iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** II **Month of publication:** February 2024

DOI: https://doi.org/10.22214/ijraset.2024.58300

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue II Feb 2024- Available at www.ijraset.com

A Hierarchical Deep Learning Architecture for Robust Intrusion Detection in Time-Evolving Security Landscapes

Shashank R R[1], Aditya A Navale[2], Indudhara S[3]
B.E in Computer Science & Engineering, Dept. of CSE, Jawaharlal Nehru New College of Engineering

Abstract: ***This paper introduces a novel architecture for Intrusion Detection Systems (IDS), designed to enhance resilience against adversarial attacks by integrating conventional machine learning (ML) models with Deep Learning (DL) models. The proposed system, termed DLL-IDS, comprises three key components: a DL-based IDS, an adversarial example (AE) detector based on local intrinsic dimensionality (LID), and an ML-based IDS.***
Initially, a novel AE detector is developed using LID to identify potential adversarial examples. This detector serves as a crucial component in identifying suspicious inputs that may be crafted to deceive the IDS. Subsequently, the system leverages the low transferability of attacks between DL and ML models to establish a robust ML model capable of discerning the maliciousness of potential AEs. The DLL-IDS architecture operates as follows: if incoming traffic is flagged as an AE by the detector, the ML-based IDS is employed to evaluate its maliciousness; otherwise, the DL-based IDS handles the prediction. By integrating both DL and ML approaches, the system benefits from the high prediction accuracy of DL models while exploiting the low susceptibility of ML models to adversarial attacks. Experimental results demonstrate significant enhancements in IDS prediction performance under adversarial conditions, achieving high accuracy with minimal resource consumption. This fusion mechanism effectively combines the strengths of DL and ML models, thereby bolstering the overall robustness of the IDS against sophisticated intrusion attempts.
Keywords: Intrusion detection system, adversarial example, adversarial detection, adversarial defense, deep learning, machine learning, classification algorithm

I. INTRODUCTION

In recent years, the proliferation of computer networks has brought significant advancements, yet it has also led to a surge in security threats targeting network devices. These threats jeopardize the confidentiality, integrity, and availability of network assets, necessitating robust security measures. Intrusion Detection Systems (IDS) have emerged as critical tools for safeguarding networks, employing signature-based and anomaly-based detection methods. While signature-based detection is effective against known threats, it falters when faced with unknown attacks. In contrast, anomaly-based detection offers resilience against unknown threats by modeling normal network behavior. The advent of high-performance hardware, particularly GPUs, has spurred researchers to explore the integration of machine learning (ML) techniques into IDS. Decision trees, Support Vector Machines, and K-Nearest Neighbor methods are common ML approaches, but their effectiveness depends on dataset complexity and size. Deep Learning (DL)-based IDS, characterized by deeper feature extraction, has gained prominence due to its superior performance. However, DL models are susceptible to adversarial attacks, where crafted perturbations manipulate IDS detection outcomes, posing a grave threat to network security. Efforts to counter adversarial attacks have led to three defense approaches: parameter protection, robustness optimization, and AE detection. While parameter protection and robustness optimization have limitations, AE detection emerges as a practical solution. However, addressing the detected adversarial traffic poses a challenge, as its malicious intent may vary. Hence, a comprehensive framework that evaluates traffic for both adversarial and malicious intent is essential.

In response, we propose the DLL-IDS system, comprising DL-based IDS, LID-based AE detector, and ML-based IDS components. The LID-based detector leverages differences in spatial attributes between clean examples and AEs to achieve high detection accuracy. Additionally, we employ the Label-spreading (LS) ML algorithm, known for its robustness against AEs, to discern the malicious nature of detected AEs. Our contributions include:

1) Introducing a novel IDS architecture addressing the specific requirements posed by adversarial attacks.
2) Proposing an innovative AE detection method based on LID, achieving high accuracy even against intense adversarial attacks.

3) Demonstrating varying levels of attack transferability between DL and traditional ML models and leveraging this insight to enhance IDS robustness.Conducting comprehensive experiments illustrating DLL-IDS's superior performance, maintaining high accuracy even under adversarial attack scenarios while exhibiting comparable detection capabilities on clean examples.

Overall, DLL-IDS represents a significant advancement in IDS resilience against adversarial threats, promising enhanced network security in the face of evolving cyber threats.

II. OBJECTIVES

1) *Enhanced Detection Accuracy:* The primary goal is to achieve high accuracy in identifying intrusions and security threats within dynamic and evolving network environments. By leveraging deep learning techniques, the architecture should be capable of effectively distinguishing between normal network behavior and malicious activities, even in the presence of sophisticated attacks.
2) *Adaptability to Time-Evolving Threats:* The architecture should demonstrate resilience against time-evolving threats and attacks. This involves the ability to continuously learn and adapt to new attack patterns, trends, and variations in network traffic over time. The model should dynamically update its detection capabilities to maintain effectiveness in evolving security landscapes.
3) *Scalability and Efficiency:* Scalability and efficiency are crucial considerations, particularly for large-scale network environments. The architecture should be capable of handling increasing volumes of network data efficiently, ensuring real-time or near-real-time intrusion detection without significant computational overhead.
4) *Hierarchical Representation of Network Data:* The architecture should employ a hierarchical representation of network data, capturing both high-level and fine-grained features relevant to intrusion detection. This hierarchical approach enables the model to extract abstract features at different levels of granularity, improving the robustness and interpretability of the detection process.
5) *Robustness Against Adversarial Attacks:* Given the prevalence of adversarial attacks targeting intrusion detection systems, the architecture should incorporate mechanisms to enhance robustness against such attacks. This includes techniques for detecting and mitigating adversarial perturbations designed to evade detection.
6) *Interpretability and Explainability:* While leveraging complex deep learning models, the architecture should maintain a degree of interpretability and explainability in its decision-making process. This ensures that security analysts can understand and trust the system's output, facilitating effective response and mitigation actions.
7) *Generalization across Diverse Network Environments:* The architecture should generalize well across diverse network environments, including different network topologies, protocols, and applications. It should be adaptable to various use cases and scenarios without significant retraining or customization efforts.
8) *Integration with Existing Security Infrastructure:* The architecture should be designed for seamless integration with existing security infrastructure and tools, enabling organizations to enhance their overall cybersecurity posture without major disruptions or additional resource requirements.

III. LIMITATIONS

While the proposed hierarchical deep learning architecture for robust intrusion detection in time-evolving security landscapes offers promising capabilities, it also has certain limitations that warrant consideration.

Firstly, despite the advancements in deep learning techniques, the effectiveness of the proposed architecture may be influenced by the availability and quality of labeled training data. Acquiring labeled datasets that accurately represent the diverse range of network activities and security threats can be challenging, particularly for rare or emerging attack scenarios. Limited or biased training data may result in suboptimal performance and reduced generalization ability.

Secondly, the scalability of deep learning models remains a concern, especially when deployed in large-scale network environments with high volumes of real-time traffic. Deep learning architectures often require significant computational resources and may struggle to process large datasets efficiently, leading to delays in intrusion detection and response. Moreover, the deployment of complex deep learning models on resource-constrained devices or networks may pose additional challenges in terms of computational overhead and energy consumption. Furthermore, while the hierarchical representation of network data enables the model to capture multi-level features, designing an optimal hierarchical architecture requires careful consideration of feature extraction and abstraction mechanisms. Inadequate feature representation or hierarchical structure may limit the model's ability to discern subtle variations in network behavior or effectively differentiate between benign and malicious activities.

Additionally, the robustness of the proposed architecture against adversarial attacks is subject to certain limitations. Adversarial attacks targeting intrusion detection systems continue to evolve, and sophisticated adversaries may exploit vulnerabilities in the model to generate adversarial examples that evade detection.

Achieving robustness against a wide range of adversarial techniques remains an ongoing research challenge, and the proposed architecture may require additional defense mechanisms to mitigate such threats effectively.

Moreover, while the architecture aims to maintain interpretability and explainability in its decision-making process, the inherent complexity of deep learning models may hinder the transparency of the detection logic. Understanding the rationale behind the model's predictions and identifying the factors influencing its decisions may be challenging, particularly for security analysts without expertise in deep learning.

Finally, the integration of the proposed architecture with existing security infrastructure and tools may encounter compatibility issues or deployment complexities. Ensuring seamless interoperability and alignment with organizational security policies and procedures is essential for effective deployment and adoption of the intrusion detection system.

IV. LITERATURE SURVEY

In [1] their comprehensive survey published in IEEE Communications Surveys & Tutorials, He, Kim, and Asghar (2023) provide a thorough examination of the field of adversarial machine learning (AML) applied to network intrusion detection systems (NIDS). The survey covers a wide range of topics, including the fundamental concepts of AML, various types of adversarial attacks and their implications for NIDS, existing defense mechanisms, and evaluation methodologies. The authors analyze the strengths and weaknesses of different AML approaches, including both evasion and poisoning attacks, and discuss the challenges and opportunities in developing robust and reliable NIDS in the face of adversarial threats. Furthermore, they highlight emerging trends, research directions, and open issues in the field, aiming to guide future research efforts towards enhancing the security and effectiveness of NIDS against adversarial attacks.

Rigaki (2017) [2] investigates the application of adversarial deep learning techniques to intrusion detection classifiers in a study conducted at Luleå University of Technology, Sweden. The research explores the vulnerabilities of intrusion detection systems (IDS) to adversarial attacks, focusing on how deep learning models employed in IDS can be manipulated or deceived by adversarial examples. Rigaki's work contributes to the understanding of the limitations and potential vulnerabilities of IDS in the face of sophisticated adversarial attacks, shedding light on the importance of developing robust defense mechanisms to mitigate such threats in network security.

In Wang's (2018) [3] study published in IEEE Access, the focus is on deep learning-based intrusion detection systems (IDS) and their susceptibility to adversarial attacks. The research delves into the challenges posed by adversaries in the context of IDS, exploring how deep learning models used for intrusion detection can be compromised or manipulated by adversarial examples. By investigating the impact of adversarial attacks on IDS performance, Wang's work contributes to the growing body of literature aimed at understanding the vulnerabilities of deep learning-based security systems and developing strategies to enhance their robustness against adversarial threats.

In Sahani et al.'s (2018) [4] contribution to the Proceedings of ICCAN 2017, the focus is on the classification of intrusion detection using data mining techniques. The study explores various data mining methods for intrusion detection, aiming to enhance the effectiveness of identifying and mitigating security threats in computer networks. By investigating the application of data mining techniques such as classification algorithms, the research aims to improve the accuracy and efficiency of intrusion detection systems. This work aligns with the broader goal of advancing cybersecurity measures by leveraging computational and analytical approaches to combat evolving threats in network environments.

In [5] their study published in IEEE Access in 2018, Yan and Han focus on enhancing intrusion detection systems (IDS) through effective feature extraction using stacked sparse autoencoders. The research addresses the crucial task of feature extraction, which plays a vital role in improving the accuracy and efficiency of IDS. By employing stacked sparse autoencoders, the study aims to learn hierarchical representations of network data, facilitating the extraction of discriminative features that capture underlying patterns indicative of network intrusions. Through their approach, Yan and Han aim to overcome the limitations of traditional feature extraction methods and contribute to the development of more robust and accurate intrusion detection systems capable of effectively identifying and mitigating security threats in complex network environments.

In [6] their paper presented at the 2017 Sensor Signal Processing for Defence Conference (SSPD), Ghanem et al. focus on the application of Support Vector Machine (SVM) techniques for network intrusion and cyber-attack detection.

The study contributes to the ongoing research efforts aimed at developing effective methods for detecting and mitigating security threats in networked environments. By leveraging SVM algorithms, which are known for their ability to handle complex data and nonlinear relationships, the authors aim to enhance the accuracy and efficiency of intrusion detection systems. Through their investigation, Ghanem et al. seek to provide insights into the potential of SVM-based approaches in addressing the evolving challenges posed by network intrusions and cyber-attacks, ultimately contributing to the advancement of cybersecurity technologies.

In [7] their study published in the International Journal of Electrical and Computer Engineering, Alalousi et al. conduct a preliminary performance evaluation of unsupervised machine learning methods, specifically K-means, K-nearest neighbors (KNN), and expectation-maximization (EM), for network flow classification. The research aims to assess the efficacy of these algorithms in categorizing network flows, which is essential for tasks like network traffic analysis and anomaly detection. By comparing the performance of these methods, the authors contribute valuable insights into their suitability for real-world applications in network security and management. This evaluation serves as a foundational step towards identifying the most effective approach for network flow classification, thereby informing the development of robust and efficient systems for network monitoring and defense.

In [8] their seminal work, Szegedy et al. investigate the intriguing properties of neural networks, shedding light on the vulnerabilities and robustness of deep learning models. Published as an arXiv preprint, their research delves into the phenomenon of adversarial examples, wherein small, imperceptible perturbations to input data can lead to misclassification by neural networks. This pioneering study highlights the susceptibility of deep neural networks to adversarial attacks, prompting further exploration into the underlying mechanisms and potential defense strategies. By uncovering these vulnerabilities, the authors catalyze a paradigm shift in understanding the security implications of artificial intelligence systems, with significant implications for fields ranging from computer vision to network intrusion detection.

In [9] their work presented at the 26th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining (PAKDD 2022), Lin, Shi, and Xue introduce IDSGAN, a novel approach that leverages generative adversarial networks (GANs) for generating adversarial attacks against intrusion detection systems (IDS). Published as part of the conference proceedings by Springer International Publishing, their research contributes to the growing body of literature on adversarial machine learning in the domain of cybersecurity. By harnessing the power of GANs, IDSGAN facilitates the automated generation of sophisticated attacks that evade detection by IDS, thereby exposing vulnerabilities in existing defense mechanisms. This study underscores the importance of developing robust intrusion detection techniques capable of withstanding increasingly sophisticated adversarial threats in network security.

In [10] their 2021 paper published in Expert Systems with Applications, Alhajjar, Maxwell, and Bastian delve into the domain of adversarial machine learning within the context of network intrusion detection systems (NIDS). Through their comprehensive survey, they explore the landscape of adversarial attacks and defense mechanisms, shedding light on the vulnerabilities and challenges faced by NIDS in the face of sophisticated adversarial threats. By synthesizing and analyzing existing research in this field, the authors provide valuable insights into the current state-of-the-art techniques, highlighting the need for robust and resilient defense strategies to mitigate the impact of adversarial attacks on network security. Their work contributes to the broader understanding of adversarial machine learning and its implications for enhancing the security posture of networked systems.

V. BACKGROUND & RELATED WORK

A. Adversarial attacks on IDS

With the increasing adoption of deep learning in intrusion detection systems (IDS), DL-based IDS offer enhanced flexibility and efficiency compared to traditional IDS by autonomously learning features and constructing behavior libraries. However, the advancement of adversarial machine learning poses new hurdles for DL-based IDS. Adversarial attacks aim to disrupt classification by introducing imperceptible perturbations, generating adversarial examples (AEs) that evade detection. Various AE generation methods, such as lp ball constraints and generative models, target feature-level attacks (FLA), which directly manipulate input characteristics. FLA, encompassing black-box (BB), gray-box (GB), and white-box (WB) attacks, perturb middle-layer characteristics to enhance AE transferability. Notably, WB attacks, like those employing the fast gradient sign method (FGSM) and Jacobian-based saliency map attack (JSMA), exhibit effectiveness against IDS classifiers. Studies comparing WB attack methods on IDS models, including deep neural networks (DNNs) and MLPs, highlight JSMA's potency in reducing accuracy. Moreover, investigations on IDS robustness against FLA, using algorithms like FGSM, JSMA, and Carlini-Wagner (CW), underscore the need for defenses that consider network constraints to prevent unrealistic perturbations. Despite the challenge posed by adversarial attacks, research on defending DL-based IDS against such threats remains a focal point, with ongoing efforts to enhance defense mechanisms.

B. *Adversarial Defense and Its Applicability in IDS*

The advent of adversarial attacks has spurred research into adversarial defense mechanisms, primarily in computer vision (CV) domains. However, these approaches have not been extensively adapted for intrusion detection systems (IDS). Existing defense strategies fall into three categories: parameter protection, robustness optimization, and adversarial detection.

Parameter protection methods, such as gradient hiding techniques, aim to safeguard model parameters from attacks, but their effectiveness in DL-IDS is limited due to the unique characteristics of network traffic data. Robustness optimization, including adversarial training, enhances classifier robustness against adversarial examples (AEs) but may not be suitable for IDS due to resource constraints and the potential decrease in accuracy for clean examples. Adversarial detection methods, on the other hand, focus on identifying AEs before classification, utilizing approaches like dynamic adversary training and density estimation. While these methods show promise in CV, their applicability to IDS remains largely unexplored. MANDA represents a notable exception, employing manifold detection and decision boundary methods for AE detection in IDS. However, manifold detection alone lacks theoretical support, and decision boundary methods suffer from efficiency issues. Our work addresses these gaps by proposing a comprehensive AE defense framework tailored to the specific requirements of IDS, including the classification of AEs generated by benign and malicious traffic.

VI. SYSTEM MODEL AND THREAT MODEL

A. *Weakness of Deep Learning*

Deep learning models exhibit vulnerabilities to adversarial attacks due to inherent characteristics of their decision boundaries and the high-dimensional nature of input data. Firstly, there exists a gap between the decision boundaries of deep learning models and the true classification boundaries of input samples, allowing adversaries to exploit this gap to generate adversarial examples (AEs) that lead to incorrect classifications. This vulnerability is influenced by the quantity and distribution of training data. Secondly, data often resides in high-dimensional spaces, making it susceptible to perturbations in each dimension. Even small perturbations in multiple dimensions can alter the classification outcome significantly. In the context of DL-based intrusion detection systems (IDS), adversaries can leverage these vulnerabilities by generating AEs from benign traffic to trigger false positives in the model or by generating AEs from malicious traffic to evade detection, thereby posing significant security threats to network systems.

B. *Threat Model*

The objective of the simulated attacker is to disrupt the classification accuracy of the IDS model by inducing misclassifications in traffic samples, regardless of their benign or malicious nature. The success of an attack is measured by the reduction of the model's accuracy below its normal range. The threat model outlines the conditions under which adversarial examples (AEs) are generated. Attacks against deep learning models can be categorized based on the attacker's prior knowledge into three types: White-Box Attack, Black-Box Attack, and Gray-Box Attack. In a White-Box Attack, the attacker possesses comprehensive knowledge about the victim model, including its training data, architecture, and weights, allowing them to exploit model information, particularly gradients, to craft AEs. In contrast, a Black-Box Attack occurs when the attacker lacks access to the victim model's parameters and instead generates AEs by observing its input-output behavior. Gray-Box Attacks involve partial knowledge, where the attacker is aware of the training data but lacks details about the model's architecture and weights, prompting them to construct an alternative model to generate AEs based on transferability. Among these categories, White-Box Attacks are deemed most potent, significantly impacting model testing outcomes. Hence, the article focuses on employing four White-Box attack methods to target deep learning models. The first method, Fast Gradient Sign Method (FGSM), is an L∞ attack that leverages DL gradients for AE generation. Its core principle involves swiftly computing gradient signs to determine the direction of gradient descent.

C. *System Model*

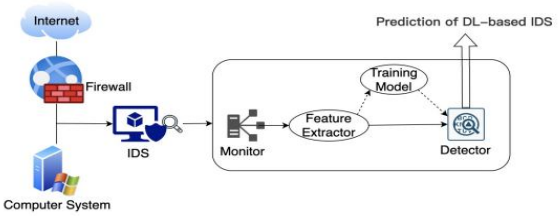


Figure 1. System model of DL-based IDS.

Figure 1 illustrates the architecture of a typical DL-based Intrusion Detection System (IDS), comprising four main components: monitor, feature extractor, training model, and Detector model. The monitor gathers data for preprocessing and detection purposes. Feature extraction involves extracting quintuples and traffic features from raw packets and formatting them. The training model trains a neural network to differentiate between normal and malicious traffic using a large dataset of examples. The DL-based IDS performs traffic classification and prediction based on the neural network's classification results. In real network setups, external traffic is usually mirrored and copied post-firewall before being transmitted to the IDS for analysis. Upon receiving traffic, the monitor immediately forwards it to the data preprocessing module. Once preprocessing is complete, the results are sent to the pre-trained DL-based IDS model for prediction. The IDS model predicts whether the traffic is malicious or benign, and this prediction is relayed to the user. However, due to the vulnerability of DL models to adversarial attacks, the DL-based IDS may misclassify adversarial traffic, leading to erroneous predictions. Adversarial Examples (AEs) of malicious traffic are crafted by attackers to conceal their true attack intent and payload, aiming to bypass IDS detection and cause harm to the system. It's crucial to recognize that both the original malicious traffic and its corresponding AEs are inherently malicious, capable of inflicting damage on the network system.

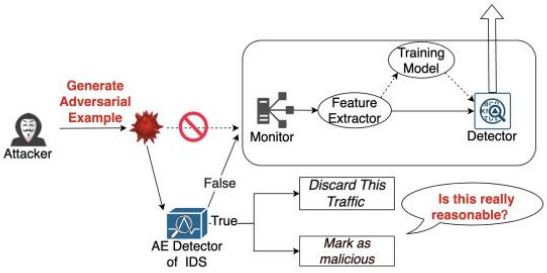


Figure 2. System model of the current AE defense for IDS

The creation of Adversarial Examples (AEs) from normal traffic aims to disrupt the regular functioning of IDS by triggering numerous false alarms categorized as 'benign,' thereby reducing its availability. It's important to note that both the original benign traffic and the AEs derived from it retain their benign nature, posing no direct threat to the network system's integrity. Research on adversarial defense strategies for DL-based IDS primarily concentrates on adversarial detection, akin to approaches employed in object detection.

The workflow depicted in Figure 2 closely resembles methods used in AE defense within object detection domains. Upon detecting adversarial attacks, the traffic is either discarded or flagged as malicious. Such decision-making is suitable for object detection, where the actual content of images is less critical than identifying adversarial attacks. However, in network traffic analysis for malicious intent, the traffic itself may exhibit malicious attributes unrelated to adversarial attacks. IDS should discern the potential maliciousness of incoming traffic. Yet, if an input example is identified as an AE, it may erroneously be classified as malicious traffic, solely focusing on the subjective intent of the AE-generating attacker, rather than its inherent nature. Consider a scenario: an attacker crafts AEs from benign traffic and injects them into the network. Despite posing no direct threat to the system, these AEs trigger a flurry of attack alarms within the IDS, overwhelming its usability and necessitating manual inspection of these false alarms, resulting in significant resource wastage. Such outcomes are untenable. Therefore, IDS should not only identify AEs but also discern whether they represent inherently malicious traffic.

VII. DLL-IDS SYSTEM

In this section, we introduce a novel IDS system architecture engineered to withstand adversarial attacks effectively. This IDS system assesses traffic from dual perspectives: discerning whether the input traffic constitutes an AE and evaluating the inherent maliciousness of the traffic. Illustrated in Figure 3, our proposed system aims to substantially bolster the lower threshold of IDS performance during attacks, leveraging the efficiency of DL models and the robustness of ML models. The upper boundary, as depicted in the figure, is defined by the model's predictive accuracy on clean examples, while the lower boundary is influenced by the severity of adversarial attacks. For instance, in the case of high-intensity attacks like CW, the model's prediction accuracy may plummet to below 10%. Our objective is to devise a novel IDS system architecture capable of fortifying IDS resilience against adversarial attacks, while satisfying the stipulated requirements: 1) discerning the inherent maliciousness of traffic and 2) accurately identifying examples as AEs.

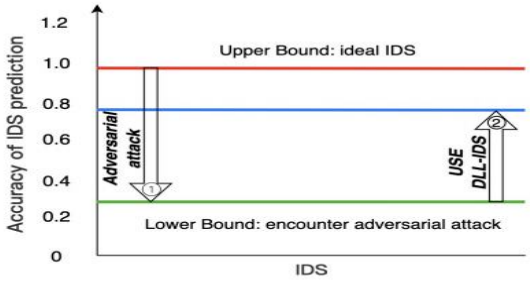


Figure 3. The system significance of DLL-IDS

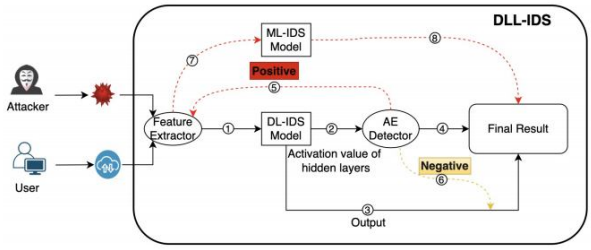


Figure 4. The realization of DLL-IDS

To realize this objective, we introduce an IDS system architecture, as depicted in Figure 4 and detailed in Algorithm 1.Departing from the conventional single-model DL-based IDS, our proposed system comprises three distinct modules: DL-based IDS, ML-based IDS, and AE detector. Upon entry into the DLL-IDS system, incoming traffic, whether clean or an AE, is initially directed to the DL-based IDS module. Here, the DL model conducts two key operations: first, it predicts the maliciousness of the sample, and second, it forwards the activation values of each layer of the sample to the AE detector for further analysis. Leveraging these activation values from the DL model, the AE detector evaluates whether the input traffic qualifies as an AE, yielding the final detection outcomes. Subsequently, the AE detector navigates two potential scenarios: if an AE is detected, the feature extraction step is revisited, and the extracted features are relayed to the ML-based IDS, assuming its robustness against AEs. The ML-based IDS then reassesses the traffic's maliciousness, with its prediction serving as the final determination. Conversely, in the absence of an AE detection, the DL-based IDS prediction takes precedence. Finally, both prediction results are aggregated and outputted. In scenarios where only clean examples are received, it's reasonable to anticipate minimal utilization of the ML-based IDS resources. Conversely, in AE-only scenarios, reliance on the AE detector's robust detection capabilities prevents interference from the DL-based IDS in the final decision-making process. This modular approach maximizes the utilization of diverse models and ensures optimal defense effectiveness with minimal system resource consumption. Below, we outline the design principles guiding each module.

Algorithm 1: DLL-IDS

Input:
X: examples of network traffic from outside
$D(x)$: a pre-trained DL-based IDS model
$Hid - Act(x)$: the Activation value of x in each hidden layer of DL model
$M(x)$: a pre-trained ML-based IDS model
$AD(lid)$: a LID-based AE detector
Output:
$isAdversarial \in \{\text{False}, \text{True}\}$
$isMalicious \in \{\text{False}, \text{True}\}$

$Result_{adv} = []$, $Result_{mal} = []$
foreach x *in* X **do**
  $Result_{mal} = D(x)$
  $Hid - Act(x) \leftarrow D(x)$
  Compute lid_x
  if $AD(lid_x)$ **then**
    $Result_{adv} = \text{True}$
    $Result_{mal} = M(x)$
  else
    $Result_{adv} = \text{False}$
  $Result_{adv}$.append($Result_{adv}$)
  $Result_{mal}$.append($Result_{mal}$)
return $Result_{adv}$, $Result_{mal}$

A. DL-based IDS

In recent years, the adoption of deep learning in Intrusion Detection Systems (IDS) has gained traction due to its capability to understand hierarchical network structures and extract features from multiple hidden layers. Deep Neural Networks (DNNs) serve as the fundamental architecture for deep learning in IDS, comprising input layers, hidden layers, and output layers. This study leverages DNNs as the primary classifier for the IDS, given their superior performance in classification tasks. Within the domain of DNNs, various network architectures such as fully-connected feedforward neural networks (FNN), convolutional neural networks (CNN), recurrent neural networks (RNN), among others, are integrated. These architectures exhibit intricate mapping relationships between input vectors and output vectors in the network model. The input vectors represent feature vectors obtained post meticulous data preprocessing, while the output consists of a probability vector characterizing different classification categories. The input-output relationship of the hidden layer can be represented as ($h = g(wx + b)$), where g signifies the activation function of the hidden layer, w denotes the weights between the input layer and the hidden layer, and b represents the biases of the hidden neurons. In this research, both FNN and CNN models are tailored to different datasets for the DL-based IDS. Specifically, for the NSL-KDD dataset, an FNN model is designed. The architecture, depicted in Figure 5, features a 128-dimensional input vector post data preprocessing, resulting in 128 neurons in the input layer. The output vector comprises two dimensions, representing benign or malicious traffic probabilities ranging from 0 to 1. The final output of the model is determined by the maximum probability value in the two-dimensional output vector. Through extensive experimentation, it was observed that increasing the number of layers enhances the prediction accuracy for the network traffic dataset, with a relative peak achieved at five layers. Thus, an FNN model with five hidden layers, as depicted in Figure 6, is chosen as the foundation for the IDS in this study. The accuracy of this FNN model for clean traffic samples of NSL-KDD is already approximately 95%. It's essential to note that while our research focuses on defending against adversarial attacks, the selection of these DL models is primarily based on their role as the DL-based IDS component in the DLL-IDS system.

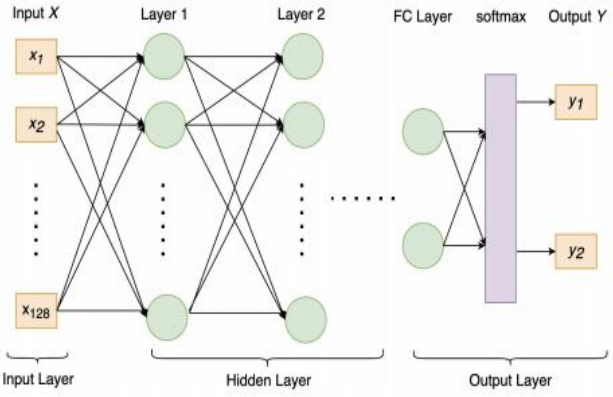


Figure 5. Designed FNN model for NSL-KDD

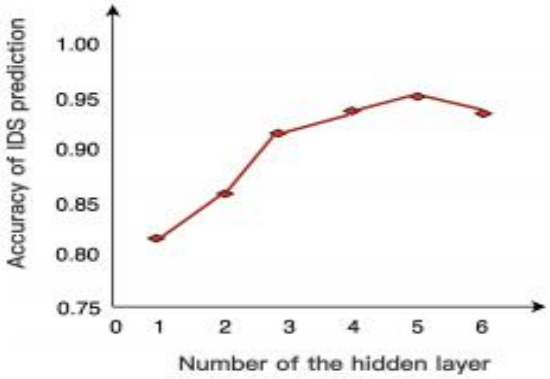


Figure 6. Network layer number and detection accuracy of the proposed FNN model

B. Adversarial Examples Detector

Previous research has elucidated that the adversarial space, a subspace of the original data, is distinguished by its low-probability region within the distribution of the original data. These adversarial boundaries closely align with the clean dataset in the adversarial direction but often belong to distinct data distributions on their submanifolds. Additionally, it has been observed that the transferability of these subspaces to other models escalates with the higher number of orthogonal adversarial directions. In this study, the objective is to devise an Adversarial Example (AE) detector for discerning whether the input traffic to IDS constitutes an AE. However, capturing the spatial distance characteristics of AEs poses a challenge due to the low dimensionality of the data and the necessity to meticulously control the magnitude of adversarial perturbations applied to network traffic. Kernel Density Estimation (KDE), a conventional method, fails to adequately capture the spatial features of AEs, resulting in unsatisfactory detection performance. Conversely, Local Intrinsic Dimensionality (LID) serves as an alternative method to KDE, focusing on the distribution of internal distances between examples to capture the intrinsic dimensionality of the data. This approach has demonstrated success across various domains, including dimension reduction, similarity search, and anomaly detection. Notably, advancements have been made in leveraging LID for extracting features of AEs in object detection scenarios. Consequently, it is plausible to construct an AE detector by extracting the LID of clean examples and AEs from traffic data and discerning the statistical disparities between them.

Algorithm 2: Training phase for LID-based adversarial detector

Input:
X: normal examples in dataset;
$D(x)$: a pre-trained DNN with L hidden layers;
k: the number of nearest neighbors for LID estimation
Output:
Detector(LID) // a result from detector

$LID_{neg} = [] , LID_{pos} = []$ // positive represents AE
foreach P_{clean} *in* X **do**
  // a minibatch of examples in X
  P_{adv} = adversarial attack P_{clean};
  $N = |P_{clean}|$ // number of examples
  LID_{neg}, LID_{pos} = zeros$[N, L]$;
  foreach i *in* $[1, L]$ **do**
    $A_{clean} = D^i(P_{clean})$;
    $A_{adv} = D^i(P_{adv})$;
    foreach $j \in [1, N]$ **do**
      $LID_{neg}[j, i] = -\left(\frac{1}{k}\sum_{i=1}^{k} \log \frac{r_i(A_{clean}[j], A_{clean})}{r_k(A_{clean}[j], A_{clean})}\right)^{-1}$;
      $LID_{adv}[j, i] = -\left(\frac{1}{k}\sum_{i=1}^{k} \log \frac{r_i(A_{adv}[j], A_{clean})}{r_k(A_{adv}[j], A_{clean})}\right)^{-1}$;
  LID_{neg}.append(LID_{neg});
  LID_{pos}.append(LID_{adv});
return Detector(LID) = train a SVM classifier on (LID_{neg}, LID_{pos})

C. ML-based IDS

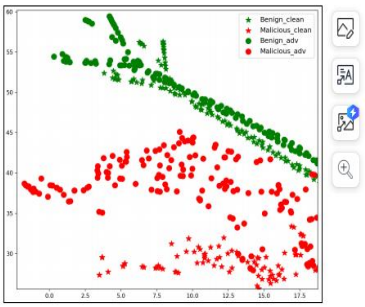


Figure 7. Projection plot of some clean examples of NSL-KDD and AEs which generated under 10% perturbation restriction with BIM attack.

Adversarial Examples (AEs) are crafted by introducing minor perturbations to clean examples, capable of traversing decision boundaries and leading DL models to erroneous classifications. For instance, an AE derived from malicious traffic might exhibit benign characteristics to the DL-based IDS. Nonetheless, we hypothesize that these generated AEs share a closer spatial distribution to the original clean class. To substantiate this conjecture, we project a subset of AEs, acquired post-attacks, onto a two-dimensional plane using t-distributed stochastic neighbor embedding (t-SNE), as illustrated in Figure 7. This depiction notably showcases that the distribution of AEs stemming from malicious traffic aligns relatively closely with that of clean examples from the same class of traffic. They coexist within the same malicious manifold space, indirectly reinforcing their classification within the same traffic category.

Given that certain machine learning algorithms possess the capability to discern and retain spatial features of samples, alongside findings indicating their resilience and transferability to AEs, we endeavor to identify an ML model proficient in making reasonably accurate predictions for AEs. Through experimentation, we compare various machine learning methods and evaluate their performance. To comprehensively capture the spatial manifolds of both malicious and benign traffic, we ultimately opt for the LS algorithm proposed by Zhu et al. This semi-supervised algorithm transforms samples into a graph representation and leverages a combination of labeled and unlabeled samples to unveil the intrinsic structure of the sample manifold. By exploiting the interrelations among graph nodes, labels are effectively propagated from labeled samples to unlabeled samples, enhancing the algorithm's ability to discern between different traffic categories.

VIII. EXPERIMENTS

A. *Datasets*

Since its inception in 1999, the KDD'99 dataset has served as a prominent benchmark for assessing anomaly detection methods. However, one significant drawback of the KDD dataset is its substantial redundancy. To address this issue, Mahbod Tavallaee introduced the NSL-KDD dataset in 2010, offering a solution to the challenges posed by the KDD dataset. Consequently, we opt for the NSL-KDD dataset as the basis for evaluating the DLL-IDS system proposed in this paper. This dataset comprises approximately 4,000,000 single connection vectors, each containing 41 features and labeled as normal or an attack. The attacks encompass four main categories: Denial of Service Attack (DoS), User to Root Attack (U2R), Remote to Local Attack (R2L), and Probing Attack, with each category further subdivided into specific attack types. For our study, we treat traffic classification as a binary classification problem.

We curate a training set of 120,000 samples and a testing set of 25,000 samples from the original data. To illustrate the effectiveness of our IDS, we focus on representative DoS attack data in presenting experimental results, noting that the defense performance against other attack types is similarly robust.

To ensure the robustness of our experimental findings, we also incorporate an additional dataset, CICIDS2018, provided by the Canadian Institute for Cybersecurity (CIC), for evaluation. This dataset mirrors real-world Packet Capture data and comprises normal network traffic along with various intrusion classes, including Brute-force, Heartbleed, DDoS, Web attacks, among others. In preprocessing the data, we eliminate features with a high number of infinite values and missing values, resulting in the retention of 78 features for training. For this dataset, we employ a deep learning model structure utilizing Convolutional Neural Networks (CNN) to train a DL-based IDS. We utilize 1.5 million network flows as the training dataset and 300,000 network flows as the testing dataset. Similar to our approach with NSL-KDD, we concentrate on the DDOS attack type when presenting experimental results for the CICIDS2018 dataset.

B. *AE Generation*

In the original NSL-KDD dataset, comprising 41 feature dimensions, we categorize features into persistent and dynamic based on whether they change in real-world scenarios. We select six persistent features, including protocol type, service, flag, land, and 35 dynamic features.

To incorporate these persistent features into model training, we normalize them and append them to each sample, expanding the original 41-dimensional feature vector to a 128-dimensional one.

When perturbing the original data, we only apply perturbations to the 35 dynamic features, respecting the nature of regular tabular data, which encompasses both discrete and continuous features. For the CICIDS2018 data, we utilize the official CICFlowMeter tool to extract relevant network traffic information. After removing features with missing or infinite values, we reshape the data into a square feature matrix with dimensions of 9×9 through zero padding. Subsequently, we employ a CNN for model training and conduct AE attack testing.

To simulate real-world scenarios, we constrain the magnitude of each perturbation to be within 20% of the original data. Adversarial attacks are exclusively applied to examples correctly predicted by the DL models. We assess the transferability of AEs between different models by statistically analyzing the performance of alternative models on these AEs. Through this analysis, we evaluate the detection capability of the LID-based AE detector and the overall robustness of the DLL-IDS system. This comprehensive approach ensures that our experiments closely mimic real-world conditions, providing valuable insights into the effectiveness and reliability of our proposed defense mechanisms.

C. Evaluation

Table 1. The DL-based IDS Accuracy under adversarial attacks

Dataset	Perturbation restriction(%)	FGSM Acc(%)	BIM Acc(%)	DEEPFOOL Acc(%)	CW Acc(%)
NSL-KDD	0	94.3	94.3	94.3	94.3
	2.0	83.6	82.1	73.0	63.4
	5.0	78.3	71.5	56.7	52.2
	10.0	74.5	62.3	31.4	18.9
	20.0	65.2	23.8	15.1	5.7
	None	46.9	7.2	0	0
CICIIDS2018	0	99.3	99.3	99.3	99.3
	2.0	79.8	78.7	69.1	50.2
	5.0	57.0	53.4	52.3	40.0
	10.0	28.6	23.3	17.9	17.3
	20.0	6.1	2.0	0	0
	None	0	0	0	0

Table 1 presents the classification accuracy of the DL-based IDS when confronted with AEs generated using various attack methods under different perturbation constraints. As all clean examples were correctly classified, the prediction accuracy on the attacked dataset (referred to as Acc) is calculated. It's known from past experience that the attack intensity of these four methods increases from left to right.

The table illustrates that as the attack intensity rises, the prediction accuracy of the model declines for higher attack intensities under the same perturbation constraint. For instance, under a 5% perturbation constraint, the DL model of NSL-KDD achieves a prediction accuracy of 78.3% for AEs generated by the FGSM algorithm. However, the CW attack, the most effective attack found in this investigation, results in a prediction success rate of only 52.2% for the DL model of NSL-KDD under the same perturbation constraint. Similarly, when subjected to the same type of attack, there's an inverse relationship between the perturbation constraint and the prediction success rate, indicating that looser perturbation constraints lead to decreased model prediction accuracy.

In the case of CICIDS2018, the prediction success rate of the FGSM attack decreases from 79.8% to 6.1% under different perturbation constraints. Additionally, when there are no limitations on the perturbation, achieving a near-zero prediction success rate becomes relatively easy, indicating almost perfect attack success. However, the rationale behind generating such adversarial examples without any perturbation constraints requires further consideration.

Following the generation of numerous AEs using different attack methods, the effectiveness of the LID AE detector, as discussed earlier, needs evaluation. Among logistic regression (LGR), decision tree classifier (DTC), Bernoulli naive Bayes (BNB), and support vector machine (SVM), we observe that BNB shows almost no effectiveness, exhibiting a very high False Positive Rate (FPR). Among the remaining three algorithms, SVM and DTC display significantly better performance than LGR, with SVM ultimately chosen due to its relatively superior performance. To ensure informative experimental results, we select DB as an artifact, a method deemed effective in IDS AE detection, for comparison. Table 3 demonstrates the performances of our LID Detector and DB Detector in AEs generated from the DL-based IDS model using different attacks, comparing their detection accuracies with two types of samples: AEs and clean examples, generated with perturbation sizes set at 5%, 10%, and 20% for each attack type.

IX. CONCLUSION

This study is dedicated to addressing the classification challenges posed by real-world traffic samples acting as adversarial examples (AEs) within the context of intrusion detection systems (IDS). Four prominent adversarial attack algorithms were carefully selected and applied to network traffic features under stringent perturbation constraints. The efficacy of these attacks was examined across various perturbation conditions, revealing their highly disruptive nature.

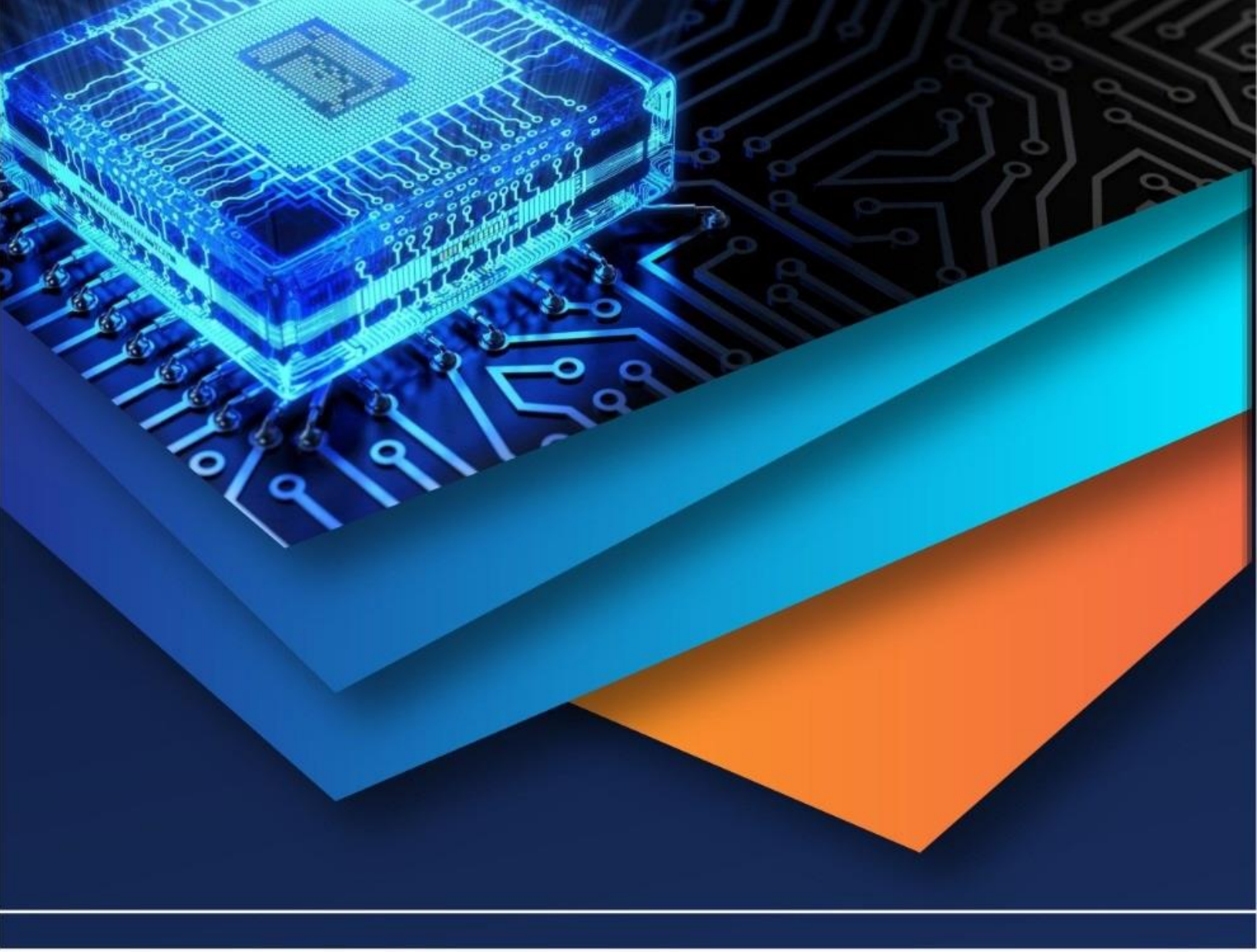
Subsequently, a novel framework called DLL-IDS was developed to counter these attacks, comprising three key components: a DL-based IDS, an AE detector, and an ML-based IDS. While investigating AEs in traffic data, it was observed that they displayed distinct spatial distribution characteristics compared to clean examples. To address this, the Local Intrinsic Dimensionality (LID) method was introduced to construct a highly effective AE detector. This detector demonstrated remarkable detection rates and required minimal prior knowledge to effectively identify the majority of AEs.

Furthermore, leveraging the limited transferability of attacks between ML and traditional DL models, a robust ML model was identified to aid in determining the maliciousness of AEs. Experimental results showcased significant enhancements in IDS performance when subjected to adversarial attacks after implementing the DLL-IDS framework. For instance, the accuracy, as measured by the highest successful attack rate using the CW attack, saw a remarkable increase from 17.9% to 71.7%.

The proposed framework holds broad applicability across various scenarios where attention is warranted on both the intrinsic properties of samples and their adversarial characteristics. Theoretically, this framework can be adapted to address security concerns introduced by deep learning models in diverse domains. It is hoped that future research will increasingly focus on exploring security issues stemming from deep learning models in other application domains.

REFERENCES

[1] He, K., Kim, D. D., & Asghar, M. R. (2023). Adversarial Machine Learning for Network Intrusion Detection Systems: A Comprehensive Survey. IEEE Communications Surveys & Tutorials, 25, 1, 538-566.
[2] Rigaki, M. (2017). Adversarial deep learning against intrusion detection classifiers. Luleå Univ. Technol., Luleå, Sweden.
[3] Wang, Z. (2018). Deep learning-based intrusion detection with adversaries. IEEE Access, 6, 38367-38384.
[4] Sahani, R., Shatabdinalini, Rout, C., Chandrakanta Badajena, J., Jena, A. K., & Das, H. (2018). Classification of intrusion detection using datamining techniques. In Progress in Computing, Analytics and Networking: Proceedings of ICCAN 2017 (pp. 753-764). Springer Singapore.
[5] Yan, B., & Han, G. (2018). Effective feature extraction via stacked sparse autoencoder to improve intrusion detection system. IEEE Access, 6, 41238-41248.
[6] Ghanem, K., Aparicio-Navarro, F. J., Kyriakopoulos, K. G., Lambotharan, S., & Chambers, J. A. (2017, December). Support vector machine for network intrusion and cyber-attack detection. In 2017 sensor signal processing for defence conference (SSPD) (pp. 1-5). IEEE.
[7] Alalousi, A., Razif, R., AbuAlhaj, M., Anbar, M., & Nizam, S. (2016). A preliminary performance evaluation of K-means, KNN and EM unsupervised machine learning methods for network flow classification. International Journal of Electrical and Computer Engineering, 6(2), 778.
[8] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2013). Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199.
[9] Lin, Z., Shi, Y., & Xue, Z. (2022, May). Idsgan: Generative adversarial networks for attack generation against intrusion detection. In Advances in Knowledge Discovery and Data Mining: 26th Pacific-Asia Conference, PAKDD 2022, Chengdu, China, May 16–19, 2022, Proceedings, Part III (pp. 79-91). Cham: Springer International Publishing.
[10] Alhajjar, E., Maxwell, P., & Bastian, N. (2021). Adversarial machine learning in network intrusion detection systems. Expert Systems with Applications, 186, 115782.
[11] Alhajjar, E., Maxwell, P., & Bastian, N. (2021). Adversarial machine learning in network intrusion detection systems. Expert Systems with Applications, 186, 115782.
[12] Sitawarin, C., & Wagner, D. (2019, May). On the robustness of deep k-nearest neighbors. In 2019 IEEE Security and Privacy Workshops (SPW) (pp. 1-7). IEEE.
[13] Carter, K. M., Raich, R., & Hero III, A. O. (2009). On local intrinsic dimension estimation and its applications. IEEE Transactions on Signal Processing, 58(2), 650-663.
[14] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2013). Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199.
[15] Carlini, N., & Wagner, D. (2017, May). Towards evaluating the robustness of neural networks. In 2017 IEEE Symposium on Security and Privacy (SP) (pp. 39-57). IEEE.
[16] Clements, J., Yang, Y., Sharma, A. A., Hu, H., & Lao, Y. (2021, December). Rallying adversarial techniques against deep learning for network security. In 2021 IEEE Symposium Series on Computational Intelligence (SSCI) (pp. 01-08). IEEE.
[17] Sheatsley, R., Papernot, N., Weisman, M. J., Verma, G., & McDaniel, P. (2022). Adversarial examples for network intrusion detection systems. Journal of Computer Security, (Preprint), 1-26.
[18] Yang, K., Liu, J., Zhang, C., & Fang, Y. (2018, October). Adversarial examples against the deep learning based network intrusion detection systems. In MILCOM 2018-2018 ieee military communications conference (MILCOM) (pp. 559-564). IEEE.
[19] Fredrikson, M., Jha, S., & Ristenpart, T. (2015, October). Model inversion attacks that exploit confidence information and basic countermea┐ sures. In Proceedings of the 22nd ACM SIGSAC conference on computer and communications security (pp. 1322-1333).
[20] Guo, C., Rana, M., Cisse, M., & Van Der Maaten, L. (2017). Countering adversarial images using input transformations. arXiv preprint arXiv:1711.00117.
[21] Athalye, A., Carlini, N., & Wagner, D. (2018, July). Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In International conference on machine learning (pp. 274-283). PMLR.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY