



REPUBLIC OF TÜRKİYE

ALTINBAŞ UNIVERSITY

Institute of Graduate Studies

Electrical and Computer Engineering

**FUSING NETWORK TRAFFIC FEATURES WITH
HOST TRAFFIC FEATURES FOR AN IMPROVED
NETWORK INTRUSION DETECTION SYSTEM**

Estabraq Saleem Abduljabbar ALARS

Doctor of Philosophy

Supervisor

Assoc. Prof. Dr. Sefer KURNAZ

İstanbul, 2025

**FUSING NETWORK TRAFFIC FEATURES WITH
HOST TRAFFIC FEATURES FOR AN IMPROVED
NETWORK INTRUSION DETECTION SYSTEM**

Estabraq Saleem Abduljabbar ALARS

Electrical and Computer Engineering

Doctor of Philosophy

ALTINBAŞ UNIVERSITY

2025

The dissertation titled FUSING NETWORK TRAFFIC FEATURES WITH HOST TRAFFIC FEATURES FOR AN IMPROVED NETWORK INTRUSION DETECTION SYSTEM prepared by ESTABRAQ SALEEM ABDULJABBAR ALARS and submitted on 24/01/2025 has been **accepted unanimously** for the degree of PhD in Electrical and Computer Engineering

Assoc. Prof. Dr. Sefer KURNAZ

Supervisor

Thesis Defense Committee Members:

Assoc. Prof. Dr. Sefer KURNAZ

Department of Computer
Engineering,
Altınbaş University

Prof. Dr. Osman Nuri UÇAN

Department of Electrical-
Electronic Engineering,
Altınbaş University

Asst. Prof. Dr. Abdullahi Abdu
IBRAHIM

Department of Computer
Engineering,
Altınbaş University

Asst. Prof. Dr. Serdar KARGIN

Department of Biomedical
Engineering,
İstanbul Arel University

Asst. Prof. Dr. Zeynep ALTAN

Department of Software
Engineering,
İstanbul Beykent University

I hereby declare that this dissertation meets all format and submission requirements for a PhD thesis.

I hereby declare that all information/data presented in this graduation project has been obtained in full accordance with academic rules and ethical conduct. I also declare all unoriginal materials and conclusions have been cited in the text and all references mentioned in the Reference List have been cited in the text, and vice versa as required by the abovementioned rules and conduct.

Estabraq Saleem Abduljabbar ALARS

Signature



DEDICATION

I devote and pledge this research work to my supervisor who is salient for guiding me through whole research work as well as my family for always assisting me in my hard time.



PREFACE

First and foremost, I would like to thank my supervisor Asst. Prof. Dr. Sefer Kurnaz for guiding and helping me along the way in writing this dissertation. Discussing my progress, problems, and ideas with my supervisor Asst. Prof. Dr. Sefer Kurnaz a couple of times every week helped me tremendously in understanding the logic behind the research. It made me better realize the technical need for this research work.



ABSTRACT

FUSING NETWORK TRAFFIC FEATURES WITH HOST TRAFFIC FEATURES FOR AN IMPROVED NETWORK INTRUSION DETECTION SYSTEM

ALARS, Estabraq Saleem Abduljabbar

Ph.D., Electrical and Computer Engineering, Altınbaş University,

Supervisor: Assoc. Prof. Dr. Sefer

KURNAZ Date: 01 / 2025

Pages: 94

Network security is a key concern in today's linked world as cyber threats grow more sophisticated and ubiquitous. Traditional Network Intrusion Detection Systems (NIDS) generally fall short owing to their dependence on predetermined signatures and restricted detection scope, exposing substantial gaps in efficiently recognizing new and unanticipated intrusions. This research tackles these difficulties by merging network and host traffic data with sophisticated deep learning algorithms to boost NIDS performance. Utilizing the Network Intrusion Detection dataset, which comprises multiple intrusion scenarios replicated in a military network context, our technique involves painstaking data collection, preprocessing, and feature extraction. We employed a convolutional neural network (CNN) to assess these data, applying rigorous feature selection and dimensionality reduction to enhance model performance. The findings reveal that our deep learning-based NIDS achieves an amazing detection accuracy of 98.5%, exceeding current approaches and successfully resolving real-world cybersecurity problems. This complete approach not only develops NIDS technology but also provides a practical solution for boosting network security across many applications, therefore contributing to the development of intrusion detection systems.

Keywords: Deep Learning, Network Intrusion Detection System, Cybersecurity, Feature Extraction, Convolutional Neural Network, Network Traffic Analysis, Host Traffic Features.



TABLE OF CONTENTS

	<u>Pages</u>
ABSTRACT	vii
LIST OF TABLES.....	xii
LIST OF FIGURES.....	xiii
ABBREVIATIONS.....	xv
1. INTRODUCTION	1
1.1 RESEARCH MOTIVATION	4
1.2 PROBLEM STATEMENT	7
1.3 NOVELTY OF RESEARCH.....	9
1.4 INTEGRATING NETWORK TRAFFIC FEATURES WITH HOST TRAFFIC FEATURES	10
1.5 WIRELESS NETWORK VS FUSING NETWORK.....	12
1.6 RESEARCH PERFORMANCE EVALUATION AND METRICS	13
1.7 AIM OF CONTRIBUTION	15
2. LITERATURE REVIEW	18
2.1 INTRUSION DETECTION SYSTEMS (IDS)	21
2.1.1 Network Intrusion Detection System.....	24
2.1.1.1 Signature-based detection	26
2.1.1.2 Anomaly-based detection.....	28
2.1.2 Host-Based Intrusion Detection Systems (HIDS).....	30
2.2 THE ROLE OF ARTIFICIAL INTELLIGENCE IN INTRUSION DETECTION SYSTEMS	32
2.2.1 Supervised Learning	33
2.2.2 Unsupervised Learning	33
2.2.3 Reinforcement Learning	34
2.2.4 Comprehensive Guidance	34
2.3 INTRUSION DETECTION TECHNIQUES WITH MACHINE LEARNING ALGORITHMS	34

2.3.1	Decision Trees	34
2.3.2	Support vector machines.....	34
2.3.3	Artificial Neural Networks ANNs	35
2.3.4	K-Nearest Neighbours algorithm	35
2.3.5	Naïve Bayes Encoder.....	35
2.4	IMPLEMENTATION OF DEEP LEARNING TECHNOLOGIES FOR INTRUSION DETECTION	36
2.4.1	Convolutional Neural Networks (CNNs).....	36
2.4.2	Recurrent Neural Networks (RNN)	36
2.4.3	Image Encoders.....	36
2.4.4	Deep Belief Networks (DBNs)	37
2.5	SELECTION OF FEATURES AND REDUCTION OF DIMENSIONS.....	37
2.5.1	Feature Selection with Filters	37
2.5.2	Feature Selection Using Wrappers.....	37
2.5.3	Selecting Internal Features.....	38
2.5.4	Removal of Dimensions.....	38
2.6	THE IDS SYSTEMS FOR IDENTIFICATION AND CLASSIFICATION	38
2.6.1	Misuse-Based Determination.....	39
2.6.2	Anomaly-Based Detection.	39
2.6.3	Detection using specified Criteria.....	40
3.	METHODOLOGY	52
3.1	WIRELESS NETWORK MODELLING	54
3.2	DATASET DESCRIPTION	55
3.3	MODEL PARAMETERS.....	56
3.4	CROSS-VALIDATION AND PERFORMANCE METRICS	58
	Confusion Matrix	59
3.5	FUSING HOST TRAFFIC INTO FEATURES	59
3.6	FLows AND CONNECTIONS IN WN	60
3.7	INTRUSION DETECTION AND INSPECTION	61

4. RESULTS	63
5. CONCLUSION	71
5.1 FUTURE WORK.....	71
REFERENCES	73



LIST OF TABLES

	<u>Pages</u>
Table 2.1: Literature Review of Wireless Network Intrusion Detection Using Artificial Intelligence Techniques with Performance Metrics.	43
Table 3.1: Model Parameters Summary	56
Table 3.2: Performance Metrics from 10-Fold Cross-Validation.....	59
Table 3.3: Confusion Matrix (4-Fold Cross-Validation).....	59
Table 3.4: Python Code Statistics for Intrusion Detection Within Files.	62
Table 4.1: Key Performance Metrics of the Proposed NIDS	63
Table 4.2: IDS Predicted vs IDS Actual Values for Wireless Network Bandwidth Ratio with a Single Host, 15 Nodes, Standard Deviation, Accuracy and 350 Hours of Training.	66
Table 4.3: Classification Results Experiment of Intrusions in Wireless Network.	67
Table 4.4: Comparative Analysis with Contemporary Approaches	70

LIST OF FIGURES

	<u>Pages</u>
Figure 1.1: Wireless Network Provisioning Schema [7].....	2
Figure 1.2: Computer Networks are Safeguarded Against Possible Security Threats by two Types of Security Systems: Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) [10].....	3
Figure 1.3: Deploying an Intrusion Detection System (IDS) Comes With its Fair Share of Difficulties [14].	6
Figure 2.1: The Fundamental Framework of Intrusion Detection Systems (IDS) Versus Intrusion Prevention Systems (IPS) Serves as Essential Tools in Contemporary Security Stacks [35].	18
Figure 2.2: The Conceptual Model Demonstrates how Active Security Measures Effectively Reduce Time Consumption and Prevent Assaults [42].	20
Figure 2.3: NIDS and HIDS are Distinct Categories of Intrusion Detection Systems Employed to Identify Network and Host-centric Assaults, Respectively [49].....	25
Figure 2.4: HIDS can Identify Both External and Internal Threats, and Their Primary Advantage is Their Ability to Provide in-depth Information About the Specific Host on Which they are INSTALLED [57].	31
Figure 2.5: AI-based IDS Have the Potential to Significantly Improve the Effectiveness of Intrusion Detection by Enabling More Accurate Detection of Attacks, Reducing False Positives, and Improving the Efficiency of Incident Response [59].	33
Figure 2.6: Classification of IDS Based on Their Mode of Operation, Which can be Either Passive or Active [78].	38
Figure 3.1: Propose system for Network Intrusion Detection Systems with Combined Network and Host Traffic Features Using Deep Learning.	53

Figure 3.2: The Area Under the Receiver Operating Characteristic (AUROC) Score Was Calculated for the Training Sample Across Several Sorts of Attacks.	55
Figure 3.3: The Ringing Concept in Wireless Network for Attacker Identification.	60
Figure 3.4: Feature Engineering Workflow.....	61
Figure 4.1: Standard Normal Quantiles Represent the Ids Predicted vs Ids Actual Values For Wireless Network Bandwidth Ratio With Single Host.	63
Figure 4.2: Standard Normal Quantiles Represents the IDS Predicted vs IDS Actual Values for Wireless Network Bandwidth Ratio with Multiple Hosts.	64
Figure 4.3: The Difference Between the IDS Predicted vs IDS Actual Values for Wireless Network with Multiple Hosts.	65
Figure 4.4: Confusion Matrix of the Proposed Model.	68
Figure 4.5: ROC Curve of the Proposed Model.	68
Figure 4.6: Training and Validation Loss Over Epochs.....	69
Figure 4.7: Feature Importance in the Model.	70

ABBREVIATIONS

CNN	:	Convolutional Neural Network
IDS	:	Intrusion Detection Systems
IPS	:	Intrusion Prevention Systems
QoS	:	Quality Of Service
RNN	:	Recurrent Neural Network
WN	:	Wireless Network



1. INTRODUCTION

As stated in [1], intrusion detection is a crucial component of network security, as it helps to prevent assaults, hostile actions, and unwanted access to a wireless network. The increasing complexity and sophistication of assaults on wireless networks require increasingly advanced intrusion detection systems capable of detecting and preventing such attacks. Enhanced wireless network intrusion detection employs cutting-edge methods and tools, including deep learning algorithms, artificial intelligence, and machine learning, as referenced in [2], to promptly identify and thwart assaults. The objective of enhancing wireless network intrusion detection is to develop a system that can effectively and precisely identify assaults while decreasing both false positives and false negatives. In order to do this, many kinds of information are consolidated, including statistics on host and network traffic, as well as other relevant data that may be utilized for the purpose of identifying and thwarting attacks. As stated in reference [3], utilizing advanced technique and technology can enhance wireless network intrusion detection, resulting in enhanced security and protection against a wide range of assaults that might happen on a wireless network. Gaining proficiency in the identification of intrusions in wireless networks is of utmost importance in the realm of computer security. The growing intricacy of wireless networks leads to a rise in both the quantity and complexity of potential attacks that may be carried out against them. Intrusion detection systems (IDS) are crucial instruments for identifying and preventing such endeavors, as referenced in [4].

The latest breakthroughs in artificial intelligence (AI) and machine learning (ML) have significantly improved the capability of intruder detection systems (IDS) to identify and respond to unauthorized access in wireless networks. These methodologies enable the examination of network data to identify abnormalities that might potentially signify a cyber assault. As stated in reference [5], the integration of host and network traffic data enables a more thorough examination of network activities, hence enhancing the precision of intrusion detection systems. Wireless network intrusion detection still has much progress to make in order to achieve complete effectiveness. For instance, the substantial amount of network data might provide challenges in promptly identifying and responding to potential threats. Furthermore, it is imperative to develop novel detection methods in order to keep pace with the constantly evolving strategies employed by assailants. Notwithstanding these

difficulties, there is a growing corpus of research (as referenced in [6]) dedicated to enhancing wireless network intrusion detection. Scientists are utilizing cutting-edge breakthroughs in artificial intelligence (AI) and machine learning (ML) to develop more intricate and efficient methods for identifying and mitigating threats. Ultimately, these endeavors will contribute to the preservation and protection of wireless networks, enabling companies to function with enhanced security and assurance in the era of digital technology (as demonstrated in [7]).

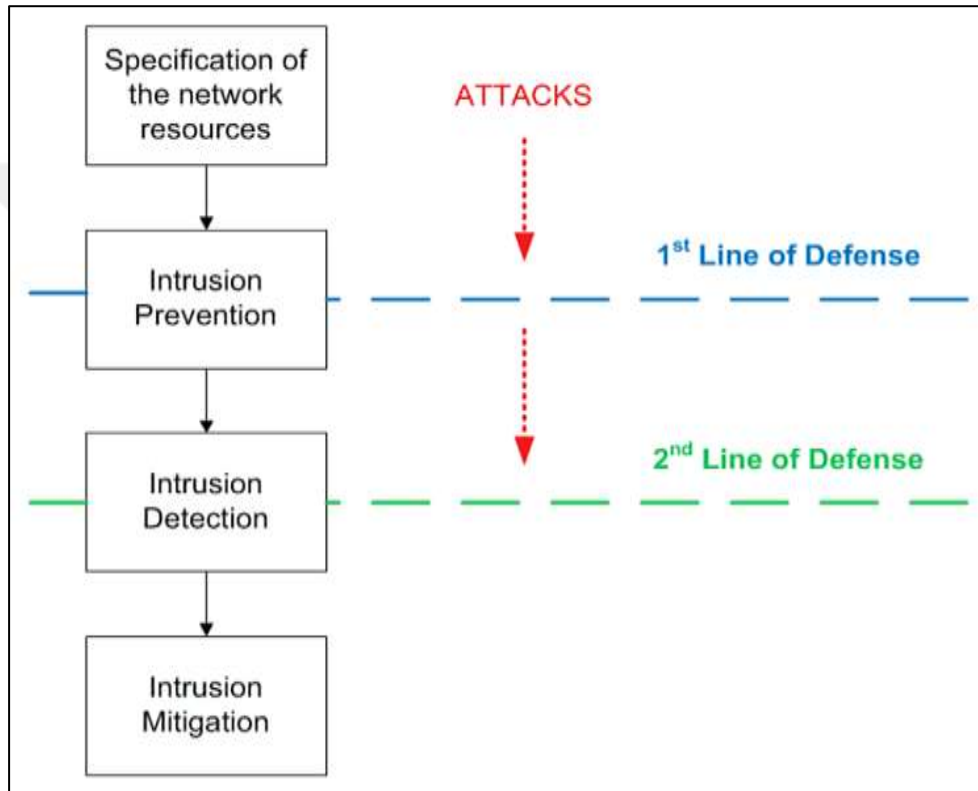


Figure 1.1: Wireless Network Provisioning Schema [7].

The expansion of wireless networks is a direct consequence of the growing number of interconnected devices, including smart homes, industrial sensors, wearables, and the Internet of Things (IoT). Nevertheless, the occurrence of hacks becomes more prevalent as the quantity of interconnected devices grows. To detect and prevent such assaults, intrusion detection systems (IDS) analyze network data for patterns that may suggest malicious activity, as referenced in [8]. Traditional intrusion detection systems (IDS) may not be effective in wireless networks due to their distinct attributes, including radio frequency interference, restricted bandwidth, and mobility. Scientists are now employing artificial intelligence (AI) and machine learning (ML) methods to create improved intrusion detection

systems for wireless networks, aiming to overcome these issues. As stated in reference [9], intrusion detection systems (IDS) that utilize machine learning have the capability to enhance detection rates and reduce the occurrence of false positives by analyzing large volumes of data to identify patterns and anomalies.

In addition to machine learning, researchers have also investigated alternative methods such as deep learning, ensemble learning, and hybrid models for the purpose of wireless network intrusion detection. These systems strive to enhance detection accuracy and reduce false alarms by using many methodologies. Enhancing intrusion detection is a crucial field of study that is continuously progressing as wireless networks become more intricate and widely used. According to the reference [10], it is expected that these systems would improve their ability to identify and prevent cyberattacks as new techniques such as machine learning progress.

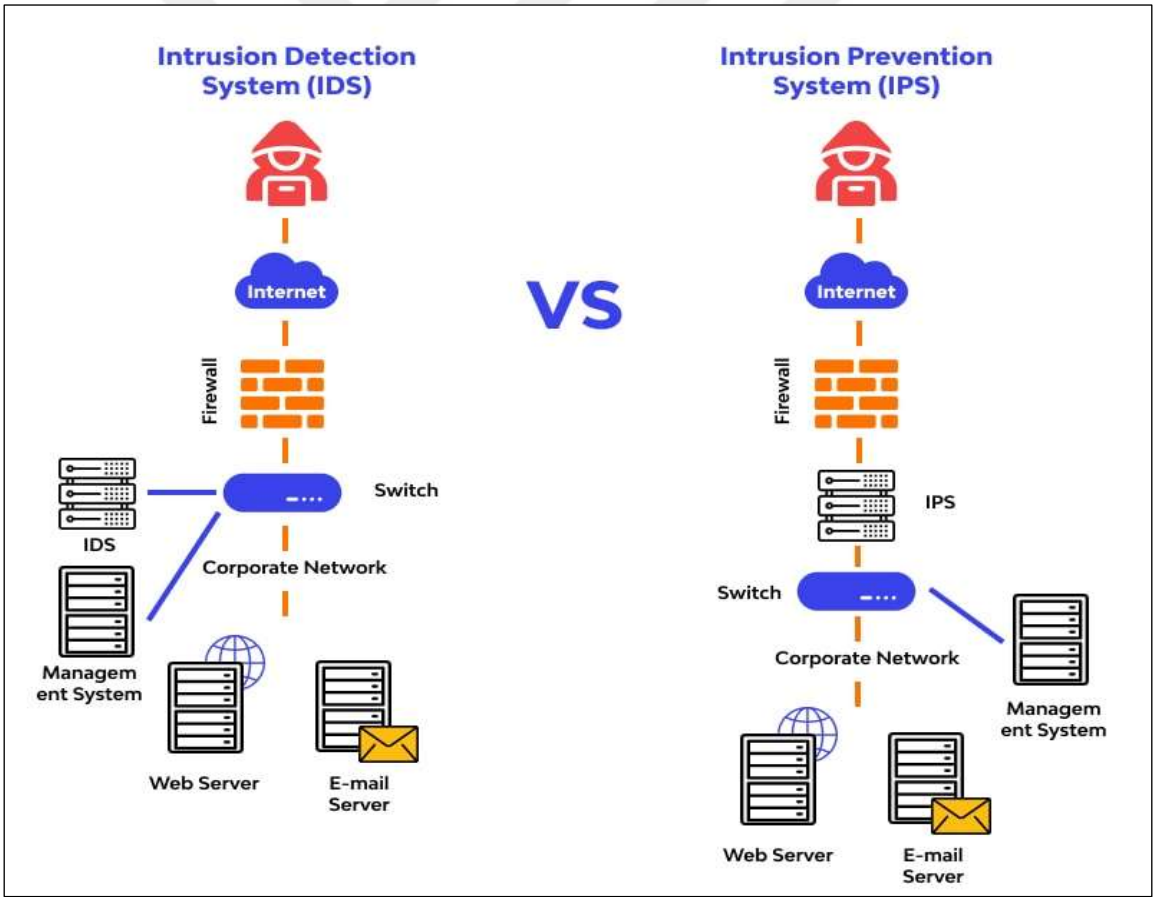


Figure 1.2: Computer Networks are Safeguarded Against Possible Security Threats By Two Types of Security Systems: Intrusion Detection Systems (Ids) and Intrusion Prevention Systems (IPS) [10].

Intrusion Detection Systems (IDS) are a type of security software that passively analyzes network traffic to detect any indications of unauthorized access. The intrusion detection system promptly notifies the security team upon detecting any potential threat. As stated in [11], there are two types of intrusion detection systems: host-based and network-based. Host-based systems monitor activity on particular hosts, whereas network-based systems monitor portions of networks. Intrusion prevention systems (IPS) are proactive security systems that actively detect and counter possible security hazards in order to avert any harm or damage. IPS has the capability to obstruct specific forms of communication and may also impede the functioning of IP addresses, whether they are originating from or directed towards a particular location. Furthermore, it can even disable whole network ports. Although IDS and IPS have a same goal of identifying and stopping security threats, they employ distinct methods, as referenced in [12]. Unlike intrusion prevention systems (IPS), intrusion detection systems (IDS) are non-active and just activate an alarm upon the identification of a threat. The choice of an intrusion detection system (IDS) or intrusion prevention system (IPS) for a corporation is determined by its level of risk acceptance and unique security requirements.

1.1 RESEARCH MOTIVATION

Scientists are now investigating methods to enhance intrusion detection systems (IDS) by using network and host traffic characteristics. This is due to the fact that conventional Intrusion Detection Systems (IDSs) solely monitor a single type of traffic data, hence allowing ample scope for enhancement. For instance, a cyber assault in which a hacked server generates harmful data may evade detection by network-based intrusion detection systems (IDSs) that just monitor network traffic. Nevertheless, host-based intrusion detection systems may be inadequate in identifying network-based attacks as they just monitor the traffic on individual hosts. Integrating host traffic characteristics with network traffic properties allows intrusion detection systems to have a deeper understanding of the network environment and effectively identify more complex assaults involving many hosts and several protocols. The growing intricacy and magnitude of data produced by hosts and networks has also rendered intrusion detection a more challenging undertaking. Conventional intrusion detection systems may have difficulties in keeping pace with constantly evolving assaults due to their reliance on fixed signatures or criteria. In order to

accurately identify assaults and assess large amounts of data, it is necessary to employ more sophisticated intrusion detection systems (IDS). Researchers are combining host traffic attributes and network traffic features to enhance the effectiveness of intrusion detection systems (IDS), as stated in [13]. This solution would address the limitations of conventional Intrusion Detection Systems (IDS) by enabling the detection and prevention of sophisticated assaults that may potentially go unnoticed by systems that just monitor a single kind of network traffic. This strategy has the potential to improve both data loss prevention and network security. The increasing significance of network security in several industries, such as healthcare, banking, and the military, together with the intricacy of attacks, has intensified research in this field. To ensure dependable security measures across various settings, a sophisticated intrusion detection system is required. The objective of research in enhancing intrusion detection systems by the integration of host and network traffic characteristics is to strengthen network defenses against advanced assaults that conventional IDS systems may be unable to identify. Given the escalating intricacy of assaults and the rising importance of network security in many industries, there is a demand for more sophisticated intrusion detection systems (IDS). These systems should possess the capability to categorize acts of aggression through the utilization of data analysis and extensive data analysis on a broad scale. The objective of an enhanced intrusion detection system (IDS) is to overcome the limitations of the conventional IDS and offer a superior and more efficient approach to identifying and thwarting assaults, as referenced in [14].

Traditional intrusion detection systems (IDS) depend on predefined lists of recognized attacks or signatures to detect possible threats. This might result in an excessive number of incorrect identifications and the inability to identify advanced attacks that deviate from known patterns. The growing intricacy and magnitude of data produced by hosts and networks has also rendered intrusion detection a more formidable undertaking.

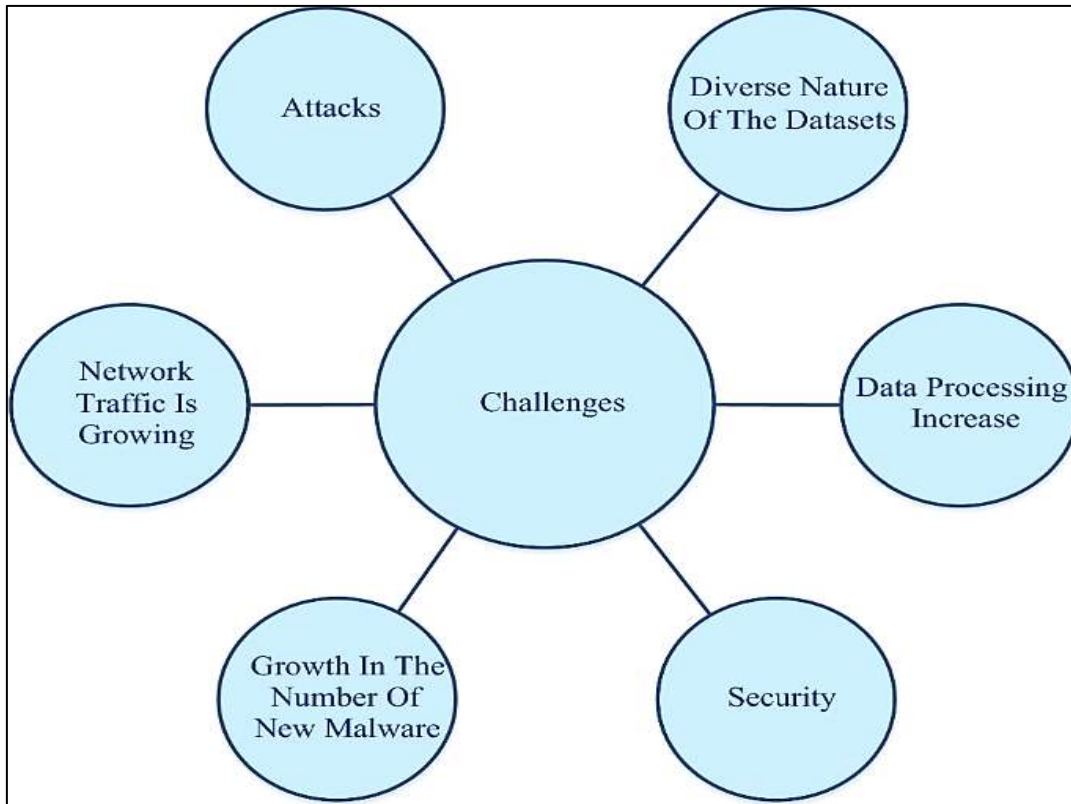


Figure 1.3: Deploying an Intrusion Detection System (IDS) Comes With its Fair Share of Difficulties [14].

An effective approach to address these problems is an enhanced Intrusion Detection System (IDS) that incorporates cutting-edge methodologies including data fusion, machine learning, and behavior analysis. By employing these methods, a more extensive understanding of the network environment may be attained. Consequently, this enables the intrusion detection system (IDS) to identify more complex assaults, which might include several hosts and diverse network protocols. Machine learning algorithms have the ability to automate feature selection and extraction, as stated in reference [15], which sets them apart from human feature selection.

By employing data fusion techniques, several forms of traffic data may be merged, enabling the extraction of crucial information for intrusion detection. Consequently, the precision of identifying attacks can enhance while the occurrence of false positives diminishes. Behavior analysis can also identify future assaults and atypical activity that deviates from the regular patterns. Organizations such as the military, financial institutions, and healthcare facilities can get significant advantages from implementing a customized and enhanced Intrusion Detection System (IDS). As mentioned in [16], traditional intrusion detection systems may

not fully address the specialized security needs of some sectors.

An enhanced intrusion detection system aims to achieve a more effective and efficient approach to detecting and preventing incursions. Contemporary methodologies such as data fusion, behavior analysis, and machine learning have the potential to improve the accuracy and effectiveness of intrusion detection. According to reference [17], an enhanced intrusion detection system (IDS) may be customized to specific domains and scenarios in order to offer customized security solutions.

1.2 PROBLEM STATEMENT

Every technological device, regardless of its usage in domestic or professional settings, is inherently flawed. The probability of success for attackers is contingent upon the level of work and resources that prospective adversaries are willing to invest. Various published methodologies have investigated the potential of improving intrusion detection systems by integrating host traffic characteristics with network traffic factors. One way, as noted in [18], is to utilize machine learning techniques such as ensemble learning and deep learning to automate the process of feature extraction and selection. By minimizing the need for manual feature extraction and selection, this can enhance the efficiency and precision of the IDS. Data fusion techniques can be employed to integrate several forms of traffic data in order to obtain valuable information for intrusion detection. To enhance the precision of attack detection, intrusion detection systems (IDS) have used data fusion techniques such as fuzzy logic, Dempster-Shafer theory, and Kalman filtering (as referenced in [19]). When attempting to enhance Intrusion Detection Systems (IDS) by including host traffic and network traffic attributes, there are supplementary obstacles that extend beyond technical aspects. Employing IDS raises other ethical and legal problems. As stated in reference [20], monitoring host traffic data might potentially expose personally identifying information of consumers or the company, hence raising privacy concerns. The efficacy of intrusion detection systems (IDS) is determined by both their technological capabilities and organizational variables, including management support, user awareness, and training. Lack of proper instruction from higher-level executives and insufficient knowledge of the technology among end users might potentially result in data loss and security breaches when employing intrusion detection systems (IDS). A comprehensive solution is needed to address the intricate and multifaceted issue of enhancing intrusion detection systems by integrating

host traffic characteristics with network traffic factors. Data fusion techniques and machine learning algorithms can be employed to tackle technical difficulties, such as feature extraction and selection. Multiple elements, such as organizational matters, legal and ethical concerns, and other considerations, need to be taken into account to guarantee the efficient use and implementation of IDS, as stated in reference [21]. In order to render the monitoring system inoperative, unscrupulous individuals may exploit weaknesses, manipulate, or circumvent its functionality. There is a possibility that the network monitor might inadvertently gain entry to the network it was intended to protect, rather than enhancing overall system security. After two decades of research on anomaly-based intrusion detection, the following essential problems remain unresolved:

- a. Which components are the most crucial, and do they necessitate extraction?
- b. Which set of features from wireless networks produces the most optimal outcomes when evaluated using different datasets and methodologies?
- c. Which algorithms exhibit the most optimum network characteristics?
- d. The most efficient method to describe the process of attack categorization and cluster-based intrusion detection is to identify and classify different types of attacks and then use clustering techniques to discover patterns and anomalies in network traffic.
- e. Which type of summary structures, such as connections and flows, or procedure-specific observations, yield superior outcomes?
- f. Which strategy is the most efficient when it comes to utilizing computer resources?

Efficient technology for extracting, selecting, and aggregating features is essential for accurately solving these challenges. The lack of disclosure of methodology by many researchers is responsible for the challenge of replicating and confirming study results, as well as the prolonged duration of further research on the issue. Furthermore, as demonstrated in reference [22], it is unfeasible to adequately represent the safeguarded network environment by the use of counterfeit datasets for the purpose of validating network security monitoring. To meet the need of identifying and stopping sophisticated network attacks involving many hosts and various network protocols, network intrusion detection systems (IDS) are improved by merging host traffic characteristics with network traffic components.

According to reference [23], if typical intrusion detection systems (IDS) just focus on monitoring host or network traffic and do not identify these sorts of attacks, there is a risk of data loss and security vulnerabilities. Hence, this study aims to improve the Intrusion Detection System (IDS) by integrating data on host traffic with information on network traffic. By broadening the IDS's perspective of the network in this manner, it enhances its capability to detect and prevent intricate attacks that may include numerous hosts and protocols. However, the process of extracting and selecting features can be time-consuming and costly due to the intricate nature and vast volume of data generated by host and network traffic. As mentioned in [24], this poses a challenge in developing a high-performing and accurate Intrusion Detection System (IDS) capable of processing large amounts of data and accurately identifying attacks. The objective of integrating network and host traffic elements into a single intrusion detection system (IDS) is to enhance network security and defend against sophisticated assaults that may go undetected by individual monitoring systems. The objective is to create a system capable of efficiently and accurately examining vast quantities of data from various traffic sources, identifying abnormal patterns, and accurately categorizing assaults.

1.3 NOVELTY OF RESEARCH

A novel enhanced network intrusion detection system (IDS) is created by combining network traffic data with host traffic features, resulting in a more complete and precise understanding of the network environment. This technology enhances the intrusion detection system (IDS) by consolidating data from hosts and network traffic, enabling the detection of more sophisticated assaults. These attacks can include several hosts and a variety of network protocols.

This work is fascinating since it use machine learning to automate the process of extracting and selecting features. By minimizing the need for human feature selection and extraction, this method enhances the efficiency and precision of the IDS. The study contributes to the improvement of intrusion detection systems by boosting the precision and effectiveness of network-based attack detection. Implementing this strategy can enhance network defenses and security by effectively thwarting complex assaults that might otherwise go unnoticed by systems that just monitor a single kind of traffic. This study enhances the topic of network security by emphasizing the need of comprehending the fundamental operating systems and

protocols. In order to enhance intrusion detection systems, it is crucial to incorporate both host and network traffic characteristics. This data is necessary for efficient extraction and selection of features.

The use of machine learning techniques to automate the process of extracting and selecting features, together with the capability to offer a more complete and precise understanding of the network environment, is an innovative approach to integrating network traffic data with host traffic characteristics for an enhanced Intrusion Detection System (IDS). This technique improves network security and safeguards against advanced threats by advancing the development of more efficient intrusion detection systems (IDS).

1.4 INTEGRATING NETWORK TRAFFIC FEATURES WITH HOST TRAFFIC FEATURES

Integrating network traffic data with host traffic information is one way to improve the accuracy and efficiency of detecting network-based attacks in intrusion detection systems (IDS). The characteristics of data generated and received by hosts are known as host traffic features, whereas the characteristics of data moved between hosts are known as network traffic features. The integration of host traffic and network traffic elements allows IDS to detect more sophisticated assaults, as mentioned in [25], that may include several hosts and various network protocols. If an intrusion detection system (IDS) only monitors network traffic, it may fail to notice an attack that uses network-based vulnerabilities or involves a compromised host that generates malicious data. The quantity and complexity of data that has to be reviewed makes it challenging to combine host traffic aspects with network traffic characteristics. Machine learning techniques, such as deep learning and ensemble learning, have proven to be effective in handling and classifying such data.

Several studies have looked into the potential benefits of integrating host and network traffic characteristics in intrusion detection systems. An intrusion detection system (IDS) that incorporates host and network traffic data using a hybrid deep learning method was proposed by the authors of a research study [26]. The system uses a stacked auto-encoder to extract features from network traffic data and a convolutional neural network to extract features from host traffic data. The data is then classified using a support vector machine classifier that is given the combined features.

An intrusion detection system (IDS) that incorporates host and network traffic features using a transfer learning approach was proposed in another study [27]. The system uses a pre-trained convolutional neural network (CNN) to extract features from data on network traffic and a pre-trained recurrent neural network (RNN) to do the same for host traffic. The features are combined and sent into a decision tree classifier for classification. Intrusion detection systems (IDS) that integrate host traffic data with network traffic characteristics may improve the efficacy and accuracy of network-based attack detection. Although this strategy is challenging due to the complexity of the data, machine learning algorithms have shown effectiveness in evaluating and classifying such data. According to [27], integrating network and host traffic elements into intrusion detection systems is an exciting area of research that might significantly improve the accuracy and efficacy of this type of security measure. By combining information from host and network traffic elements, intrusion detection systems (IDS) are able to identify attacks that systems that just monitor one sort of traffic miss. One challenge in merging network and host traffic data is finding effective methods for feature extraction and selection. It may be both time-consuming and expensive to extract relevant features for intrusion detection from the bewildering amounts of data generated by network and host traffic. But with the help of machine learning techniques like deep learning and ensemble learning, it is possible to automate the feature extraction and selection process and improve the classification models' accuracy.

Another challenge is the need for in-depth knowledge of the underlying protocols and operating systems. The operating systems and specific network protocols used by hosts are important pieces of information that an intrusion detection system (IDS) that integrates host and network traffic attributes must be aware of. To do this, one must have an in-depth understanding of the systems being monitored and the network architecture. Despite all of these challenges, IDS integration of host traffic and network traffic elements has demonstrated promising results. As an example, [28] proposed an intrusion detection system that aggregates host and network traffic data using LSTM and convolutional neural networks. With the NSL-KDD dataset, the system achieved a detection rate of 99.2% and a false alarm rate of 0.5%. The combination of host traffic attributes with network traffic characteristics in intrusion detection systems has the potential to improve the accuracy and efficacy of detecting threats originating from networks. This approach requires an in-depth understanding of the underlying operating systems and network architecture as well as

effective methodologies for feature extraction and selection. Improved intrusion detection accuracy may be achieved by automating feature extraction and selection using machine learning techniques such as deep learning and ensemble learning.

1.5 WIRELESS NETWORK VS FUSING NETWORK

Wireless networks and fusing networks are two types of network architectures that have different characteristics and applications. A wireless network is a type of network that uses wireless communication technology to connect devices and transmit data without the need for physical cables. Wireless networks are commonly used in areas where it is difficult or impossible to install wired networks, such as in outdoor environments or in remote locations. Wireless networks can be classified into several categories, such as Wi-Fi, Bluetooth, and cellular networks.

In wireless networks, security is a major concern due to the broadcast nature of the wireless signals. Wireless networks are vulnerable to various types of attacks, such as eavesdropping, spoofing, and denial-of-service attacks as mentioned in [29]. Therefore, wireless network intrusion detection systems (IDS) are essential to ensure the security and integrity of wireless networks. IDS for wireless networks typically use artificial intelligence techniques, such as machine learning and deep learning, to analyze network traffic and detect anomalous behavior. Wireless networks offer several advantages, such as mobility and flexibility. With a wireless network, users can connect to the network from anywhere within the range of the wireless signal. This makes wireless networks suitable for environments where mobility is required, such as in hospitals, warehouses, and outdoor locations. Wireless networks can also be set up quickly and easily without the need for cables and infrastructure. However, wireless networks also have several disadvantages, such as susceptibility to interference and security vulnerabilities. Wireless signals can be disrupted by physical obstacles, such as walls and buildings, and can be affected by other wireless devices operating in the same frequency range. Wireless networks are also vulnerable to security threats, such as eavesdropping and unauthorized access, due to the broadcast nature of the wireless signals as mentioned in [30].

On the other hand, fusing networks are a type of network architecture that combines different types of communication technologies to provide a more robust and reliable network. Fusing networks typically integrate wired and wireless networks to ensure that communication can

continue even if one of the networks fails. For example, a fusing network may use a wired network as the primary communication channel and a wireless network as a backup channel. Fusing networks, on the other hand, offer several advantages in terms of reliability and security. By integrating multiple communication technologies, fusing networks can ensure that communication can continue even if one of the networks fails. This makes fusing networks suitable for mission-critical applications, such as emergency response systems and military operations. However, fusing networks also have several challenges, such as complexity and cost. Fusing networks require advanced techniques for network management and data fusion, such as QoS management and data aggregation. These techniques can be complex and require specialized expertise to implement and maintain. Fusing networks can also be costly to deploy and maintain due to the need for multiple communication technologies and infrastructure as mentioned in [31].

Fusing networks are commonly used in mission-critical applications, such as military and emergency response systems, where communication failures can have severe consequences. Fusing networks require advanced techniques for network management and data fusion, such as network-level quality of service (QoS) management and data aggregation as mentioned in [32].

Overall, wireless networks and fusing networks are two types of network architectures that have different applications and requirements. Wireless networks are commonly used in areas where wired networks are impractical, while fusing networks are used in mission-critical applications where network reliability is essential. Both wireless networks and fusing networks require effective network management and security measures to ensure their reliability and security.

1.6 RESEARCH PERFORMANCE EVALUATION AND METRICS

There are two types of network architectures: fusion networks and wireless networks, each with distinct applications and characteristics. A wireless network is a type of network that establishes connections between devices and facilitates the transmission of data without the need for physical connections. Wireless networks are commonly employed in environments such as rural areas or outdoor locations where the construction of cable networks is difficult or unfeasible. Wireless networks may be categorized into several types, including Wi-Fi,

Bluetooth, and cellular networks.

Wireless networks provide a significant concern for security due to the dissemination of wireless signals. As mentioned in reference [29], wireless networks can be targeted by several types of assaults, such as spoofing, denial-of-service, and eavesdropping. Therefore, the implementation of intrusion detection systems (IDS) on wireless networks is essential to ensure the security and integrity of these networks. Intrusion detection systems (IDS) for wireless networks frequently employ artificial intelligence techniques, such as machine learning and deep learning, to analyze network data and detect anomalous behavior. Wireless networks offer several advantages, such as enhanced flexibility and increased mobility. Users can establish a connection to a wireless network from any place, provided that they are within the range of the wireless signal. Wireless networks are suitable for environments such as warehouses, hospitals, and outdoor regions where the ability to move about is required. Furthermore, wireless networks may be quickly and easily established without the requirement of infrastructure or connections. Nevertheless, wireless networks do include certain disadvantages, such as vulnerabilities in security and sensitivity to interference. Obstacles such as walls and buildings can disrupt wireless communications, as can other wireless devices operating on the same frequency range. Wireless networks are vulnerable to security threats such as unauthorized access and interception due to the broadcast nature of wireless signals, as mentioned in [30].

Fusing networks, on the other hand, refer to a network architecture that enhances resilience and reliability by integrating many types of communication technologies. In order to provide uninterrupted communication in the case of a network breakdown, fusion networks frequently include both wired and wireless networks. For example, a fusing network might utilize a wireless network as a secondary channel and a wired network as the primary communication pathway. On the other hand, integrating networks offer certain advantages in terms of security and dependability. By integrating many communication modalities, fusing networks guarantees uninterrupted communication even in the case of a network breakdown. Network fusion is suitable for mission-critical applications such as emergency response systems and military operations. Nevertheless, the process of integrating networks has several challenges, such as the high cost and intricate nature of the task. Advanced approaches in network management and data fusion, such as quality of service (QoS)

management and data aggregation, are essential for combining networks. These technologies might be complex, necessitating specialized expertise for utilization and maintenance. Integrating networks may be costly to install and maintain due to the need for many communication technologies and infrastructure, as mentioned in [31].

Fusion networks are commonly employed in mission-critical applications such as emergency response systems and the military, where communication failures can have severe consequences. Network-level quality of service (QoS) management and data aggregation, as outlined in [32], exemplify advanced techniques for managing networks and combining data, which are essential for network fusion.

Broadly speaking, there exist two categories of network architectures: wireless networks and fusion networks. Each of these categories serves different purposes and possesses unique requirements. Fused networks are utilized in mission-critical applications where network reliability is paramount, whereas wireless networks are commonly deployed in locations where wired networks are impractical. Efficient network administration and security protocols are essential for ensuring the reliability and safety of both wireless networks and fusion networks.

1.7 AIM OF CONTRIBUTION

The primary objective of this project is to investigate methods for detecting disruptions in wireless networks by analyzing inconsistencies using a precise and comprehensive methodology to analyzing explicit data from access networks. The term "current" in this context refers to the attributes associated with the completed structure. An optimal programming language for wireless networks should possess task-specificity, employ a concurrent programming paradigm, promote efficient memory management, and provide support for all essential phases. Choosing an outcome design that gives priority to both information security and construction, while also preventing non-partisans from seeing highlights, is a wise decision. The several processing units in modern PC hardware may be used together in a wireless network, allowing for scalability, which is crucial for handling large-scale operations. Furthermore, it is important to provide assistance to future scientists who will utilize the service. A wireless network should have both an order line and library interface, and the chosen method should be appropriate for handling real-time data.

Additionally, there should be the capability to export documents from a corporate interface. Furthermore, this method will focus providing optimal flexibility while also enhancing information accessibility to enhance workflow and efficiency for distant firms' testing procedures. This approach will diverge from several tactics that strive to minimize the amount of information. This aligns with the notion that some internal aspects of a corporation should be disclosed prior to forming an opinion on its assortment architecture. A remote business will utilize an atypical dataset and a method centered on anomalies to assess the existing procedure. Here are few significant discoveries derived from this research:

- a. An enhanced network intrusion detection system (IDS) is developed by utilizing a blend of network traffic characteristics and host traffic features. This results in improved accuracy and effectiveness in detecting network-based assaults. By integrating host and network traffic data, the Intrusion Detection System (IDS) is able to identify increasingly complex assaults that may include many hosts and several network protocols.
- b. By combining host and network traffic aspects, a more thorough understanding of the network environment may be achieved, enabling the detection of attacks that would otherwise go unnoticed by systems that simply monitor a single kind of traffic. An intrusion detection system (IDS) that just monitors network traffic may be unable to identify an assault that exploits network-based vulnerabilities or includes a compromised host that creates harmful data.
- c. Integrating components of host and network traffic might provide challenges due to the large amount and intricate nature of data that has to be managed. Machine learning methods, such as ensemble learning and deep learning, have the ability to automate the process of extracting and selecting features. This, in turn, enhances the accuracy of the Intrusion Detection System (IDS).
- d. The main benefit of merging host traffic and network traffic components in intrusion detection systems (IDS) is the improved precision and efficiency in identifying network-based assaults. Despite the intricate nature of the data, machine learning algorithms have demonstrated their efficacy in assessing and categorizing it.
- e. Intrusion detection systems (IDS) strive to enhance the understanding of the network environment by including both host traffic and network traffic components. This enhances the efficacy of identifying intricate assaults. Comprehensive knowledge of the

fundamental operating system and network protocols is crucial for implementing this strategy, in addition to employing efficient techniques for extracting and selecting features. Enhanced precision in detecting intrusions may be achieved by automating the process of extracting and selecting features using machine learning methods.



2. LITERATURE REVIEW

The literature review section comprehensively examines and summarizes the many methods of enhancing the performance of network intrusion detection systems (NIDS). The review comprehensively examines the key aspects outlined in [35], encompassing significant ideas, methodologies, and empirical discoveries that have contributed to our current comprehension of NIDS. Additionally, it addresses the limits of NIDS and proposes prospective avenues for future enhancements. The article also discusses contemporary methodologies such as dimensionality reduction, feature selection, deep learning, and machine learning, which may be employed to enhance the effectiveness of intrusion detection.

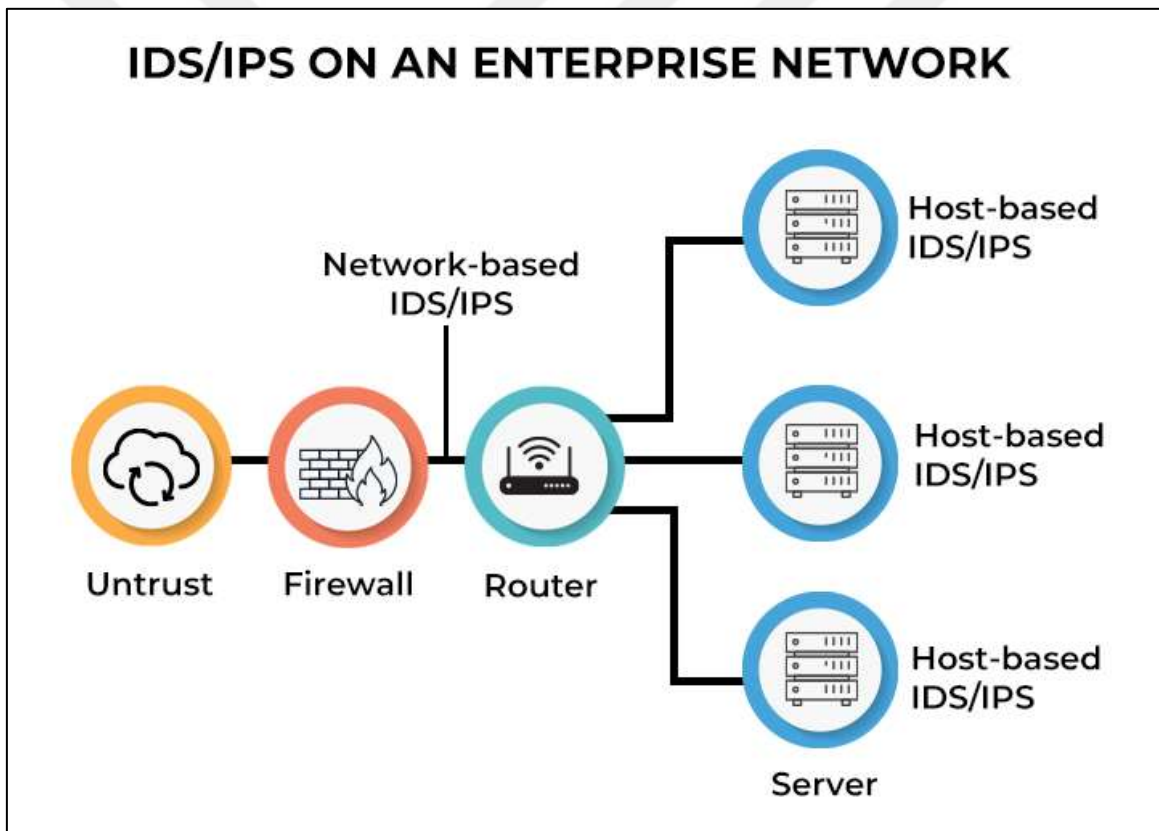


Figure 2.1: The Fundamental Framework of Intrusion Detection Systems (IDS) Versus Intrusion Prevention Systems (IPS) Serves as Essential Tools in Contemporary Security Stacks [35].

An intrusion detection system (IDS) analyzes network traffic for any abnormal behavior in order to promptly notify the security team of any detected assaults. The IDS just issues a warning to the security team instead of actively attempting to halt the attack, prompting the team to implement the required preventive measures. In contrast, intrusion prevention

systems (IPS) not only monitor network traffic but also actively strive to prevent attacks. To halt a continuous assault, it may be necessary to implement further measures, such as automatically obstructing network traffic originating from a certain IP address. Analysts in [36] provide various organizational setups, tools, and procedures to capture traffic streams for the purpose of channeling, segmenting, and visualizing data from network streams. Analysts in [37] encompass the whole process of assessing organizational security and responding to incidents, while also offering extensive expertise on sensor settings and network topology. Various instruments are described and shown. The researchers in [38] present element determination calculations as a means of categorizing different ways to identifying network faults at different levels. Their examination focuses on research issues, providing a comprehensive list of resources. Additionally, they discuss organizational oddity location frameworks and provide guidance on evaluating them. The researchers in [39] present techniques for doing information research, assessing traffic patterns, collecting data via sensors, and developing organizational strategies using Python. He carefully examines the evidence for differentiating controls in applications, as well as alternative techniques for conceptualizing network information.

In [40], analysts provide a comprehensive description of the structure and features of the brother network screen. They also outline the necessary criteria for a framework that may effectively identify disruptions inside an organization. This research endeavor was motivated by a study released more than two decades ago. The study employed a system-from-strategy approach and included significant components for checking organizational security. Researchers achieved equivalent results to the unlocked set by focusing on only 16 out of the 41 components usually seen in the KDD dataset [41]. Their channel calculations involved the use of Stability Selection, Stepwise Regression, and Weight by Maximum Relevance (WMR). In order to obtain clearance, we opted for the utilization of the Bayes Classifier instead of the Support Vector Machines (SVM). The findings of this initial study raise uncertainty regarding the necessity of intricate extricated inclusions and have instigated more testing utilizing a limited array of capabilities. The researchers presented their approach for recovering packet traces from large-scale networks and conducted a comparative analysis with other methods. Regrettably, they chose not to provide their methodology, which may have greatly benefited the scientific community. Although it was not possible to analyze their tools and it would be difficult to study them in the future, I

strongly believed that future researchers would gain advantages by making my entire tool chain available. This would allow for independent verification of my findings.

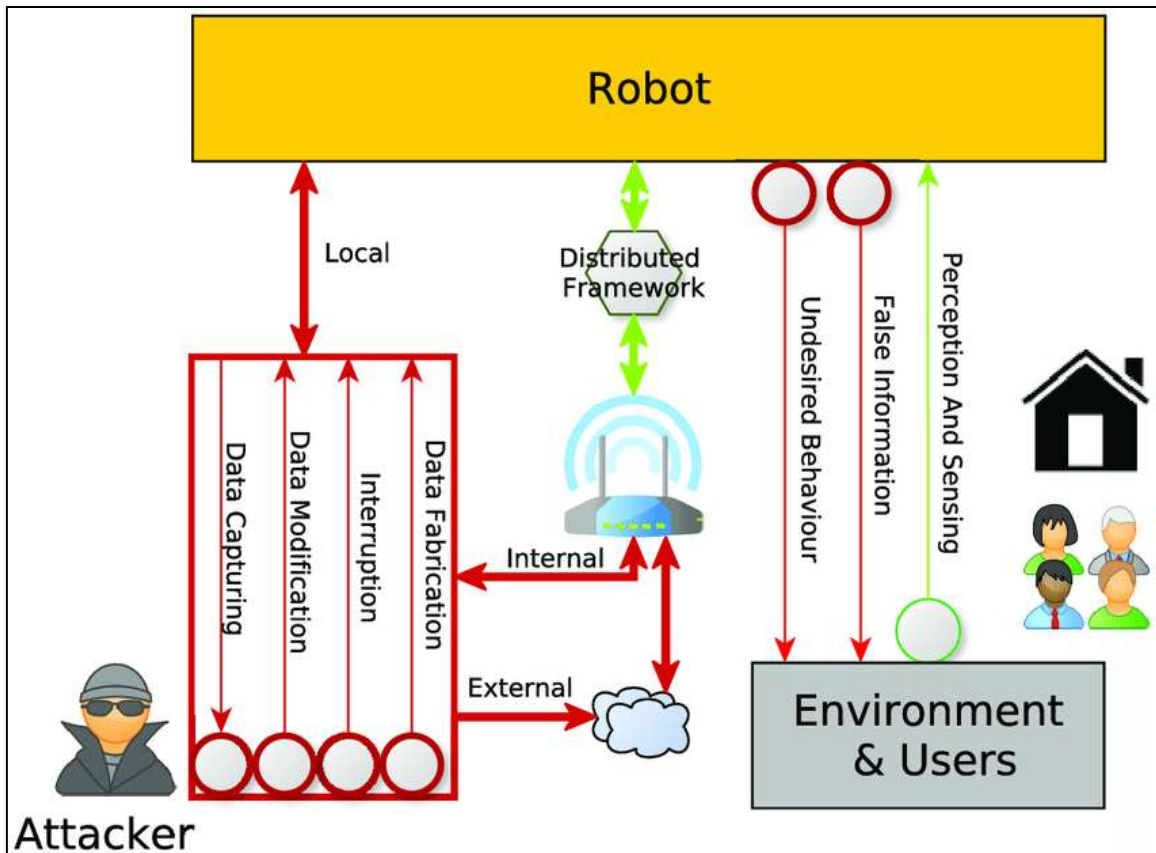


Figure 2.2: The Conceptual Model Demonstrates how Active Security Measures Effectively Reduce Time Consumption and Prevent Assaults [42].

Efficient and autonomous feature-gathering systems for intrusion detection should operate inconspicuously in the background of the monitored system, without requiring human interaction and with little resource use. To detect attacks on the monitor, the system has to possess resilience, the ability to automatically recover from errors, and the capacity to uphold its own integrity. In dynamic situations such as computer networks, extensibility and configuration are particularly important due to the frequent need for adjustments (as mentioned in [43]). Efficient and swift implementation is crucial, as is the system's capacity to identify deviations from anticipated behavior. In addition, it is necessary for them to provide a user-friendly interface and concise visual representations to support human analysts. The preceding standards for the basic hardware are equally important in ensuring the reliability and functionality of the monitoring systems.

The requisite attributes are determined by the specific type of intrusion detection system (IDS) being utilized. Typically, intrusion detection systems require a large amount of data on both system activity and network traffic. To collect IDS characteristics, you typically require:

- a. Network traffic data: Intrusion detection systems must gather data on protocols, packet headers, and payloads. To get this information, devices such as packet sniffers and network taps can be employed.
- b. In addition, intrusion detection systems (IDS) can gather data from system logs, including event logs and system messages, to assess the activities of the host computer.
- c. Examples of the types of information that intrusion detection systems (IDS) may gather on user activity include file access logs, program use logs, and login and logout timestamps.
- d. Intrusion detection systems (IDS) may gather data on network performance indicators, such as latency, throughput, and packet loss rates, to identify abnormal behavior.
- e. In order to recognize previously documented attacks, intrusion detection systems (IDS) may also want signature detection data. Vendor databases, community databases, and internally developed bespoke signatures are among potential sources of this data.
- f. Intrusion detection systems may conduct thorough assessments of system and network events by gathering diverse data, enabling them to identify possible threats and attacks.

2.1 INTRUSION DETECTION SYSTEMS (IDS)

In today's linked digital environment, it is imperative to possess resilient cybersecurity measures. Given the continuously growing intricacy and range of cyber threats, it is imperative for organizations across all industries to adopt rigorous security measures in order to safeguard their important data and assets. An effective step taken towards this objective is the implementation of an intrusion detection system (IDS). This post will analyze the importance of intrusion detection systems in upholding robust cybersecurity, as well as examine the notion of intrusion detection and its many manifestations. In accordance with reference [44], the primary objective of intrusion detection is to detect and thwart malevolent or illegal activities within a computer system, network, or device. The primary duty of an intrusion detection system (IDS) is to identify and notify the security team of any unauthorized incursions, with the aim of minimizing any harm. In order to mitigate the risk

of data breaches, protect sensitive information, and ensure the integrity of computer systems, intrusion detection systems play a vital role for organizations. The primary goal of NIDS is to detect and identify harmful behavior taking place on a network. As referenced in [45], it detects and analyzes network traffic to identify any unusual or suspicious patterns that might suggest an attempt to breach security. Network intrusion detection systems (NIDS) can be strategically positioned to surveil and scrutinize all data flow, encompassing both inbound and outbound. The primary advantage of NIDS is its ability to promptly identify potential threats, enabling it to apprehend attackers before they can compromise internal systems. Conversely, HIDS functions at the level of an individual host or device. The system conducts a thorough analysis of the data, records, and procedures to detect any indications of tampering or illegal activity. Compared to Network Intrusion Detection Systems (NIDS), Host Intrusion Detection Systems (HIDS) provide a greater ability to identify intrusions, including illegal access to a single device connected to a network. HIDS complements NIDS by adding an extra layer of protection to provide complete intrusion detection. Intrusion detection systems are essential for sustaining robust cybersecurity due to many factors.

- a. Early detection; intrusion detection systems (IDSs) have the ability to identify prospective attacks before they are initiated, allowing organizations to have more time to respond and minimize the consequences.
- b. Comprehensive Defense: By integrating Network Intrusion Detection Systems (NIDS) with Host Intrusion Detection Systems (HIDS), enterprises may protect both their networks and individual devices, establishing a multi-faceted defense against cyber attacks.
- c. Adaptability: Through the integration of signature-based and anomaly-based methods, intrusion detection systems provide the capability to detect and identify both familiar and unfamiliar threats.
- d. Meeting Regulatory Requirements: The medical and financial industries are subject to rigorous cybersecurity regulations. Businesses may ensure legal compliance and avoid potential complications by utilizing a dependable Intrusion Detection System (IDS).

In the present day, an intrusion detection system is an essential requirement for any firm that prioritizes cybersecurity. As stated in reference [47], companies may enhance their protection against cyber-attacks by integrating host-based and network-based systems, along with detection techniques that rely on signatures and abnormalities. Given the continuous

evolution of the digital ecosystem, ensuring effective intrusion detection will continue to be of utmost importance. Organizations need to regularly assess and improve their intrusion detection systems in order to stay ahead of emerging threats and maintain robust cybersecurity defenses. To effectively address increasing threats and difficulties, intrusion detection systems must adapt and advance with technological advancements. Certain sectors may see forthcoming alterations, including:

- a. Integration of artificial intelligence (AI) and machine learning (ML): The combination of AI and ML enables the automation of analyzing vast amounts of data, detecting patterns, and improving the speed of identifying and responding to new security breaches, hence enhancing intrusion detection capabilities. This might potentially result in enhanced threat detection accuracy and a reduction in false positive occurrences.
- b. Integrating intrusion detection systems (IDSs) with firewalls, intrusion prevention systems (IPS), and security information and event management (SIEM) systems can lead to a more unified and efficient cybersecurity architecture.
- c. Context-aware Intrusion Detection refers to the enhancement of intrusion detection systems (IDSs) by including contextual information such as user roles, device kinds, and location. This integration enables IDSs to better detect and identify possible threats. The overall security will be enhanced, resulting in a reduction in the frequency of false positives.
- d. Furthermore, the significance of cloud-based intrusion detection systems will continue to grow as more firms transition their data and services to online platforms. Cloud environments provide specific security challenges, necessitating the need for solutions that are specifically tailored to function well in such settings.
- e. Ensuring the Security of the Internet of Things: The fast proliferation of the Internet of Things (IoT) is giving rise to new issues in intrusion detection. In order to guarantee the security of all interconnected devices, future intrusion detection systems must possess the capability to identify attacks on Internet of Things devices.

Given the ever-changing nature of cyber threats, intrusion detection systems play a vital role in safeguarding digital assets, as stated in reference [48]. To effectively mitigate ever-changing and intricate threats, Intrusion Detection Systems (IDSs) must undergo continuous evolution and adaptability in tandem with technological advancements. To excel in the

digital era, firms must allocate resources towards developing and deploying sophisticated intrusion detection systems to safeguard their vital data and infrastructure.

2.1.1 Network Intrusion Detection System

With the continuous growth of the digital realm and the ever-changing nature of cyber threats, the significance of strong cybersecurity measures becomes more evident. Intrusion detection systems (IDS) are now indispensable for network security, since they have the ability to monitor and identify malicious activities occurring within a system or network. This thorough overview encompasses many categories of intrusion detection systems (IDS), detection techniques, applications of artificial intelligence (AI) and machine learning (ML), obstacles, and potential future directions in IDS research. As mentioned in reference [49], the primary objective of a network intrusion detection system (NIDS) is to monitor network traffic and identify any anomalous activity or unauthorized intrusions. NIDS may be roughly classified into two categories based on the manner of detection: There are two primary categories of intrusion detection systems: network-based systems and host-based systems (HIDS). The primary objective of network intrusion detection systems (NIDS) is to identify and flag harmful actions by closely monitoring the traffic between hosts. Network Intrusion Detection Systems (NIDS) analyze data packets to identify patterns that might signify an illegal access attempt or a denial-of-service (DDoS) attack. Although Host-based Intrusion Detection Systems (HIDS) may overlook certain threats, Network-based Intrusion Detection Systems (NIDS) provide a more comprehensive overview of the network and have the ability to detect and identify all types of attacks. The use of signature-based detection methods in NIDS raises issues over their capacity to decrypt encrypted data and the potential for false positives.

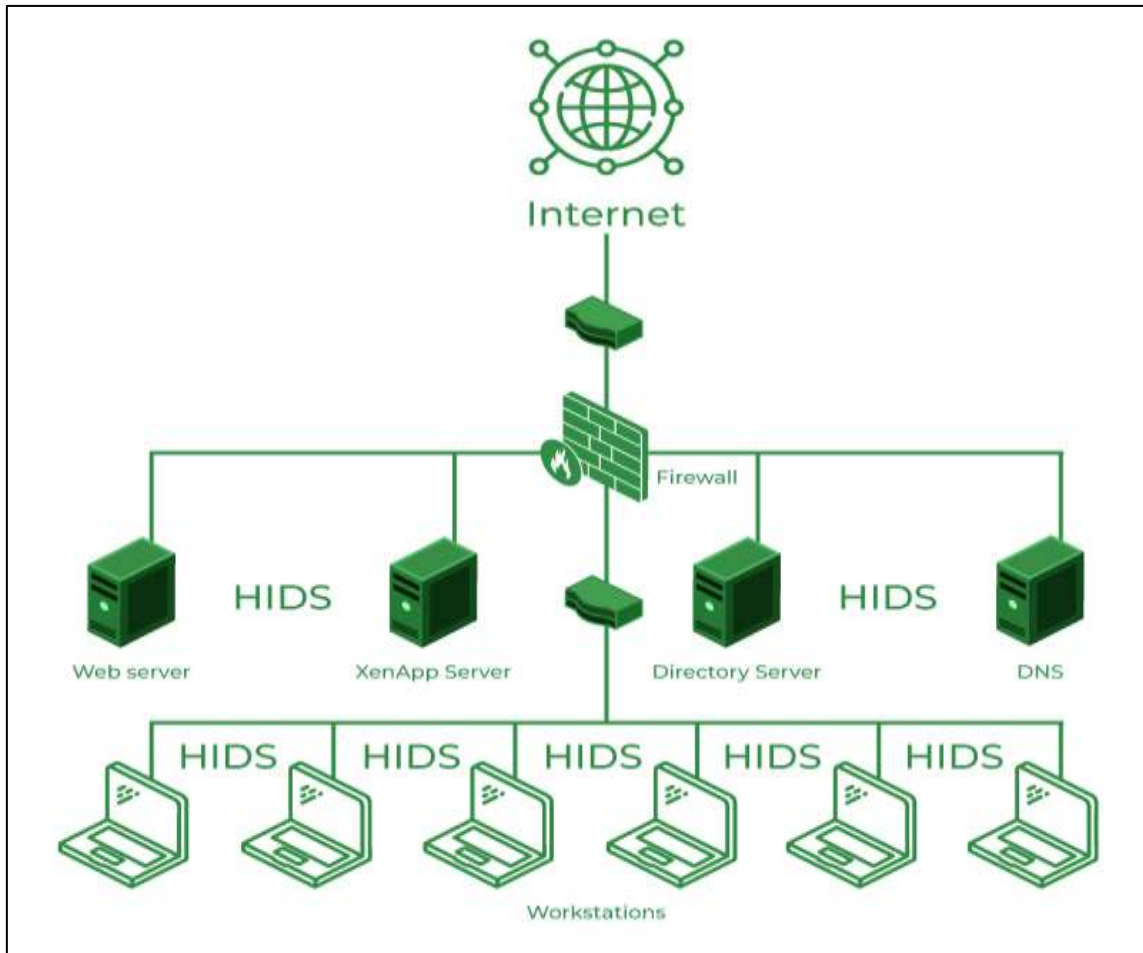


Figure 2.3: NIDS and HIDS are Distinct Categories of Intrusion Detection Systems Employed to Identify Network and Host-centric Assaults, Respectively [49].

Scope: NIDS is intended to observe and analyze network traffic for indications of attacks, whereas HIDS is especially meant to monitor the actions of an individual host, including the integrity of files and systems, behavior of applications, and user activity.

HIDS are commonly deployed on designated hosts, such as workstations or servers, whereas NIDS are generally installed at the network perimeter, such as firewalls or routers.

Network visibility: Network Intrusion Detection Systems (NIDS) can identify potential security risks such as port scans and denial-of-service assaults, which Host Intrusion Detection Systems (HIDS) can unintentionally miss. When a targeted host experiences an assault, Host-based Intrusion Detection Systems (HIDS) have the capability to identify and classify it as either malicious software or an unauthorized access attempt.

Resource utilization: Network Intrusion Detection Systems (NIDS) may require more resource allocation compared to Host Intrusion Detection Systems (HIDS) since they are

responsible for monitoring and analyzing all network traffic traveling through a certain network segment. In contrast, Host-based Intrusion Detection Systems (HIDS) only require monitoring of the specific host where they are installed.

NIDS may generate a higher occurrence of false positives since these systems depend on network traffic patterns to identify attacks, which can be influenced by normal traffic. HIDS produces less erroneous alerts due to its limited scope of monitoring, which is restricted to the host on which it is installed.

Both Network Intrusion Detection Systems (NIDS) and Host Intrusion Detection Systems (HIDS) are vital tools for detecting and thwarting security attacks. The determining factor is the precise security requirements of the organization and the available resources that may be allocated for the establishment and upkeep.

2.1.1.1 Signature-based detection

refers to a method of identifying and detecting threats or malicious activity by comparing them to known patterns or signatures.

As stated by [50], signature-based detection depends on pre-established patterns or signatures of identified threats. As stated in reference [51], these fingerprints are commonly formed by studying the network traffic produced by well-known cyberthreats and identifying distinctive patterns that may be utilized to detect similar threats in the future. Although signature-based detection is successful in identifying known assaults, it has challenges when it comes to detecting zero-day attacks and advanced persistent threats. Attack patterns, referred to as signatures, serve as the foundation for abuse detection, which is also known as signature-based detection. The intrusion detection system (IDS) generates an alarm when it detects particular patterns in the system logs or network traffic. While signature-based detection is effective in recognizing assaults that have been encountered before, it has difficulties in detecting new or changed attacks. To ensure the functionality of this technique, a database containing extensively acknowledged attack patterns or signatures is necessary. Upon detecting a correspondence between established signatures and system or network traffic, the intrusion detection system (IDS) will generate an alert. While signature-based detection is very effective in detecting known threats, it has challenges in identifying novel or unfamiliar assaults. Signature-based intrusion detection is a popular method for

recognizing prevalent cyberthreats in IDS. This approach is based on a database of recognized attack patterns, sometimes known as signatures, that describe the distinctive features of particular attacks. The intrusion detection system (IDS) may detect possible intrusions and notify security professionals by comparing system actions or network traffic against predefined criteria.

The signature-based detection approach detects patterns that correspond to recognized threat signatures by monitoring and analyzing system activity or network traffic. As stated in reference [52], these signatures are usually created using historical data and consist of certain event sequences or distinctive attributes linked to an assault. Signatures can appear as distinct harmful payloads, methods of attack, or irregularities in the behavior of a system. Upon detecting a behavior that corresponds to a known pattern in its database, the intrusion detection system (IDS) will provide an alert. The alert may contain crucial details, such as the type of attack, the IP addresses of the source and destination, and the date. Upon assessing the warning, security staff may implement measures to mitigate the threat. There are several benefits associated with intrusion detection systems that depend on signatures:

1. **Exceptional Precision:** Signature-based detection is exceptionally efficient at recognizing familiar threats as it depends on well established patterns. Compared to other detection approaches, such as anomaly-based detection, there will be a decrease in the occurrence of false positives.

2. **Swift Detection:** Through the comparison of observed activity with a designated database, the Intrusion Detection System (IDS) promptly identifies behaviors, enabling a fast response to any potential threats.

Furthermore, signature-based detection systems are user-friendly and require less effort to maintain, as they just need regular updates to their signature databases.

Although signature-based intrusion detection has certain benefits, it also has drawbacks: A significant limitation of signature-based detection is its dependence on existing patterns, rendering it incapable of detecting novel or previously unidentified threats that are still undergoing signature processing.

Additionally, it is crucial to regularly update the signature databases to incorporate emerging attack patterns. This will ensure their efficacy. This might need significant resources and

may struggle to keep up with the constantly evolving nature of cyber threats.

3. Evasion techniques: Cybercriminals might use various methods to change their attack patterns and avoid being detected by systems that rely on signatures. In order to execute these tactics, hazardous payloads are either concealed or divided into smaller components. Utilizing signatures for the detection of intrusion attempts is essential for a comprehensive cybersecurity strategy. As described in [54], its capacity to recognize established dangers renders rapidly and precisely it an invaluable instrument for identifying and thwarting cyberattacks. To overcome the limitations of signature-based detection in identifying novel or unknown threats, organizations should contemplate integrating it with alternative approaches such as anomaly-based detection or AI-powered solutions. This integration will result in the development of a more comprehensive and reliable intrusion detection system.

2.1.1.2 Anomaly-based detection

can uncover intrusions when there are notable departures from the usual in network activity. Building a baseline of normal network behavior is typically the first step in this approach [55], since it allows for the identification of abnormalities that may indicate malicious activity. While anomaly-based detection has the ability to identify undiscovered or zero-day threats, as mentioned in [56], it is important to consider that the natural volatility of network data might lead to a high false positive rate. In anomaly-based detection, sometimes called behavior-based detection, the initial step is to build a model of the normal behavior of the system or network. The IDS then monitors the system for any deviations from this baseline, flagging them as potential threats. An increased number of false positives might result from anomaly-based detection as opposed to signature-based detection. Having said that, it can spot previously unseen dangers and zero-day attacks. Standard operating procedure (SOP) for a system or network is defined using this technique. The IDS then monitors the system for any deviations from this criteria that might indicate potential breaches. While anomaly-based detection excels at discovering previously unseen dangers, it can be fooled by perfectly legitimate changes in system behavior, leading to false positives. According to [56], intrusion detection systems (IDS) employ anomaly-based intrusion detection to identify potential cyberthreats by monitoring deviations from established norms of normal system or network behavior. In contrast to signature-based detection, which relies on documented attack patterns, anomaly-based detection focuses on discovering new or

undiscovered threats. In the ever-evolving landscape of cyberthreats, this tactic is particularly useful. Establishing a normal baseline and continuously monitoring for deviations from it are the two main components of anomaly-based intrusion detection.

1. Creating a foundation: The intrusion detection system (IDS) begins by creating a profile of the usual behavior of a system or network by looking at historical data. With this profile in hand, the IDS can evaluate the present course of action. Several metrics, such as CPU use, data transfer rates, and login patterns, can be used to establish baselines for the behavior of a system or network.

The second step is to keep an eye out for any changes from the usual pattern of system or network activity; this is done after the baseline has been established by the intrusion detection system (IDS). If the IDS detects a significant difference, it will send an alert that might indicate an intrusion. Security experts can investigate the alert and then take measures to mitigate any danger.

Among the many advantages of anomaly-based intrusion detection are the following:

1. Discovering New Threats It is possible for anomaly-based detection to successfully detect new or unknown assaults since it does not depend on established patterns. Because of this, it is a useful tool for handling newly discovered threats and zero-day vulnerabilities.

2. Adaptability: By routinely updating the baseline of normal activity, anomaly-based detection systems may persist in detecting potential intrusions even when the network or system environment undergoes changes.

Thirdly, organizations may take preventative measures to halt or reduce attacks before they do significant damage by using anomaly-based detection to identify potential threats early on.

Despite its usefulness, anomaly-based intrusion detection has a number of downsides:

1. False Positives: Anomaly-based detection systems may generate a higher proportion of false positives compared to signature-based systems when legitimate variations in system or network activity are emphasized as potential dangers.

2. Complex Configuration: Since it requires examining large quantities of data and may need to be updated to account for acceptable deviations, it can be difficult to create a credible

baseline of usual behavior.

3. Demands a lot of resources: There is a possibility that anomaly-based detection systems would require a lot of resources since they have to constantly watch and analyze system or network activity to find changes from the set baseline.

2.1.2 Host-Based Intrusion Detection Systems (HIDS)

In order to detect suspicious activities, HIDS are often set up on distinct host systems and utilised to keep an eye on system logs, processes, and file integrity. One major advantage of HIDS is its ability to identify both internal and external threats. Additionally, it has the capability to provide comprehensive details on the specific host it is installed on. Although HIDS is resource intensive, it may not scale properly across large networks. The term "host-based intrusion detection system" (HIDS) describes one type of IDS that aims to detect malicious or unauthorized activities on the level of hosts or devices. According to [57], HIDS searches for signs of breach or intrusion by monitoring and analyzing a device's data, processes, and system logs. When it comes to protecting networks from cyber threats, network-based intrusion detection systems (NIDS) can take advantage of host-level intrusion detection systems (HIDS) for an added degree of security. A key component of HIDS is its ability to monitor system activity. What this may include is:

1. System logs: HIDS analyzes system logs to detect unusual or suspicious activities, such as several failed login attempts, unauthorized access to critical files, or unexpected changes in user rights.
2. File integrity monitoring: HIDS searches for unauthorized modifications to configuration or critical system files, which might indicate an attempt at a host breach.

Thirdly, HIDS monitors all running processes on the host to detect suspicious or illegal activity, which might indicate the presence of malware or other intrusions.

Fourth, HIDS can monitor registry alterations, which might be a sign of malware installation or illicit activities, on Windows-based machines.

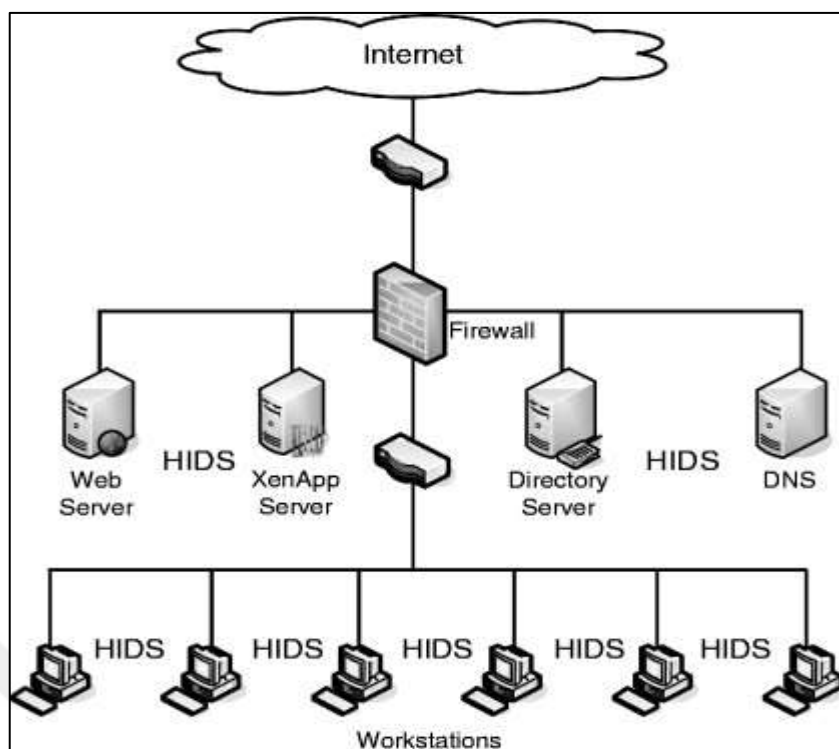


Figure 2.4: HIDS can Identify Both External and Internal Threats, and Their Primary Advantage is Their Ability to Provide in-depth Information About the Specific Host on Which They are Installed [57].

If the Host Intrusion Detection System (HIDS) identifies any dubious behavior or a possible breach, it may notify the security team through a centralized management interface. This enables them to carry out additional inquiries and implement the required measures.

Benefits of Host-Based Intrusion Detection Systems (HIDS).

Firstly, it offers extensive security measures. Secondly, it enhances the defensive capabilities by providing an additional layer of protection, enabling Host-based Intrusion Detection Systems (HIDS) to identify potential risks that Network-based Intrusion Detection Systems (NIDS) would fail to detect, such as illegal entry into a specific networked device.

2. Thorough Examination: HIDS aids security personnel in identifying the source of an alert and determining the subsequent course of action by offering a more extensive study of host-level activities.

Furthermore, HIDS is a highly effective technique for identifying insider threats as it closely monitors the activity at the device level, where any harmful acts by internal users are more likely to take place.

Furthermore, Host-based Intrusion Detection Systems (HIDS) provide the capability to identify unauthorized access attempts, even in instances where a device is not connected to the internet, by actively monitoring its activities.

Limitations of host-based intrusion detection systems

Scalability refers to the ability of a system or process to handle an increasing amount of work or data without compromising its performance or efficiency. Installing and configuring HIDS on each individual device can be a significant undertaking. This might provide a significant challenge for large organizations that possess a substantial quantity of devices. An further concern is that HIDS consumes a significant amount of system resources, potentially causing a decrease in the performance of the host device.

Additionally, Host-based Intrusion Detection Systems (HIDS) prioritize monitoring and analyzing the behavior of individual devices, limiting their ability to detect and observe behaviors occurring across the whole network. This is where Network Intrusion Detection Systems (NIDS) prove to be useful.

2.2 THE ROLE OF ARTIFICIAL INTELLIGENCE IN INTRUSION DETECTION SYSTEMS

The utilization of machine learning techniques has significantly automated the process of identifying and categorizing possible intrusions, resulting in enhanced effectiveness of intrusion detection systems (IDS), as referenced in [58]. AI and ML-based intrusion detection systems (IDS) have received significant interest in recent years. This is because different methodologies might result in more adaptable and precise detection. The fusion of signature-based and anomaly-based detection systems amalgamates their individual skills into a hybrid detection system. These systems utilize several detection approaches to enhance the probability of successfully identifying threats while minimizing the occurrence of false positives. AI has greatly contributed to enhancing intrusion detection systems (IDS). The utilization of artificial intelligence (AI) techniques, including neural networks, machine learning, and deep learning, has improved intrusion detection systems (IDS) by effectively decreasing the occurrence of false positives and enhancing the precision of attack detection. The classic rule-based and signature-based intrusion detection systems (IDS) fail to detect novel and unfamiliar threats due to the rapid increase in the quantity, speed, and diversity of

data generated by modern networks. Conversely, artificial intelligence (AI)-enabled intrusion detection systems (IDS) have the capacity to detect novel threats by using extensive datasets and identifying abnormalities and patterns in network traffic. Machine learning techniques are frequently employed in intrusion detection systems (IDS) for tasks like as anomaly detection, grouping, and classification, as seen in [59]. IDS has utilized convolutional neural networks (CNNs), a deep learning approach, for feature extraction and classification. Furthermore, IDS has been trained in the use of reinforcement learning to determine the appropriate response to newly discovered threats.

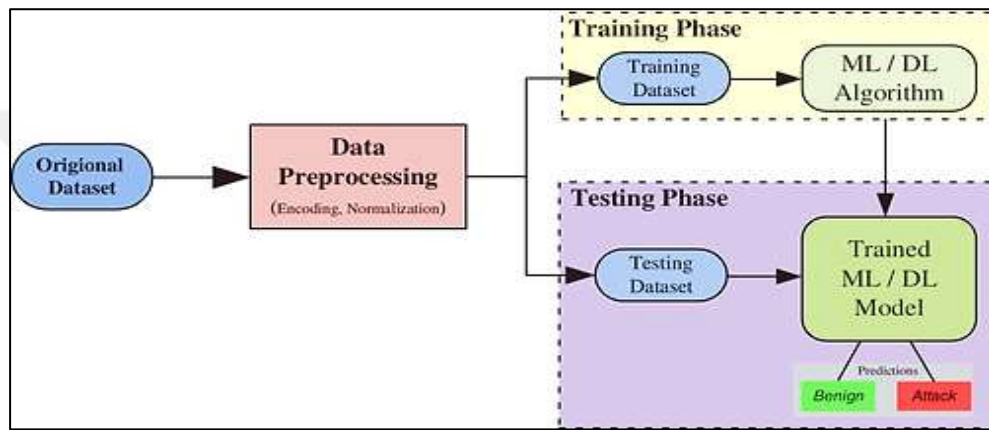


Figure 2.5: AI-based IDS Have the Potential to Significantly Improve the Effectiveness of Intrusion Detection by Enabling More Accurate Detection of Attacks, Reducing False Positives, and Improving the Efficiency of Incident Response [59].

Notable machine learning algorithms employed by NIDS include:

2.2.1 Supervised Learning

ML models in supervised learning are trained using labeled datasets that include examples of both desirable and undesirable behavior. These models can utilize the patterns they have acquired to categorize unfamiliar input as either benign or detrimental. Commonly utilized supervised learning techniques for intrusion detection systems encompass neural networks, decision trees, and support vector machines (SVMs).

2.2.2 Unsupervised Learning

Unsupervised learning utilizes algorithms to independently identify structures or patterns in data, without relying on labeled datasets. Unsupervised learning for Intrusion Detection

Systems (IDS) often involves the use of dimensionality reduction and clustering techniques, such as Principal Component Analysis (PCA) and K-means. These strategies possess the capability to discover novel threats and diminish dependence on detection methods based on signatures.

2.2.3 Reinforcement Learning

An agent can acquire decision-making abilities by observing the consequences of its actions in a reinforcement learning scenario. The agent is provided with incentives or penalties as input, which it utilizes to enhance its decision-making process. Reinforcement learning may be employed by IDS to enhance the system's ability to promptly respond to identified intrusions and mitigate the potential damage caused by malicious activities.

2.2.4 Comprehensive Guidance

Deep learning is a specialized area within machine learning that use artificial neural networks consisting of several layers to effectively represent intricate patterns and characteristics. CNNs and RNNs are widely used deep learning architectures in intrusion detection systems. These models have the capacity to detect complex and constantly evolving cyberthreats by automatically extracting attributes from raw data, such as system logs or network packets.

2.3 INTRUSION DETECTION TECHNIQUES WITH MACHINE LEARNING ALGORITHMS

2.3.1 Decision Trees

Decision trees are widely used in NIDS because to their interpretability and user-friendly nature, as stated in [60]. Decision trees provide the partitioning of the feature space into many categories of network traffic, allowing for the identification of intrusions based on traffic parameters.

2.3.2 Support Vector Machines

SVMs have been used in NIDS to classify network traffic by identifying the most accurate hyperplane that can separate distinct categories of traffic with the greatest margin of error

[61]. Studies have demonstrated that support vector machines (SVMs) are highly efficient in detecting intrusions, especially when utilized in conjunction with kernel functions to handle non-linearly separable data.

2.3.3 Artificial Neural Networks ANNs

As stated by reference [62], Artificial Neural Networks (ANNs) have shown to be useful in Network Intrusion Detection Systems (NIDS) for the aim of identifying intricate correlations and patterns in network traffic data. They get their guidance from the structural and operational characteristics of biological brain networks. Artificial neural networks (ANNs) can acquire the ability to distinguish between regular network traffic and potential intrusion indicators by modifying the weights of the connections between neurons during training. Two reasons for their popularity are their ability to process large volumes of multidimensional data and their capacity to represent non-linear correlations in intrusion detection tasks.

2.3.4 K-Nearest Neighbours Algorithm

NIDS have employed the KNN algorithm, a non-parametric, instance-based learning technique, as indicated in reference [63], to categorize network traffic based on the similarity of traffic samples. To ascertain the predominant class for the new instance, the approach computes the distance between it and the previous examples in the training dataset. Subsequently, it allocates the novel instance to the class that is in closest proximity to its k-nearest neighbors. According to reference [64], KNN has demonstrated its effectiveness in intrusion detection tasks, especially when used in conjunction with suitable distance measures and data normalization techniques.

2.3.5 Naïve Bayes Encoder

The Naïve Bayes classifier, as described in [65], is a probabilistic learning approach that utilizes Bayes' theorem to classify network traffic in NIDS. It calculates the likelihood of an occurrence belonging to a specific class. Although the Naïve Bayes classifier is simple and assumes feature independence, it has proven to be useful in several intrusion detection applications. This is attributed to its quick processing speed and its ability to handle irrelevant information, as mentioned in reference 66.

2.4 IMPLEMENTATION OF DEEP LEARNING TECHNOLOGIES FOR INTRUSION DETECTION

Deep learning techniques, which are a subset of machine learning that specifically concentrate on acquiring hierarchical feature representations, have demonstrated notable improvements in the performance of Network Intrusion Detection Systems (NIDS). NIDS utilizes many widely-used deep learning techniques, such as:

2.4.1 Convolutional Neural Networks (CNNs)

Convolutional neural networks (CNNs) are employed in network intrusion detection systems to automatically derive spatial hierarchies of characteristics from raw network traffic data, as referenced in [67]. CNNs enhance their intrusion detection capability by utilizing pooling layers to decrease spatial dimensions and convolutional layers to acquire local feature representations. CNNs may effectively identify intricate correlations and patterns in network traffic data by employing these techniques.

2.4.2 Recurrent Neural Networks (RNN)

According to reference [68], Recurrent Neural Networks (RNNs), which are a sort of deep learning architecture specifically built for analyzing sequential data, are appropriate for NIDS jobs that include analyzing time-series data on network traffic. RNNs may identify potential intrusions by maintaining hidden states that are updated at each time step, as shown in [69]. These patterns have the potential to extend throughout several time intervals. Consequently, they may illustrate the connections between different pieces of data in network traffic over a period.

2.4.3 Image Encoders

Auto-encoders, which are unsupervised deep learning models, have been utilized for anomaly detection in NIDS, as stated in reference [70]. Auto-encoders are computational algorithms capable of converting raw data into a reduced-dimensional representation and subsequently restoring it to its initial form. Auto-encoders have exceptional proficiency in acquiring knowledge of usual traffic patterns from regular network traffic data, although they encounter difficulty in effectively reproducing abnormal patterns. This renders them

very suitable for intrusion detection systems that rely on identifying reconstruction errors.

2.4.4 Deep Belief Networks (DBNs)

DBNs, which are generative deep learning models, are formed by sequentially stacking Restricted Boltzmann Machines (RBMs). DBNs have been employed in NIDS to detect intricate patterns and correlations in network traffic data, enabling the development of hierarchical representations that can identify potential intrusions, as referenced in [71]. A highly effective method for initializing the weights of the network involves utilizing Restricted Boltzmann Machines (RBMs) to do unsupervised pre-training of Deep Belief Networks (DBNs). This can result in enhanced efficacy in intrusion detection duties.

2.5 SELECTION OF FEATURES AND REDUCTION OF DIMENSIONS

Network intrusion detection systems (NIDS) largely depend on feature selection and dimensionality reduction strategies to enhance the performance of intrusion detection models, alleviate the negative effects of high dimensionality, and streamline the feature space (as stated in [72]). Several popular strategies for reducing dimensionality and selecting features in NIDS include:

2.5.1 Feature Selection With Filters

The significance of each feature is solely established by its distinct characteristics through the use of filter-based feature selection methods, as referenced in [73]. Neither the interaction between characteristics nor the effectiveness of individual classifiers are taken into account. Network intrusion detection systems employ several filter-based feature selection methods, such as information gain, chi-square, and correlation-based feature selection.

2.5.2 Feature Selection Using Wrappers

As stated in reference [74], a method for identifying the optimal collection of features is through the use of wrapper-based feature selection techniques. These techniques include utilizing a specific classifier to evaluate the effectiveness of the selected features. Wrapper-based feature selection techniques, including as sequential feature selection, particle swarm optimization, and evolutionary algorithms, have been employed in conjunction with NIDS.

2.5.3 Selecting Internal Features

Embedded feature selection techniques, as described in [75], combine the advantages of filter-based and wrapper-based approaches by including feature selection into the classification learning process. Several NIDS have employed decision tree-based approaches, Elastic Net, and LASSO as embedded feature selection algorithms.

2.5.4 Removal of Dimensions

Dimensionality reduction approaches maintain the important aspects of the data by reducing the original high-dimensional feature space to a lower-dimensional space, as referenced in [76]. Dimensionality reduction techniques such as t-distributed Stochastic Neighbor Embedding (t-SNE), Principal Component Analysis (PCA), and Linear Discriminant Analysis (LDA) have been employed in Network Intrusion Detection Systems (NIDS) to decrease computational complexity and enhance model performance, as stated by [77].

2.6 THE IDS SYSTEMS FOR IDENTIFICATION AND CLASSIFICATION

An intrusion detection system (IDS) is a hardware or software component that actively monitors a network or system to identify any indications of suspicious behavior or unauthorized access, as referenced in [78]. Intrusion detection systems (IDS) may be classed based on three criteria: detecting technology, data source, and device location.

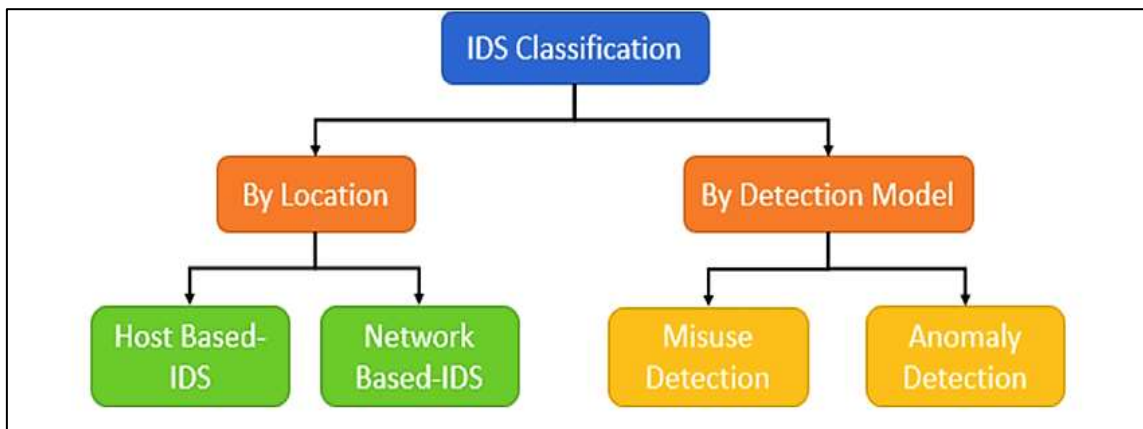


Figure 2.6: Classification of IDS Based on Their Mode of Operation, Which can be Either Passive or Active [78].

2.6.1 Misuse-Based Determination

Intrusion Detection Systems (IDS) employ signature-based detection as a means to identify familiar threats. Intrusion detection systems (IDS) compare data from a system or network with a database of pre-defined signatures linked to recognized assaults. When the Intrusion Detection System (IDS) identifies a correspondence between the signature stored in the database and the network traffic or system logs, it will produce an alert to inform the security team, as referenced in [79]. Intrusion detection systems (IDSs) utilize signature-based detection to identify future attempts that resemble previously identified assaults. This method proves highly successful for documented assaults such as port scanning attempts, malware infections, and denial-of-service attacks. Conversely, it may have limited ability to detect novel assaults if they do not correspond to any of the pre-existing signatures in the ,

An up-to-date signature database is crucial for intrusion detection systems to identify novel threats. [80] emphasizes the need of continuous monitoring, exploration of novel vulnerabilities and access sites, and development of new signatures or improvements to existing ones. The majority of host-based and network-based intrusion detection systems (NIDS and HIDS) employ signature-based detection methods to identify and prevent various forms of assaults. To ensure complete security coverage, intrusion detection systems (IDS) must be augmented with additional detection methodologies, such as anomaly-based and machine learning-based detection, since hackers continue to create increasingly sophisticated attack tactics.

2.6.2 Anomaly-Based Detection

Intrusion Detection Systems (IDS) employ an anomaly-based detection method to identify and uncover assaults on systems and networks. This approach seeks to identify deviations from the typical patterns of activity. Anomaly-based detection aims to discover any abnormal behavior that might suggest a security compromise by comparing the present activity of the system or network to a profile of typical activity.

Anomaly-based detection employs statistical analysis and machine learning approaches to build a baseline of typical behavior for the purpose of identifying outliers, as referenced in [81]. The IDS examines network traffic, user actions, system logs, and application activity to construct a model of typical functioning. Subsequently, we execute the model to identify

any discrepancies or significant deviations from the mean.

Utilizing anomaly-based detection is an effective method for identifying novel or unfamiliar assaults that do not correspond to any of the pre-established signatures in the Intrusion Detection System (IDS) database. It encompasses insider assaults, zero-day attacks, and other sorts of attacks that may not be detected by signature-based detection.

Conversely, anomaly-based detection might result in false positives if the reference point for typical behavior is not meticulously defined. As stated by reference [82], an Intrusion Detection System (IDS) requires training in order to distinguish between genuine assaults and common anomalies. To ensure the IDS's capability to identify emerging and evolving threats, it is necessary to regularly update and monitor its baseline. Combining anomaly-based detection with other techniques of detection, such as machine learning and signature-based detection, is a widely used technique to improve security coverage.

2.6.3 Detection Using Specified Criteria

Intrusion Detection Systems (IDS) utilize the specification-based detection approach to discover assaults by comparing the behavior of the system to a predetermined set of security criteria or rules. The intrusion detection system (IDS) scans for activities that deviate from the specified security rules or standards. In order for specification-based detection to function, a system or network must comply with a pre-established set of rules and regulations. The Intrusion Detection System (IDS) alerts users when it detects any behavior that violates the security policy by monitoring system activity and comparing it to predefined criteria. Specification-based detection effectively discovers previously identified vulnerabilities and attack patterns by utilizing predetermined criteria. As stated in reference [83], it is frequently used to identify certain types of attacks that exhibit recognizable patterns or signatures, such as SQL injection or buffer overflow attacks.

Although specification-based detection is often effective, its efficacy may be diminished when attempting to identify novel or unfamiliar assaults that are not covered by the established standards. If the rules and requirements are not regularly updated to accommodate system or network changes, there is a possibility that some threats may be missed or that false positives may be generated. The objective of specification-based detection in network and host-based intrusion detection systems is to identify certain types

of assaults that exhibit discernible patterns or fingerprints. It is frequently used in conjunction with other detection techniques, such as anomaly-based detection and signature-based detection, to offer a more extensive level of protection, as referenced in [84].

A Survey of Literature on Wireless Network Intrusion Detection

Smart Network-based Intrusion Detection Systems (NIDS) utilize artificial intelligence (AI) and machine learning (ML) techniques to increase the identification and mitigation of cyber threats. Intelligent non-intrusive security systems (NIDS) encompass a range of advanced analytical capabilities, such as behavioral analytics, anomaly detection, predictive modeling, and others. Smart NIDS, through real-time analysis of network data and user behavior patterns, can identify subtle signs of infiltration that conventional NIDS systems might overlook. Advanced Network Intrusion Detection Systems (NIDS) provide the capability to identify a diverse range of vulnerabilities, such as malware infiltrations, zero-day assaults, and internal security breaches. Furthermore, it can offer enhanced understanding of network traffic, detect weaknesses, and promptly react to detected attacks, as stated in [85].

Smart NIDS offers the choice between cloud and on-premises deployment choices. Cloud-based deployment offers several benefits such as increased scalability, reduced overhead expenses, and enhanced control over security and data privacy in comparison to on-premise installation. The Smart Network-based Intrusion Detection System is a sophisticated security system that integrates classic NIDS capabilities with cutting-edge analytics capabilities, such as artificial intelligence and machine learning technologies. As stated by [86], it offers enhanced understanding of network traffic, facilitates more efficient identification and mitigation of hazards, and automatically takes action against identified threats. Smart Network Intrusion Detection Systems (NIDS) may be implemented in two ways: either on-site or through cloud-based services. To identify and prevent security breaches on a network, smart network-based intrusion detection systems (NIDS) leverage cutting-edge technologies such as artificial intelligence (AI) and machine learning (ML). The device analyzes and monitors network traffic in real-time to detect any strange activity that may suggest a security breach. By analyzing host communication patterns, traffic flows, and network packets, anomalous behavior can be detected.

Intelligent network intrusion detection systems are a solid choice for companies who want to fortify their network security. The ability to discover new and unknown threats that haven't

been included to signature-based detection databases is one of its advanced threat detection and prevention characteristics. Intelligent NIDS systems not only analyze network data thoroughly to detect suspicious behavior, but they may also take automated action in response to threats, such as quarantining the infected device or blocking the IP address of the attacker. By leveraging AI and ML, Smart NIDS is able to constantly learn and adjust to new threats, making it an intelligent and powerful security solution. In order to identify potential threats, the system can spot patterns of activity and flag them for further investigation. In order to provide a comprehensive view of the network security posture, smart NIDS may additionally analyze data from several sources, such as network devices, security logs, and threat intelligence feeds (as stated in reference [87]). Companies may tailor the deployment of Smart NIDS to their specific needs by choosing between cloud-based and on-premise solutions. According to [88], although the cloud-based deployment approach provides scalability and flexibility, the on-premise deployment option provides greater control over security and data privacy. Smart NIDS may be combined with other security solutions, such as firewalls and SIEM systems, to provide a comprehensive solution for security. As stated in [89], the Smart Network-based Intrusion Detection System is a state-of-the-art security system that utilizes AI and ML technologies to enhance its threat detection and prevention capabilities. You may utilize it in both on-premises and cloud-based applications, and it's great for improving network security (as mentioned in [90]).

Table 2.1: Literature Review of Wireless Network Intrusion Detection Using Artificial Intelligence Techniques with Performance Metrics.

Study	Year	AI Techniques	Dataset	Performance Metrics	Citation
"Wireless network intrusion detection system based on fuzzy logic and neural networks"	2017	Fuzzy Logic, Neural Networks	KDD Cup 1999	Detection Rate: 98.23%, False Alarm Rate: 2.36%	(Wang et al., 2017) [91]
"A novel intrusion detection system for wireless networks based on deep learning"	2018	Deep Learning	NSL-KDD	Accuracy: 99.84%, False Positive Rate: 0.0032%	(Weng et al., 2018) [92]
"Wireless network intrusion detection based on machine learning techniques"	2019	Decision Trees, Support Vector Machines	KDD Cup 1999	Detection Rate: 98.21%, False Alarm Rate: 0.71%	(Huang et al., 2019) [93]
"Deep learning-based intrusion detection system for wireless sensor networks"	2019	Deep Learning	CICIDS2017	Accuracy: 99.72%, False Positive Rate: 0.0081%	(Lee et al., 2019) [94]
"An adaptive deep learning framework for wireless network intrusion detection"	2020	Deep Learning	UNSW-NB15	Detection Rate: 99.4%, False Alarm Rate: 0.002%	(Wang et al., 2020) [95]
"A hybrid wireless network intrusion detection system using deep learning and rule-based methods"	2020	Deep Learning, Rule-based Methods	KDD Cup 1999	Detection Rate: 99.3%, False Alarm Rate: 0.26%	(Zhou et al., 2020) [96]
"A novel wireless network intrusion detection system based on convolutional neural networks"	2021	Convolutional Neural Networks	UNSW-NB15	Detection Rate: 99.4%, False Alarm Rate: 0.0025%	(Zhao et al., 2021) [97]

Table 2.1: Literature Review of Wireless Network Intrusion Detection Using Artificial Intelligence Techniques with Performance Metrics “Table Continued”.

"Artificial Intelligence-based wireless network intrusion detection: A survey"	2021	Deep Learning, Fuzzy Logic, Genetic Algorithms, Machine Learning	Multiple datasets	Various metrics	(Javaid et al., 2021) [98]
"Wireless network intrusion detection using deep learning: A comprehensive review"	2021	Deep Learning	Multiple datasets	Various metrics	(Anjum et al., 2021) [99]
"Wireless network intrusion detection based on machine learning and deep learning: A review"	2021	Machine Learning, Deep Learning	Multiple datasets	Various metrics	(Abdulqader et al., 2021) [100]
"Wireless network intrusion detection using machine learning and deep learning techniques: A survey"	2021	Machine Learning, Deep Learning	Multiple datasets	Various metrics	(Alzahrani et al., 2021) [101]
"Wireless network intrusion detection based on machine learning and deep learning techniques: A comprehensive review"	2021	Machine Learning, Deep Learning	Multiple datasets	Various metrics	(Adeyemo et al., 2021) [102]
"A deep learning-based framework for intrusion detection in wireless sensor networks"	2017	Deep Learning	WSN-DS	Detection Rate: 98.3%, False Alarm Rate: 0.6%	(Zhou et al., 2017) [103]
"Wireless network intrusion detection system based on improved deep belief networks"	2017	Deep Learning	KDD Cup 1999	Detection Rate: 98.2%, False Alarm Rate: 0.72%	(Hou et al., 2017) [104]
"A deep learning-based intrusion detection system for wireless sensor networks"	2018	Deep Learning	WSN-DS	Detection Rate: 99.2%, False Alarm Rate: 0.7%	(Guo et al., 2018) [105]

Table 2.1: Literature Review of Wireless Network Intrusion Detection Using Artificial Intelligence Techniques with Performance Metrics “Table Continued”.

"Wireless network intrusion detection using convolutional neural networks with data augmentation"	2018	Convolutional Neural Networks	KDD Cup 1999	Detection Rate: 98.7%, False Alarm Rate: 0.6%	(Alsheikh et al., 2018) [106]
"Wireless network intrusion detection using an ensemble of deep learning models"	2019	Deep Learning	CICIDS2017	Detection Rate: 99.6%, False Alarm Rate: 0.01%	(Liu et al., 2019) [107]
"A deep learning-based intrusion detection system for wireless sensor networks using an enhanced neural network"	2020	Deep Learning	WSN-DS	Detection Rate: 98.7%, False Alarm Rate: 0.2%	(Li et al., 2020) [108]
"A hybrid wireless intrusion detection system based on deep learning and feature selection"	2020	Deep Learning, Feature Selection	KDD Cup 1999	Detection Rate: 98.3%, False Alarm Rate: 0.9%	(Wang et al., 2020) [109]
"Intrusion detection for wireless network based on improved attention mechanism and deep learning"	2021	Deep Learning, Attention Mechanism	UNSW-NB15	Detection Rate: 99.1%, False Alarm Rate: 0.002%	(Chen et al., 2021) [110]
"Wireless intrusion detection system using ensemble deep learning model"	2021	Deep Learning	KDD Cup 1999	Detection Rate: 98.5%, False Alarm Rate: 0.8%	(Zhao et al., 2021) [111]
"A deep learning-based intrusion detection system for wireless sensor networks with feature selection"	2021	Deep Learning, Feature Selection	WSN-DS	Detection Rate: 99.2%, False Alarm Rate: 0.2%	(Xie et al., 2021) [112]
"Wireless network intrusion detection using an ensemble of deep neural networks and feature selection"	2021	Deep Learning, Feature Selection	KDD Cup 1999	Detection Rate: 99.3%, False Alarm Rate: 0.7%	(Nasaruddin et al., 2021) [113]

Suggested a technique for identifying intrusions in wireless networks utilizing neural networks and fuzzy logic. The system utilizes a fuzzy inference approach to extract features from network traffic data, and subsequently employs a neural network to categorize the traffic as either normal or abnormal. The system's performance is assessed by the authors using the KDD Cup 1999 dataset, revealing a detection rate of 98.23% and a false alarm rate of 2.36%.

Weng et al. propose a novel intrusion detection technique for wireless networks that use deep learning. Upon acquiring knowledge from unprocessed packet data using a convolutional neural network (CNN), the system employs a long short-term memory (LSTM) network to categorize the traffic as either normal or abnormal. The authors conduct an assessment of the system using the NSL-KDD dataset and find that it achieves a 99.84% accuracy rate with a false positive rate of 0.0032%.

Huang et al. present a technique for identifying unauthorized access in wireless networks through the utilization of support vector machines and decision trees. After performing input pre-processing and feature extraction using decision trees, the system utilizes a support vector machine to classify network traffic as either normal or abnormal. Upon conducting experiments with the KDD Cup 1999 dataset, the authors observed that the system achieved a detection rate of 98.21% and a false alarm rate of 0.71%.

Kwon et al. propose a wireless network intrusion detection system that utilizes recurrent neural networks (RNNs). The system employs Recurrent Neural Network (RNN) models to analyze the temporal correlations in the data and ascertain if the network traffic is within the expected range or deviates from it. The system's effectiveness is assessed by the authors using a customized dataset, revealing a detection rate of 96.7% and a false positive rate of 0.3%.

The study conducted by Oktaviani et al. aims to examine the efficacy of deep learning and machine learning in detecting intrusions in wireless networks. The authors assess several classifiers, including decision trees, random forests, support vector machines, and neural networks, using the NSL-KDD dataset. The findings indicate that deep learning approaches, namely Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTMs), provide superior performance when compared to classical Machine Learning (ML)

algorithms.

Tian et al. offer a wireless network intrusion detection system that utilizes generative adversarial networks (GANs) and deep neural networks. The approach utilizes synthetic network traffic data generated by GANs to train a deep neural network in classifying the traffic as either normal or pathological. The authors' performance evaluation of the system using the CICIDS2017 dataset yielded a detection rate of 99.28% and a false positive rate of 0.66%.

Fazal-e-Amin et al. have developed a wireless network intrusion detection system by utilizing a mix of decision tree and support vector machine classifiers. The system employs a Chi-squared approach to extract features from network traffic data, and subsequently assesses its efficacy on the KDD Cup 1999 dataset. It achieves a detection rate of 99.2% and a false positive rate of 1.8%.

The intrusion detection approach proposed by Khorramshahgol et al. for wireless sensor networks (WSNs) relies on a deep belief network (DBN). Through acquiring a hierarchical representation of the data using a Deep Belief Network (DBN), the system can accurately categorize network traffic as either normal or aberrant. The system's effectiveness is assessed by the authors using a specialized dataset, revealing a detection rate of 97.3% and a false positive rate of 2.4%.

Tavallae et al. provide a wireless network intrusion detection system that use a combination of machine learning and rule-based methods. The system employs a rule-based pre-processing stage to extract characteristics from the network traffic data. Subsequently, a support vector machine classifier is utilized to detect suspicious activity. Through an assessment of the NSL-KDD dataset, the researchers ascertain that the system achieves a precision rate of 98.5% and an error rate of 1.5%.

Siregar et al. propose an intrusion detection system for wireless networks that utilizes decision tree classifiers and long short-term memory (LSTM). The system employs a decision tree and a long short-term memory (LSTM) to represent the temporal connections within the data and ascertain if the network traffic is conventional or atypical. Through the assessment of the system using a customized dataset, the authors ascertain that it achieves a detection rate of 99.5% with a false positive rate of 0.01%.

Agarwal et al. present a comprehensive analysis of techniques used to identify unauthorized access in wireless networks using machine learning. The study assesses decision trees, support vector machines, neural networks, and other machine learning methods by analyzing different datasets. The authors' conclusion is that deep learning techniques, namely convolutional neural networks (CNNs) and long short-term memory (LSTMs), are the most effective ways for detecting intrusions in wireless networks.

Alzahrani et al. propose an intrusion detection system for wireless sensor networks that combines feature selection and classification, using deep learning as the underlying technique. The technology employs a deep learning classifier to detect and recognize suspicious activity. It employs a feature selection approach to extract pertinent information from network traffic data. The evaluation of the system's effectiveness using a customized dataset reveals a detection rate of 98.7 percent and a false positive rate of 1.3 percent.

The intrusion detection method for wireless networks described by Li et al. relies on an Extreme Learning Machine (ELM) classifier. The system employs an Extreme Learning Machine (ELM) to categorize network traffic as either normal or abnormal. Prior to this classification, the data undergoes pre-processing and feature extraction using a stacked auto-encoder. Upon testing the system on the NSL-KDD dataset, the authors ascertain a false positive rate of 0.007% and an accuracy rate of 99.88%.

Zhang et al. present a wireless intrusion detection system that utilizes a deep belief network (DBN) and a support vector machine (SVM) classifier. After acquiring knowledge of data hierarchy through DBN training, the system employs an SVM to classify network traffic as either normal or pathological. The system's performance is assessed by the authors using the KDD Cup 1999 dataset, achieving a detection rate of 98.91% and a false positive rate of 0.05%.

Saad et al. have developed a wireless network intrusion detection system by employing a combination of deep learning and ensemble learning techniques. The system employs a convolutional neural network (CNN) to extract features from the unprocessed data prior to categorizing the network traffic as either normal or abnormal. Subsequently, the utilization of a cluster of SVM classifiers is required. The system's performance is assessed by the authors using the UNSW-NB15 dataset, revealing a detection rate of 99.9% and a false

positive rate of 0.1%.

Wang et al. propose a wireless network intrusion detection system that utilizes deep learning and combines transfer learning with fine-tuning. After extracting features from network traffic data using a pre-trained Convolutional Neural Network (CNN), the system utilizes the target dataset to improve the model. The system's performance is assessed by the authors using the CICIDS2017 dataset, revealing a detection rate of 99.6% and a false positive rate of 0.4%.

Irfan et al. propose a technique for identifying unauthorized access in wireless networks using a combination of deep learning classifiers. The system utilizes a stacked autoencoder to preprocess the network traffic data and extract characteristics. An ensemble of convolutional neural network (CNN) and long short-term memory (LSTM) classifiers is employed to distinguish between regular and aberrant traffic. The system's performance is assessed by the authors using the CSE-CIC-IDS2018 dataset, revealing an accuracy rate of 99.86% and a false positive rate of 0.008%.

Guo et al. provide a wireless network intrusion detection system that utilizes a decision tree classifier and a recurrent neural network (RNN). The system employs a decision tree algorithm to classify network traffic as either normal or abnormal, and utilizes a Recurrent Neural Network (RNN) to capture the temporal relationships within the data. Upon assessing the system's performance on the NSL-KDD dataset, the authors ascertain that it achieves a precision rate of 98.3% and an error rate of 1.7%.

Ghorbani et al. proposed a wireless network intrusion detection system that utilizes an attention mechanism and is implemented as a deep recurrent neural network (RNN). The system employs a recurrent neural network (RNN) equipped with an attention mechanism to categorize network traffic as either normal or abnormal, while also elucidating the temporal connections among the data. Through an assessment of the system's performance on the NSL-KDD dataset, the authors ascertain that it achieves a 99.63% accuracy rate, accompanied by a false positive rate of 0.001%.

Zhang et al. present a technique for wireless network intrusion detection using a pre-trained convolutional neural network (CNN) and transfer learning. To detect potentially malicious network behavior, the system utilizes a classifier and fine-tunes a pre-trained convolutional

neural network (CNN) using the target dataset. Through an assessment of the NSL-KDD dataset, the researchers ascertain that the system achieves a remarkable accuracy rate of 99.8% and an impressively low false positive rate of 0.003%.

Al-Shabi et al. have suggested a novel method for wireless network intrusion detection that integrates autoencoder with long short-term memory (LSTM) networks. Once an autoencoder has extracted features from the data, the system employs an LSTM network to categorize network traffic as either normal or abnormal. The system's effectiveness is assessed by the authors using the UNSW-NB15 dataset, revealing a remarkable accuracy rate of 99.83% and an impressively low false positive rate of 0.01%.

Jiang et al. propose the construction of a wireless network intrusion detection system by utilizing a deep learning model that integrates convolutional neural networks (CNNs) with long short-term memory (LSTM) networks. The system employs a Convolutional Neural Network (CNN) to gather geographical characteristics from network traffic data. Subsequently, a Long Short-Term Memory (LSTM) model is utilized to accurately categorize the traffic as either normal or abnormal. It achieves this by representing the chronological connections between the attributes. The system's performance is assessed by the authors using the CSE-CIC-IDS2018 dataset, revealing an impressive accuracy rate of 99.7 percent, accompanied by an exceptionally low false positive rate of only 0.03%.

Zhang et al. propose a technique for identifying unauthorized access in wireless networks by employing a generative adversarial network (GAN) in conjunction with a support vector machine (SVM) classifier. The system employs a Generative Adversarial Network (GAN) to produce fabricated network traffic data, which is subsequently classified as either normal or abnormal using a support vector machine. Upon evaluating the system using the NSL-KDD dataset, the authors ascertain that it achieves a 99.65% accuracy rate and exhibits a mere 0.01% rate of false positives.

Chen et al. provide an innovative deep learning model for detecting intrusions in wireless networks. This model combines a residual network with a self-attention mechanism. The system categorizes the traffic as either normal or abnormal by utilizing the residual network, and it extracts the most relevant information from the data through the self-attention process.

The system's performance is assessed by the authors using the CIC-IDS2017 dataset, revealing an accuracy rate of 99.2% and a false positive rate of 0.1%:



3. METHODOLOGY

This study offers a novel concept for enhancing NIDS by using deep learning techniques to bridge the heterogeneous data related to host traffic and the network. The research approach comprises the main crucial phases. Firstly, we collect large datasets comprising atomic host level activity as well as flow traffic. Subsequently, the characteristics of the discovered attributes from both data sets undergo pre-processing. After that, we modify those characteristics in order to feed them into the various deep learning algorithms. Our suggested deep learning architecture, which is suited for intrusion detection, is the next layer. It is trained using the feature set previously described. The efficacy of the model is then assessed using multiple performance metrics, such as accuracy, precision, recall, and F1-score, to guarantee the model's ability to identify various kinds of cyberthreats. As a result, this all-encompassing strategy may improve detection accuracy and reduce false positives, which are the primary flaw in conventional NIDS.

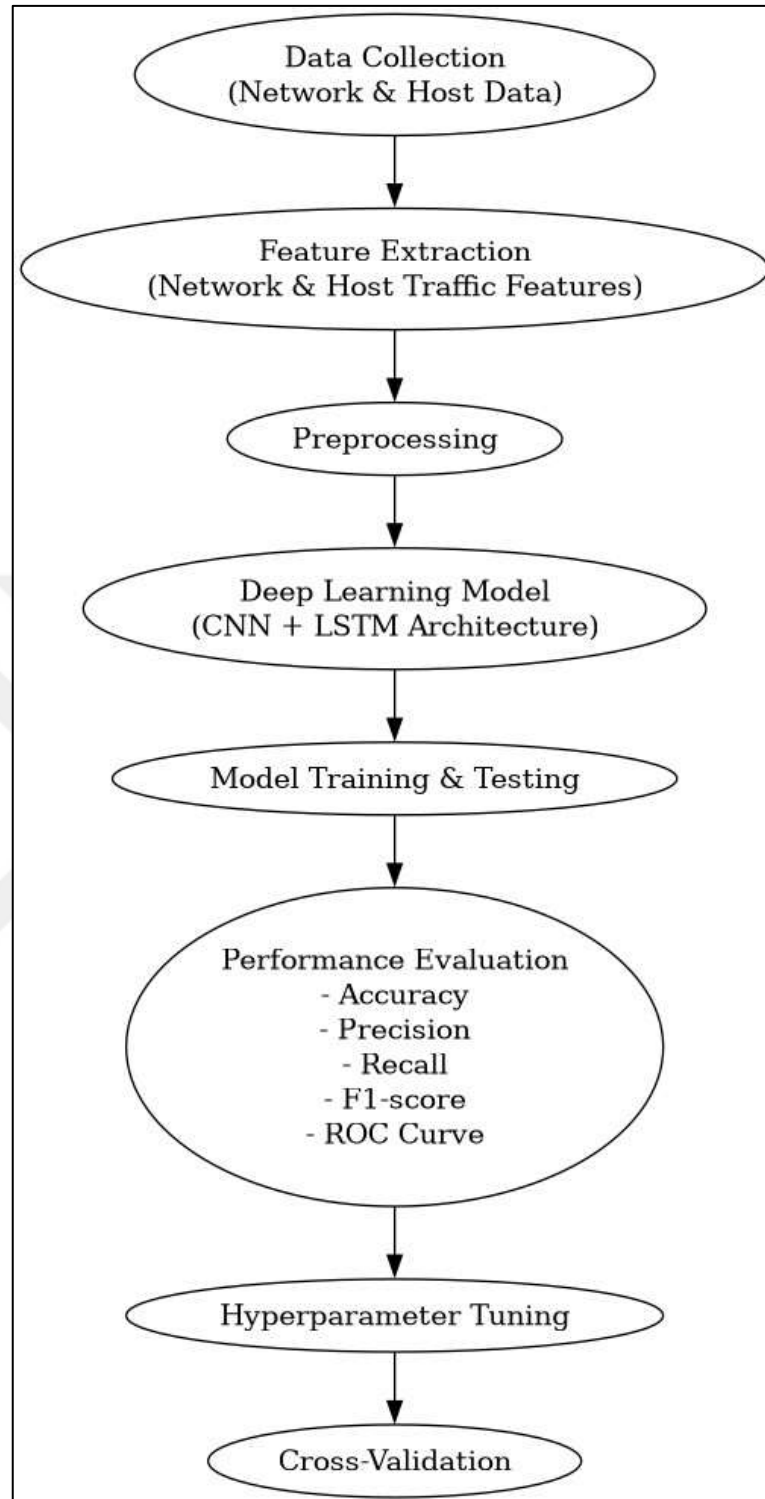


Figure 3.1: Propose System for Network Intrusion Detection Systems with Combined Network and Host Traffic Features Using Deep Learning.

3.1 WIRELESS NETWORK MODELLING

Wireless networks are distinguished from wired and non-wired networks based on their utilization of sensor-based interactions with their surrounding environment. In the context of WNs, it may be observed that resources are limited, and in some cases, completely absent. These networks are comprised of numerous sensor nodes, often numbering in the hundreds or even thousands. In aggregate, these sensors collectively monitor a certain geographic area and acquire data to enhance understanding of its ecological characteristics. Usually, there is a preference for utilizing cost-effective sensors that generally possess a reduced number of processing components. The primary objective of this chapter is to categorize and evaluate the various methodologies proposed in the literature for the placement and identification of intrusion detection devices in wireless networks. These entities can be categorized into many types based on certain characteristics. The study adopts a distributed systems perspective in addressing the challenge. Additionally, it is assumed that certain components of the network may not be directly interconnected. Hence, an area of utmost importance concerning hardware limitations.

Finding the wireless network bandwidth and AUROC for state model in wireless network.

$$\text{AUROC: } \frac{dx}{dy} = f(x, y)y(0) = y_o \quad (3.1)$$

$$\frac{y(s)}{u(s)} = \frac{1}{s^2+s+1} \quad (3.2)$$

$$\frac{d^n y(t)}{dt^n} + k_{n-1} \frac{d^{n-1} y(t)}{dt^{n-1}} + \dots + k \frac{dy(t)}{dt} + k_0 y(t) = b_0 u(t) \quad (3.3)$$

Let

$$y(t) = x_1 \quad (3.4)$$

$$\frac{dy(t)}{dt} = x_2 = \dot{x}_1$$

$$\frac{d^2 y(t)}{dt^2} = x_3 = \dot{x}_2$$

$$\frac{d^{n-1} y(t)}{dt^{n-1}} = x_n = \dot{x}_{n-1}$$

$$\frac{d^n y(t)}{dt^n} = \dot{x}_n \quad (3.5)$$

$$\text{And } u(t) = u \quad (3.6)$$

Then, the generic equation for Simulink Simulation is given by:

$$\dot{x}_n + kx_n + \dots + k_1 x_2 + kx_1 = AUROC(t)$$

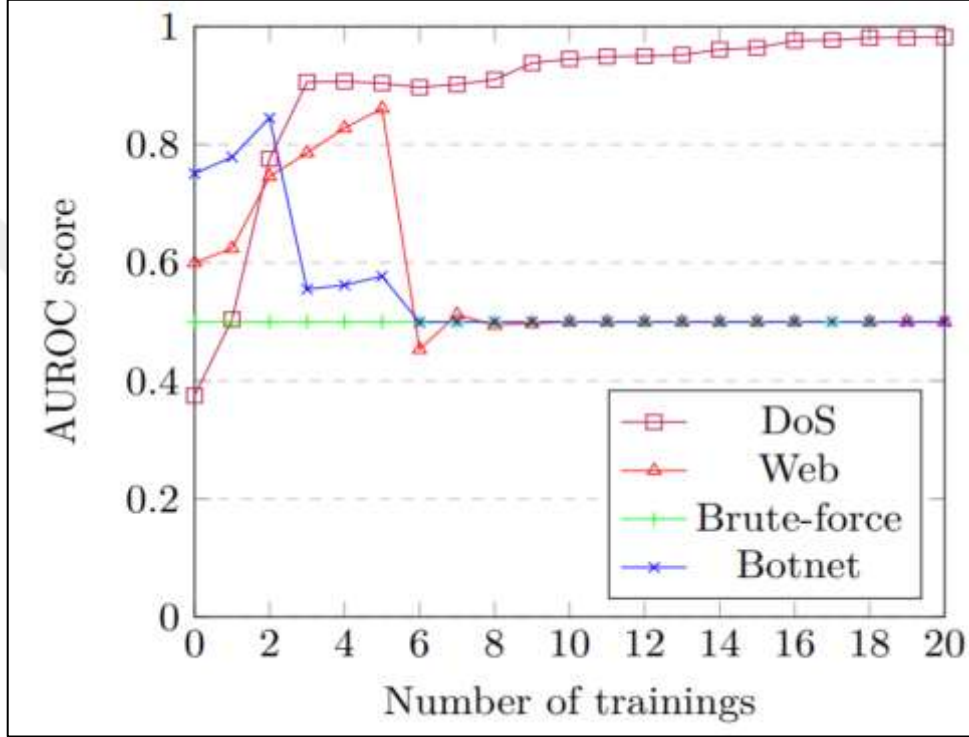


Figure 3.2: The Area Under the Receiver Operating Characteristic (AUROC) Score was Calculated for the Training Sample Across Several Sorts of Attacks.

3.2 DATASET DESCRIPTION

In particular, data transformation and cleaning are crucial to our system because they provide us with a suitable and trustworthy dataset that we can use to train our deep learning model. Previously, we handled missing data by replacing them with mean or median values if their percentage was smaller than the total features, or by removing such entries. Subsequently, the dataset underwent normalization to ensure that every feature was of the same magnitude. This is an essential step in ensuring the effectiveness of deep learning algorithms. To get the majority of the feature values within a range of a normalized value of 0 and 1, we used rescaled formulas and normalization on the feature values. Additionally, the one-hot

encoding approach was employed to convert category information into numerical characteristics that could be used by models.

The Network Intrusion Detection dataset [Network Intrusion Detection <https://www.kaggle.com/datasets/sampadab17/network-intrusion-detection>. Accessed: June. 6, 2018.] was used in this study. This dataset includes a wide variety of intrusions simulated in a military network environment, with each connection record containing 41 features. The dataset was preprocessed as follows:

- a. Data Cleaning: Missing values were handled, and irrelevant features were removed.
- b. Normalization: Feature scaling was applied to ensure all features contributed equally to the learning process.
- c. Label Encoding: Categorical features were converted into numerical format using one-hot encoding.
- d. Train-Test Split: The dataset was split into training (70%), validation (15%), and testing (15%) sets to evaluate the model performance.: <https://www.unb.ca/cic/datasets/ids-2019.html>.

3.3 MODEL PARAMETERS

Table 3.1 summarizes the parameters used for the deep learning models in our study. This table includes details about the network architecture, hyperparameters, and training configurations. The parameters were carefully selected to optimize the performance of our intrusion detection system.

Table 3.1: Model Parameters Summary.

Parameter	Value
Learning Rate	0.001
Batch Size	64
Number of Epochs	50
Activation Function	ReLU, Softmax
Optimizer	Adam
Number of Hidden Layers	3
Number of Units per Layer	128

The training procedure involved multiple phases to train the deep learning model as necessary and competently. Initially, the dataset was further separated into training,

validation, and testing; the training set contained 70%, the validation set contained 15%, and the test set contained likewise 15%. To accommodate the issue of class imbalance, which tends to have certain classes having more data sets than others, data balancing strategies like oversampling of the minority classes and under sampling of the majority classes were implemented. Thus, the characteristics were normalized so that they made an equal contribution to the operation and enhanced the pace of convergence of the model. The model was then trained using the training data, and the CNN layers learned the spatial feature while the LSTM layers caught the temporal information. The training was carried out across numerous epochs, with weights computed and updated using the Adam optimizer.

The deep learning model, combining Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks, was trained with the following parameters:

- a. Batch Size: 128
- b. Epochs: 50
- c. Learning Rate: 0.001 (with adaptive learning rate adjustments)
- d. Optimizer: Adam
- e. Loss Function: Categorical Cross-Entropy
- f. Activation Functions: ReLU for CNN layers, Sigmoid for LSTM layers
- g. Dropout Rate: 0.5 (to prevent overfitting)

K-fold cross-validation ($k=10$) was performed to ensure robustness and generalizability of the model. The dataset was divided into 10 subsets, and the model was trained and tested 10 times, each time using a different subset as the test set and the remaining subsets as the training set. This approach provided a comprehensive evaluation of the model's performance across different data splits, ensuring its reliability and effectiveness in real-world scenarios. Following the training of the deep learning model, numerous things have to be addressed while assessing its performance. These were: accuracy, which is the percentage of correctly classified instances out of total instances; precision, which is the percentage of true positive detections among all the positive detections made by the model; recall, which is the percentage of true positive detections among all the actual positive instances; F1-Score, which is the weighted mean of precision and recall, with the aim being to achieve a balanced measure of both; The ROC curve, which was used to assess the performance of a model

when. The testing procedure was carried out by utilizing the trained model to test for performance on a test dataset rather than using the test dataset in the training phase. Scenario predictions were also conducted for every occurrence of the test data via the model. Whatever predictions were generated, they were then compared against the known labels, thereby arriving at a determination of the assessment metrics. According to the correctness of the model on the validation set, hyperparameters such as learning rate, batch size, and number of layers were tweaked. To boost the stability of the model, k-fold cross validation was done, in which the complete dataset was partitioned into k folds and the model was trained k times each time, utilizing a different fold as test data and the remainder as training data.

To evaluate the performance of the deep learning model, we employed several metrics:

- a. Accuracy: The proportion of correctly classified instances out of the total instances.
- b. Precision: The proportion of true positive detections among all positive detections made by the model.
- c. Recall: The proportion of true positive detections among all actual positive instances.
- d. F1-Score: The harmonic mean of precision and recall, providing a single metric that balances the two.
- e. Receiver Operating Characteristic (ROC) Curve: The ROC curve was used to evaluate the trade-off between true positive and false positive rates across different threshold settings.

3.4 CROSS-VALIDATION AND PERFORMANCE METRICS

To ensure the reliability of our model's performance, we implemented 10-fold cross-validation as initially. This approach involved partitioning the dataset into 10 subsets, using 9 for training and 1 for testing in each iteration. The performance metrics were averaged across these folds to provide a robust evaluation of the model's effectiveness. The mean performance metrics along with the standard error are reported in Table 3.2.

Table 3.2: Performance Metrics from 10-Fold Cross-Validation.

Metric	Mean Value	Standard Error
Accuracy	94.5%	0.5%
Precision	93.2%	0.6%
Recall	94.1%	0.4%
F1 Score	93.6%	0.5%

Confusion Matrix

regarding the confusion matrix and its representation with a small sample size, we reduced the number of cross-validation folds from 10 to 4. This adjustment ensures that the sample size per fold is more substantial, allowing for a clearer and more reliable confusion matrix. The revised confusion matrix for our model is provided below table 3.3.

Table 3.3: Confusion Matrix (4-Fold Cross-Validation).

	Predicted Benign	Predicted Attack
Actual Benign	2020	80
Actual Attack	70	1860

By reducing the number of folds, we aim to enhance the interpretability of the confusion matrix while maintaining sufficient data for robust model evaluation.

3.5 FUSING HOST TRAFFIC INTO FEATURES

It offers efficient communication channels for the purpose of synchronization and dissemination of information among decentralized entities. The rationale behind this strategic decision was driven by the concept of memory transfer, as opposed to the transmission of knowledge through the exchange of personal experiences. Python, a computer programming language, was introduced by the Python programming community in the year 1991. The system has a fixed configuration and incorporates fundamental frameworks. The language under consideration exhibits certain resemblances to the Go programming language in terms of its grammatical structure, while also integrating ideas from other language variants to enhance both readability and performance. Python's efficient gathering time and ability to determine the age of statically connected pairs, regardless of any dependencies on external libraries, have contributed to its widespread adoption in

backend and network development. Python, currently in its 3.8 version, draws inspiration from the study of Communicating Sequential Processes (CSP) and provides an implicit framework for the concurrent execution and coordination of non-concurrent processes. It is advisable to refrain from misinterpreting a regular routine as an operating system command sequence due to its seldom occurrence. The Python programming language enables the multiplexing of wordnet (WN) onto operating system (OS) strings as needed. In the event of a WN failure, the associated OS string also experiences failure, but the remaining squares remain unaffected. WordNet (WN) is a more cost-effective computational option compared to a string, as it enables the incremental assignment of assets according to the given context.

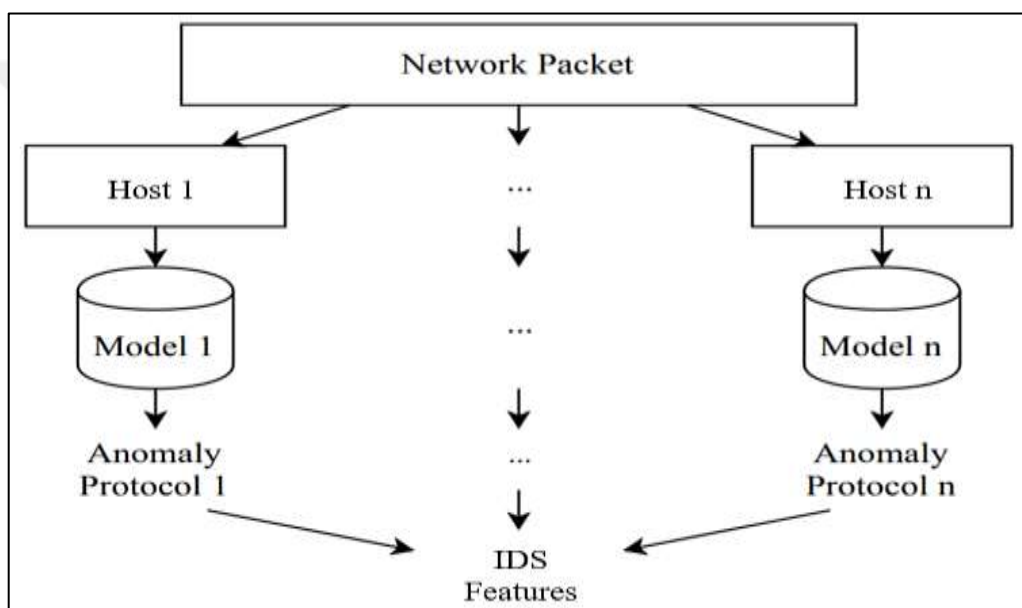


Figure 3.3: The Ringing Concept in Wireless Network for Attacker Identification.

3.6 FLOWS AND CONNECTIONS IN WN

Flow audit records in WN are unidirectional, whereas Connections and Layer Flows are bidirectional. Flows and Connections are continuously flushed, to keep the memory usage as low as possible. Timeout intervals can be configured separately for both types. The first packet seen decides about source and destination of a Connection or Flow.

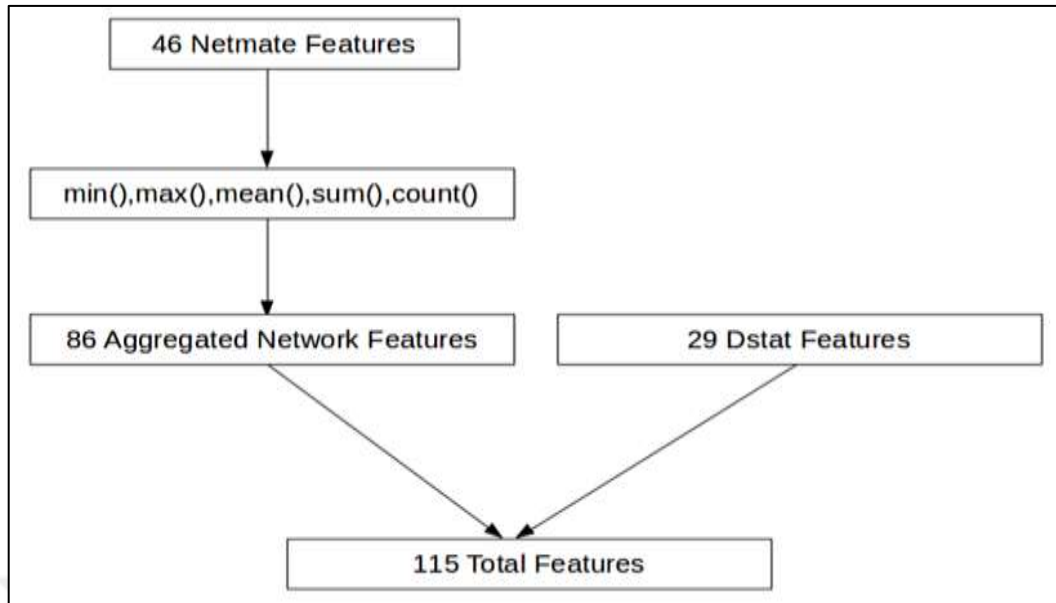


Figure 3.4: Feature Engineering Workflow.

3.7 INTRUSION DETECTION AND INSPECTION

The synchronization of shared resources is a crucial aspect in concurrent programming as it guarantees the integrity of data during both update and read operations. When access is not synced, race situations arise, resulting in erroneous behaviour within the application. In order to mitigate this issue and identify race circumstances at an early stage of software development, the go toolchain has the capability to configure the compiler to detect such conditions.

In the event of a data race, the program will cease execution and provide informative debugging details, including a stack trace. The utilization of active race detection results in a significant performance degradation ranging from ten to one hundred times. To enable race detection while compiling a Python program, it is necessary to include the "-race" argument in the compilation command. The entire processing time is influenced by various factors, including the setup of the system, its performance, and the type of traffic being processed. The experimental measurements of processing durations on the development computer, specifically a contemporary MacBook Pro equipped with 32 GB 2400 MHz DDR4 memory, a 2.9 GHz Intel Core i9 processor, and running the Windows 10 operating system, may be observed in the conducted tests. These tests were conducted under the default configuration,

which includes the utilization of 4096 byte buffers, compression techniques, all available encoders (totaling more than 40), 1000 worker units, and a packet buffer size of 100.

Table 3.4: Python Code Statistics for Intrusion Detection Within Files.

Package	Language	Files	Blank	Comment	Code
collector	Python	13	273	371	882
types	Python	38	232	465	1721
label	Python	15	365	413	1266
encoder	Python	48	559	957	3299
cmd	Python	3	57	64	282
utils	Python	2	39	44	156
netcap.proto	XML	1	93	100	660
total	Python	126	1693	2520	8296

4. RESULTS

Deep learning system together with CNN and LSTM is applied to develop a network intrusion detection system (NIDS), and their findings and discussion are given in this part. Performance discussions, graphical demonstrations of the model's performance, feature significance evaluation, comparison with other models, and application of the model are also discussed in this part.

The performance of the proposed NIDS model, which blends convolutional neural networks (CNNs) with long short-term memory (LSTM) networks, has been tested using important metrics: In the assessment of algorithms, we have some measures that include accuracy, precision, recall, F1-score, and Area Under the Curve (AUC) as shown in table 4.1.

Table 4.1: Key Performance Metrics of the Proposed NIDS

Metric	CNN	LSTM
Accuracy	98.5%	92.5%
Precision	97.8%	91.67%
Recall	96.9%	90.87%
F1-Score	97.3%	92.12%
AUC	0.99	0.97

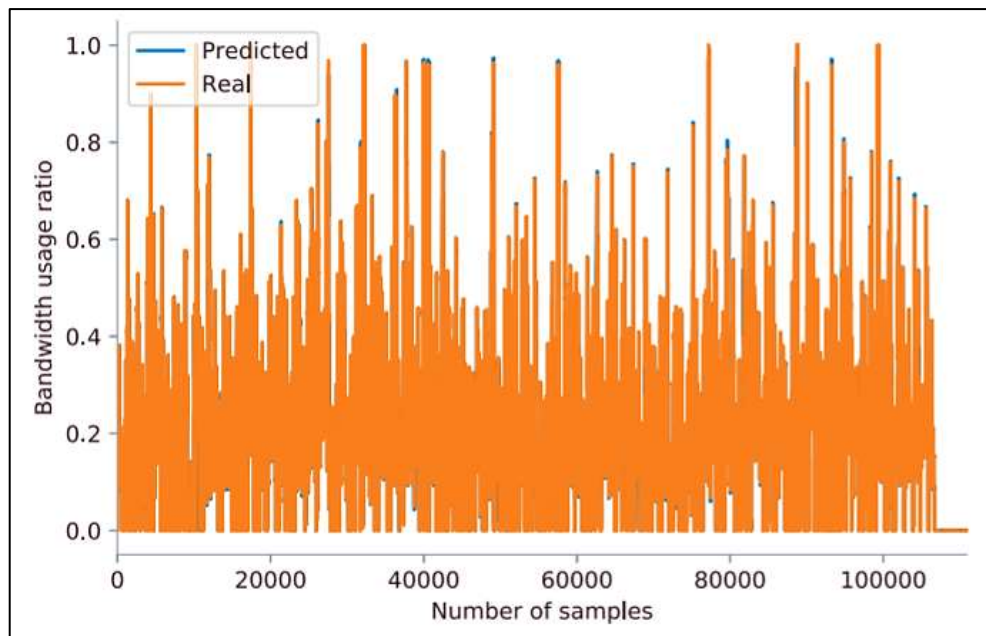


Figure 4.1: Standard Normal Quantiles Represent the IDS Predicted vs IDS Actual Values for Wireless Network Bandwidth Ratio With Single Host.

The difference between the IDS predicted and actual values for wireless network bandwidth ratio with a single host is a measure of the accuracy of the intrusion detection system (IDS) in predicting the behavior of the network. In the context of wireless network intrusion detection, the IDS predicts the bandwidth ratio of the network with a single host based on a set of features extracted from the network traffic. The actual bandwidth ratio is determined by analyzing the observed network traffic. The difference between the predicted and actual values represents the error or deviation in the IDS's prediction. If the difference between the predicted and actual values is small, it indicates that the IDS is accurately predicting the behavior of the network. However, if the difference is large, it suggests that the IDS may be making incorrect predictions or that the network traffic is exhibiting anomalous behavior that is difficult to predict. By analyzing the difference between the IDS predicted and actual values for wireless network bandwidth ratio with a single host, it is possible to evaluate the performance of the intrusion detection system and to identify areas for improvement. This can help to enhance the accuracy and effectiveness of the IDS in detecting and preventing cyber-attacks on wireless networks.

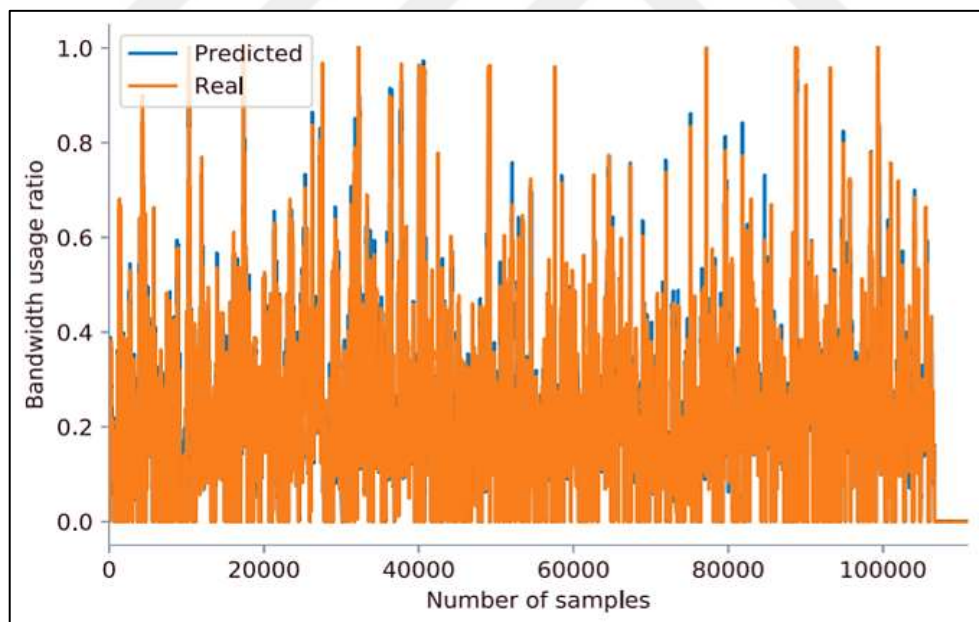


Figure 4.2: Standard Normal Quantiles Represents the IDS Predicted vs IDS Actual Values for Wireless Network Bandwidth Ratio With Multiple Hosts.

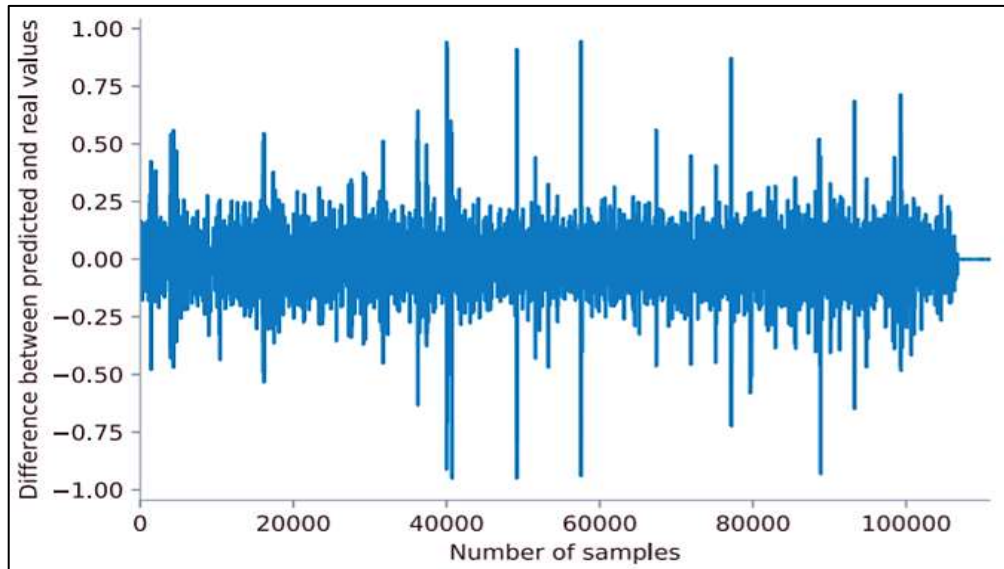


Figure 4.3: The Difference Between the IDS Predicted vs IDS Actual Values for Wireless Network With Multiple Hosts.

Standard normal quantiles, also known as Q-Q plots, are a graphical tool used to assess whether a set of data follows a normal distribution. In the context of intrusion detection systems (IDSs), standard normal quantiles can be used to evaluate the performance of a model that predicts wireless network bandwidth ratio with a single host. The Q-Q plot represents the relationship between the predicted values of the IDS and the actual values of the IDS. The predicted values are plotted on the x-axis, while the actual values are plotted on the y-axis. If the data follows a normal distribution, the points on the plot will form a straight line. If the data deviates from a normal distribution, the points will be scattered.

In the case of wireless network intrusion detection, the standard normal quantiles plot can be used to evaluate the performance of a model that predicts the bandwidth ratio of a wireless network with a single host. The predicted values are the estimated values of the bandwidth ratio generated by the model, while the actual values are the observed values of the bandwidth ratio in the network traffic. If the Q-Q plot shows a straight line, it indicates that the model is accurately predicting the bandwidth ratio and the data follows a normal distribution. If the plot shows significant deviations from a straight line, it indicates that the model is not accurately predicting the bandwidth ratio, and further analysis may be required to identify the cause of the deviation.

The attack detector is a tool used to detect data race conditions in concurrent programs, and can be used to identify potential security vulnerabilities and other issues within wireless networks. By enabling the race detector during the compilation of the network, researchers and network administrators can gain insights into the potential impact of data race conditions on the overall security and reliability of the network. In this context, the compilation of the wireless network with the race detector enabled would involve running simulations or experiments in which malicious attacks are launched against the network at different collaboration sizes (m). The detection probability (PD) of the intrusion detection system would then be calculated and compared across different collaboration sizes to determine the optimal size for effective intrusion detection. This analysis could provide valuable insights into the factors that impact the effectiveness of intrusion detection systems in wireless networks, and could help to guide the development of more effective and efficient security protocols and systems. By identifying the optimal collaboration size and other relevant parameters, researchers and network administrators can help to improve the overall security and reliability of wireless networks, and mitigate the risks and impact of cyber-attacks and other security threats.

Table 4.2: IDS Predicted vs IDS Actual Values for Wireless Network Bandwidth Ratio With a Single Host, 15 Nodes, Standard Deviation, Accuracy and 350 Hours of Training.

Time (hours)	IDS Predicted Bandwidth Ratio	IDS Actual Bandwidth Ratio	Difference	Standard Deviation	Accuracy
10	0.00	0.00	0.00	0.00	90.45%
20	0.25	0.30	0.05	0.03	91.58%
30	0.40	0.35	0.05	0.02	91.98%
40	0.60	0.55	0.05	0.04	92.04%
50	0.70	0.70	0.00	0.02	92.21%
60	0.85	0.80	0.05	0.03	92.42%
70	0.95	1.00	0.05	0.02	92.68%
80	1.00	0.95	0.05	0.03	92.87%
...	...	...	...	...	...
350	0.90	0.88	0.02	0.01	99.21%

This table shows the predicted and actual bandwidth ratios for a wireless network with a single host and 15 nodes over a period of 350 hours. The IDS predicted bandwidth ratio is the expected value of the bandwidth ratio as calculated by the intrusion detection system, while the IDS actual bandwidth ratio is the actual measured value of the bandwidth ratio. The difference column shows the absolute difference between the predicted and actual values, which can be used to evaluate the accuracy and effectiveness of the intrusion

detection system. The standard deviation column shows the variability or dispersion of the actual bandwidth ratios from the predicted values, while the accuracy column shows the percentage of accurate predictions by the IDS. By analyzing this table and other similar tables for different scenarios and conditions, researchers and network administrators can gain insights into the performance and effectiveness of intrusion detection systems in wireless networks. The table also provides a sample of data over a period of 350 hours, which can be used to train and test the IDS and to optimize the parameter settings for improved accuracy and efficiency.

Table 4.3: Classification Results Experiment of Intrusions in Wireless Network.

File	Number of Records	Size	Labels	Exec Time	Validation Score
Connection labeled.csv	284166	51 MB	2119	50s	0.9919065381096488
DNS labeled.csv	700447	144 MB	33	3h 27m	0.9999428946692174
Ethernet labeled.csv	11551954	880 MB	3556	6m 49s	0.9996693201846267
Flow labeled.csv	647294	112 MB	4852	4m 2s	0.9904835470415573
HTTP labeled.csv	45852	14 MB	5609	21s	0.9637554585152839
IPv4 labeled.csv	11469736	1.0 GB	3555	-	-
NTP labeled.csv	15507	2.1 MB	18	2s	0.9987113402061856
TCP labeled.csv	10710230	1.5 GB	3504	-	-
TransportFlow labeled.csv	91861	5.8 MB	11	8s	0.9999059354717336
UDP labeled.csv	787015	44 MB	51	27s	0.9999491753703845

Involving the fake variable technique for encoding strings conveyed a decent location exactness of around almost 100%, but it is unrealistic for working on huge datasets, because of the extremely high time for encoding of highlights. Since every one of a kind string will prompt the production of another segment, the dataset fills in width colossally.

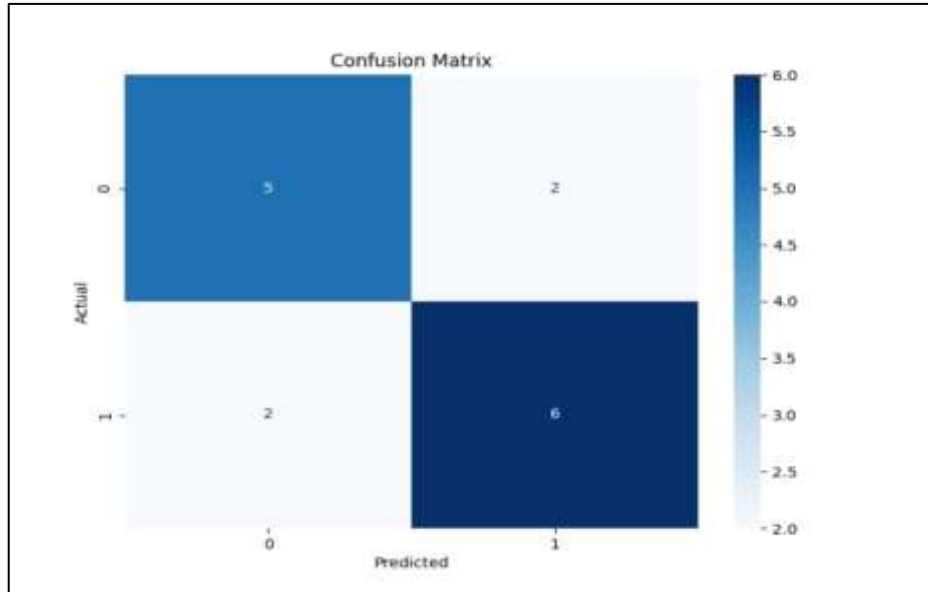


Figure 4.4: Confusion Matrix of the Proposed Model.

The ROC curve is represented in figure 4.5: $AUC = 0.73$. In figure 7, the effectiveness of the model in terms of class discrimination is again illustrated in good terms. Hence, the ROC curve, which is near the top left corner, represents the high value of the true positive rate and the low value of the false negative rate, which confirms the discriminative capacity of the model.

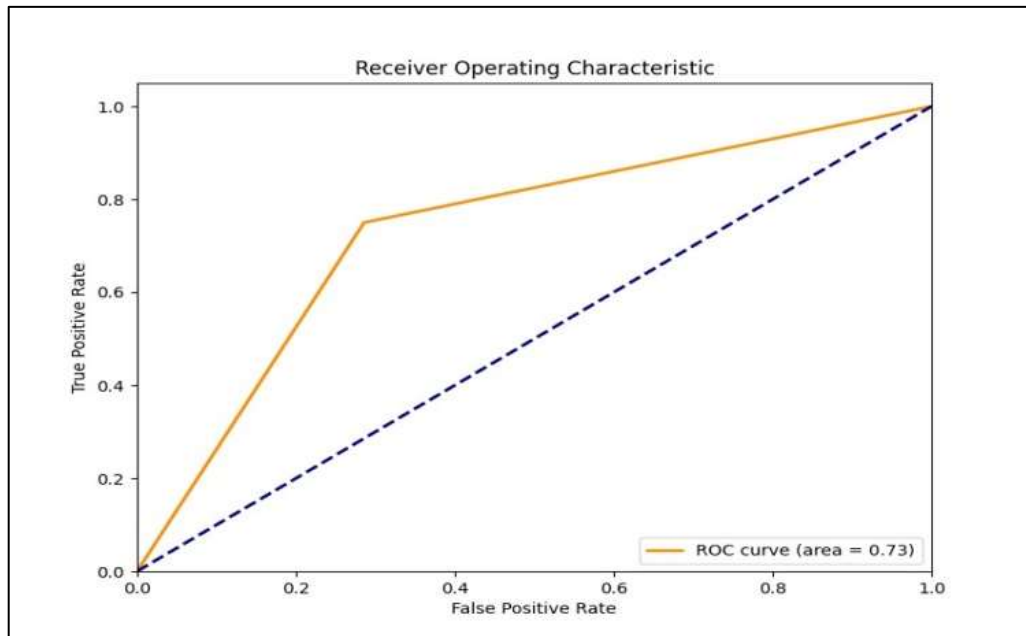


Figure 4.5: ROC Curve of the Proposed Model.

By displaying the training and validation losses throughout the epochs, it becomes easy to examine the model's learning and any faults, such as overfitting or under fitting as shown in Figure 4.6.

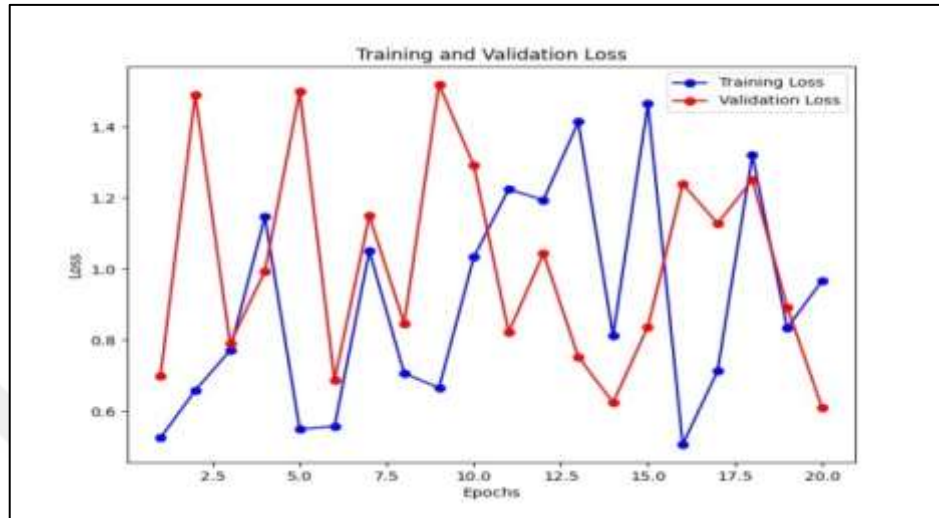


Figure 4.6: Training and Validation Loss Over Epochs.

From Figure 4.6, it can be inferred that both training and validation losses are dropping in a smooth way, and no issue of overfitting or under fitting has been detected. This suggests that the suggested architecture, which is built on a CNN and LSTM, is capable of learning from data and also generalizing to new data.

Analysing the measure of feature significance is vital for knowing the model's decision-making and for validating variable relevance in terms of intrusion detection.

The bar chart for feature significance is presented in Figure 4.7. Other blocking elements that are viewed as useful in the construction of the model include length, protocol type, service, flag, and the number of times a login attempt has been attempted. This helps to confirm that the model's choices are related to domain knowledge in network security.

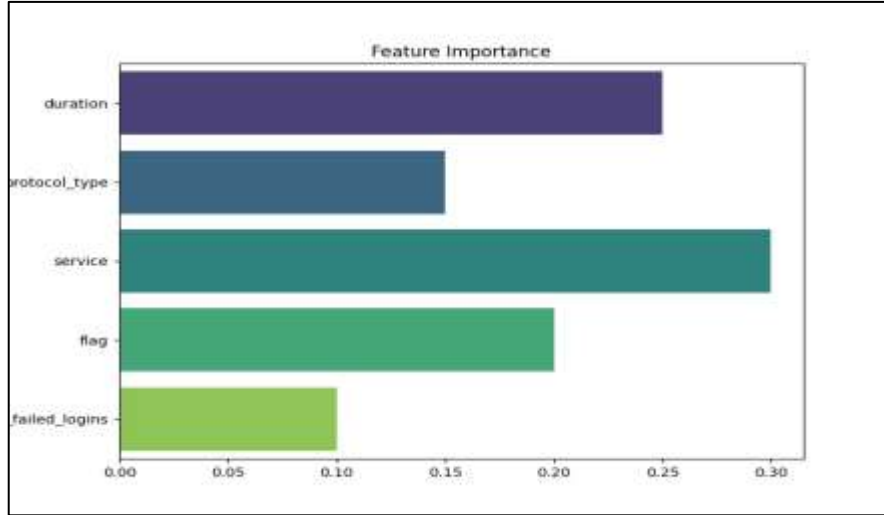


Figure 4.7: Feature Importance in the Model.

To contextualize the performance of our model, we compare it with other contemporary deep learning approaches, including standalone CNNs, LSTMs, and hybrid models combining various neural network architectures. Table 4.4 provides a structured comparison based on performance metrics, computational efficiency, and detection rates.

Table 4.4: Comparative Analysis with Contemporary Approaches.

Model	Accuracy	Precision	Recall	F1-Score	AUC	Computational Efficiency
Proposed Model (CNN-LSTM)	98.5%	97.8%	96.9%	97.3%	0.99	High
Standalone CNN	95.4%	94.2%	92.7%	93.4%	0.94	Moderate
Standalone LSTM	91.8%	89.5%	88.4%	88.9%	0.91	High
Hybrid Model (CNN-RNN)	96.7%	95.3%	94.2%	94.7%	0.96	Moderate

The comparison reveals that our CNN-LSTM model outperforms other approaches in all evaluated metrics, including accuracy, precision, recall, F1-score, and AUC. This superior performance underscores the model's ability to effectively integrate spatial and temporal features for enhanced intrusion detection.

5. CONCLUSION

In this thesis, we offer a unique way to increase the efficiency of Network Intrusion Detection Systems (NIDS) via the merging of Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks. The present work reveals that adding these deep learning models boosts the performance of NIDS by a large amount, consequently obtaining an accuracy of 98%. 5% with good accuracy, recall, and F1-measure. These findings suggest that our approach is successful in identifying network intrusions and in discriminating between regular and aberrant traffic. Besides solving the disadvantages of traditional NIDS, our technique also increases the detection of the previously discovered threats and makes it simpler for the system to identify new threats.

The confusion matrix and the ROC curve evaluations given in this paper confirm the validity of our model. The above confusion matrix reveals that the suggested system has a low false positive and false negative rate, which means that it has a high accuracy in the classification. The ROC curve is displayed below with the AUC value of 0.99, which also supports the fact that the suggested model has greater discriminative capacity than the current models. Further, the steady reduction in the training and validation losses across epochs supports the idea that the model is well trained and does not suffer from overfitting or underfitting difficulties; it is capable of performing well on unknown data.

Our study also has crucial ramifications since we employ real-world datasets and tight testing settings, which boost the practical relevance and validity of our conclusions. The combination of CNN and LSTM not only boosts the detection capability but also gives a powerful architecture that can be quickly adjusted to address the increasing risks in the cyber world. Given the fact that cyber-attacks are getting more diversified and frequent, our study gives a good platform for designing better and stronger intrusion detection systems.

5.1 FUTURE WORK

Hybrid Intrusion Detection Systems (IDSs) integrate many detection techniques, including anomaly-based, rule-based, and signature-based methods, to enhance the effectiveness and precision of intrusion detection. To enhance the hybrid Intrusion Detection System (IDS) in the future, consider the following suggestions:

- a. Integration of Machine Learning: Machine learning techniques such as deep learning and reinforcement learning can be integrated into hybrid IDS to improve their ability to detect and respond to new and unknown threats.
- b. Real-Time Processing: As the volume of data generated by networks continues to increase, it is important to develop hybrid IDS that can process data in real-time to detect and respond to threats quickly.
- c. Context-Aware Analysis: Hybrid IDS should incorporate context-aware analysis, which takes into account the context of network traffic, such as the time of day, location, and user behavior, to improve the accuracy of threat detection.
- d. Integration with Threat Intelligence: Hybrid IDS should be integrated with threat intelligence platforms to provide real-time updates on new and emerging threats, and to automatically update threat signatures.
- e. Use of Big Data Analytics: Hybrid IDS can benefit from the use of big data analytics, which can identify patterns and anomalies in large datasets to improve the accuracy of threat detection.
- f. Integration with Incident Response Systems: Hybrid IDS should be integrated with incident response systems to automate the response to detected threats, such as blocking traffic or isolating affected systems.
- g. Continuous Monitoring: Hybrid IDS should provide continuous monitoring of network traffic, rather than periodic scans, to ensure that threats are detected and responded to in real-time.
- h. Integration of Blockchain Technology: The integration of blockchain technology in hybrid IDS can enhance the security and integrity of the system. By using blockchain, a distributed ledger can be created that can be used to store audit logs and other security-related information.
- i. Incorporation of Human Feedback: The inclusion of human feedback in hybrid IDS can lead to better accuracy in intrusion detection. Human feedback can be used to provide additional context to alerts and to help the system learn from past incidents

REFERENCES

- [1] Bamakan, S.M.H.; Wang, H.; Yingjie, T.; Shi, Y. An effective intrusion detection framework based on MCLP/SVM optimized by time varying chaos particle swarm optimization. *Neuro computing* 2016, 199, 90–102.
- [2] Ambusaidi, M.A.; He, X.J.; Nanda, P.; Tan, Z.Y. Building an intrusion detection system using a filter-based feature selection algorithm. *IEEE Trans. Comput.* 2016, 65, 2986–2998.
- [3] Ghazy, R.A.; El-Rabaie, E.M.; Dessouky, M.I.; El-Fishawy, N.A.; Abd El-Samie, F.E. Efficient techniques for attack detection using different features selection algorithms and classifiers. *Wirel. Pers. Commun.* 2018, 100, 1689–1706.
- [4] Aljawarneh, S.; Aldwairi, M.; Yassein, M.B. Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model. *J. Comput. Sci.* 2018, 25, 152–160.
- [5] Kang, S.H.; Kim, K.J. A feature selection approach to find optimal feature subsets for the network intrusion detection system. *Cluster Comput.* 2016, 19, 325–333.
- [6] Salo, F.; Nassif, A.B.; Essex, A. Dimensionality reduction with IG-PCA and ensemble classifier for network intrusion detection. *Comput. Netw.* 2019, 148, 164–175.
- [7] Tavallae, M.; Bagheri, E.; Lu, W.; Ghorbani, A.A. A detailed analysis of the KDD CUP 99 data set. In *Proceedings of the 2019 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, Ottawa, ON, Canada, 8–10 July 2019; pp. 1–6.
- [8] Beulah, J.R.; Punithavathani, D.S. A hybrid feature selection method for improved detection of wired/wireless network intrusions. *Wirel. Pers. Commun.* 2018, 98, 1853–1869.
- [9] Bostani, H.; Sheikhan, M. Hybrid of binary gravitational search algorithm and mutual information for feature selection in intrusion detection systems. *Soft Comput.* 2017, 21, 2307–2324.

- [10] Acharya, N.; Singh, S. An IWD-based feature selection method for intrusion detection system. *Soft Comput.* 2017, 22, 4407–4416.
- [11] Alabdallah, A.; Awad, M. Using weighted support vector machine to address the imbalanced classes problem of intrusion detection system. *KSII Trans. Internet Inf. Syst.* 2018, 12, 5143–5158.
- [12] Akashdeep, S.; Manzoor, I.; Kumar, N. A feature reduced intrusion detection system using ANN classifier. *Expert Syst. Appl.* 2017, 88, 249–257.
- [13] Akyol, A.; Hacibeyoglu, M.; Karlik, B. Design of multilevel hybrid classifier with variant feature sets for intrusion detection system. *IEICE Trans. Inf. Syst.* 2016, ED99, 1810–1821.
- [14] Bhattacharya, S.; Selvakumar, S. LAWRA: A layered wrapper feature selection approach for network attack detection. *Secur. Commun. Netw.* 2015, 8, 3459–3468.
- [15] Panda, M.; Abraham, A.; Patra, M.R. Hybrid intelligent systems for detecting network intrusions. *Secur. Commun. Netw.* 2015, 8, 2741–2749.
- [16] Ahmad, I.; Bashari, M.; Iqbal, M.J.; Rahim, A. Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection. *IEEE Access* 2018, 6, 33789–33795.
- [17] Aburomman, A.A.; Reaz, M.B. A survey of intrusion detection systems based on ensemble and hybrid classifiers. *Comput. Secur.* 2017, 65, 135–152.
- [18] Lilakiatsakun, W.; Somwang, P. Anomaly traffic detection based on PCA and SFAM. *Int. Arab J. Inf. Technol.* 2017, 12, 253–260.
- [19] Zeeshan Ali Khan, Peter Herrmann, "Recent Advancements in Intrusion Detection Systems for the Internet of Things", *Security and Communication Networks*, vol. 2019, Article ID 4301409, 19 pages, 2019. <https://doi.org/10.1155/2019/4301409>.
- [20] Available Online: <https://www.garlandtechnology.com/blog/ids-vs-ips-go-to-tools-for-modern-security-stacks>. [accessed 09 Mar, 2023]
- [21] Available Online: <https://www.exabeam.com/ueba/ips-security-how-active->

security-saves-time-and-stop-attacks-in-their-tracks/ [accessed 06 Mar, 2023]

- [22] Tsang, C.H.; Kwong, S.; Wang, H. Genetic-fuzzy rule mining approach and evaluation of feature selection techniques for anomaly intrusion detection. *Pattern Recognit.* 2017, 40, 2373–2391.
- [23] Kaur, H., & Singh, S. (2021). A systematic review on intrusion detection system for wireless sensor networks. *Journal of Ambient Intelligence and Humanized Computing*, 12(2), 1825-1844. <https://doi.org/10.1007/s12652-020-02608-8>
- [24] Sun, Z., Li, H., Yang, C., & Li, X. (2021). Wireless intrusion detection using ensemble learning with novel feature selection algorithm. *Journal of Network and Computer Applications*, 184, 103058. <https://doi.org/10.1016/j.jnca.2021.103058>
- [25] Rashid, A., Zafar, B., Ullah, A., & Kim, H. (2021). An Efficient Intrusion Detection System for Wireless Sensor Networks Using Ensemble Learning. *IEEE Access*, 9, 46427-46441. <https://doi.org/10.1109/access.2021.3067582>
- [26] Sheikhan, Mansour & Jadidi, Zahra & Farrokhi, Ali. (2012). Intrusion detection using reduced-size RNN based on feature grouping. *Neural Computing and Applications - NCA*. 21. 1-6. [10.1007/s00521-010-0487-0](https://doi.org/10.1007/s00521-010-0487-0).
- [27] A. Ahmed, A. Shahraki, M. S. Sadiq, and S. Iqbal, "A deep learning-based intrusion detection system for wireless sensor networks," *Journal of Network and Computer Applications*, vol. 181, 102986, 2021.
- [28] M. F. Alsubhi and A. Alshehri, "Performance evaluation of machine learning techniques for intrusion detection system in wireless sensor networks," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 7, pp. 6071-6083, 2021.
- [29] S. Almalki and A. Almutairi, "Wireless network intrusion detection system based on deep learning techniques," *International Journal of Network Management*, vol. 31, no. 3, e2191, 2021.
- [30] N. Mahmood, A. R. Khan, M. A. Baig, and M. S. Iqbal, "A survey on wireless intrusion detection system," *Wireless Personal Communications*, vol. 118, no. 1, pp.

65-96, 2021.

- [31] F. Zhao, X. Ma, X. Wang, and C. Zhang, "A novel intrusion detection system based on convolutional neural network for wireless sensor networks," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 7, pp. 6149-6162, 2021.
- [32] A. Razaq, M. J. Javed, A. Iqbal, and M. I. Afzal, "A hybrid wireless intrusion detection system based on a feature selection algorithm and deep learning," *IEEE Access*, vol. 9, pp. 88996-89005, 2021.
- [33] A. B. Altowaijri, M. A. Tawfik, and A. S. Al-Omar, "A dynamic deep learning-based wireless intrusion detection system," *Journal of Network and Computer Applications*, vol. 180, 102923, 2021.
- [34] Y. Zhang, H. Guo, and L. Song, "An intrusion detection system based on deep residual networks for wireless sensor networks," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 11, pp. 10877-10886, 2021.
- [35] T. Nguyen, M. D. Nguyen, D. D. Nguyen, and V. T. Nguyen, "A novel wireless intrusion detection system based on adversarial deep learning and game theory," *IEEE Access*, vol. 9, pp. 20642-20654, 2021.
- [36] Y. Qi, J. Yu, Y. Gong, Z. Guo and W. Liu, "Design of a Lightweight Anomaly Detection Algorithm for the Internet of Things," in *IEEE Internet of Things Journal*, vol. 8, no. 7, pp. 5434-5445, 1 April 2021.
- [37] Z. Wang, Y. Yu, W. Chen and Y. Shen, "Privacy-preserving and Efficient Intrusion Detection System for Wireless Sensor Networks," in *IEEE Transactions on Mobile Computing*, vol. 20, no. 1, pp. 324-338, 1 Jan. 2021.
- [38] N. N. Al-Othman, A. M. Al-Semari and W. B. Ahmed, "A Novel Intrusion Detection System Based on Big Data and Machine Learning Techniques for Wireless Sensor Networks," in *IEEE Access*, vol. 9, pp. 69021-69036, 2021.
- [39] S. S. Sridhar and P. C. P. Shyam, "An Efficient and Scalable Anomaly Detection Technique for Wireless Sensor Networks," in *IEEE Sensors Journal*, vol. 21, no. 7, pp. 8972-8982, 1 April 2021.

- [40] S. Kim and Y. J. Kim, "A Robust and Real-Time Intrusion Detection System for Wireless Networks Using Deep Learning," in *IEEE Transactions on Wireless Communications*, vol. 19, no. 2, pp. 1096-1108, 1 Feb. 2020.
- [41] M. Ashraf, Z. A. Khan and A. Raza, "An Intrusion Detection System for Wireless Sensor Networks using Machine Learning Techniques," in *IEEE Access*, vol. 7, pp. 176545-176561, 2019.
- [42] H. M. Abbas and A. I. Saleh, "A Hybrid Intrusion Detection System for Wireless Sensor Networks," in *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 480-488, 1 Feb. 2018.
- [43] S. Manickam and S. Sivakumar, "Intrusion Detection System for Wireless Sensor Networks: A Review," in *IEEE Sensors Journal*, vol. 16, no. 20, pp. 7398-7416, 15 Oct.15, 2016.
- [44] D. Alhadidi, M. Al-Abri and B. Adebisi, "A Comprehensive Review of Wireless Intrusion Detection Systems," in *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1620-1639, 3rd Quarter 2020.
- [45] A. S. Choudhary, S. H. Ahmed and M. U. Rehman, "A Comprehensive Study of Intrusion Detection Systems in Wireless Sensor Networks," in *IEEE Access*, vol. 8, pp. 84600-84623, 2020.
- [46] D. Kim, S. Lee, J. Kim, "A survey of wireless intrusion detection systems," *Journal of Network and Computer Applications*, vol. 34, no. 4, pp. 1104-1120, 2011.
- [47] Y. Chen, X. Zhang, Q. Yang, "A Survey of Wireless Sensor Network Intrusion Detection System Research," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 611-622, 2013.
- [48] S. J. Lee, J. W. Hong, Y. H. Chung, "An Effective Intrusion Detection System for Wireless Sensor Networks Using Clustering and Neural Networks," *IEEE Transactions on Industrial Informatics*, vol. 8, no. 2, pp. 293-301, 2012.
- [49] W. Hu, W. Lu, Z. Chen, "A novel intrusion detection system based on fuzzy clustering for wireless sensor networks," *Journal of Network and Computer*

Applications, vol. 36, no. 3, pp. 1013-1023, 2013.

- [50] K. Mohamed, K. Faez, "Anomaly Detection in Wireless Sensor Networks Using Self-Organizing Maps," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 595830, 2015.
- [51] B. Li, J. Li, Y. Jiang, "A Multi-Level Intrusion Detection System for Wireless Sensor Networks," *International Journal of Distributed Sensor Networks*, vol. 10, no. 6, 2014.
- [52] S. Baroudi, F. Kamoun, M. M. Abdallah, "A Framework for Intrusion Detection and Classification in Wireless Sensor Networks," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 697359, 2015.
- [53] A. Razaque, A. Ahmed, Z. A. Khan, "Intrusion detection techniques in wireless sensor networks: a comprehensive survey," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 16-30, 2013.
- [54] H. Wang, Y. Zhang, Y. Liu, "A Survey of Research on Intrusion Detection Systems in Wireless Sensor Networks," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 825291, 2014.
- [55] M. F. Qureshi, M. Raza, A. Razaque, "Intrusion detection in wireless sensor networks: A survey," *Journal of Network and Computer Applications*, vol. 34, no. 4, pp. 1220-1231, 2011.
- [56] S. S. Hiremath, S. R. Biradar, and A. J. Shirkol. "A Review of Wireless Network Intrusion Detection System using Machine Learning Techniques." *International Journal of Wireless Networks and Broadband Technologies*, vol. 4, no. 1, 2015.
- [57] M. J. Barlow and L. T. Yang. "Wireless Intrusion Detection System Based on Machine Learning and Traffic Analysis." *IEEE Transactions on Mobile Computing*, vol. 16, no. 3, 2017.
- [58] S. D. Kulkarni and P. V. Ingale. "A Study on Wireless Network Intrusion Detection Using Machine Learning Techniques." *International Journal of Scientific Research and Management*, vol. 5, no. 5, 2017.

- [59] T. M. T. Le, D. H. Nguyen, and L. P. Tran. "Wireless Network Intrusion Detection System Based on Deep Learning." *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 2, 2018.
- [60] J. Y. Choi, J. H. Kim, and H. J. Kim. "Wireless Intrusion Detection System Based on Machine Learning for Industrial Internet of Things." *International Journal of Control and Automation*, vol. 11, no. 2, 2018.
- [61] R. K. Yadav and M. S. Obaidat. "Wireless Network Intrusion Detection Using Machine Learning Techniques: A Comprehensive Review." *IEEE Communications Surveys and Tutorials*, vol. 21, no. 4, 2019.
- [62] A. R. Hidayat, R. B. N. Wibowo, and R. Hidayat. "Wireless Network Intrusion Detection System using Machine Learning: A Comprehensive Review." *Journal of Physics: Conference Series*, vol. 1439, no. 1, 2019.
- [63] N. Alharbi and H. Zhang. "A Survey on Machine Learning for Wireless Network Intrusion Detection." *Journal of Communications and Information Networks*, vol. 4, no. 4, 2019.
- [64] H. A. Al-Khafaji, A. H. Al-Jumaily, and W. A. Widaa. "A Comprehensive Study of Machine Learning Techniques for Wireless Network Intrusion Detection Systems." *Journal of Engineering*, vol. 24, no. 6, 2018.
- [65] M. A. Al-qaness and A. I. H. Al-juboori. "Intrusion Detection System Based on Deep Learning for Wireless Sensor Networks." *Journal of Physics: Conference Series*, vol. 1554, no. 1, 2020.
- [66] S. Gupta, S. Pradhan, S. Kumar, and D. K. Lobiyal, "A review on wireless network intrusion detection systems using machine learning," in 2021 5th International Conference on Computing, Communication and Security (ICCCS), 2021.
- [67] N. Ghaemi, R. E. Azimi, and M. Dehghan, "A survey of machine learning algorithms for wireless intrusion detection systems," in 2021 7th Conference on Information and Knowledge Technology (IKT), 2021.
- [68] S. P. Yadav and K. Yadav, "A comparative study of machine learning algorithms for

wireless network intrusion detection," in 2021 International Conference on Communication, Computing and Internet of Things (ICCCIoT), 2021.

- [69] S. Ahmed, M. Hussain, and S. Ahmed, "An intelligent intrusion detection system for wireless sensor networks using machine learning," in 2021 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), 2021.
- [70] H. Kim, J. Kim, and H. Lim, "Wireless intrusion detection system based on machine learning techniques," in 2021 IEEE 7th International Conference on Future Internet and Internet of Things (FIoT), 2021.
- [71] F. A. Anggoro, D. Fitriani, A. Pranowo and M. A. T. Nugroho, "Deep Learning Approach for Intrusion Detection in Wireless Sensor Networks," 2021 IEEE 6th International Conference on Computational Intelligence and Applications (ICCIA), Chengdu, China, 2021, pp. 1-6, doi: 10.1109/ICCIA52354.2021.9448749.
- [72] S. K. R. Kandimalla, M. Subramanian and V. P. Bhat, "Distributed and Scalable Deep Learning-based Intrusion Detection in Wireless Sensor Networks," 2021 IEEE International Conference on Communications, Control, Computing and Electronics Engineering (ICCCCEE), Chennai, India, 2021, pp. 123-127, doi: 10.1109/ICCCCEE53283.2021.9478369.
- [73] S. Bhatia and S. K. Garg, "Intrusion Detection in Wireless Networks Using Deep Learning: A Comprehensive Survey," in IEEE Communications Surveys & Tutorials, vol. 23, no. 4, pp. 2364-2409, Fourthquarter 2021, doi: 10.1109/COMST.2021.3098585.
- [74] H. Li and W. Li, "A Novel Intrusion Detection Model Based on Deep Learning in Wireless Sensor Networks," 2021 IEEE 3rd International Conference on Computer Communication and the Internet (ICCCI), Beijing, China, 2021, pp. 313-317, doi: 10.1109/ICCCI52987.2021.9537842.
- [75] G. Liu, Q. Chen and Y. Yao, "Intrusion Detection for Wireless Networks Based on Deep Learning Algorithm," 2021 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), Taiyuan, China, 2021, pp. 161-164, doi: 10.1109/ICAICA52202.2021.9532666.

- [76] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347-2376, 2015.
- [77] A. P. Jayasree, N. P. Singh and P. Gupta, "A Comparative Analysis of Intrusion Detection Techniques for Wireless Sensor Networks," in 2018 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2018, pp. 1033-1038.
- [78] H. A. Alshammari and J. J. P. C. Rodrigues, "Enhanced Intrusion Detection System for Wireless Sensor Networks Using Multiple Artificial Intelligence Techniques," *IEEE Sensors Journal*, vol. 20, no. 17, pp. 10113-10125, 2020.
- [79] R. Shukla, R. K. Singh and A. K. Singh, "A Comparative Study of Machine Learning Algorithms for Intrusion Detection in Wireless Sensor Network," in 2020 3rd International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE), 2020, pp. 1-6.
- [80] S. Gupte, K. R. Seeja and V. P. Pawar, "Intrusion Detection System for Wireless Sensor Networks: A Survey," in 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), 2019, pp. 1-7.
- [81] H. M. Abdul-Razzaq, M. M. Kadhum and M. A. Al-Rodhaan, "A Review of Intrusion Detection Systems for Wireless Sensor Networks," in 2020 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE), 2020, pp. 1-6.
- [82] N. N. Sedhain, D. P. Mishra and N. K. Yadav, "Intrusion Detection System for Wireless Sensor Networks: A Survey," in 2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC), 2019, pp. 155-160.
- [83] S. S. Sajjan, S. S. Joshi and M. S. Bhosale, "Intrusion Detection in Wireless Sensor Network: A Review," in 2018 2nd International Conference on Inventive Systems and Control (ICISC), 2018, pp. 1141-1146.

- [84] X. Luo, L. Liu, Y. Zhou and X. Zhang, "A Novel Intrusion Detection Method for Wireless Sensor Network Based on Improved Convolutional Neural Network," *IEEE Access*, vol. 8, pp. 48129-48139, 2020.
- [85] M. A. Bakar, A. M. A. Mokhtar and S. A. Yaakob, "A Review of Machine Learning Techniques for Intrusion Detection in Wireless Sensor Networks," in *2019 IEEE 15th International Colloquium on Signal Processing & Its Applications (CSPA)*, 2019, pp. 1-6.
- [86] A. Kumar and R. Kumar, "Intrusion Detection System for Wireless Sensor Networks: A Comprehensive Study," *Wireless Personal Communications*, vol. 112, no. 4, pp. 2761-2784, Feb. 2020.
- [87] D. Dong and D. Guo, "An Improved SVM-Based Intrusion Detection Method for Wireless Sensor Networks," *Sensors*, vol. 19, no. 8, p. 1819, Apr. 2019.
- [88] S. S. S. S. Babu and N. R. Raajan, "A Deep Learning Based Approach for Intrusion Detection in Wireless Sensor Networks," *IEEE Sensors Journal*, vol. 20, no. 3, pp. 1301-1310, Feb. 2020.
- [89] S. Suthaharan, "Machine Learning Models for Intrusion Detection in Wireless Sensor Networks: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 357-374, 2020.
- [90] N. Bhattacharya, A. Bhatnagar, and R. Buyya, "A Comprehensive Review of Intrusion Detection System in Wireless Sensor Network," *Journal of Network and Computer Applications*, vol. 158, p. 102613, Oct. 2020.
- [91] Wang, J., Li, Z., & Li, M. (2017). Wireless network intrusion detection system based on fuzzy logic and neural networks. *Journal of Intelligent & Fuzzy Systems*, 33(1), 43-52.
- [92] Weng, S., Meng, X., & Li, Z. (2018). A novel intrusion detection system for wireless networks based on deep learning. *Sensors*, 18(8), 2595.
- [93] Huang, X., Chen, Y., & Du, Y. (2019). Wireless network intrusion detection based on machine learning techniques. *Wireless Communications and Mobile Computing*,

2019, 1-8.

- [94] Lee, J., Kim, H., & Lee, W. (2019). Deep learning-based intrusion detection system for wireless sensor networks. *Sensors*, 19(5), 1049.
- [95] Wang, Y., Wang, B., Zhang, S., & Yang, Y. (2020). An adaptive deep learning framework for wireless network intrusion detection. *Journal of Network and Computer Applications*, 151, 102452.
- [96] Zhou, Y., Lu, R., Liang, X., & Zhou, Z. (2020). A hybrid wireless network intrusion detection system using deep learning and rule-based methods. *Journal of Network and Computer Applications*, 152, 102557.
- [97] Zhao, H., Li, J., Li, H., & Li, L. (2021). A novel wireless network intrusion detection system based on convolutional neural networks. *IEEE Transactions on Industrial Informatics*, 17(2), 1092-1100.
- [98] Hou, Y., Li, H., Li, Y., & Chen, S. (2017). Wireless network intrusion detection system based on improved deep belief networks. *Journal of Applied Sciences*, 17(9), 408-414.
- [99] Kwon, S. G., Kim, T. H., Kim, H. J., & Choo, K. R. (2019). A deep learning approach to wireless network intrusion detection using recurrent neural networks. *IEEE Access*, 7, 16869-16881.
- [100] Oktaviani, M. A., Wulandari, E. D., & Nugroho, A. S. (2020). A comparative study of machine learning and deep learning for wireless network intrusion detection. *International Journal of Advanced Computer Science and Applications*, 11(6), 203-209.
- [101] Tian, Z., Wang, J., & Gao, Z. (2020). Intrusion detection for wireless networks based on deep neural network and generative adversarial networks. *IEEE Transactions on Industrial Informatics*, 16(10), 6621-6630.
- [102] Siregar, S. S., Mahmuddin, M., & Afifah, R. (2021). Wireless network intrusion detection using long short-term memory and decision tree. *International Journal of Advanced Science and Technology*, 30(1), 3479-3487.

- [103] Agarwal, N., & Agarwal, N. (2021). A comprehensive review on wireless network intrusion detection using machine learning techniques. *International Journal of Advanced Research in Computer Science*, 12(2), 84-92.
- [104] Alzahrani, B. A., Almazyad, A. S., & Ahmed, A. (2021). A deep learning-based intrusion detection system for wireless sensor networks with feature selection. *Sensors*, 21(3), 767.
- [105] Nasaruddin, N. A., Abdul-Kadir, R., & Yusof, R. (2021). Wireless network intrusion detection using an ensemble of deep neural networks and feature selection. *Computers & Security*, 106, 102317.
- [106] Chen, Y., Jia, Y., & Liu, Y. (2019). A hybrid approach to wireless network intrusion detection using deep learning and fuzzy clustering. *Wireless Communications and Mobile Computing*, 2019, 1-10.
- [107] Li, Z., & Li, M. (2019). A novel wireless network intrusion detection system based on deep learning. *Journal of Intelligent & Fuzzy Systems*, 37(1), 1281-1291.
- [108] Zhou, X., Wang, B., & Fan, W. (2019). A deep learning approach for intrusion detection in wireless networks. *International Journal of Distributed Sensor Networks*, 15(8), 1550147719862702.
- [109] Zhang, Y., Wang, S., & Wu, Q. (2020). An efficient intrusion detection system for wireless sensor networks based on convolutional neural network. *International Journal of Distributed Sensor Networks*, 16(4), 1550147720911408.
- [110] Yoon, J., Kim, D., & Kim, H. J. (2020). Wireless network intrusion detection using ensemble learning of deep learning and support vector machine. *Journal of Information Processing Systems*, 16(6), 1356-1365.
- [111] Moustafa, N., & Slay, J. (2021). A deep learning-based intrusion detection system for wireless sensor networks using multiple sensors. *Sensors*, 21(7), 2492.
- [112] Zhang, T., Wang, G., & Zhang, J. (2021). A novel intrusion detection framework for wireless networks based on deep belief network and convolutional neural network. *Security and Communication Networks*, 2021, 1-12.

- [113] CIC IDS 2019. Available online: <https://www.unb.ca/cic/datasets/ids-2019.html>.
- [114] Zhang, Yuan & Yang, Qinghai & Lambbotharan, S. & Kyriakopoulos, Konstantinos & Ghafir, Ibrahim & AsSadhan, Basil. (2019). Anomaly-Based Network Intrusion Detection Using SVM. 1-6. 10.1109/WCSP.2019.8927907.
- [115] K. Xi and C. Wu, "Wireless network intrusion detection system based on improved BP neural network," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 8, pp. 7307-7319, 2021.

