sharafaldin, A. Habibi Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proceedings of the 4th International Conference on Information Systems Security and Privacy - ICISSP*, 2018, pp. 108–116. [Online]. Available: https://doi.org/10.5220/0006639801080116

[32] H. V. Le, H. P. Du, H. N. Nguyen, C. N. Nguyen, and L. V. Hoang, "A proactive method of the webshell detection and prevention based on deep traffic analysis," *International Journal of Web and Grid Services*, vol. 18, no. 4, p. 361–383, jan 2022. [Online]. Available: https://doi.org/10.1504/ijwgs.2022.126117

[33] J. Wang, J. Pan, I. AlQerm, and Y. Liu, "Def-IDS: An ensemble defense mechanism against adversarial attacks for deep learning-based Network intrusion detection," in *2021 International Conference on Computer Communications and Networks (ICCCN)*, 2021, pp. 1–9. [Online]. Available: https://doi.org/10.1109/ICCCN52240.2021.9522215

[34] R. Shu, T. Xia, L. Williams, and T. Menzies, "OMNI: Automated ensemble with unexpected models against adversarial evasion attack," *Empirical Software Engineering*, vol. 27, no. 1, p. 103644, 2022. [Online]. Available: https://doi.org/10.1007/s10664-021-10064-8

[35] X. Yuan, S. Han, W. Huang, H. Ye, X. Kong, and F. Zhang, "A simple framework to enhance the adversarial robustness of deep learning-based intrusion detection system," *Computers and Security*, vol. 137, p. 103644, 2024. [Online]. Available: https://doi.org/10.1016/j.cose.2023.103644

Received on May 15, 2025
Accepted on August 18, 2025