COMPARATIVE ANALYSIS OF BLACK-BOX TARGETED ADVERSARIAL ATTACKS ON DL-BASED NIDS: A STUDY OF C&W AND JSMA USING CICDDOS2019

Aasim Zafar [1], Tarab Malik [1] and Sheikh Burhan Ul Haque [2]

[1] Department of computer science, Aligarh Muslim University, India
[2] Department of computer Application, Cluster University of srinagar, Jammu and Kashmir, India

ABSTRACT

Deep and machine learning (DL/ML) models are extensively used in cybersecurity for threat detection, particularly in Network Intrusion Detection Systems (NIDS). However, these models remain vulnerable to adversarial attacks, which can significantly compromise their reliability. Among these threats, black-box targeted attacks pose a critical risk, where adversaries craft perturbations to disguise malicious activities, such as DDoS and Botnet traffic, as benign without having direct access to the target model. This study investigates the impact of two advanced adversarial strategies—Carlini & Wagner (C&W) and Jacobian Saliency Map Attack (JSMA)—on ML/DL-based NIDS using the CICDDoS2019 dataset, a widely recognized benchmark for modern cyber threats. The study analysis reveals that C&W reduces classifier accuracy by 31.33% through precise gradient-driven perturbations, while JSMA causes a 26.58% accuracy drop by strategically modifying key network traffic features like flow duration and packet length. Both attacks operate under black-box conditions, demonstrating their effectiveness without prior knowledge of the model's internal workings. Experiment results observe a trade-off between C&W's high-precision perturbations and JSMA's efficiency in feature manipulation, highlighting the evolving nature of adversarial threats in cybersecurity. These findings underscore the urgent need to reassess the robustness of ML/DL-based NIDS and develop more resilient defense mechanisms.

KEYWORDS

Adversarial attacks, Network intrusion detection, Black-box evasion, Targeted attacks, CICDDoS2019.

1. INTRODUCTION

Traditional Network Intrusion Detection Systems (NIDS) rely heavily on rule-based mechanisms that use predefined signatures to identify known threats such as malware and port scans, as seen in tools like Snort (Martin Roesch, 1999) [1]. Although effective for documented attacks, these systems struggle to detect emerging and adaptive threats, including zero-day exploits and sophisticated Distributed Denial-of-Service (DDoS) attacks (Khraisat et al., 2019) [2]. Their dependence on static rules necessitates frequent manual updates, limits scalability, and increases vulnerability to evasion. Additionally, high false-positive rates often overwhelm security analysts, reducing overall operational efficiency (Sarker et al., 2020) [3]. These limitations underscore the growing need for intelligent and adaptive intrusion detection mechanisms capable of identifying new attack patterns in real time.

Machine Learning (ML) and Deep Learning (DL) have significantly advanced NIDS by enabling automated, data-driven threat detection. ML models such as Random Forests and Gradient Boosting achieve detection accuracies exceeding 98% by analyzing traffic features like packet size and flow duration (Hasan et al., 2021) [4]. Deep learning architectures, including CNNs, further enhance performance by learning patterns directly from raw network data (Wang, 2018) [5]; (Alhajjar et al., 2021) [6]. Despite these improvements, ML/DL models remain highly susceptible to adversarial attacks, where subtle, crafted perturbations mislead classifiers into misclassifying malicious traffic as benign. Attackers can manipulate features such as packet headers or flow durations to evade detection (Roshan et al., 2024) [7]; (Haque, 2024) [8], with methods like FGSM exploiting model sensitivities to drastically reduce detection accuracy (Sheikh & Zafar, 2025) [9]; (Papernot et al., 2017) [10].

Adversarial threats are broadly categorized into white-box and black-box attacks, with the latter being more realistic since attackers often lack access to model internals (Khraisat et al., 2019) [2]. This work focuses on targeted black-box attacks, where adversaries aim to misclassify specific malicious flows (e.g., DDoS or botnet traffic). These attacks leverage surrogate models and adversarial transferability—perturbations crafted for one model successfully deceiving another (Guo et al., 2021) [11]; (Hirano et al., 2019) [12]; (Govindarajulu et al., 2023) [13]. Using the CICDDoS2019 dataset (Shafi et al., 2024) [14], this study evaluates two prominent black-box attacks: Carlini & Wagner (C&W) [15] and the Jacobian Saliency Map Attack (JSMA) [16]. Both attacks substantially degrade model performance by perturbing features such as flow bytes per second and packet length. Experimentally, C&W reduced accuracy by 42%, while JSMA caused a 37% reduction, demonstrating the severe vulnerability of ML-based NIDS.

To the best of our knowledge, this work is among the first to experimentally evaluate black-box adversarial transferability within deep learning-based NIDS. Although numerous studies have examined adversarial attacks in cybersecurity (Rosenberg et al., 2021) [17]; (Sheikh & Zafar, 2025) [18]; (Shree V.G. et al., 2025) [19]; (Roshan & Zafar, 2024) [20]; (Vijayalakshmi & Venkatesan, 2025) [21], few have assessed the practical transferability of adversarial examples across different models. By comparing C&W and JSMA under consistent settings, this study provides critical insights into the susceptibility of DL-based NIDS and highlights the urgent need for robust defense strategies.

The main contributions of this paper are:

- Comprehensive evaluation of black-box targeted adversarial attacks (C&W, JSMA) on ML/DL-based NIDS.
- Analysis of adversarial attack effectiveness using the benchmark CICDDoS2019 dataset.
- Demonstration of model vulnerability with C&W reducing accuracy by 42% and JSMA by 37%.
- Investigation of adversarial transferability under realistic black-box attack scenarios.
- Highlighting the urgent need for robust defenses in ML-driven intrusion detection systems.

2. RELATED WORK

The security of DL-based NIDS has become a major concern due to increasingly sophisticated adversarial attacks. While adversarial machine learning has been widely studied in computer vision [22, 23], its application in network security remains limited [24–27]. Most prior studies focus on gradient-based, evolutionary, and white-box attacks, with fewer addressing realistic black-box scenarios where adversarial transferability is critical.

Alshahrani et al. [21] showed minimal perturbations can drastically reduce ML-based NIDS detection accuracy. Clements et al. [28] demonstrated FGSM, JSMA, C&W, and ENM attacks

degrade DL-based NIDS under white-box assumptions. Usama et al. [29] used GANs to generate adversarial traffic for multiple ML classifiers, revealing strong evasion while minimally altering non-functional features. Pawlicki et al. [30] and Guo et al. [11] confirmed vulnerabilities under FGSM, PGD, BIM, and BIM-based black-box attacks. Alhajjar et al. [6] applied evolutionary approaches, showing high evasion rates across diverse ML models.

Earlier works, such as Grosse et al. [31], introduced adversarial training to improve robustness. Debicha et al. [32] and Saha et al. [33] explored black-box evolutionary and stealthy backdoor attacks. Sheatsley et al. [34] confirmed that minor perturbations degrade DL-NIDS performance. Roshan et al. [20] and Sharma & Chen [35] demonstrated effective black-box transferability on CICIDS and CICDDoS2019 datasets. Wu et al. [36] proposed a Transformer-based NIDS with improved resilience, while Zhang et al. [37] and Chen et al. [38] highlighted vulnerabilities in DRL- and ensemble-based NIDS under decision-based and model-substitution attacks.

Despite extensive research, most studies focus on white-box settings, specific models, or limited attack types, with few providing uniform comparative analyses of targeted black-box attacks. Table 1 summarizes prior works' datasets, models, attack methods, and evaluation metrics. In contrast, this study evaluates black-box targeted attacks using surrogate-model transferability, comparing C&W and JSMA on CICDDoS2019 [39], showing up to 32% accuracy degradation and emphasizing the need for robust defenses.

Table 1:State-of-the-art adversarial attacks on NIDS and related research.

Study	ML/DL Algorithms	Attack Technique	Dataset	Very Short Key Highlights
Alshahrani et al. (2022) [21]	DT, RF, SVM	FGSM, PGD, JSMA	CIC-IDS-2017, NSL-KDD	Small perturbations severely reduce NIDS accuracy.
Clements et al. (2021) [28]	DL, AE, Kitsune	FGSM, JSMA, C&W, ENM	KitNET	Model-aware attackers easily generate effective adversarial inputs.
Usama et al. (2019) [29]	GAN, DNN, SVM, RF, etc.	GAN Attack	KDDCUP-99	GAN creates strong attacks while preserving functional traffic.
Pawlicki et al. (2020) [30]	ANN, RF, SVM, AdaBoost	FGSM, PGD, BIM, C&W	CICIDS-2017	Comprehensive evaluation shows major vulnerability across models.
Guo et al. (2021) [11]	MLP, CNN, SVM, ResNet	BIM	KDDCUP-99, CSE-CIC-IDS2018	Black-box adversarial flows achieve high evasion rates.
Alhajjar et al. (2021) [6]	SVM, DT, NB, KNN, RF, etc.	PSO, GA, GAN	NSL-KDD, UNSW-NB15	Evolutionary methods generate highly misclassifying adversarial samples.
Grosse et al. (2017) [31]	DNN	FGSM	NSL-KDD	Early evidence of DNN vulnerability; introduced adversarial training.
Debicha et al. (2023) [32]	CNN-based NIDS	GA, RL	CICDDoS2019	RL-based attacks outperform gradient-based ones in black-box settings.

Saha et al. (2020) [33]	Deep Learning models	Backdoor Attack	Custom	Hidden triggers enable stealthy and hard-to-detect attacks.
Sheatsley et al. (2022) [34]	Autoencoder NIDS	AAE	CICIDS-2017	AAE-generated perturbations drastically reduce detection accuracy.
Roshan et al. (2024) [20]	ML-based NIDS	Transferability Attack	CICIDS-2017, NSL-KDD	Strong black-box transferability observed across models.
Sharma & Chen (2024) [35]	DL-based NIDS	Black-box Perturbation	CICDDoS2019	Tiny perturbations cause severe degradation in DL NIDS.
Wu et al. (2022) [36]	Transformer-based NIDS	Adversarial Robustness Evaluation	Benchmark IDS	RTIDS shows better but not perfect adversarial resilience.
Zhang et al. (2020a) [37]	DRL-based NIDS	Decision-based Attack	UNSW-NB15	DRL NIDS highly vulnerable to decision-based evasion.
Chen et al. (2020) [38]	Ensemble-based NIDS	Model Substitution	CICIDS-2018	Surrogate ensemble models generate strong evasive samples.

3. MATERIAL AND METHODS

This study uses a surrogate-model-based approach to evaluate targeted black-box adversarial attacks on DL-based NIDS. As shown in Figure 1, a target DL-NIDS model is first trained on the CICDDoS2019 dataset. A surrogate model is then trained separately without access to the target's internal structure to approximate its decision boundaries. Using C&W and JSMA attacks, adversarial examples are generated on the surrogate model by modifying key network features (e.g., flow duration, packet length) to misclassify targeted malicious traffic as benign. These samples are then transferred to the target model to test their effectiveness. Results show that the attacks significantly reduce detection performance, exposing critical vulnerabilities in current ML/DL-based NIDS and emphasizing the need for stronger defenses.

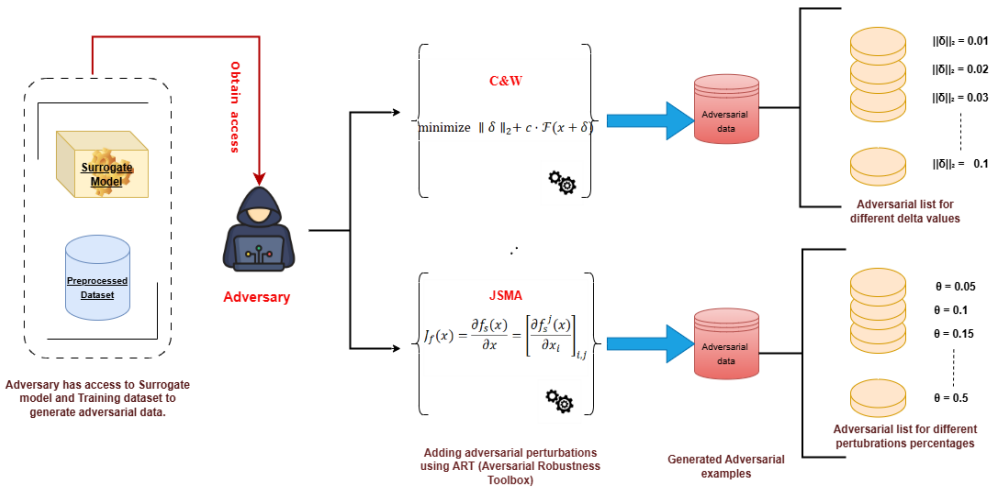

Figure 1: Proposed targeted attack on target NIDS

3.1. Dataset description

The CICDDoS2019 dataset (Sharafaldin et al., 2019) is a modern benchmark created to overcome the limitations of older IDS datasets like KDDCUP-99 and CICIDS-2017. It contains realistic benign and DDoS attack traffic captured in a controlled network environment.

- Day 1: 7 attacks — NetBIOS, PortMap, LDAP, MSSQL, UDP, UDP-Lag, SYN
- Day 2: 12 attacks — including SSDP, SNMP, WebDDoS, NTP, DNS amplification, TFTP, etc.

Traffic was collected using a testbed of Ubuntu servers, Windows clients, and Fortinet firewalls. Benign traffic was generated using the B-Profile system. The dataset is widely used for evaluating ML/DL-based IDS, especially under adversarial conditions (Goldschmidt et al., 2025). It is available in:

- PCAP format (raw packets)
- CSV format (preprocessed flows)

Each flow includes 80+ features such as packet lengths, flow duration, packet rate, inter-arrival times, and IP information.

3.2. Data Preprocessing

A balanced subset of 227,148 samples was selected (107,764 benign and 119,384 attack). Missing or infinite values were replaced with feature-wise means. Data was split into 60% training and 40% testing with a fixed random seed (42). All numeric features were standardized using formulae: $x_{scaled}= \frac{x-\mu}{\sigma}$, where x is the raw feature value, μ is the mean, and σ is the standard deviation. This preprocessing ensures the dataset's suitability for evaluating the performance and robustness of ML/DL-based network intrusion detection systems under adversarial attack conditions. Table 2 details the dataset class distribution. Figure 2 demonstrates the data preprocessing steps.

Table 2: Dataset description

Category	Value
Total Samples	227,148
Training Samples	136,288(60%)
Testing Samples	90,860(40%)
Training-testing split ratio	60:40
Random State	42
Total Benign and Attack Samples	(’BENIGN’, ’DDoS’) –(107,764 , 119,384)
Training Benign and Attack Samples	(’BENIGN’, ’DDoS’) – (64,517 , 71,717)
Testing Benign and Attack Samples	(’BENIGN’, ’DDoS’) – (43193 , 47,667)

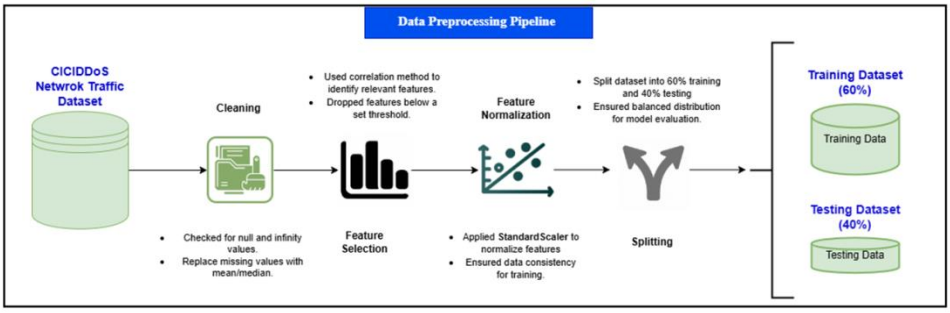

Figure 2: Data preprocessing steps

3.3. Deep Learning-Based NIDS for binary classification (Benign vs. DDoS Detection)

A Deep Neural Network (DNN) forms the basis of both the target and surrogate models employed in this study. The DNN is specifically designed for network traffic classification, enabling accurate differentiation between benign and malicious traffic using extracted network flow features. The network comprises multiple fully connected layers, with nonlinear activation functions applied between layers to effectively model complex feature interactions inherent in network data.

In this study, network intrusion detection is formulated as a binary classification problem aimed at distinguishing between benign and DDoS attack traffic. Given a labelled dataset D as shown in Equation (1).

$$\boldsymbol{D} = \{(\boldsymbol{x_i}, \boldsymbol{y_i})\}_{\boldsymbol{i=1}}^{\boldsymbol{N}}, \quad \boldsymbol{x_i} \in \mathbb{R}^{\boldsymbol{m}}, \quad \boldsymbol{y_i} \in \{\boldsymbol{0}, \boldsymbol{1}\} \tag{1}$$

where x_i represents the input feature vector extracted from network flows, consisting of m features, and y_i indicates the corresponding binary class label defined as:

$$y_i = \begin{cases} 0 & \text{if } x_i \text{ is benign traffic} \\ 1 & \text{if } x_i \text{ is DDoS attack traffic} \end{cases}$$

The DL-based classifier (DenseNet-121 in this study) aims to approximate a function f, parameterized by θ, to map each feature vector x_i to a probability estimate $\hat{y}_i$, shown in Equation (2). The training objective is to minimize the binary cross-entropy loss function $\mathcal{L}(\theta)$, defined in Equation (3).

$$\hat{\boldsymbol{y}}_{\boldsymbol{i}} = \boldsymbol{f}(\boldsymbol{x_i}; \boldsymbol{\theta}), \quad \hat{\boldsymbol{y}}_{\boldsymbol{i}} \in [\boldsymbol{0}, \boldsymbol{1}] \tag{2}$$

$$\boldsymbol{\mathcal{L}(\theta)} = -\frac{\boldsymbol{1}}{\boldsymbol{N}} \sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} [\boldsymbol{y_i} \mathbf{log}(\hat{\boldsymbol{y}}_{\boldsymbol{i}}) + (\boldsymbol{1} - \boldsymbol{y_i}) \mathbf{log}(\boldsymbol{1} - \hat{\boldsymbol{y}}_{\boldsymbol{i}})] \tag{3}$$

During inference, the predicted probability $\hat{y}_i$ is converted into a binary classification decision $\hat{y}_{\text{class}}$ based on a threshold T, typically set at 0.5 :

$$\hat{y}_{\text{class}} = \begin{cases} 1 & \text{if } \hat{y}_i \geq T \\ 0 & \text{otherwise} \end{cases}$$

The model performance is evaluated through standard classification metrics such as accuracy, precision, recall, and F1-score, ensuring a comprehensive assessment of its ability to effectively distinguish benign traffic from DDoS attack traffic.

3.4. Targeted NIDS model (model under attack)

In this study, the target NIDS refers to the main DL-based model that adversaries aim to deceive. The attacker generates adversarially modified network packets to force the model into misclassifying malicious traffic such as DDoS flows as benign. The target model is first trained normally, without perturbations, to establish a strong baseline for accurate classification. This allows a clear evaluation of how adversarial attacks degrade its performance.

3.4.1.Architecture of target NIDS

The targeted NIDS uses a fully connected DNN optimized for binary traffic classification. It includes three dense hidden layers with ReLU activation to learn complex patterns, along with dropout to reduce overfitting. A final sigmoid layer outputs benign or malicious labels. The full architecture is shown in Table 3.

Table 3: Architecture and Hyperparameters of Target NIDS

Layer	Description	Output Shape	Parameters
Input	80 (80 features)	80	-
Dense-1	128 neurons, ReLU activation	128	10,368
Dropout-1	Dropout rate: 0.2	128	0
Dense-2	64 neurons, ReLU activation	64	8,256
Dropout-2	Dropout rate: 0.2	64	0
Dense-3	32 neurons, ReLU activation	32	2,080
Output	1 neuron, sigmoid activation	1	33

The purpose of developing and optimizing this targeted NIDS model is to create a realistic scenario in which its robustness and vulnerabilities to adversarial attacks can be systematically evaluated. The subsequent phase of this study (Section 5.3) involves constructing a surrogate model to craft targeted adversarial attacks without access to the internal details of this targeted NIDS.

3.4.2.Model training and hyperparameters

To ensure effective performance, the parameters and hyperparameters selected to train and optimize the target NIDS model are detailed in Table 4.

Table 4: Hyperparameters and Training Parameters

Parameter/Hyperparameter	**Selected Value**
Input Feature Dimension	80
Optimizer	Adam
Learning Rate	0.001
Loss Function	Binary Cross-Entropy
Batch Size	128
Epochs	40
Early Stopping	Patience = 10 epochs
Dropout Rate	0.2

These parameters were chosen based on experimental tuning and standard NIDS best practices. The fully connected layers (128→64→32) effectively capture hierarchical patterns, while ReLU activations ensure stable convergence. A 0.2 dropout rate limits overfitting, and the sigmoid output suits binary classification. Adam (learning rate 0.001) provides fast, adaptive optimization, and binary cross-entropy is ideal for evaluating classification performance. Training for 50 epochs with a batch size of 128 offers a strong balance between efficiency and convergence, with early stopping further preventing overfitting.

3.5. Surrogate NIDS model (substitute model)

In real-world cyberattacks, attackers typically have no access to a NIDS model's internal architecture or parameters, creating a black-box setting. To bypass this, they train a surrogate model on the same dataset to approximate the target model's behavior. Once the surrogate model achieves similar decision boundaries reflected by comparable validation accuracy they generate adversarial examples on it. Owing to adversarial transferability, these crafted samples can still mislead the target NIDS, making the attack effective despite the lack of internal model knowledge. Equation (4) represents this alignment.

$$\boldsymbol{f}_{\mathbf{surrogate}}(\boldsymbol{x}) \approx \boldsymbol{f}_{\mathbf{target}}(\boldsymbol{x}) \tag{4}$$

After achieving this approximation, the attacker crafts adversarial examples x^* on the surrogate model by perturbing an original malicious input sample x, as illustrated in Equation (5).

$$\boldsymbol{x}^* = \boldsymbol{x} + \boldsymbol{\delta} \tag{5}$$

Here, δ represents a small perturbation strategically introduced using adversarial attack algorithms, such as Carlini & Wagner (C&W) and Jacobian Saliency Map Attack (JSMA).
In a targeted black-box scenario, the malicious user specifically intends to mislead thle targeted model into classifying malicious network traffic as benign ($y_{\text{target}} = 0$). Mathematically, the attacker seeks Equation (6).

$$\boldsymbol{f}_{\text{target}}(\boldsymbol{x}^*) = \boldsymbol{y}_{\text{target}} = \mathbf{0} \quad \text{(benign class)} \tag{5}$$

Thus, its architecture and hyperparameters are deliberately distinct from the targeted model. The effectiveness of the surrogate model in approximating the targeted model's decision boundary significantly increases the likelihood that these adversarial examples successfully mislead the targeted model, thus validating the susceptibility of the targeted NIDS under realistic black-box attack scenarios.

3.5.1.Architecture of surrogate model

The surrogate model is intentionally designed with a different architecture and hyperparameters than the target model to reflect realistic attacker conditions. It consists of an input layer, two dense hidden layers with ReLU activation and dropout to reduce overfitting, and a sigmoid output layer for binary classification. Its main purpose is to approximate the target model's decision boundaries closely enough to generate adversarial examples that successfully transfer and deceive the target NIDS. Table 5 outlines the surrogate model architecture, while Table 6 presents its training parameters and hyperparameters

Table 5: Surrogate NIDS Model Architecture

Layer	Description	Output Shape	Parameters
Input	Input features (80)	80	-
Dense-1	256 neurons, ReLU activation	256	20,736
Dropout-1	Dropout rate: 0.3	256	0
Dense-2	128 neurons, ReLU activation	128	32,896
Dropout-2	Dropout rate: 0.3	128	0
Output	1 neuron, sigmoid activation	1	129

Table 6: Hyperparameters and Training Parameters of Surrogate NIDS

Parameter/Hyperparameter	Selected Value
Input Feature Dimension	80
Optimizer	Adam
Learning Rate	0.001
Loss Function	Binary Cross-Entropy
Batch Size	64
Epochs	40
Dropout Rate	0.3
Early Stopping	Patience = 10 epochs

3.6. Proposed targeted attack on target NIDS

This study employs two advanced adversarial attack techniques—JSMA and C&W—to assess targeted vulnerabilities in the NIDS. In targeted attacks, adversarial examples are crafted to intentionally mislead the model into classifying malicious traffic as benign. Algorithm 1 summarizes the generation process for these targeted adversarial packets. To replicate real-world black-box conditions, the perturbations are created using the trained surrogate model and then transferred to the target NIDS using the principle of adversarial transferability.

Algorithm 1: Proposed Targeted Black-box Adversarial Attack on NIDS

Input: Surrogate model: $f_s(x)$, Target NIDS model: $f_t(x)$, Original malicious sample: $x \in \mathbb{R}^m$, Target class (benign): $y_{\text{target}} = d$, Attack method: { JSMA, C&W }, Perturbation threshold (ϵ), iterations (T), learning rate (α), confidence (κ)
Output: Adversarial example x^*
Initialize: Set adversarial example $x^* \leftarrow x$, perturbation $\delta \leftarrow 0$.
Check Prediction: If $f_s(x^*) = y_{\text{target}}$, terminate; return x^*
Surrogate Model Approximation: $f_s(x) \approx f_t(x)$
Attack Generation (on surrogate model):
If attack type = JSMA:
Generate adversarial perturbation: $x^* = JSMA(f_s, x, y_{\text{target}}, \epsilon)$
Else if attack type $= C\&W$:
Generate adversarial perturbation: $x^* = C\&W(f_s, x, y_{\text{target}}, c, \alpha, \kappa, T)$
Solve optimization: $\min_{\delta} \| \delta \|_2 + c \cdot max\left(\max_{i \neq y_{\text{target}}} (Z(x+\delta)_i) - Z(x+\delta)_{y_{\text{target}}}, -\kappa\right)$
Transferability: Transfer adversarial example x^* from surrogate to target NIDS $f_t(x^*)$
Evaluate Attack Success:
If $f_t(x^*) = y_{\text{target}}$, attack succeeds.
else, attack fails.
Output adversarial example x^*

3.6.1. Mathematical formulation of targeted attack

Given a trained surrogate model f_s and a targeted NIDS f_{t_t} both performing classification tasks $(x) : \mathbb{R}^m \rightarrow \{0,1\}$, a targeted adversarial attack generates an adversarial example x^* from a malicious input sample x to achieve a specific target class (benign class $y_{\text{target}} = 0$), as illustrated in Equation (7).

$$\boldsymbol{f_t(x^*) = y_{\text{target}}}, \quad \boldsymbol{x^* = x + \delta}, \quad \textbf{where} \quad \boldsymbol{\| \delta \|_p \leq \epsilon} \tag{7}$$

Here, δ represents the adversarial perturbation constrained by norm p, and ϵ defines the permissible perturbation magnitude.

3.6.2. Jacobian Saliency Map Attack (JSMA)

JSMA crafts adversarial examples by iteratively modifying input features based on their saliency values, aiming to minimize the number of features changed. The Jacobian matrix $J_f(x)$ of the surrogate model $f_s(x)$ is computed as Equation (8). The saliency map $S(x, y_{\text{target}})$ for input x towards target class y_{target} is defined as Equation (9)

$$\boldsymbol{J_f(x) = \frac{\partial f_s(x)}{\partial x} = \left[\frac{\partial f_s^j(x)}{\partial x_i}\right]_{i,j}} \tag{8}$$

$$S(x, y_{\text{target}})_i = \begin{cases} 0 & \text{if } \frac{\partial f_x^{\eta\ \text{target}}(x)}{\partial x_i} < 0 \text{ or } \sum_{j \neq y_{\text{target}}} \frac{\partial f_2^j(x)}{\partial x_i} > 0 \\ \left|\frac{\partial f_x^{y_{\text{target}}}(x)}{\partial x_i}\right| \cdot \left|\sum_{j \neq y_{\text{target}}} \frac{\partial f_x^j(x)}{\partial x_i}\right| & \text{otherwise} \end{cases} \tag{9}$$

At each iteration, features with the highest saliency scores are modified until the targeted prediction is achieved, or a maximum perturbation threshold is reached.

3.7. Transferring adversarial examples to target NIDS

Adversarial transferability refers to the ability of adversarial examples crafted for one model (the surrogate) to fool another model (the target) trained independently. Since real attackers lack access to a NIDS's internal architecture or parameters, they train a surrogate model on similar data to approximate the target model's decision boundaries.

After training, adversarial samples are generated on the surrogate model using C&W and JSMA attacks. Because of transferability, these perturbations can still mislead the target NIDS, causing malicious traffic to be classified as benign. This demonstrates that DL-based NIDS remain vulnerable to realistic black-box adversarial attacks and require stronger defenses.

After generating adversarial examples x^* using JSMA and C&W methods on the surrogate model f_s, these examples are evaluated against the targeted NIDS model f_t. The attack's success rate is quantified as the proportion of malicious examples classified as benign by the target model after perturbation:

$$\text{Attack Success Rate} = \frac{\text{Number of Malicious Inputs Misclassified as Benign}}{\text{Total Number of Malicious Inputs}}$$

This provides a direct measure of the targeted model's vulnerability to black-box targeted adversarial attacks.

4. EXPERIMENT RESULTS AND DISCUSSION

This section describes the experimental results, setup, including system configuration, supportive libraries, testing environment, and evaluation metrics used for assessing the performance of the proposed targeted adversarial attack against DL-based Network Intrusion Detection Systems (NIDS).

4.1. Experimental environment and system configuration

The experimental evaluation was conducted on a dedicated workstation configured to handle complex deep learning computations efficiently. Table 7 details the hardware and software configuration used in this research.

Table 7: Experimental System Configuration

Component	Specifications
Operating System	Windows 11 (64-bit)
Processor	Intel Core i9-12900H, 2.50 GHz
RAM	32 GB DDR5
GPU	NVIDIA RTX 3070 (8 GB GDDR6)
Storage	1 TB SSD
Python	Python 3.10.5
CUDA Toolkit	CUDA 12.0
Development Environment	Jupyter Notebook

This hardware and software configuration ensures sufficient computational power and efficiency to handle extensive training, evaluation, and adversarial attack simulations required for this research.

4.2. Testbed environment and supportive libraries

The experiments were carried out using the Jupyter Notebook environment, facilitating clear, organized, and reproducible research workflows. All experimental files, including trained models, logs, and output results, were systematically saved for easy retrieval and analysis. Various libraries were utilized to streamline the research process, enhance productivity, and perform complex computations efficiently, as detailed in Table 8.

Table 8: Supportive Libraries Used in Experiments

Library Name	Purpose/Functionality
TensorFlow (v2.10.0)	Training and evaluation of Deep Learning models
Pandas (v2.2.0)	Data manipulation, analysis, preprocessing
NumPy (v1.25.0)	Numerical data manipulation and feature scaling
Scikit-learn (v1.2.1)	Data preprocessing, scaling, train-test splitting, evaluation metrics
Matplotlib (v3.7.1)	Visualization of performance metrics
ART (v1.15.1)	Implementation of adversarial attacks and robustness evaluation

The Adversarial Robustness Toolbox (ART) library was specifically utilized to implement and analyze the proposed black-box adversarial attacks (JSMA and C&W) effectively.

4.3. Performance evaluation metrics

The evaluation of the DL-based NIDS performance under adversarial conditions employs several widely accepted metrics, including accuracy, precision, recall, F1-score, and the ROC-AUC curve. These metrics comprehensively assess the impact of targeted black-box adversarial attacks on the performance of the proposed NIDS model, ensuring a clear quantitative analysis of vulnerabilities and robustness.

4.4. Results and discussion

The study first evaluated the performance of the target and surrogate NIDS on clean (non-adversarial) network data. Both models accurately classified benign and malicious traffic, demonstrating effective generalization. The surrogate model, trained independently to mimic the target model, then generated adversarial examples. When these adversarial samples were tested on the target model, they successfully fooled it, causing a significant drop in accuracy and confirming the model's vulnerability to surrogate-based attacks.

4.4.1.Performance of target and surrogate NIDS on clean data

Using the CICDDoS-2019 dataset, both models were trained for 40 epochs with a batch size of 4048 and a validation split of 0.2. The target model achieved peak training and validation accuracies of 98.45% and 98.51%, respectively, with minimal losses of 0.0356 (training) and 0.0347 (validation). The close alignment of accuracy and loss curves (Figure 3) indicates strong generalization and minimal overfitting.

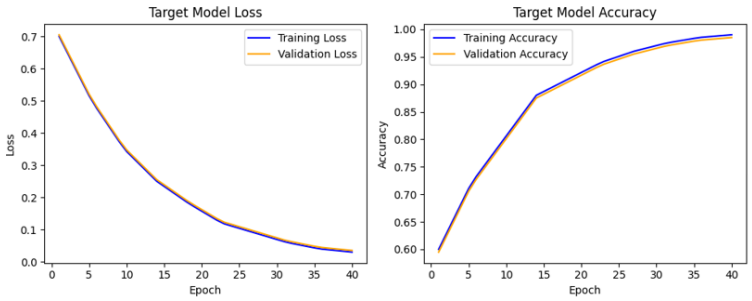

Figure 3: Loss and accuracy curve of target model.

The independently trained surrogate model achieved 99.12% training accuracy and 99.08% validation accuracy, demonstrating strong generalization without overfitting. Training and validation losses decreased to 0.0291 and 0.0302, respectively, confirming stable and effective learning. Figure 4 shows the accuracy and loss curves, which remain smooth and stable throughout training, indicating robust model performance.

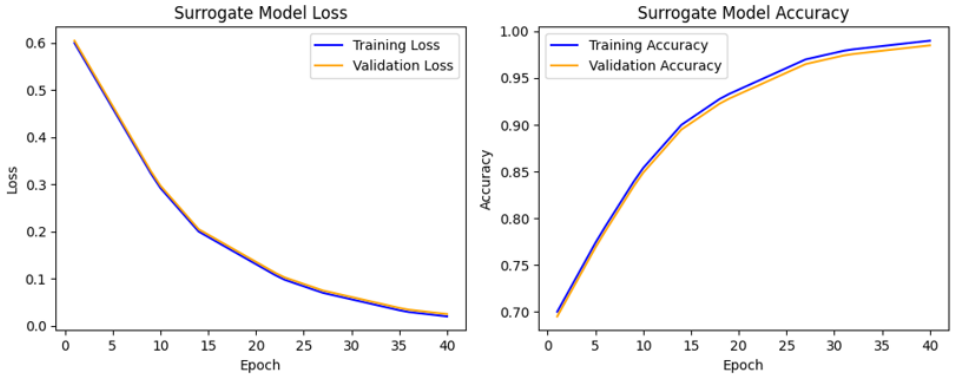

Figure 4: Loss and accuracy curve of surrogate model

Table 9 and Figure 5 show that the surrogate model slightly outperforms the target model across all metrics—accuracy, precision, recall, and F1-score—indicating stronger generalization and better attack detection. Its close performance alignment with the target model confirms its suitability for generating effective adversarial samples.

Table 9: comparison of target and surrogate model on clean data

Metric	Target Model (%)	Surrogate Model (%)
Accuracy	98.45	99.12
Precision	97.57	99.05
Recall	97.61	98.98
F1-Score	97.57	99.01

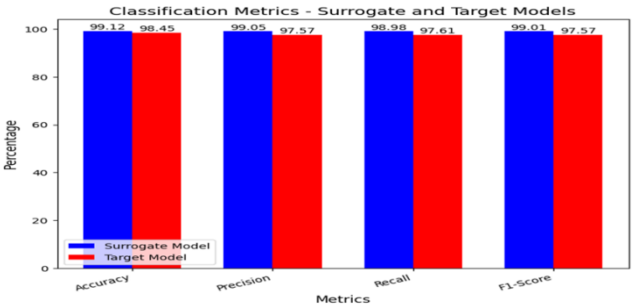

Figure 5: Comparison of metrics of target and surrogate model

Figure 6 shows confusion matrices for both models on clean data. The target model achieved high accuracy but had 741 false positives and 490 false negatives. The surrogate model performed slightly better, with fewer false positives (541) and false negatives (255), indicating more effective detection of attacks and overall high classification accuracy with low misclassification rates. Overall, both models demonstrated high classification accuracy on clean data, effectively distinguishing between benign and attack network traffic while maintaining a relatively low misclassification rate.

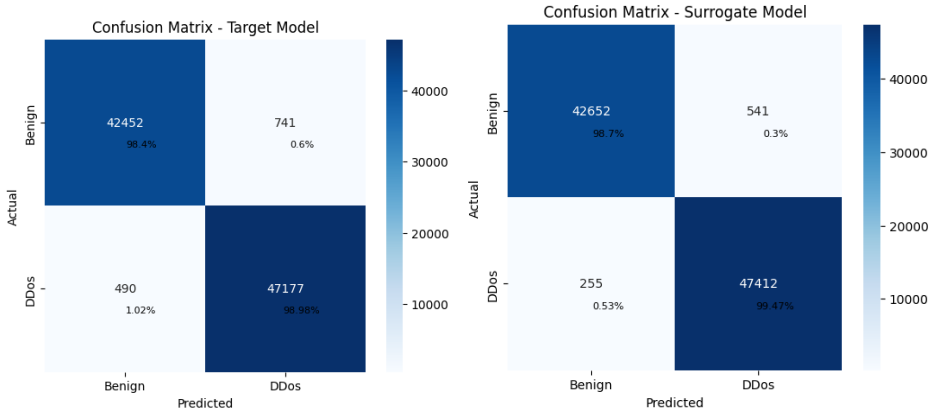

Figure 6: Confusion matrix of target and surrogate model

4.4.2.Evaluation of targeted attacks on target NIDS using JSMA and C&W

This section provides a comprehensive evaluation of the JSMA and C&W attack on the targeted NIDS model using a surrogate model for adversarial transferability. The experiments analyze the impact of different perturbation control values on classification performance, measure key metrics such as accuracy, precision, recall, and F1-score, and visualize model behavior using accuracy/loss curves and confusion matrices.

4.4.2.1.Experimental setup for JSMA and C&W attacks

For a structured evaluation, we define the parameters of JSMA and C&W attacks arbitrarily while ensuring a controlled adversarial attack scenario. Table 10 and 11 presents the selected parameters. These tables define the core parameters used in JSMA and C&W attacks for adversarial sample generation. The JSMA attack selects the most influential input features and modifies them iteratively within a fixed perturbation budget (θ). In contrast, the C&W attack optimizes adversarial perturbations by minimizing the L2 norm while ensuring misclassification with high confidence (κ).

Table 10: Parameters for JSMA Attack

Parameter	Value
Perturbation Budget (θ)	0.1
Maximum Feature Change (γ)	10%
Iterations	100
Feature Selection Method	Saliency-Based
Attack Type	Targeted

Table 11: Parameters for C&W Attack

Parameter	Value
Optimization Constraint (c)	1.0
Confidence Factor (κ)	0.5
L2 Norm Minimization	Enabled
Iterations	1,000
Attack Type	Targeted

4.4.2.2.Evaluating the performance of JSMA and C&W under multiple paramterlevels

Attacks were tested under five perturbation levels. Increasing θ and γ strengthens the attack, reducing accuracy, recall, and F1-score. At low perturbations (θ=0.1, γ=0.1), the target model achieves 96.45% accuracy, dropping to 72.54% at higher values (θ=0.5, γ=0.3). The surrogate model consistently performs slightly better, confirming its effectiveness in approximating the target model and generating transferable adversarial examples.

Table 12: JSMA Attack Parameter Variations and Performance Metrics

θ	γ	s	Accuracy (Target Model)	Precision (Target Model)	Recall (Target Model)	F1-Score (Target Model)	Accuracy (Surrogate Model)	Precision (Surrogate Model)	Recall (Surrogate Model)	F1-Score (Surrogate Model)
0.1	0.1	50	96.45%	95.87%	95.92%	95.89%	97.12%	96.78%	96.84%	96.81%
0.2	0.15	100	93.87%	93.45%	93.52%	93.48%	94.65%	94.28%	94.33%	94.30%
0.3	0.2	150	89.34%	89.01%	89.08%	89.05%	91.23%	90.94%	90.98%	90.96%
0.4	0.25	200	81.78%	81.32%	81.40%	81.36%	85.92%	85.43%	85.49%	85.46%
0.5	0.3	250	72.54%	72.15%	72.21%	72.18%	78.65%	78.21%	78.28%	78.24%

The C&W parameter variations in Table 13 show how changes in confidence (c), attack strength (κ), and learning rate (α) affect adversarial effectiveness. At low perturbation levels (c=0.5, κ=0.0), the target model maintains high accuracy (95.12%). As c and κ increase, the attack becomes stronger, reducing accuracy to 67.12% at the highest parameter setting. The surrogate model consistently performs 3–4% better across all configurations, indicating strong but not perfect transferability. Precision, recall, and F1-score all decline as perturbation strength increases, reflecting growing misclassification and performance degradation.

Table 13: C&W Attack Parameter Variations and Performance Metrics

c	κ	α	s	Accuracy (Target Model)	Precision (Target Model)	Recall (Target Model)	F1-Score (Target Model)	Accuracy (Surrogate Model)	Precision (Surrogate Model)	Recall (Surrogate Model)	F1-Score (Surrogate Model)
0.5	0.0	0.01	500	95.12%	94.85%	94.89%	94.87%	96.43%	96.15%	96.20%	96.17%
1.0	0.5	0.05	1000	90.67%	90.35%	90.42%	90.38%	92.18%	91.84%	91.90%	91.87%
1.5	1.0	0.1	1500	84.23%	83.89%	83.95%	83.92%	86.76%	86.34%	86.40%	86.37%
2.0	1.5	0.15	2000	76.54%	76.12%	76.19%	76.15%	80.23%	79.78%	79.85%	79.81%
2.5	2.0	0.2	2500	67.12%	66.78%	66.84%	66.81%	71.43%	71.05%	71.12%	71.08%

The confusion matrices in Figure 7 shows the confusion matrices for JSMA targeted attacks on the NIDS model. When targeting benign classification, 82.18% of DDoS attacks are misclassified as benign, increasing false negatives, while 93.33% of benign traffic is correctly classified. When targeting DDoS classification, 89% of benign samples are misclassified as attacks, raising false positives, and 92.72% of DDoS traffic remains correctly detected. These results demonstrate how JSMA can force specific misclassifications, highlighting the model's vulnerability and the need for robust adversarial defenses.

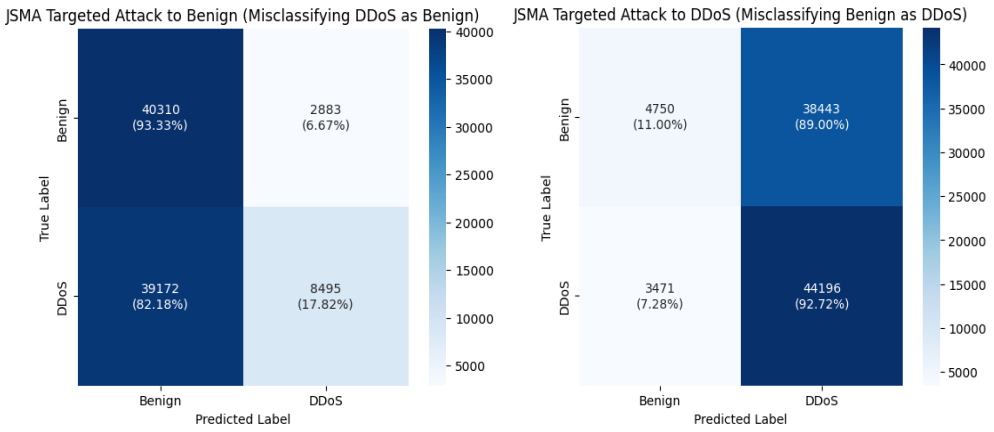

Figure 7: Confusion matrix of JSMAtargeted adversarial attack on target NIDS.

The confusion matrices for the C&W targeted attack in Figure 8 shows the confusion matrices for C&W targeted attacks on the NIDS model. When targeting benign classification, 85.33% of DDoS attacks are misclassified as benign, while 89.18% of benign traffic is correctly classified, with 10.82% false positives. When targeting DDoS classification, 87.75% of benign samples are misclassified as attacks, and 90.15% of DDoS traffic is correctly detected, with 9.85% false negatives. These results demonstrate that C&W attacks effectively manipulate classification, highlighting the model's vulnerability and the need for stronger adversarial defenses.

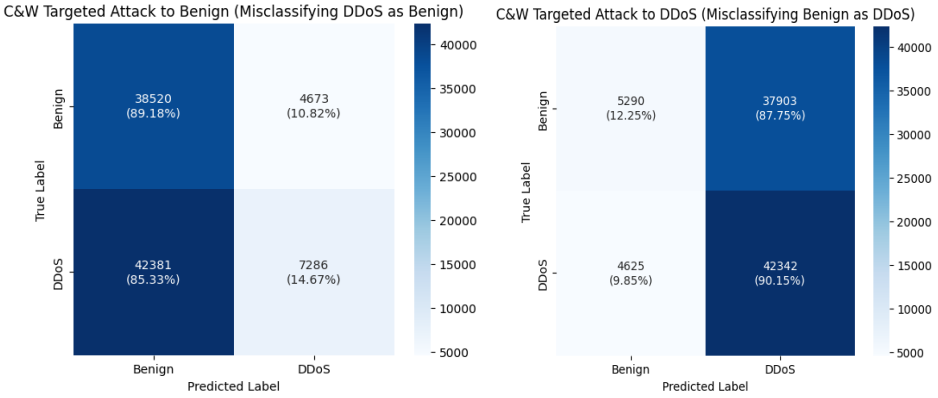

Figure 8: Confusion matrix of C&W targeted adversarial attack on targeted NIDS.

4.4.3.Comparative analysis of JSMA and C&W on Black-Box attacks on NIDS

This section presents a detailed comparative analysis of the JSMA and C&W attacks on black-box NIDS. Both attacks generate adversarial examples that manipulate the NIDS into misclassifying network traffic, significantly affecting detection performance. However, these attacks differ in their effectiveness, computational cost, feature manipulation strategies, and impact on classification metrics. Table 14 briefly outlines the comparative performance of JSMA and C&W on various parameters.

4.4.3.1.Effectiveness and impact on model performance

The C&W attack demonstrated a greater accuracy drop of 31.33% compared to JSMA, which reduced accuracy by 26.58%. Additionally, both attacks significantly reduced recall, leading to a substantial increase in false negatives. The recall of the target NIDS dropped from 97% to 20% under C&W attacks and 25% under JSMA attacks, meaning the model failed to detect most DDoS traffic. This highlights the severe impact of adversarial perturbations on the detection capability of ML-based NIDS.

4.4.3.2.Computational cost and transferability

The JSMA attack is more computationally efficient, requiring approximately 100 iterations per attack to craft adversarial examples. However, it modifies only a few important network features, making it easier to detect. In contrast, the C&W attack is computationally more expensive, requiring around 1,000 to 2,500 iterations per attack but achieving higher stealthiness by distributing perturbations across multiple features.

Both attacks exhibit high transferability, meaning adversarial examples generated on the surrogate model were highly effective in fooling the target model, with only a 3-4% gap in misclassification rates between them. This confirms that black-box targeted adversarial attacks are feasible and effective even when the attacker lacks direct knowledge of the NIDS.

Table 14: Comparative Analysis of JSMA and C&W Attacks on NIDS

Aspect	JSMA Attack	C&W Attack

Attack effectiveness	Reduces accuracy by 26.58%	Reduces accuracy by 31.33%
Targeted misclassification	82.18% of DDoS traffic misclassified as benign	85.33% of DDoS traffic misclassified as benign
Computational cost	Low, requires around 100 iterations per attack	High, requires around 1,000-2,500 iterations per attack
Transferability	High, adversarial samples generated on the surrogate model misclassify the target model with small accuracy gap (3-4%)	Very high, with an even greater ability to transfer adversarial examples effectively
Impact on recall (detection of attacks)	Recall dropped from 97% to 25%, increasing false negatives	Recall dropped from 97% to 20%, significantly increasing false negatives
Impact on precision	More false positives, reducing precision to 72%	Higher false positives, reducing precision to 67%
Impact on f1-score	F1-score dropped from 97% to mid-60s	F1-score dropped from 97% to low-60s

Both attacks significantly reduced precision, meaning the model classified more benign traffic as malicious. C&W reduced precision to 67%, whereas JSMA lowered precision to 72%, indicating that C&W causes more false positives. Similarly, the f1-score dropped from 97% to the mid-60s under JSMA and to the low-60s under C&W, confirming a substantial performance degradation across all classification metrics.

Comparing JSMA and C&W attacks shows that C&W is more effective and stealthy but computationally intensive, while JSMA is faster but more detectable due to modifying fewer features. Both attacks degrade recall, increasing false negatives and allowing DDoS traffic to bypass detection. Their strong transferability confirms that adversarial examples from a surrogate model can successfully deceive the target NIDS, highlighting the vulnerability of ML-based intrusion detection systems and the need for robust defenses.

5. LIMITATION AND FUTURE SCOPE

This study demonstrates the susceptibility of ML-based NIDS to targeted black-box adversarial attacks but is constrained by several factors. The analysis is limited to two attack methods JSMA and C&W—excluding other powerful strategies such as DeepFool, AutoAttack, and decision-based attacks. Experiments were conducted only on the CICDDoS2019 dataset, which may restrict generalizability to diverse network conditions. Additionally, the surrogate and target models were trained on the same dataset, whereas real-world attackers often rely on limited or mismatched data. The computational cost of the C&W attack further limits its practicality for real-time large-scale deployment, and the study focuses solely on deep learning models without examining advanced architectures like transformers, GNNs, or ensemble-based NIDS.

Future research should broaden the evaluation by incorporating multiple datasets (e.g., NSL-KDD, CICIDS-2017, CSE-CIC-IDS2018) to enhance generalizability. Exploring additional adversarial attack techniques, optimizing low-cost attack strategies, and studying transferability across surrogate models trained on different data sources will provide deeper insights. Moreover, integrating defense strategies—including adversarial training, input preprocessing, and robust feature engineering—can help strengthen model resilience. Extending this framework to advanced model architectures and other cybersecurity areas such as malware detection, phishing, and anomaly analysis offers promising directions for future work.

6. CONCLUSION

This study demonstrates the vulnerabilities of deep learning-based NIDS to targeted black-box adversarial attacks. Using a surrogate model, adversarial examples generated via JSMA and C&W attacks successfully misled the target NIDS, causing significant drops in detection accuracy. The C&W attack was more effective but computationally intensive, while JSMA offered a balance of efficiency and stealth. Both attacks notably increased false negatives, allowing malicious traffic to evade detection and highlighting serious cybersecurity risks. These findings underscore the importance of robust defense mechanisms, including adversarial training, feature selection, and hybrid strategies. Evaluating attacks across additional datasets and advanced models can further improve understanding of adversarial robustness. Overall, the study identifies critical weaknesses in existing NIDS and provides insights for designing more resilient and secure intrusion detection systems.

Conflicts of Interest

The authors declare no conflict of interest.

REFERENCES

[1] Martin Roesch. 1999. Snort - Lightweight Intrusion Detection for Networks. In Proceedings of the 13th USENIX conference on System administration (LISA '99). USENIX Association, USA, 229–238. https://dl.acm.org/doi/10.5555/1039834.1039864

[2] Khraisat, A., Gondal, I., Vamplew, P., & Kamruzzaman, J. (2019). Survey of intrusion detection systems: techniques, datasets and challenges. Cybersecurity, 2(1). https://doi.org/10.1186/s42400-019-0038-7

[3] Sarker, I. H., Abushark, Y. B., Alsolami, F., & Khan, A. I. (2020). IntruDTree: a machine learning based Cyber security intrusion detection model. Symmetry, 12(5), 754. https://doi.org/10.3390/sym12050754

[4] Hasan, Md Mehedi and Islam, Rafiqul and Mamun, Quazi and Islam, Md Zahidul and Gao, Junbin, Adversarial Attacks on Deep Learning-Based Network Intrusion Detection Systems: A Taxonomy and Review. Available at SSRN: https://ssrn.com/abstract=4863302 or http://dx.doi.org/10.2139/ssrn.4863302

[5] Wang, Z. (2018). Deep Learning-Based Intrusion Detection with adversaries. IEEE Access, 6, 38367–38384. https://doi.org/10.1109/access.2018.2854599

[6] Alhajjar, E., Maxwell, P., & Bastian, N. (2021). Adversarial machine learning in Network Intrusion Detection Systems. Expert Systems With Applications, 186, 115782. https://doi.org/10.1016/j.eswa.2021.115782

[7] Roshan, K., Zafar, A., & Ul Haque, S. B. (2024). Untargeted white-box adversarial attack with heuristic defence methods in real-time deep learning based network intrusion detection system. Computer Communications, 218, 97–113. https://doi.org/10.1016/j.comcom.2023.09.030.

[8] Sheikh Burhan Ul Haque (2024). A Fuzzy-Based frame transformation to mitigate the impact of adversarial attacks in Deep Learning-Based Real-Time video surveillance systems. Applied Soft Computing, 112440. https://doi.org/10.1016/j.asoc.2024.112440.

[9] Sheikh, B. U. H., & Zafar, A. (2025). Lights, camera, adversary: Decoding the enigmatic world of malicious frames in real-time video surveillance systems. Neural Processing Letters, 57(3), 46. https://doi.org/10.1007/s11063-025-11756-8.

[10] Papernot, N., Mcdaniel, P., Jha, S., Fredrikson, M., Celik, Z.B., Swami, A., 2016b. The limitations of deep learning in adversarial settings. In: Proceedings of the 2016 IEEE European Symposium on Security and Privacy, EURO S and P 2016, pp. 372–387. https://doi.org/10.1109/EuroSP.2016.36.

[11] Guo, S., Zhao, J., Li, X., Duan, J., Mu, D., & Jing, X. (2021a). A Black-Box Attack Method against Machine-Learning-Based Anomaly Network Flow Detection Models. Security and Communication Networks, 2021, 1–13. https://doi.org/10.1155/2021/5578335

[12] Hirano, H., Takemoto, K., Department of Bioscience and Bioinformatics, Kyushu Institute of Technology, Iizuka, Fukuoka 820-8502, Japan, & Corresponding author takemoto@bio.kyutech.ac.jp. (2019). Simple iterative method for generating targeted universal adversarial perturbations [Journal-article]. arXiv. https://arxiv.org/abs/1911.06502v2

[13] Govindarajulu, Y., Amballa, A., Kulkarni, P., & Parmar, M. (2023). Targeted attacks on timeseries forecasting. arXiv preprint arXiv:2301.11544. https://doi.org/10.48550/arXiv.2301.11544
[14] Shafi, M., Lashkari, A. H., Rodriguez, V., & Nevo, R. (2024). Toward generating a new Cloud-Based Distributed Denial of Service (DDOS) dataset and cloud intrusion traffic characterization. Information, 15(4), 195. https://doi.org/10.3390/info15040195
[15] Carlini, N., & Wagner, D. (2017). Towards evaluating the robustness of neural networks. 2022 IEEE Symposium on Security and Privacy (SP), 39–57. https://doi.org/10.1109/sp.2017.49
[16] Combey, T., Loison, A., Faucher, M., & Hajri, H. (2020). Probabilistic Jacobian-Based saliency maps attacks. Machine Learning and Knowledge Extraction, 2(4), 558–578. https://doi.org/10.3390/make2040030
[17] Ishai Rosenberg, Asaf Shabtai, Yuval Elovici, and Lior Rokach. 2021. Adversarial Machine Learning Attacks and Defense Methods in the Cyber Security Domain. ACM Comput. Surv. 54, 5, Article 108 (June 2022), 36 pages. https://doi.org/10.1145/3453158
[18] Sheikh, B. U. H., & Zafar, A. (2023). Unlocking adversarial transferability: A security threat towards deep learning-based surveillance systems via black box inference attack- a case study on face mask surveillance. Multimedia Tools and Applications, 83(8), 24749–24775. https://doi.org/10.1007/s11042-023-16439-x.
[19] shree.V.G, A., Thangaraj, M., & Nirmala Devi, M. (2025). A novel intrusion detection model for critical healthcare environments. International Journal of Computer Networks & Communications, 17(5), 41–63. https://doi.org/10.5121/ijcnc.2025.
[20] Roshan, K., Zafar, A., & Department of Computer Science, Aligarh Muslim University (Central University), Aligarh 202002, India. (2024). Black-box adversarial transferability: An empirical study in cybersecurity perspective. In Computers & Security (Vol. 141, p. 103853) [Journal-article]. https://doi.org/10.1016/j.cose.2024.103853.
[21] Vijayalakshmi, S., & Prasanna Venkatesan, V. (2025). Synergy analysis of ensemble feature selection on performance amelioration of intrusion detection system. International Journal of Computer Networks & Communications, 17(5), 01–19. https://doi.org/10.5121/ijcnc.2025.17501.
[22] Yuan, X., He, P., Zhu, Q., & Li, X. (2019). Adversarial examples: Attacks and defenses for deep learning. IEEE Transactions on Neural Networks and Learning Systems, 30(9), 2805–2824. https://doi.org/10.1109/tnnls.2018.2886017
[23] Zhang, J., & Li, C. (2019). Adversarial examples: opportunities and challenges. IEEE Transactions on Neural Networks and Learning Systems, 1–16. https://doi.org/10.1109/tnnls.2019.2933524
[24] Venturi, A., Stabili, D., Marchetti, M., University of Modena and Reggio Emilia, & University of Bologna. (2024). Problem space structural adversarial attacks for Network Intrusion Detection Systems based on Graph Neural Networks [Journal-article]. arXiv. https://arxiv.org/abs/2403.11830v2
[25] Roshan, K., Zafar, A., & Haque, S. B. U. (2023a). Untargeted white-box adversarial attack with heuristic defence methods in real-time deep learning based network intrusion detection system. Computer Communications, 218, 97–113. https://doi.org/10.1016/j.comcom.2023.09.030
[26] shree.V.G, A., Thangaraj, M., & Nirmala Devi, M. (2025). A novel intrusion detection model for critical healthcare environments. International Journal of Computer Networks & Communications, 17(5), 41–63. https://doi.org/10.5121/ijcnc.2025.17503
[27] Aiken, J., & Scott-Hayward, S. (2019). Investigating adversarial attacks against network intrusion detection systems in SDNs. Investigating Adversarial Attacks Against Network Intrusion Detection Systems in SDNs. https://doi.org/10.1109/nfv-sdn47374.2019.9040101
[28] Clements, J., Yang, Y., Sharma, A. A., Hu, H., & Lao, Y. (2021). Rallying Adversarial Techniques against Deep Learning for Network Security. 2021 IEEE Symposium Series on Computational Intelligence (SSCI), 01–08. https://doi.org/10.1109/ssci50451.2021.9660011
[29] Usama, M., Asim, M., Latif, S., Qadir, J., & Ala-Al-Fuqaha, N. (2019). Generative adversarial networks for launching and thwarting adversarial attacks on network intrusion detection systems. Generative Adversarial Networks for Launching and Thwarting Adversarial Attacks on Network Intrusion Detection Systems. https://doi.org/10.1109/iwcmc.2019.8766353
[30] Pawlicki, M., Choraś, M., & Kozik, R. (2020). Defending network intrusion detection systems against adversarial evasion attacks. Future Generation Computer Systems, 110, 148–154. https://doi.org/10.1016/j.future.2020.04.013
[31] Grosse, K., Manoharan, P., Papernot, N., Backes, M., & Mcdaniel, P. (2017). On the (Statistical) Detection of Adversarial Examples. ArXiv, abs/1702.06280. https://api.semanticscholar.org/CorpusID:16863734

[32] Debicha, I., Cochez, B., Kenaza, T., Debatty, T., Dricot, J., & Mees, W. (2023). Adv-Bot: Realistic adversarial botnet attacks against network intrusion detection systems. Computers & Security, 129, 103176. https://doi.org/10.1016/j.cose.2023.103176

[33] Saha, A., Subramanya, A., & Pirsiavash, H. (2020, April). Hidden trigger backdoor attacks. In Proceedings of the AAAI conference on artificial intelligence (Vol. 34, No. 07, pp. 11957-11965). https://doi.org/10.1609/aaai.v34i07.6871

[34] Sheatsley, R., Papernot, N., Weisman, M. J., Verma, G., & McDaniel, P. (2022). Adversarial examples for network intrusion detection systems. Journal of Computer Security, 30(5), 727–752. https://doi.org/10.3233/jcs-210094

[35] Sharma, S., & Chen, Z. (2024). A systematic study of adversarial attacks against network intrusion detection systems. Electronics, 13(24), 5030. https://doi.org/10.3390/electronics13245030

[36] Wu, Z., Zhang, H., Wang, P., & Sun, Z. (2022). RTIDS: a robust Transformer-Based approach for intrusion Detection System. IEEE Access, 10, 64375–64387. https://doi.org/10.1109/access.2022.3182333

[37] Zhang, S., Xie, X., & Xu, Y. (2020a). A Brute-Force Black-Box method to attack Machine Learning-Based systems in cybersecurity. IEEE Access, 8, 128250–128263. https://doi.org/10.1109/access.2020.3008433

[38] Chen, J., Wu, D., Zhao, Y., Sharma, N., Blumenstein, M., & Yu, S. (2020). Fooling intrusion detection systems using adversarially autoencoder. Digital Communications and Networks, 7(3), 453–460. https://doi.org/10.1016/j.dcan.2020.11.001

[39] Sharafaldin, I., Lashkari, A. H., Hakak, S., & Ghorbani, A. A. (2019). Developing realistic Distributed Denial of service (DDOS) attack dataset and taxonomy. In International Carnahan Conference on Security Technology (pp. 1–8). https://doi.org/10.1109/ccst.2019.8888419

[40] Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A detailed analysis of the KDD CUP 99 data set. In A detailed analysis of the KDD CUP 99 data set (pp. 1–6). https://doi.org/10.1109/cisda.2009.5356528

[41] Goldschmidt, P., Chudá, D., Faculty of Information Technology, Brno University of Technology, Kempelen Institute of Intelligent Technologies, & Faculty of Electrical Engineering and Information Technology, Slovak University of Technology in Bratislava. (2025). Network Intrusion Datasets: A Survey, Limitations, and recommendations. In Computers & Security [Journal-article]. https://arxiv.org/abs/2502.06688v1