INSE – 6610 CYBERCRIME INVESTIGATIONS

Summer Term - 2023

Log Analysis of Intrusion Detection System using Machine learning – A survey and Review

Submitted to:

Professor Ivan Pustogarov, PhD

Submitted by:

Prithvik Adithiya Ravindran
(40195464)

Arundhathi Sivaprasad
(40194782)

Varun Venkat Gururajan
(40218830)

Gayathri Venkataramana
(40195042)

Shivendra Arulalan
(40197455)

Chris Melvin Franklin
(40188797)

Gurpreet Kaur
(40226346)

Adnaan Khan
(40185329)

Yaswanth Kalyanam
(40194341)

Arjith Colapakkam Anandaraj
(40203509)

Submitted on:

09th of August 2023, Summer Term - 2

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Abstract-- This report presents a comprehensive analysis of Intrusion Detection Systems (IDS), shedding light on their evolution, classifications, architectures, and inherent challenges and limitations. With a primary focus on the incorporation of machine learning (ML) techniques, the study dives into the ways in which ML has reshaped IDS, enhanced detection capabilities and improving overall system efficacy. Various ML techniques, datasets, and performance comparisons are discussed, offering a multidimensional view of contemporary IDS. The report also explores the application of IDS in different contexts like network monitoring systems and control systems, illustrating the versatile nature of these security measures. A comparative analysis underscores key findings and notable innovations, while the discussion segment synthesizes major themes and observations. Through an intricate blend of historical understanding, technical assessment, and forward-looking insight, the report contributes valuable knowledge to the existing literature on IDS and the transformative potential of machine learning.

1. INTRODUCTION

1.1 Background

The Evolution of Network Security:

Network security has always been paramount to ensuring the integrity and confidentiality of information across systems. Over the years, as the digital landscape expanded and became more intricate, so did the threats that sought to compromise it. Intrusion Detection Systems (IDS), born out of this necessity, have become instrumental in the proactive defense of these networks. These systems are more than mere digital guards; they are the eyes and ears of the network, constantly monitoring and detecting any possible threats.

The Era of Transformations:

From their rudimentary inception where IDS primarily relied on predefined rules and signatures to detect threats, we've now entered an era where the systems are far more sophisticated. One of the reasons for such a transition is the ever-evolving nature of cyber threats. Static defences were no longer sufficient as the malicious actors innovated and evolved their techniques. This urged IDS developers to think beyond the traditional, and thus began the integration of technological advancements into IDS, with machine learning (ML) being the most significant.

Machine Learning - The Game Changer:

In the realm of IDS, machine learning marked a revolutionary shift. Unlike conventional methods that leaned heavily on previously known patterns and signatures, ML-enabled IDS can learn from the data. This not only makes them efficient at detecting known threats but also gives them the capability to identify new, unknown threats by understanding patterns and anomalies.

1.2 Objectives

This report, in its essence, seeks to unfurl the world of IDS, presenting its multifaceted nature to the reader. With a blend of historical, technical, and futuristic insights, the study aims to provide a well-rounded understanding of intrusion detection systems.

Decoding IDS:

The initial segment of the report takes a deep dive into the standalone nature of IDS. From a historical perspective that traces the roots and evolution of these systems to their classifications that define their operational methodologies, this section is foundational. Furthermore, it brings to light the inherent challenges faced by IDS, emphasizing their limitations in a contemporary setting.

Expanding Horizons:

IDS are not confined to a singular application. Their versatility is what makes them so invaluable in the modern digital ecosystem. This section ventures into the myriad applications of IDS, examining their role in diverse environments such as network monitoring systems and control systems. Each application comes with its unique challenges and benefits, a spectrum this segment endeavours to explore.

The Convergence of IDS and ML:

The core of this report is the union of IDS with machine learning. It's where the past meets the future. This section delves deep into the intricacies of how ML techniques are being harnessed in IDS. From the fundamental principles that drive this integration to the tangible challenges faced during its implementation, this part of the report is comprehensive and insightful.

1.3 Importance of Machine Learning

Redefining Cybersecurity with AI:

The digital age has brought with it a new set of challenges, demanding solutions that evolve and adapt. Machine learning, a subset of artificial intelligence, has emerged as a beacon in this dynamic landscape. Its ability to learn, adapt, and predict has brought transformative changes to many sectors, with cybersecurity being one of the most impacted.

Benefits Galore:

Machine learning's integration with IDS brings forth numerous benefits. Firstly, the adaptability allows for the detection of not just known threats, but also new ones that might not have a predefined signature. This drastically reduces false negatives. Secondly, by analyzing vast datasets, ML can fine-tune its detection algorithms, thereby reducing false positives. Moreover, by continually learning, ML ensures that the IDS remains relevant, even as threat actors innovate their malicious techniques.

Navigating the Study:

Following this elaborate introduction, readers will journey through the vast landscape of IDS. The report, structured meticulously, guides the audience through a progression that starts with basic understanding and culminates into advanced insights.

The subsequent sections ensure a fluid narrative, each building on the other. From the standalone nature of IDS to their varied applications and eventually the revolutionary integration with machine learning, the report promises a holistic view. The comparative analysis and concluding discussion further distil the insights, offering actionable takeaways and food for thought.

Contribution to the Dialogue:

In an era where cyber threats are continually evolving, understanding the defenses becomes pivotal. Through this report's broad yet detailed examination, we aim to enrich the ongoing discourse surrounding IDS and emphasize the transformative potential of machine learning, offering a reservoir of knowledge to both experts and novices alike in this rapidly evolving field.

2. LITERATURE SURVEY

2.1 IDS STANDALONE SYSTEM

2.1.1 Introduction to Intrusion Detection Systems (IDS)

Intrusion Detection Systems (IDS) have rapidly become an essential component in the vast landscape of cybersecurity. With the rise of digital infrastructures, protecting sensitive information from malicious entities has taken precedence. IDS are deployed to monitor networks and systems, ensuring that malicious or unauthorized activities are promptly detected. This article delves into the background of IDS and underscores the importance of standalone IDS in ensuring robust security.

2.1.2 Background of Intrusion Detection Systems (IDS)

The concept of Intrusion Detection Systems has evolved over time, catering to the ever-changing cyber-threat landscape. The early history of intrusion-detection stems from the realm of expert systems. Yost (2016) provided an in-depth exploration into the early history of intrusion-detection expert systems (IDES), shedding light on the pioneering attempts to deploy expert systems in detecting unauthorized activities [5].

Mohamed et al. offered a primer on IDS, introducing them as systems specifically designed to detect unauthorized intrusion into computer networks and systems [2]. Over the years, research has broadened the scope of IDS, categorizing them based on different criteria. For instance, IDS can be classified into host-based (HIDS) and network-based (NIDS). While HIDS monitor activities on individual devices or hosts, NIDS analyse traffic on the entire network [7]. The former, as discussed by Ou et al. (2010), focuses on system-level activities, including system

calls, application logs, and file-system modifications [85]. On the other hand, NIDS, as highlighted by Guillen et al. (2009), focus on capturing and analysing packets in network traffic, which provides a more holistic view of the network activities [79].

With a surge in the number and complexity of cyber-attacks, researchers started exploring unique methodologies and techniques for IDS. Anomaly-based systems, for instance, look for patterns that deviate from established norms, making them particularly potent against novel attacks [9][11]. On the contrary, signature-based systems detect known patterns associated with specific threats, but their strength lies in the speed and accuracy of detection, given that the attack's signature is known [13].

2.1.3 Importance and Focus on Standalone IDS

The significance of Intrusion Detection Systems can't be understated. With organizations relying more than ever on digital tools, ensuring that their networks are free from unauthorized activities becomes imperative. As highlighted by Ashoor and Gore, the importance of IDS stems from their ability to not only detect but also potentially prevent malicious activities, saving organizations from financial losses and reputational damage [6].

The debate over the effectiveness of standalone IDS versus hybrid models has been an ongoing topic in the research community. Efe and Abacı (2022) undertook a comprehensive comparison of HIDS and NIDS, shedding light on their individual strengths and weaknesses [83]. While hybrid models, as discussed by Wang and Zhang (2012), advocate for a combined approach harnessing both HIDS and NIDS for optimal results [84], standalone IDS have their distinct advantages.

A standalone IDS, being specialized, can be tailored to the specific needs of an organization. Host-based systems, for example, can be fine-tuned to the intricacies of the host system, ensuring a depth of monitoring that might be impossible with broader, more generalized systems. Similarly, a standalone NIDS can be designed to capture and analyse every packet on a network, ensuring no data packet goes unchecked. Moreover, standalone systems can be optimized.

2.1.4 Historical Context and Evolution of IDS

Intrusion Detection Systems (IDS) have evolved significantly since their inception, primarily driven by the exponential growth in computer networks and the increasing sophistication of cyber threats. Understanding the history and progression of IDS offers insights into the motivations behind its development and the directions in which it's headed.

The genesis of intrusion detection can be traced back to the era when computer systems began to interact over shared networks. One of the first forays into this domain was the Intrusion-Detection Expert Systems (IDES), which represented the preliminary stages of automated monitoring and threat analysis [5].

IDEs were the pioneering phase, using heuristic approaches to detect unauthorized activities on computer systems. Systems like these relied heavily on defined rule sets and exhibited patterns to identify anomalous behaviour. The primary motivation at this time was to automate the task of monitoring vast amounts of log data, which would have been impossible for humans to analyse exhaustively.

Throughout the late 20th century, the complexity and scale of cyberattacks increased. Hackers were not just independent actors but organized groups or even state-sponsored entities. This period saw the transition from simple, rule-based systems to more advanced models, such as anomaly-based detection, which aimed to detect threats by identifying deviations from a 'normal' baseline [9].

Signature-based models also became popular, where specific patterns or 'signatures' of known threats were used to detect and possibly prevent intrusions. However, while these models were highly effective against known threats, they often struggled against new, unidentified threats, leading to the exploration of heuristics and more adaptive systems [8].

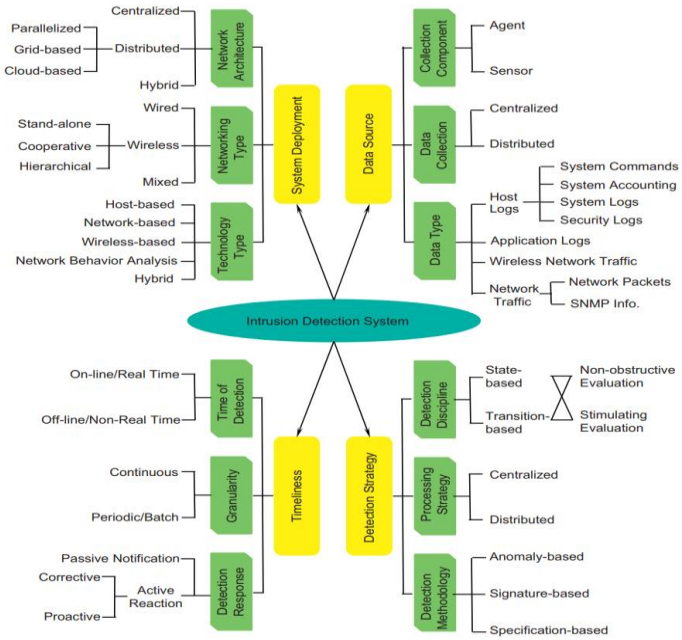

Figure 1: Overview of IDS Taxonomy [8]

2.1.5 Evolution of Standalone IDS to Meet Cybersecurity Needs

The initial IDS models were predominantly standalone, operating either on specific devices (host-based) or at certain network points (network-based). However, the complexity of cyber environments and the multifaceted nature of threats soon necessitated a more integrated approach.

The comparison between host-based IDS (HIDS) and network-based IDS (NIDS) revealed unique strengths and weaknesses inherent to each system [83]. While HIDS provided deep insights into system-specific anomalies by monitoring system calls and internal processes [86], NIDS offered a broad overview, monitoring traffic across the entire network and identifying potentially malicious patterns [79, 80].

Recognizing the value in both approaches, researchers began to advocate for hybrid models, which combined the strengths of HIDS and NIDS [84]. This evolution was emblematic of the broader shift in cybersecurity towards layered defence strategies. Rather than relying on a single point of detection or prevention, security experts started to see the value in multiple,

overlapping layers of security, ensuring that even if one layer were compromised, others would still be operational.

In tandem with these developments was the recognition that the domain of threats had expanded beyond traditional computing devices. The advent of the Internet of Things (IoT) introduced a plethora of new devices, each a potential point of vulnerability [14]. Modern IDS had to evolve to protect not just computers but a myriad of interconnected devices, each with its unique operating environment.

Another significant challenge emerged in the form of false alarms or false positives. High rates of false alarms can reduce user trust in a system, and lead to potential threats being overlooked due to alarm fatigue [13]. Addressing this challenge has been one of the focal points in recent IDS research.

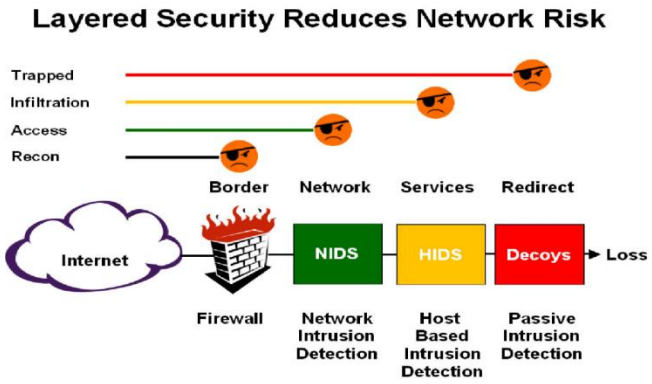

Figure 2:Layered Security approaches for reducing risk [13]

Lastly, the incorporation of advanced techniques, including game theory and machine learning, indicates the trajectory of IDS towards more autonomous and intelligent systems capable of adapting to emerging threats in real-time [81].

2.1.6 Fundamentals of Intrusion Detection Systems (IDS)

Intrusion Detection Systems (IDS) serve as a critical line of defence in today's digital landscape, actively monitoring for suspicious activities within a system or network and signalling alerts when potential threats are identified. The intricacies of IDS lie in their design, approach, and capabilities, as explored through a series of studies.

At its core, an IDS is a mechanism designed to detect unauthorized access or misuse of a system [2]. They can be broadly categorized into two types based on their monitoring approach:

- *Host-based IDS (HIDS):* Monitors individual host systems, like computers or servers, by analysing system calls, logs, and other system-level events [85][86].
- *Network-based IDS (NIDS):* Monitors network traffic, examining data packets in the network to identify suspicious patterns [79][80].

The two primary methodologies behind intrusion detection are:

- *Signature-based detection:* Identifies intrusions based on known patterns or 'signatures' of previous attacks [13].
- *Anomaly-based detection:* Identifies intrusions by detecting patterns of activity that deviate from established norms [9].

2.1.7 Core Functionalities and Characteristics of Standalone IDS

- *Real-time Monitoring and Analysis:* IDS continually surveys and analyses the data, ensuring that threats are detected in real-time or near-real-time. This function is vital to quickly addressing and neutralizing potential security threats [1].
- *Alarm and Alert Generation:* Upon detection of a suspicious or malicious activity, the IDS triggers an alert, notifying the system or network administrators. This alert can range from simple log entries to more advanced notifications like emails or messages [6].
- *Data Storage and Logging:* IDS systems maintain logs of monitored activities, which can later be analysed to understand attack patterns, improve system defences, or aid in digital forensic investigations [3].
- *Attack Recognition:* One of the most fundamental tasks of an IDS is to recognize patterns associated with known attacks. This is where the signature-based detection technique plays a pivotal role, identifying malicious activities based on pre-existing patterns [7].
- *Behavioural Analysis:* Anomaly-based IDS monitors the behaviour of the system or network and raises alarms when activities deviate from established norms or baseline behaviours. Such systems utilize machine learning and statistical models to differentiate between normal and anomalous behaviours [9].
- *Adaptability:* A robust IDS should be adaptable, adjusting to the evolving threat landscape. This means frequently updating the known signatures of malicious activities and continually learning new behavioural patterns for more accurate anomaly detection [8].
- *False Alarm Reduction:* Reducing false positives is essential for efficient IDS operation. Constant false alarms can desensitize administrators to threats, potentially overlooking actual security breaches. Techniques have been devised to minimize these false alarms, especially in signature-based IDS systems [13].
- *Scalability:* As networks and systems grow, the IDS should be able to scale accordingly, ensuring that the expanded infrastructure remains under surveillance without compromising performance [83][84].
- *Integration Capabilities:* While a standalone IDS provides comprehensive security coverage, the ability to integrate with other security systems such as intrusion prevention systems (IPS) can provide an additional layer of defence, thereby enhancing the overall security posture [79][10].
- *Archival and Recovery:* A comprehensive IDS not only detects and alerts but also archives these instances for future reference and potential recovery needs. This archival can prove invaluable for post-incident analysis and for devising improved security strategies [1].

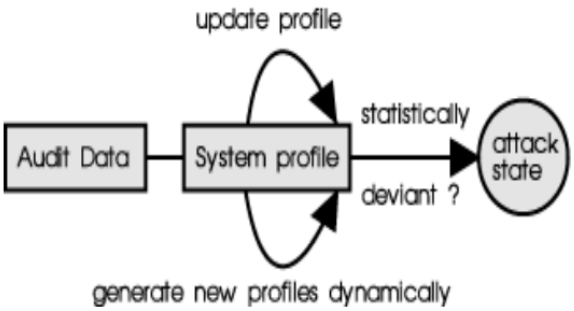

Figure 3: Typical anomaly detection system [10]

In conclusion, the importance of Intrusion Detection Systems in the contemporary cybersecurity landscape cannot be overstated. From their foundational concepts to their diverse characteristics and functionalities, IDSs stand as vigilant sentinels, safeguarding our digital assets and data from an ever-evolving array of cyber threats [6].

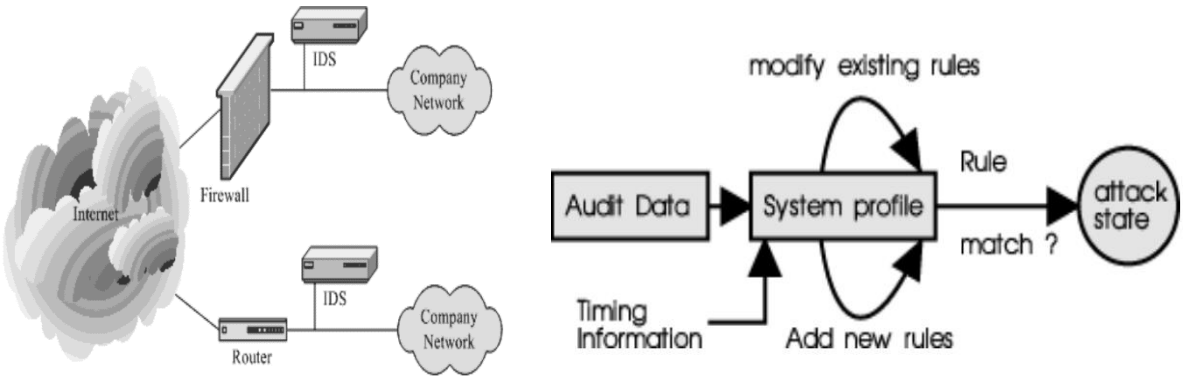

Figure 4: Typical locations of an IDS and Typical misuse detection system [10]

2.1.8 Classification and Techniques

Intrusion Detection Systems (IDS) serve as pivotal components in the cybersecurity arsenal. They are systems designed to monitor, detect, and signal any unauthorized, suspicious, or anomalous activity within a network or a host system. Based on an array of research articles, the classifications and techniques of IDS can be comprehensively analysed.

Host-based IDS (HIDS) and Network-based IDS (NIDS)

Host-based IDS (HIDS): This system predominantly operates on specific host systems such as computers or servers. It keeps tabs on system-level events, system calls, and logs to detect intrusive activities [85][86].

Advantages:

- Precise: HIDS can provide details about which files were accessed or modified during an intrusion [86].
- Independent of Network Topology: HIDS is unaffected by encrypted network traffic or switched network topologies [83].

Disadvantages:

- Resource Intensive: Monitoring at the system level can be resource-intensive [85].
- Limited Scope: It only detects attacks on individual hosts, not on the network as a whole [83].

Network-based IDS (NIDS): NIDS scrutinizes data packets traversing within a network, searching for any malign patterns or anomalies [79][80].

Advantages:

- Wide Coverage: NIDS can monitor an entire network, making them effective for early intrusion detection [79].
- Performance: Usually less resource-intensive compared to HIDS since they only examine packet headers [80].

Disadvantages:

- Encryption: Struggles with encrypted traffic, potentially missing intrusions in such data [80].
- Potential for Blind Spots: If not positioned correctly, some malicious traffic might evade detection [83].

Methodological Techniques in IDS

Signature-based detection: Recognizes malicious activities based on known attack patterns or signatures [13].

Advantages:

- Accuracy: Highly accurate when dealing with known attack vectors [5].
- Efficiency: Quick in detection as it matches patterns with known signatures [13].

Disadvantages:

- Limited to Known Threats: Cannot detect zero-day attacks or new malicious patterns [7].
- Maintenance Overhead: Regular updates are required to keep the signature database current [13].

Anomaly-based detection: Operates by benchmarking normal behaviour and flagging deviations or anomalies from this established baseline [9].

Advantage:

- Adaptive: Can identify previously unknown threats or zero-day attacks [9]. - Comprehensive: Monitors all activities, not limited to known signatures [11].

Disadvantages:

- False Positives: An increase in false alarms due to legitimate activities sometimes deviating from the norm [13].
- Requires Baseline: Initial profiling of "normal" activity is necessary, which can be time-consuming [11].

Heuristic-based detection: Utilizes algorithms to determine the likelihood of an action being malicious based on various attributes [8].

Advantages:

- Adaptive: Can evolve with changing threat landscape by adjusting heuristics [8].
- Proactive: Might identify novel threats before they become widespread [8].

Disadvantages:

- Complexity: More complex to develop and maintain [8].
- Potential for Errors: Depending on the quality of the heuristic, there could be false positives or negatives [8].

Application to Specialized Domains: The IoT Example

With the expanding Internet of Things (IoT) landscape, specialized IDS techniques are also emerging. Anomaly and signature-based IDS are being amalgamated to offer protection to IoT devices, with considerations for their unique operational and resource constraints [14].

Advantages:

- Comprehensive: Combines the strengths of both anomaly and signature-based approaches [14].
- Suitable for Dynamic Environments: Given the diverse and dynamic nature of IoT devices [14].

Disadvantages:

- Resource Constraints: IoT devices might lack the resources for complex IDS operations [14].
- Evolving Landscape: IoT represents a rapidly changing field, making it challenging to keep IDS methodologies up to date [14].

In summation, Intrusion Detection Systems, with their varied classifications and techniques, play a critical role in fortifying cyber defences. Each approach and classification have its set of merits and demerits. A judicious blend, tailored to specific environments and needs, often offers the most effective protection.

Fundamentals of IDS

Intrusion Detection Systems (IDS) are paramount in safeguarding the integrity, confidentiality, and availability of information systems. At its core, an IDS monitors network traffic or system behaviour to detect suspicious activities that could signify attacks, such as security threats or policy violations. According to Mohamed et al., an IDS serves as a fundamental primer in cyber defence by sounding alarms and potentially taking preventative action based on detected threats [2].

2.1.9 Anomaly-based IDS: A Deeper Dive

Anomaly-based Intrusion Detection Systems (AIDS) is a sophisticated method that seeks to identify any deviations from established normal behaviour. Instead of relying on known attack signatures, it models the "normal" pattern of system behaviour and then flags any deviations as potential intrusions. This methodology offers a proactive approach, as it can detect previously unknown threats or zero-day attacks [9].

Features of Anomaly-based IDS:

- Adaptive Learning: Unlike signature-based systems that need frequent updates for new signatures, anomaly-based systems learn and adapt to changing environments over time. They analyse the traffic or behaviour over a period and set a baseline [9].

- Proactivity: This type of IDS is known to catch novel attacks or zero-day vulnerabilities because it doesn't rely on predefined patterns. Instead, it identifies deviations from the 'norm'[11].
- Granularity: Anomaly-based IDS can potentially identify not only broad attacks but also fine-grained abnormal activities, down to the level of unusual system calls as mentioned by Liu et al. in their review on HIDS [86].

Benefits of Anomaly-based IDS:

- *Mitigation of Unknown Threats:* The most prominent advantage of anomaly-based IDS is its ability to detect new or modified threats that signature-based systems might miss [9].
- *Lower Maintenance:* Since anomaly-based systems don't rely on attack signatures, they don't require constant updates with new threat definitions. They learn and adapt automatically [11].
- *Comprehensive Analysis:* By examining system behaviour holistically, these systems provide a broad perspective on system health and security, as highlighted by Samrin and Vasumati [11].

Challenges of Anomaly-based IDS:

- False Positives: One significant challenge in anomaly-based systems is the potential for high false positive rates. Any deviation from the norm, even if benign, can trigger an alarm. This issue was notably discussed by Hubballi & Suryanarayanan, emphasizing techniques to reduce such false alarms in IDS [13].

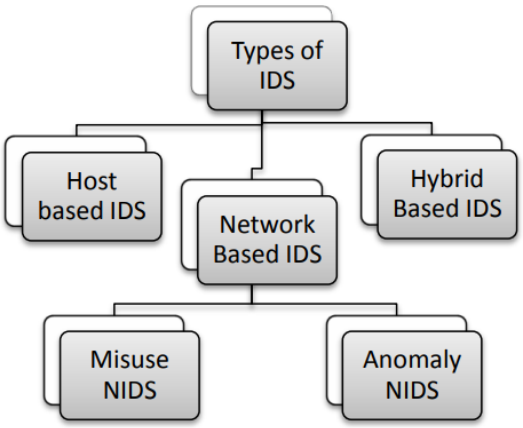

Figure 5: Classification of IDS [12]

- *Complex Implementation:* Building an accurate and effective model that captures all aspects of 'normal' behaviour can be intricate. Lalduhsaka et al. discuss the challenges in building a model for IDS, including defining the bounds of normality [12].

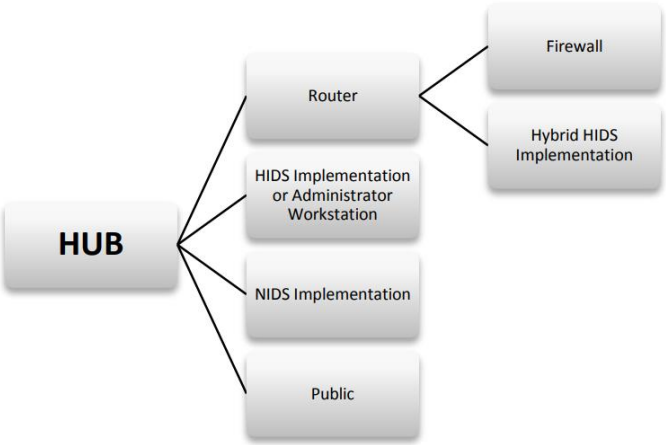

Figure 6: Types of IDS

- *High Resource Consumption:* Continual monitoring and analysis of behaviour against the baseline can be resource-intensive, requiring significant computational power and storage [12].
- *Adversarial Attacks:* Attackers might exploit the learning phase of the IDS, feeding it with malicious behaviour to be considered as 'normal', thus compromising the system [81].

In conclusion, while anomaly-based IDS offers promising features and benefits, especially in detecting novel threats, it is not without challenges. Balancing its capabilities with potential pitfalls requires a comprehensive understanding of its functionalities and inherent limitations.

- With the growth of the Internet of Things (IoT), the application of IDS, especially the anomaly-based approach, becomes even more critical. As pointed out by Otoum & Nayak, combining anomaly and signature-based methodologies might be the key to securing our increasingly interconnected world [14].

2.1.10 Design Considerations for Standalone IDS

Intrusion Detection Systems (IDS) are pivotal tools in the cyber defence arsenal. While there are many configurations and deployment strategies, the design of a standalone IDS is particularly crucial, given its independent operation. A well-crafted standalone IDS can significantly enhance system security, but its creation comes with a set of challenges and unique design elements, especially when juxtaposed against integrated systems.

Key Factors in Designing Effective Standalone IDS:

- *Determining System Type:* IDS can be broadly classified into Network-based (NIDS) and Host-based (HIDS). The choice between these is often predicated on the specific security requirements and the nature of the assets to be protected [83].
- *Detection Methodology:* Whether to use a signature-based method, which detects known patterns of attacks, or an anomaly-based method, which identifies deviations from typical behaviours, is a foundational design decision [9].
- *Scalability:* As systems grow, the IDS should have the capacity to handle an increasing amount of data without compromising performance [1].

- *Alert Management:* An efficient standalone IDS needs a well-designed alert system. Without the integration of other systems to validate alerts, the standalone IDS must prioritize alerts to minimize false positives [13].
- *Ease of Update:* Threat landscapes evolve; hence, the IDS must be designed with easy-to-update mechanisms, especially for signature-based systems that require updated databases of known attack patterns [8].
- *Customization:* Given the variance in network topologies and user behaviour, customization features can ensure that the IDS is tailored to the specific environment in which it operates [5].
- *Resource Efficiency:* A standalone IDS, especially a HIDS, should be designed to minimize its consumption of system resources, so it doesn't impede regular operations [85].

Challenges in Designing Standalone IDS:

- *False Positives and Negatives:* Striking a balance to minimize both false positives and false negatives is challenging, especially in anomaly-based systems [13].
- *Evolving Threat Landscape:* The continuous emergence of new threats means that the IDS must regularly evolve, posing challenges in keeping it updated [8].
- *Performance Overheads:* Especially for HIDS, there's a need to ensure that the IDS don't introduce significant performance degradation [86].
- *Bypass Techniques:* Attackers often employ techniques to bypass IDS, like fragmentation attacks. Designing an IDS to counter such techniques is challenging [79].
- *Adversarial Machine Learning:* In anomaly-based systems, attackers might poison the learning phase, causing malicious activities to be seen as normal [81].

Design Element	Standalone System	Integrated System
Self-sufficiency	Independent in detection, analysis, alert	Can rely on integrated components
Adaptability	Must adapt to changes in environment	Access broader dataset from components
Response Mechanism	Focuses on detection and alerting	Can initiate automated responses
Configuration & Deployment	Simple deployment process	Integration complexity and cross-checking
Maintenance & Updates	Independently updated	Updates may affect multiple components

Table 1: Standalone vs Integrated systems

2.1.11 Signature Based Detection

Using Signature Based Detection we search network traffic for a sequence of bytes that could represent an attack on the system. It is one of the easiest ways of detection as all it needs is a pattern for the IDS to identify. Moreover, pattern matching can be more efficient as very low amount of power is required by the system to identify a matching sequence. Vulnerabilities have a particular string sequence that can be fed to the Signature Based IDS which will then alert whenever a similar sequence occurs in the network [2].

However, the issue with this kind of IDS is that all the signatures must store in the database and any exploit that occurs outside its scope will never be detected. Zero-day attacks are software vulnerabilities that are new and are unknown to the software vendors, leaving the users defenceless meaning IDS cannot pick such an attack. Also, against attacks generated by humans or a worm, a Signature Based IDS is not very effective. To bypass such IDS methods, attackers may use advanced techniques such as encrypted channel, a No-Operation generator (NOP) which places sequences of no-operation instructions into a program's code, payload encoders [2]. As more variants of an attack emerges, more signatures have to be developed to keep the system protected which reduce the efficiency and performance of the system. The efficiency of the system depends on how quick a signature is developed by the developers before an attack has been performed.

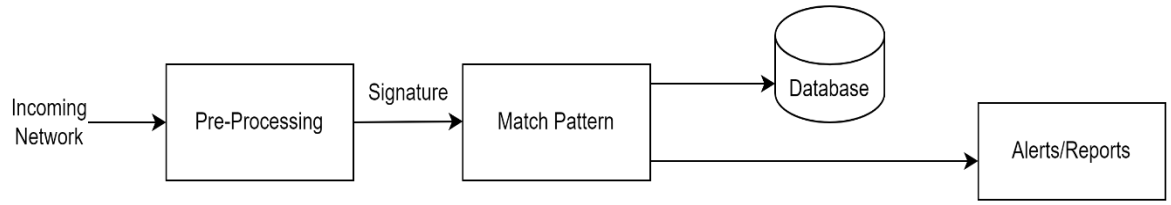

Figure 7: Signature Based Detection Architecture [4]

2.1.12 Anomaly Based Detection

Since Signature Based IDS has its disadvantages, an improved technique known as the Anomaly based IDS can be used. This technique uses machine learning algorithms and statistical analysis to understand the behavioural pattern of the network and alerts the user when the system identifies anything that deviates from the normal baseline. This system monitors network traffic, system behaviour, or user activities to find and identify abnormal patterns from the expected baseline. The baseline which is defined as normal for a system is prepared by the network administrators and is considered as a crucial phase [2].

IDS should be capable of understanding various protocols and understand its goals to reduce the number of false positives. With dynamic networks and large inputs rate of false positives could be high. To overcome this, we need a system that can adapt to any dynamic environment using self-learning techniques like Deep learning [87].

One major drawback in Anomaly Based IDS comes from how well a baseline is defined by the network administrators. The efficiency of this system depends how well protocols are defined along with custom rules making it a difficult job.

When used together as a single system, the combination of signature and anomaly methods can become overwhelmed and face restrictions. A hybrid model uses machine learning like Support Vector Machine (SVM), C5 decision tree, k-Nearest Neighbour eliminating all the disadvantages and provides optimum results. However, a hybrid model that uses deep learning can be effective in identifying intruders and protecting the system [7].

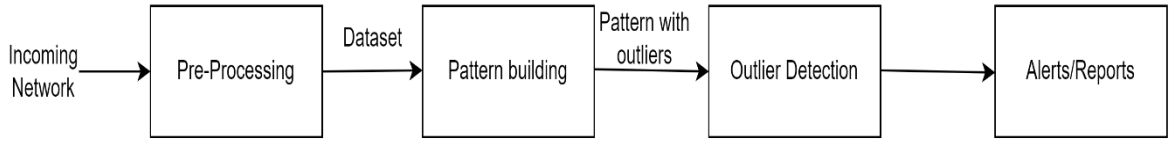

Figure 8: Anomaly Based Detection Architecture [4]

2.1.13 Intrusion Prevention System

IPS or Intrusion Prevention System is a more advanced version of IDS which is coupled with a firewall. As we know, IDS helps in identifying an intrusion within a network and alerting the user, an IPS goes one step ahead and prevents the attack from happening. To improve IPS, it has to run on stable and reliable platform, hardware should be supported to run IPS.

When compared with IDS, IPS has features like In-line and traffic isolation. An inline Intrusion Prevention System (IPS) is a security solution that operates at the network level and is placed directly in the data path to inspect and block malicious or unauthorized network traffic in real-time. Unlike intrusion detection systems (IDS) that only monitor and raise alerts, inline IPS actively takes action to prevent the detected threats from reaching their intended targets[3].

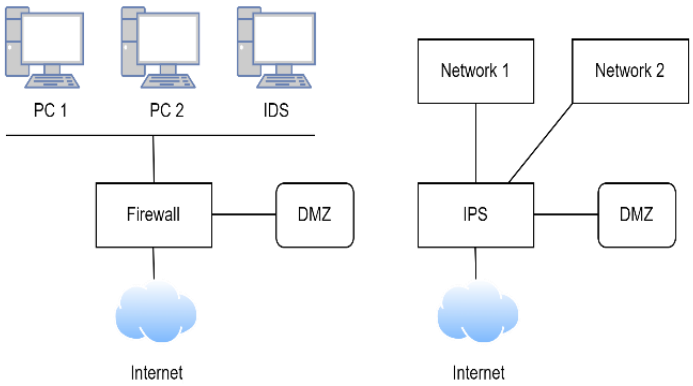

Figure 9: IDS and IPS system [3]

2.1.14 Heuristic Intrusion Detection and Prevention System

The Heuristic Intrusion Detection and Prevention System (HIDPS) is an intelligent security system designed to scan for malicious behaviour in programs attempting to access or operate within the system. Unlike traditional systems that rely on virus definitions, HIDPS utilizes heuristics and online scanning techniques to make intelligent decisions about the nature of a program (malicious or non-malicious). It operates as a standalone script, eliminating the need for frequent updates. By monitoring all active processes and network packets, the system minimizes the chances of undetected rootkits or Trojans [1].

2.1.15 Issues and Challenges with IDS

With increase on different variants of attacks and improvement in technology, it has become easier for attackers to find ways to get past defensive systems like IDS in a network. Moreover, the dataset used to train an Intrusion Detection model presents a significant challenge due to the sheer size of raw network traffic data, which can become incredibly large even when collected over a few days. Moreover, this data requires extraction steps to be compatible with the model [5].

To effectively train machine learning models for an Intrusion Detection System (IDS), a dataset containing both normal and malicious data is necessary. These models learn to recognize patterns in both types of data, allowing the system to identify malicious data and raise alarms when analysing new data in the network. However, results based on such datasets might not precisely reflect real-time situations. Gathering real-time data is problematic since it can grow rapidly in size, and there is no assurance of encountering attacks during the data collection process [5].

The focus is on enhancing the detection rate by employing an efficient algorithm that accurately identifies attacks from the dataset. Achieving a higher detection rate is crucial as it necessitates a precise analysis of packet features. There are two primary methods for detection: signature-based or anomaly-based. The former focuses on identifying known attacks, while the latter can detect both known and unknown attacks. However, given the substantial volume of packets in the IoT environment, a slow processing algorithm for attack detection can significantly hamper system performance. Hence, there is a pressing need to develop faster and more efficient methods to detect abnormal packets [7].

To make IDS ineffectual cyber-criminals use techniques called the evasion techniques to do undetected by the IDS. Techniques like Encryption, Fragmentation, Flooding and Obfuscation can be used.

Encryption: Encryption is a concept used for confidentiality which encodes a packet using encryption algorithm, making it difficult for a middleman to read the contents of a packet. IDS which cannot decrypt the data cannot match this encrypted data to its signature database which makes it ineffective [88].

Flooding: The attacker initiates the attack with the intention of overwhelming the detector, leading to a failure of the control mechanism. Consequently, when the detector malfunctions, all traffic is permitted without scrutiny. One prevalent technique for generating a flooding scenario involves spoofing the legitimate User Datagram Protocol (UDP) and Internet Control Message Protocol (ICMP). The flooding of traffic serves to camouflage the cybercriminal's abnormal activities [88].

Obfuscation: Obfuscation techniques can be employed to avoid detection, involving methods that make the attack message harder to comprehend. The term "obfuscation" refers to altering the program code in a manner that retains its functionality while reducing the possibility of being detected through static analysis or reverse engineering. This process aims to make the code obscure and less readable. By obfuscating malware, attackers can successfully evade existing Intrusion Detection Systems (IDS) in use today [88].

Detecting attacks concealed by evasion techniques poses a significant challenge for both Signature-based Intrusion Detection Systems (SIDS) and Anomaly-based Intrusion Detection Systems (AIDS). The effectiveness of evasion techniques depends on the IDS's capability to either recover the original attack signatures or generate new ones to accommodate the attack modifications. The resilience of IDS to different evasion techniques requires further exploration. For instance, SIDS using regular expressions can identify deviations resulting from basic mutations like altering space characters, but they remain ineffective against various encryption techniques [88].

2.2 IDS IN VARIOUS APPLICATIONS

2.2.1 Introduction to IDS (Intrusion Detection Systems) in Various Applications

Intrusion Detection Systems (IDS) are a crucial component of modern cybersecurity, designed to identify and respond to suspicious activities or potential threats within a computer network or system. IDS plays a vital role in safeguarding data, protecting sensitive information, and ensuring the integrity and availability of resources. These systems work by monitoring and analysing network traffic, system logs, and other data sources to detect any signs of unauthorized access or malicious activities.

There are various types of IDS, each tailored to specific environments and applications. Let's explore some common applications of IDS:

- *Network-based IDS (NIDS):* NIDS is deployed at strategic points within a network to monitor incoming and outgoing traffic. It analyses packets passing through these points and compares them against known patterns or signatures of known threats. When suspicious activity is detected, NIDS can generate alerts, logs, or even take proactive measures like blocking the traffic.
- *Host-based IDS (HIDS):* HIDS operates directly on individual hosts or endpoints, continuously monitoring system logs, file integrity, user activities, and other host-specific data. By comparing this data to established baseline behaviours or predefined rules, HIDS can detect anomalies or signs of potential intrusions on a specific host.
- *Cloud-based IDS:* As cloud computing gains popularity, cloud-based IDS is becoming more important. It offers network and application monitoring tailored for cloud environments, providing security insights and protection for cloud-hosted assets.
- *Application-level IDS:* Application-level IDS focuses on monitoring the application layer of a network to detect and prevent attacks targeting specific applications, such as web applications or databases. This helps protect against application-layer vulnerabilities and unauthorized access attempts.
- *Wireless IDS (WIDS):* WIDS is designed for wireless networks, including Wi-Fi and Bluetooth. It monitors wireless traffic for signs of unauthorized access, rogue devices, or other wireless-specific security issues.
- *Signature-based IDS:* Signature-based IDS relies on predefined signatures or patterns of known threats. When it detects a match between the observed data and a signature, it raises an alert or takes appropriate action.
- *Anomaly-based IDS:* Anomaly-based IDS focuses on detecting deviations from normal network or system behaviour. It establishes a baseline of "normal" behaviour and raises

alerts when it encounters activities that deviate significantly from the norm, potentially indicating an intrusion.

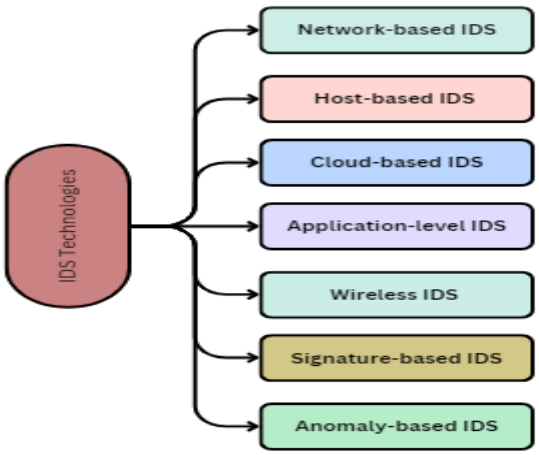

Figure 10: Types of IDS Technologies

2.2.2 IDS in Context

Intrusion Detection Systems (IDS) serve as integral components of larger cybersecurity systems, enhancing threat detection and response capabilities. Whether integrated with network monitoring tools, control systems, IoT and cloud environments, web and network applications, or industrial control systems (ICS), IDS plays a vital role in safeguarding critical assets and data. In network monitoring systems, IDS continuously analyzes network traffic for suspicious activities, alerting administrators about potential security breaches. For critical infrastructure and ICS, IDS monitors networks for anomalies and unauthorized access, ensuring the integrity of operations. In IoT and cloud environments, IDS detects abnormal device behavior and safeguards cloud resources from unauthorized access. In the realm of web and network applications, IDS mitigates web-based attacks like SQL injection and DDoS, while in MANETs, IDS secures mobile networks from potential disruptions. Host-based IDS provides added security for individual systems, detecting intrusions and insider threats. In combination, IDS reinforces overall cybersecurity by detecting and responding to emerging threats promptly.

We can classify the papers we have studied based on their application into 5 distinct sections.

2.2.4 Intrusion Detection in IoT and Cloud Environments

1. Hierarchical Federated Learning for Collaborative IDS in IoT Applications [15]
2. Lightweight Intrusion Detection System(L-IDS) for the Internet of Things [17]
3. A collaborative framework for intrusion detection (C-NIDS) in Cloud computing [28]
4. Intrusion Detection System in Cloud Computing: Challenges and opportunities [29]
5. Machine learning based intrusion detection system for software-defined networks [32]
6. A collaborative intrusion detection and Prevention System in Cloud Computing [35]

Intrusion Detection Systems (IDS) play a vital role in safeguarding the security of IoT and cloud environments. As the cyber threat landscape continues to evolve, the integration of IDS into these ecosystems becomes increasingly critical. The analyzed research papers propose innovative approaches to enhance the effectiveness of IDS in IoT and cloud applications.

Addressing the security concerns within IoT, a hierarchical federated learning strategy is introduced for collaborative IDS [15]. This IDS aims to protect interconnected IoT devices from unauthorized access and potential threats. The challenges addressed involve resource constraints of IoT devices and ensuring secure communication between devices and the central IDS server. By leveraging hierarchical federated learning, the IDS learns from local device data while preserving privacy and minimizing communication overhead [15]. The benefits of this approach include decentralized learning capabilities, improved scalability, and reduced reliance on a central server, resulting in enhanced detection of IoT-specific attacks and anomalies.

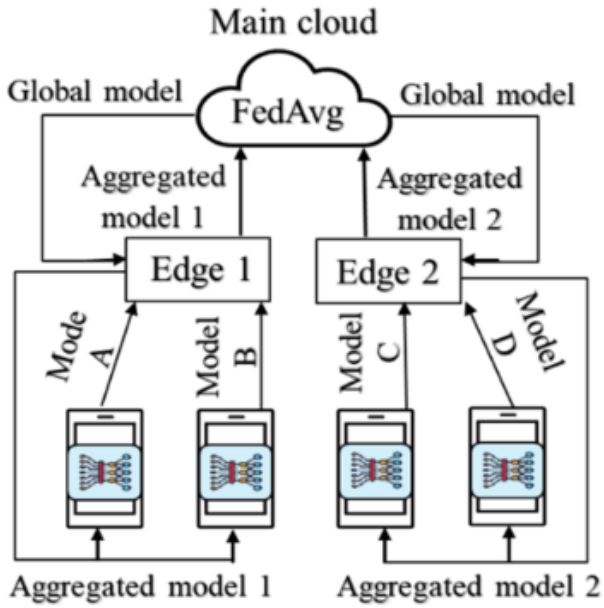

Figure 11:Hierarchical Federated Learning [15]

In this study we focus on a lightweight IDS specifically tailored for IoT environments. This IDS efficiently monitors and analyses IoT device behaviour, detecting abnormal activities that could indicate security breaches [17]. Challenges revolve around achieving accurate detection without overwhelming device resources. However, the lightweight design ensures efficiency and low overhead, making it suitable for resource constrained IoT devices [17]. The effectiveness of this IDS is demonstrated through its ability to detect and mitigate IoT-specific threats while minimizing the impact on device performance.

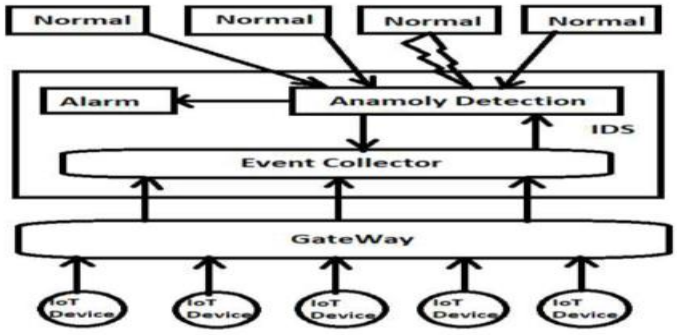

Figure 12: LIDS Architecture [17]

This research paper proposes a collaborative framework for IDS in cloud computing environments. This IDS aims to protect cloud resources and data from unauthorized access and potential breaches. Challenges involve handling vast amounts of data and distinguishing legitimate cloud activities from potential attacks. The collaborative approach allows multiple IDS instances to work together, improving threat detection accuracy and response time. The benefits lie in its ability to detect sophisticated attacks and the collaborative nature that enhances overall cloud security. The effectiveness of this collaborative IDS is evident in its proactive defence against various cloud-based threats.[28]

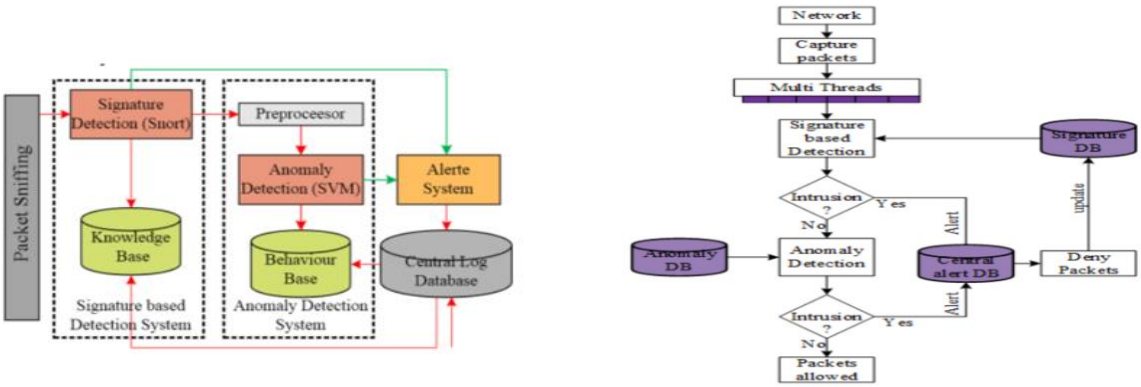

Figure 13: Architecture framework and Workflow for C-NIDS [28]

Within the context of IoT, the fourth paper explores the challenges and opportunities of integrating IDS into cloud computing environments. Challenges include ensuring customer data privacy, scalability of the IDS, and handling false positives to avoid disrupting legitimate cloud activities. The benefits of IDS in cloud environments lie in providing enhanced security and compliance with regulatory requirements. The effectiveness of IDS in cloud computing depends on its adaptability to dynamic cloud environments, swift detection, and response to potential threats, while minimizing false positives to maintain operational efficiency. The tables below demonstrate the various challenges faced by certain types of IDS techniques in cloud computing. [29]

Highlighting the potential of machine learning-based IDS for Software-Defined Networks (SDN). This IDS is integrated into SDN controllers, continuously monitoring network traffic, and identifying suspicious activities. Challenges include ensuring the accuracy of machine learning models and efficiently handling large-scale SDN environments. The benefits lie in its ability to adapt to evolving threats and anomalous network behaviours, making it an effective tool to detect and mitigate attacks in dynamic SDN environments. [32]

The sixth paper introduces a collaborative IDS and Prevention System for cloud computing environments. This IDS continuously monitors network traffic, identifying potential threats, while the prevention system responds with security measures. Challenges involve coordinating response actions among different cloud nodes and minimizing false negatives to ensure comprehensive threat detection. The benefits of this collaborative system lie in its proactive defence, swift response to emerging threats, and distributed nature, enhancing overall cloud security. The effectiveness of this system is evident in its ability to prevent and mitigate potential security incidents in cloud environments. The figure below shows the architecture for the proposed security framework that integrates Hybrid-NIDS to cloud. It has higher detection rate, higher accuracy, and low false alerts.[35]

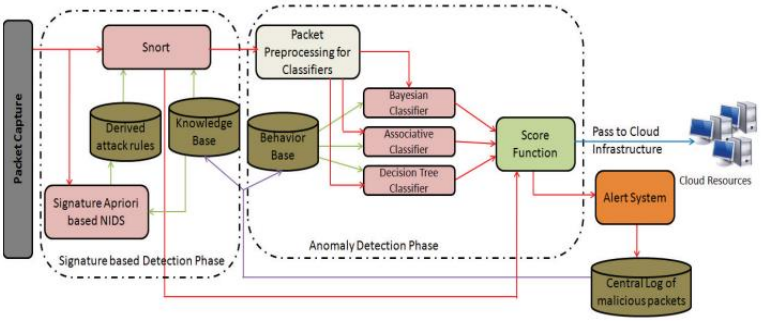

Figure 14: Architecture of proposed H-NIDS [35]

Features / References	Detection Technique	IDS Type	Positioning	Detection Time	Data Source	Attacks covered	Limitations/ Challenges
CIDS for Cloud Computing Networks, 2010 [2]	Signature based	Distributed	Each Cloud region	Real time	Network traffic, signatures of known attacks	Protects system from single point of failure, DoS and DDoS	Can't detect unknown attacks, High computational overhead
Securing cloud from DDOS Attacks using IDS in VMs, 2010 [18]		Network based	Virtual Switch	Real time	Network packets, signatures of known intrusions	Secures VMs from DDoS attacks	Detects only known attacks
Integrating a NIDS into an Open Source Cloud Computing Environment, 2010 [17]		Network based	At each node	Real time	Network traffic, normal usage of resources like CPU	Only Known attacks particularly SIP flooding	can't detect unknown attacks,
Autonomic Agent-Based Self-Managed IDPS, 2010 [20]	Anomaly based	Host based	N/A	Real time	Network traffic, System activities (system calls etc.)	Can detect all types of attacks in real-time	Implementation details are not given
Multi-level IDS and Log Management in CC, 2011 [21]		Host based	At each guest OS	Real time	User behaviors, known attack patterns	Can detect both known and unknown attacks at a fast rate	Consumes more resources for high level users
Distributed Intrusion Detection in Clouds using MAs, 2009 [22]		Distributed	At each VM	Real time	Audit data, known intrusion patterns, system logs	Can detect both known and unknown attacks	There is a limit on the number of VMs to be visited
Collabra: Xen Hypervisor based Collaborative IDS, 2011 [29]		VMM based, Distributed	At each VMM	Real time	Audit data, anomaly database	Can detect hyper-call based attacks on VMM and host OS	Cannot detect other types of attacks
IDS for Cloud Computing, 2012 [19]	Hybrid	Distributed	At the processing server	Real time	Audit data, user profiles, signatures of known intrusions	Can help CSP to improve its quality of service, detects unknown attacks	The proposed idea is theoretical, No implementation provided
Bayesian Classifier and Snort based NIDS in Cloud Computing, 2012 [5]		Network based	At the processing servers	Real time	Network packets, known attack signatures, prior events	Detects all types of attacks	Complexity increased due to integration of both, signatures and anomalies
IDS in Cloud Computing Environment, 2011 [28]		Host based and Network based	At each node	Real time	Logs of user activities, signatures of known attacks	Can detect all known attacks, may detect unknown attacks using ANN	Experimental results are not given
GCCIDS, 2010 [27]		Host based	At each node	Real time	Audit data, user profiles	Known attacks, Unknown attacks using ANN	Accurate detection requires more training time, there is a limit on number of rules.

Figure 15: Challenges faced by IDS [29]

The integration of IDS into IoT and cloud environments is vital for cybersecurity enhancement. These studies introduce inventive solutions to challenges like resource limits, privacy worries, and data scalability. IDS benefits from its unique threat detection and response capabilities in IoT and cloud contexts, safeguarding assets and data. As technology advances, IDS remains crucial for digital security, and ongoing improvements and collaborative strategies bolster its role in proactive defence against evolving cyber threats.

2.2.5 Intrusion Detection for Web and Network Applications

1. OWADIS: Rapid Discovery of OWASP10 Vulnerability based on Hybrid IDS [16]
2. A Signature-Based Intrusion Detection System for Web Applications based on Genetic Algorithm [19]
3. Specification-based intrusion detection for unmanned aircraft systems [21]
4. Research Trends in Network-Based Intrusion Detection Systems: A Review [27]
5. Integrating intrusion detection and network management [31]
6. The network management design integrated with the intrusion detection system [34]

Intrusion Detection Systems (IDS) are critical components in safeguarding web and network applications against cyber threats. The analysed research papers propose innovative approaches to enhance IDS integration, addressing challenges, and reaping benefits for improved effectiveness.

Introducing OWADIS, a Hybrid IDS, the study focuses on rapid discovery of OWASP10 vulnerabilities in web applications. This IDS employs a blend of signature-based and anomaly-based detection methods. Challenges encompass precise identification of OWASP10 vulnerabilities and managing the substantial web traffic load. The advantages manifest in rapid vulnerability detection, enabling timely mitigation, and fortifying web applications against prevalent threats [16].

Highlighting a Signature-Based IDS tailored for web applications through a Genetic Algorithm, the study underscores attack detection using predefined attack signatures while optimizing the database. Challenges encompass the need for constant signature database updates to counter emerging threats. The merits are seen in the effective identification of established web application attacks and the system's flexibility to evolving attack patterns [19].

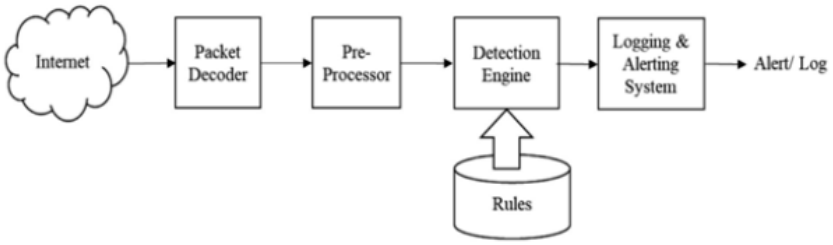

Figure 16: Signature-based Intrusion Detection System [19]

Centred on Specification-Based IDS for unmanned aircraft systems. Challenges revolve around ensuring accurate specification models and minimizing false positives to prevent disrupting legitimate aircraft operations. The benefits lie in its proactive defence, identifying deviations from specified behaviours, and promptly responding to potential threats, safeguarding unmanned aircraft systems [21].

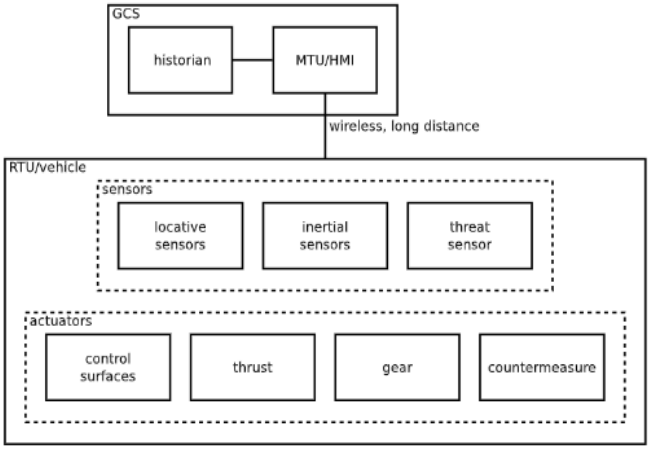

Figure 17: Unmanned Aircraft Systems [21]

This paper presents a Review of Network-Based IDS (NIDS) research trends. Challenges include handling large-scale network traffic and efficiently analysing network packets. The benefits of NIDS include real-time monitoring, early threat detection, and the ability to detect network-specific attacks and anomalous behaviours. The graph below states the trends followed by number of research articles on N-IDS since 1972 to 2020. [27]

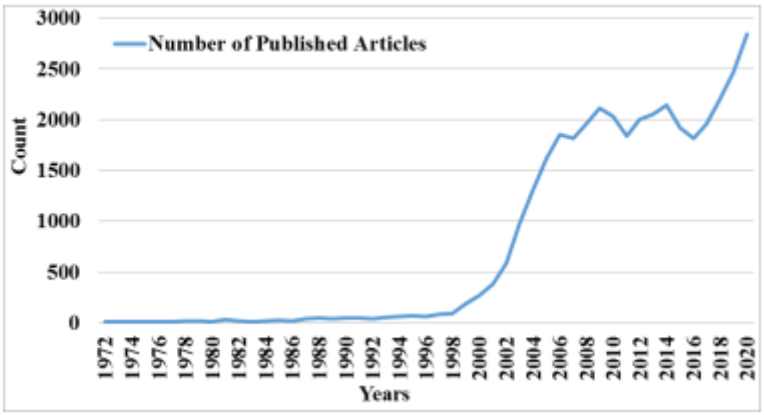

Figure 18: Count of Research publications on N-IDS [27]

Highlighting the integration of IDS with network management systems. Challenges involve coordinating network management actions and IDS responses to potential threats. The benefits lie in enhanced network security, seamless management, and quick mitigation of security incidents. This paper provides a detailed knowledge of the architecture of NIDS as well as the positions of NIDS in Cloud. [31]

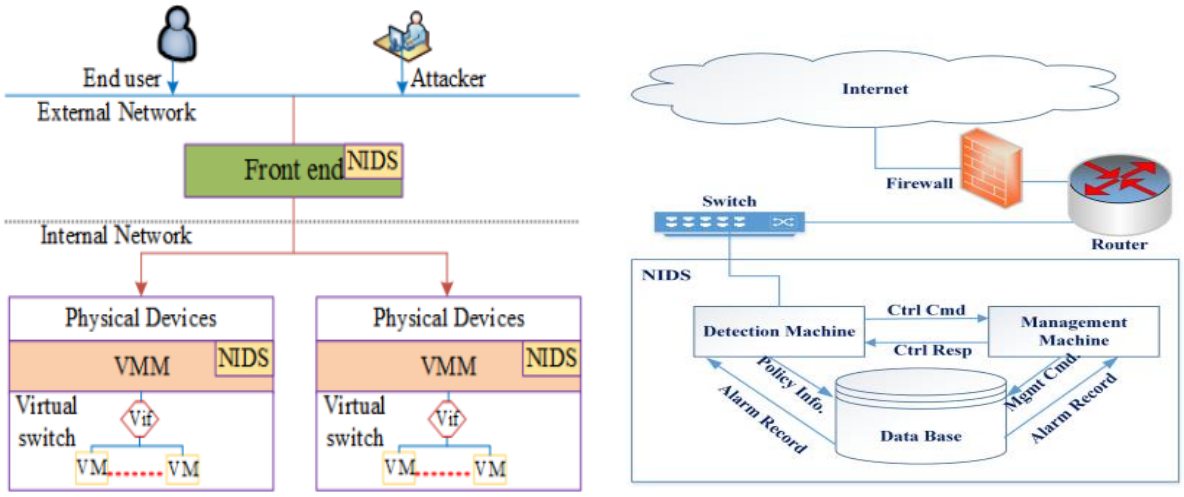

Figure 19: Components of Network-based IDS and Positions of NIDS in Cloud.[31]

The sixth paper discusses the integration of IDS with network management systems for comprehensive security. Challenges include effectively combining network management functionalities with IDS capabilities. The benefits include centralized monitoring, rapid threat detection, and coordinated incident response, ensuring robust network security. This paper provides a brief explanation of some popular NIDS that are used commercially and their corresponding advantages.

S. No.	NIDS	Manufacturer	Approaches Used	Advantage
1	Snort: Created by Martin Roesch, 1998 [26]	Cisco Systems, Sourcefire https://www.snort.org/	Signature-based, network intrusion detection, pattern matching Aho-Corasick algorithm [27].	Free open-source, real-time alerting, and packet logging
2	OSSEC: Open-Source HIDS Security Daniel B. Cid owned the copyrights of the OSSEC project, 2008.	AlienVault® OSSIM™, in 2008 Currently maintained by Atomicorp https://atomicorp.com/about-ossec/	Correlate and analyze logs, log-based intrusion detection	File Integrity Monitoring (FIM), log monitoring, rootkit detection, auditing, export to SIEMs, active response, process monitoring, time-based alerting, and log analysis
3	OSSIM: Open-Source Security Information and Event Management (SIEM)	AlienVault® OSSIM™, in 2008 Currently, AT &T Cybersecurity in 2019	Log processing, correlation directives (rules), behavioral monitoring, SIEM event correlation	Lacks support for Cloud-based servers and applications Reports are heavy and detailed, and tedious to parse through
4	Suricata: Free and open-source, a real-time intrusion detection system	Owned and supported by the Open Information Security Foundation (OISF) www.openinfosecfoundation.org	Signature-based intrusion detection, process multithreading to improve processing speed [28]	Suricata can handle larger volumes of traffic as compared to Snort
5	Bro: An open-source software framework that detect behavioral abnormalities on a network	Initially written by Vern Paxson Later in 2018, Paxson and the project's leadership team gave a new name to this project Zeek for developing the IDS. Like Suricata or Snort, it is also rules-based IDS. https://bricata.com/blog/what-is-bro-ids/	Script interpretation.	Transforms network traffic data into higher-level events. Offers a script interpreter
6	Fragroute/ Fragrouter: A network intrusion detection evasion toolkit	D. Son https://monkey.org/~dugsong/fragroute/	When Fragroute initialize, it deletes the route to the target Intercepts network traffic and modifies the packets before forwarding	Probe packets can be fragmented easily with Fragrout ICMP echo request messages are used by fragtest
7	BASE: Basic Analysis and Security Engine (BASE) offers a web-based front end for examining the alerts produced by Snort.	https://sourceforge.net/projects/secureideas/	It offers a web front-end to query and analysis the alerts produced by a SNORT IDS.	User authentication and role-based system Search interface and Query-builder for identical alerts matching from the alert meta information Packet viewer (decoder)
8	Sguil: Built by a group named network security analysts	https://github.com/bammv/sguil/releases/tag/v0.9.0	Event-driven analysis Network Security Monitoring	Captures raw packet, session data, and Real-time events Compatible on the operating system that supports TCL/TK Receive alerts from OSSEC, Zeek, Suricata, Snort, and other data sources.

Figure 20: Popular Commercially used NIDS [34]

In conclusion, the integration of IDS into web and network applications offers significant advantages in identifying and mitigating cyber threats. Challenges vary from accurately

identifying vulnerabilities and maintaining updated signature databases to handling vast amounts of network traffic. The benefits of IDS include real-time monitoring, proactive defence, efficient threat detection, and swift incident response. The effectiveness of IDS is evident in its ability to detect known and unknown threats, safeguarding critical web and network assets. As the cyber threat landscape evolves, continuous research and development in IDS techniques will further strengthen their effectiveness in securing web and network applications against emerging cyber threats.

2.2.6 Intrusion Detection in Industrial Control Systems (ICS)

1. A modern and sophisticated host-based intrusion detection dataset [38]
2. Context-aware intrusion detection in automotive control systems [39]
3. A survey of intrusion detection on industrial control systems [40]
4. iFinger: Intrusion detection in industrial control systems via register-based fingerprinting [41]
5. On cyber-attacks and signature-based intrusion detection for Modbus-based industrial control systems [43]
6. SCADA-specific intrusion detection/prevention systems: a survey and taxonomy [44]
7. Sequence-aware intrusion detection in industrial control systems [45]

Integration into Industrial Control Systems (ICS): The reviewed papers focus on Intrusion Detection Systems (IDS) integrated into ICS, which are used to monitor and protect critical infrastructure and industrial processes from cyber threats.

Challenges:

- Complexity: ICS environments are complex, with numerous interconnected devices and protocols, making it challenging to detect and respond to intrusions effectively.
- Legacy Systems: Many ICS components and devices may be legacy systems with limited security features, making them vulnerable to attacks.
- Real-Time Requirements: Intrusion detection in ICS requires real-time capabilities to prevent disruptions to critical processes.
- Anomalous Behaviour Detection: Distinguishing normal behaviour from potentially malicious activities in ICS can be challenging due to the diverse nature of industrial operations.

Benefits:

- Enhanced Security: IDS provides an additional layer of security to detect and prevent unauthorized access and potential attacks on critical infrastructure.
- Proactive Defense: IDS helps in identifying potential threats early, allowing for timely response and mitigation measures to prevent damage.
- Compliance: Integration of IDS assists organizations in meeting regulatory and compliance requirements, ensuring a secure and safe industrial environment.

Effectiveness:

- Improved Threat Detection: IDS specialized for ICS can effectively detect and alert on ICS-specific threats, providing more accurate threat detection.
- Real-Time Response: IDS with real-time capabilities allows for swift response to potential security incidents, reducing the impact of attacks on critical processes.

- Anomaly Detection: Some IDS systems leverage anomaly detection techniques to identify unusual behaviour in ICS networks, providing a proactive defense against emerging threats.

Examples from the Papers:

- The "A modern and sophisticated host-based intrusion detection dataset" [38] presents a dataset designed for evaluating host-based IDS in ICS, offering a valuable resource for testing and improving IDS performance in ICS environments.
- The paper on "Context-aware intrusion detection in automotive control systems" [39] focuses on IDS specialized for automotive ICS, considering the unique requirements and challenges of the automotive industry.
- "A survey of intrusion detection on industrial control systems" [40] provides an overview of existing IDS approaches in ICS, highlighting their strengths and weaknesses.
- "iFinger: Intrusion detection in industrial control systems via register-based fingerprinting" [41] introduces a register-based fingerprinting approach for IDS, aiming to detect anomalies in ICS communications.
- "On cyber-attacks and signature-based intrusion detection for Modbus-based industrial control systems" [43] discusses signature-based IDS for Modbus-based ICS, targeting specific communication protocols used in the industrial domain.
- The paper on "SCADA-specific intrusion detection/prevention systems: a survey and taxonomy" [44] offers a comprehensive review of SCADA-specific IDS and prevention systems, highlighting their applications and effectiveness.
- "Sequence-aware intrusion detection in industrial control systems" [45] proposes a sequence-aware approach for IDS in ICS to detect attacks based on the sequence of commands and operations, improving the accuracy of threat detection.

In summary, IDS plays a crucial role in securing Industrial Control Systems (ICS) against cyber threats. The challenges lie in the complexity of ICS environments, legacy systems, and the need for real-time detection. However, the integration of IDS brings enhanced security, proactive defence, and compliance adherence, resulting in improved threat detection and timely response to potential incidents. The reviewed papers offer valuable insights into specialized IDS approaches for ICS, contributing to the development of robust cybersecurity measures in critical infrastructure and industrial operations.

2.2.7 Intrusion Detection for Mobile Adhoc Networks (MANETs)

1. Multi-Layer Integrated Anomaly Intrusion Detection System for Mobile Adhoc Networks [33]

Mobile Adhoc Networks (MANETs) are dynamic, self-configuring networks without a fixed infrastructure, making them susceptible to various security threats. Securing MANETs against intrusions presents unique challenges, such as limited resources and the absence of centralized authority. To address these issues, the paper titled "Multi-Layer Integrated Anomaly Intrusion Detection System for Mobile Adhoc Networks" proposes a comprehensive approach - the Multi-Layer Integrated Anomaly Intrusion Detection System (MLI-IDS).

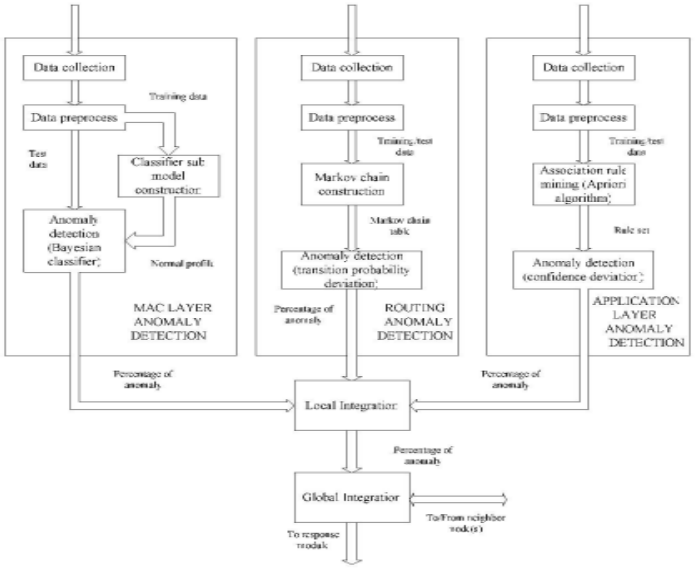

Figure 21: Proposed System Architecture [33]

The MLI-IDS is designed with multiple layers of defence, each focusing on specific aspects of MANET security. The first layer involves traffic analysis, examining packet headers, source-destination relationships, and routing information to identify any malicious packets or patterns. The second layer employs behaviour analysis, establishing baseline behaviour for each node to detect deviations such as data flooding, route poisoning, or packet dropping, indicative of potential intrusions. The third layer utilizes anomaly detection techniques to identify previously unknown and emerging threats. By continuously learning from network behaviour, the IDS can adapt to new attack patterns and detect zero-day attacks, enhancing its effectiveness against evolving threats.

A notable strength of the proposed IDS lies in its resource efficiency. Considering the resource constraints of MANETs, the MLI-IDS operates with minimal overhead, optimizing resource utilization to ensure it does not burden the network's performance.

The paper provides a comprehensive evaluation of the MLI-IDS through simulation experiments and performance metrics. The results demonstrate the system's effectiveness in detecting various types of intrusions while maintaining low false positive rates and minimal impact on network performance.

In conclusion, the "Multi-Layer Integrated Anomaly Intrusion Detection System for Mobile Adhoc Networks" presents a practical and adaptive solution for securing MANETs against cyber threats. By integrating multiple layers of defence, including traffic analysis, behaviour analysis, and anomaly detection, the proposed MLI-IDS offers a robust approach to detect and mitigate intrusions in the dynamic environment of MANETs. Its resource efficiency and ability to identify both known and unknown threats, including zero-day attacks, make it a promising tool for enhancing the security and resilience of mobile adhoc networks.

2.2.8 Miscellaneous Intrusion Detection Studies

1. Host-Based Intrusion Detection System with System Calls: Review and Future Trends [22]
2. A Comparative Study of AI-Based Intrusion Detection Techniques in Critical Infrastructures Towards a Novel Intrusion Detection Architecture using Artificial Intelligence [23]

3. SIP Based Intrusion Detection System for VoIP based Applications. In Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies [18].
4. A real-time and robust intrusion detection system with commodity wi-fi [20].
5. Towards a Novel Intrusion Detection Architecture using Artificial Intelligence. [24]
6. Intrusion detection system in cloud environment: Literature survey & future research directions. [25]
7. Integrating Intrusion Detection System with Network monitoring. [26]
8. A novel hybrid-network intrusion detection system (H-NIDS) in cloud computing. [30]
9. A Hybrid Intrusion Detection System Based on Feature Selection and Weighted Stacking Classifier. [36]
10. A research paper on hybrid intrusion detection system. [37]
11. Analysis of host-based and network-based intrusion detection system. [42]
12. A closer look at intrusion detection system for web applications.[46]

This section talks about miscellaneous IDS as these studies couldn't be classified under a specific section of application, all the above-mentioned papers propose or talk about a novel application of IDS which can be beneficial.

Host-Based Intrusion Detection System with System Calls: Review and Future Trends [22]

- *Integration:* The paper focuses on host-based intrusion detection systems that monitor system calls to detect unauthorized activities and potential intrusions on individual hosts or endpoints.
- *Challenges:* Challenges in host-based IDS include the overhead of monitoring system calls, ensuring real-time detection, and handling large volumes of system call data efficiently.
- *Benefits:* Host-based IDS provides detailed insights into the behavior of individual hosts, allowing for targeted detection of intrusions specific to each host's environment.
- *Effectiveness:* The paper reviews existing host-based IDS approaches, highlighting their strengths and limitations. Future trends discussed aim to improve detection accuracy and reduce false positives by leveraging advanced machine learning techniques.

A Comparative Study of AI-Based Intrusion Detection Techniques in Critical Infrastructures and Towards a Novel Intrusion Detection Architecture using Artificial Intelligence [23]

- *Integration:* Both papers explore AI-based intrusion detection techniques for critical infrastructures. They propose novel architectures leveraging artificial intelligence to enhance intrusion detection capabilities.
- *Challenges:* Challenges involve developing efficient AI models, training data availability, and ensuring robustness against adversarial attacks.
- *Benefits:* AI-based IDS can adapt to evolving threats, identify unknown attack patterns, and reduce false positives through sophisticated learning algorithms.
- *Effectiveness:* The papers present comparative studies, evaluating the performance of various AI models and their potential for improving intrusion detection in critical infrastructures.

SIP Based Intrusion Detection System for VoIP based Applications [18]

- *Integration:* The paper presents a SIP-based intrusion detection system specifically tailored for Voice over Internet Protocol (VoIP) applications.
- *Challenges:* Challenges include handling real-time VoIP traffic and accurately distinguishing between legitimate and malicious SIP messages.
- *Benefits:* A SIP-based IDS enables early detection of VoIP-specific threats, such as call hijacking and service abuse.
- *Effectiveness:* The paper evaluates the IDS's performance through experiments and demonstrates its capability to detect VoIP-based attacks, ensuring the integrity and availability of VoIP services.

A real-time and robust intrusion detection system with commodity Wi-Fi [20]

- *Integration:* The paper proposes a real-time intrusion detection system utilizing commodity Wi-Fi devices for wireless network security.
- *Challenges:* Challenges involve achieving real-time processing of Wi-Fi data streams and differentiating between benign and malicious Wi-Fi activities accurately.
- *Benefits:* Commodity Wi-Fi-based IDS offers cost-effectiveness and widespread deployment potential in various environments.
- *Effectiveness:* The paper validates the IDS's real-time detection capabilities through experiments, showing its potential in protecting wireless networks from intrusions and unauthorized access.

Intrusion Detection System in Cloud Environment: Literature Survey & Future Research Directions [25]

- *Integration:* The paper explores IDS in cloud environments, emphasizing the need for effective intrusion detection to safeguard cloud resources and data.
- *Challenges:* Challenges include handling vast amounts of data in the cloud, identifying novel cloud-based attacks, and ensuring IDS scalability.
- *Benefits:* Cloud IDS enhances the security posture of cloud infrastructures, providing proactive defence against emerging cloud-specific threats.
- *Effectiveness:* The paper presents a literature survey on cloud IDS techniques, offering insights into existing approaches and future research directions to strengthen cloud security.

Integrating Intrusion Detection System with Network Monitoring [26]

- *Integration:* The paper discusses integrating IDS with network monitoring to improve threat detection and response capabilities.
- *Challenges:* Challenges involve managing the integration of IDS and network monitoring tools, as well as minimizing false positives from network events.
- *Benefits:* Integration enhances the overall network security posture, providing real-time detection and automated response to potential intrusions.
- *Effectiveness:* The paper highlights the advantages of combining both systems, providing a more comprehensive view of network activity and enhancing intrusion detection effectiveness.

A Novel Hybrid-Network Intrusion Detection System (H-NIDS) in Cloud Computing [30]

- *Integration:* The paper introduces a novel H-NIDS that combines different intrusion detection techniques specifically for cloud computing environments.
- *Challenges:* Challenges include coordinating response actions among different cloud nodes and optimizing the hybrid IDS's performance.
- *Benefits:* H-NIDS improves threat detection accuracy in the cloud, ensuring comprehensive security coverage for diverse cloud-based attacks.
- *Effectiveness:* The paper presents the design and evaluation of H-NIDS, demonstrating its effectiveness in preventing and mitigating security incidents in cloud environments.

A Hybrid Intrusion Detection System Based on Feature Selection and Weighted Stacking Classifier [36]

- *Integration:* The paper proposes a hybrid intrusion detection system that combines feature selection and a weighted stacking classifier to improve detection accuracy.
- *Challenges:* Challenges involve selecting optimal features and ensuring the classifier's adaptability to diverse attack patterns.
- *Benefits:* The hybrid IDS achieves higher detection rates by leveraging the strengths of both feature selection and stacking classifier techniques.
- *Effectiveness:* The paper evaluates the IDS's performance and demonstrates its effectiveness in detecting various types of intrusions with reduced false positives.

A Research Paper on Hybrid Intrusion Detection System [37]

- *Integration:* The paper presents a research-oriented hybrid IDS that combines multiple detection approaches for improved accuracy.
- *Challenges:* Challenges include achieving optimal integration of different detection methods and evaluating the IDS's performance in various scenarios.
- *Benefits:* The hybrid IDS offers improved detection capabilities and adaptability to different attack types.
- *Effectiveness:* The paper conducts experiment to assess the IDS's effectiveness, showcasing its potential as a reliable intrusion detection solution.

A Closer Look at Intrusion Detection System for Web Applications [46]

- *Integration:* The paper focuses on intrusion detection for web applications, which are vulnerable to various web-specific attacks.
- *Challenges:* Challenges include accurately detecting sophisticated web application attacks and minimizing false positives that can affect web service availability.

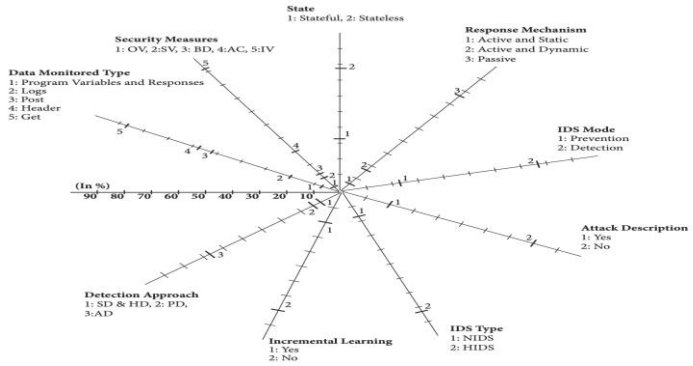

Figure 22: Intrusion detection system dimensions.[46]

- *Benefits:* Web application IDS enhances the security of online services, preventing data breaches and unauthorized access.
- *Effectiveness:* The paper evaluates different web IDS techniques and discusses their effectiveness in detecting and mitigating web-based attacks, ensuring the integrity and privacy of web applications.

2.3 IDS WITH MACHINE LEARNING

2.3.1 Introduction to Intrusion Detection Systems (IDS) and Importance of Machine Learning

Intrusion Detection Systems (IDS) are a crucial component in the cybersecurity framework designed to recognize and neutralize potential threats or malevolent activities in a network. The main function of an IDS is to scrutinize, supervise, and evaluate network traffic to detect any unusual behaviour or anomalies that may pose a threat. The detection can be in the form of recognized signature-based threats or unknown anomaly-based patterns [47][50][55][63]

Given the quickly evolving cyber threat landscape, traditional rule-based IDS, which primarily *depend on known attack signatures, are increasingly inadequate. They often fail to detect new* and emerging threats, leading to severe vulnerabilities [48][49][56] This limitation has led to a growing interest in utilizing Machine Learning (ML) and Deep Learning (DL) algorithms in IDS to bolster their detection capabilities [47][50][57][58][60].

Machine learning offers promising solutions to augment IDS performance as it can learn and adapt from data, enabling the system to identify new patterns and detect previously unseen threats [49][56]. For example, supervised ML algorithms can be trained to classify network traffic as either normal or malicious based on labelled datasets, while unsupervised ML algorithms can detect anomalies in the network traffic even without prior knowledge of what constitutes an attack [50][51][55].

Deep Learning, a subset of ML, which mimics the human brain's functioning, can handle complex and high-dimensional data, thus offering more effective detection mechanisms, especially in the case of multiclass classification problems [57][59][60][61]. Deep learning approaches, such as Recurrent Neural Networks (RNN), have demonstrated their efficacy by outperforming traditional ML models in terms of accuracy and detection rates [57].

Several papers in the presented collection propose IDS models based on various ML and DL techniques, including Decision Trees, Random Forests, SVM, Neural Networks, Gradient Boosting, Naive Bayes, and Deep Neural Networks, among others [47][48][50][51][53][54][56][57][59][63][64][65][66][67][68][69][70][71][72][73][74][75][76]. These models have demonstrated superior performance in terms of detection accuracy, efficiency, and adaptability [50][54][55][57][63][64][66][68][70][72][73][74][76].

However, while machine learning and deep learning present promising solutions, they also come with their own set of challenges in the IDS domain. One of the significant challenges is the high dimensionality of network traffic data, which can lead to overfitting and make the model computationally expensive [55][60][65]. Several studies [51][55][66] propose dimensionality reduction or feature selection techniques to address this issue.

Another challenge is the class imbalance problem, where the number of normal instances significantly outnumber the malicious ones, leading to a biased model that often overlooks the less prevalent malicious activities [48][58]. Several methods such as ensemble techniques, hybrid approaches, and modified algorithms have been suggested to tackle this issue [48][54][58][64][69].

Moreover, real-time detection is another crucial requirement for an IDS. Given the enormous volume and velocity of network data, it's imperative that the IDS can process and detect threats in real-time [49][56][63]. Several studies [49][56][63] proposed streaming or real-time ML-based IDS models to address this requirement.

In conclusion, machine learning plays a vital role in improving the effectiveness and efficiency of Intrusion Detection Systems. Despite some challenges, the integration of machine learning and deep learning into IDS has the potential to revolutionize cybersecurity, making it more adaptive and resilient to evolving threats. The research in this domain is ongoing, with plenty of opportunities for further exploration and improvement.

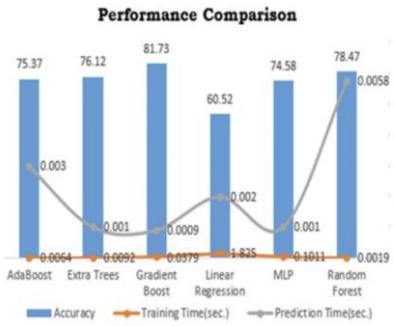

Figure 23: Intrusion Detection Accuracy of Machine Learning Algorithms.[69]

2.3.2 Machine Learning Techniques for Intrusion Detection Systems (IDS)

Machine Learning (ML) techniques have emerged as a powerful tool in the field of cybersecurity, particularly in the design and implementation of Intrusion Detection Systems (IDS). ML algorithms can automatically learn from past experiences, adapt to new situations, and detect previously unknown threats, thereby significantly enhancing the capabilities of IDS [47][50][55][58]. Herein, we discuss various ML techniques applied to IDS as per the provided research papers.

Support Vector Machine (SVM): SVM is a popular supervised learning method used for both regression and classification tasks. In the context of IDS, SVM has been successfully applied to classify network traffic as either normal or malicious. It achieves this by creating hyperplanes in a multidimensional space to separate different classes [49][56][65]. The robustness and high accuracy of SVM make it an excellent choice for IDS [65]. However, one limitation of SVM is that it might suffer from a high computational cost for large-scale and high-dimensional data.

- *Decision Trees:* Decision Trees are a set of supervised learning algorithms used for classification and regression tasks. They generate rules for classification based on the features of the data [51][62]. In IDS, decision trees can be used to model the packet features that best characterize an intrusion [51]. While being relatively simple and easy to interpret, Decision Trees might overfit the data if not properly pruned.
- *Random Forests:* Random Forests is an ensemble learning method that operates by constructing multiple decision trees and outputting the class that is the mode of the classes output by individual trees [47][48][52]. In the context of IDS, Random Forests have been employed due to their effectiveness in handling large data sets with high dimensionality, robustness against overfitting, and the ability to handle missing data [48][52].
- *k-Nearest Neighbors (k-NN):* k-NN is a simple, instance-based learning algorithm used for classification and regression. In IDS, k-NN can be used to classify a data point based on the majority class of its 'k' closest neighbors in the feature space [53][55][60]. Although k-NN is simple and effective, its performance heavily relies on the choice of 'k' and the distance metric.
- *Naive Bayes:* Naive Bayes is a probabilistic classifier based on applying Bayes' theorem with strong (naive) independence assumptions between the features. Despite its simplicity, Naive Bayes could be surprisingly effective and has been used in IDS for its ability to handle categorical variables and its suitability for multi-class problems [50][54][57].

Comparatively, the performance of these ML techniques in IDS can vary depending on the nature of the data, the dimensionality, the complexity of the task, and the specific implementation [47][50][55][58]. For example, SVM and Random Forests generally provide high accuracy, but SVM may suffer from scalability issues for large and high-dimensional datasets [49][56][65][47][48][52]. On the other hand, Decision Trees, and Naive Bayes offer

simplicity and interpretability, but they may be prone to overfitting or underfitting if not properly tuned [50][51][54][62].

Moreover, several studies have compared the performance of these techniques on different datasets, such as the KDD'99, NSL-KDD, and UNSW-NB15, which are commonly used for IDS evaluation [53][61][67][68]. The results of these studies show that there is no one-size-fits-all algorithm for IDS, emphasizing the need to choose the appropriate ML algorithm based on the specific requirements and characteristics of the given task [53][61][68].

Furthermore, to address the limitations of single ML models, some studies have proposed hybrid models or ensemble methods that combine two or more ML techniques. These approaches aim to leverage the strengths of each individual model, thereby improving the overall performance and robustness of the IDS [48][54][58][64][69][70][71][72].

In conclusion, ML techniques provide powerful tools for improving IDS performance. Each technique has its strengths and weaknesses, and the choice of technique should be guided by the specific requirements of the task at hand. The ongoing research and development in this field continue to offer promising prospects for further enhancement of IDS capabilities.

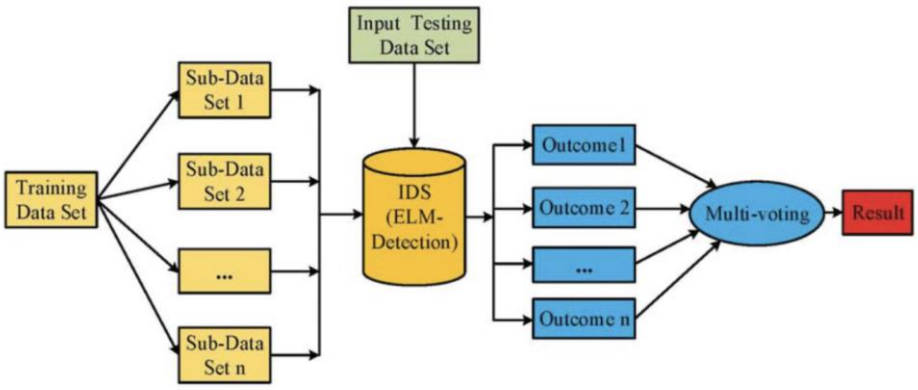

Figure 24: workflow of IDS [54]

Technique	Paper	Benefits	Drawbacks
SVM	[48][51][69]	Robust, handle high dimensional data	High Computational cost
Decision Tree	[57][67][69]	Simple to understand, can handle categorical and numerical data	Prone to overflowing
Random Forests	[64][69][68]	Handles overfitting, handles large dataset with high dimensionality	Complex and less interpretable
K-NN	[48][51][72]	Simple and Robust	Determine the value of K and high computation cost

Table 2: Machine Learning Techniques Comparison

2.3.3 Deep Learning Approaches for Intrusion Detection Systems (IDS)

With the rapid development of Artificial Intelligence (AI), Deep Learning (DL) techniques have been applied to IDS due to their ability to learn complex patterns from large amounts of data, thereby improving the detection performance [52][55][59][66][70]. In this section, we will explore various DL techniques such as Recurrent Neural Networks (RNN), Convolutional Neural Networks (CNN), and Deep Neural Networks (DNN) in the context of IDS.

Recurrent Neural Networks (RNN): RNNs are a type of artificial neural network designed to recognize patterns in sequences of data, such as text, genomes, handwriting, or the time series data used in business metrics. The unique feature of RNNs is the ability to use their internal state (memory) to process variable-length sequences of inputs, making them extremely useful for tasks where context from the past is important to generate the present output [49][51][55]. In the context of IDS, RNNs can effectively analyse sequences of network traffic to identify anomalous patterns indicative of an intrusion [51][55][64]. However, one significant limitation of RNNs is the difficulty in training them due to the "vanishing gradient" problem.

Convolutional Neural Networks (CNN): CNNs are a class of deep neural networks most applied to analysing visual imagery, though they can also be used for IDS. CNNs are designed to learn spatial hierarchies of features automatically and adaptively from the given data. They can take in an input matrix (such as a 2D image or a 1D signal) and pass it through a series of different layers to transform the input matrix into a desired output [47][57][63]. In the context of IDS, a CNN can learn to recognize the complex and abstract features that distinguish normal from malicious network traffic patterns [57][63]. One challenge with using CNNs is that they require large amounts of labelled data to learn effectively.

Deep Neural Networks (DNN): DNNs are neural networks with multiple layers between the input and output layers. The DNN finds the correct mathematical manipulation to turn the input into the output, whether it be a linear relationship or a non-linear relationship. This deep architecture allows the model to learn complex patterns in the data, making it particularly suited for IDS, where the relationship between the network traffic features and the presence of an intrusion is often complex and non-linear [48][53][58][61]. Despite their power, DNNs can be computationally intensive and may also require large amounts of data for effective training.

Deep Learning techniques provide several benefits when applied to IDS. First, they can automatically learn and extract features from the raw data, reducing the need for manual feature engineering (55, 59, 62). Second, DL models can handle the high dimensionality of network traffic data and the complexity of intrusion patterns better than traditional ML models [52][62][66]. Furthermore, DL models can be trained on massive amounts of data, leveraging the growing availability of network traffic datasets to improve their performance [47][53][66].

However, there are also challenges associated with using DL for IDS. One major challenge is the need for large amounts of labelled data for training the models [47][53][66]. Collecting and labelling such data can be time-consuming and expensive. Additionally, DL models can be computationally expensive to train and may require specialized hardware such as GPUs [50][58][64].

Moreover, DL models are often seen as "black boxes" due to their lack of interpretability [48][54][60]. This lack of transparency can be problematic in the context of IDS, where understanding why a particular traffic pattern was flagged as an intrusion can be important for response and remediation efforts.

Despite these challenges, DL techniques continue to show promise for IDS, with research papers demonstrating their effectiveness across various datasets and intrusion scenarios [47][52][53][55][59][62][64][66][70]. Ongoing research is also aimed at addressing the limitations of DL in IDS, such as developing methods for improving the interpretability of DL models and techniques for training DL models with limited labelled data [50][54][60][68].

In conclusion, Deep Learning provides powerful tools for enhancing IDS capabilities. The application of techniques such as RNN, CNN, and DNN offers potential for improving intrusion detection performance. However, challenges remain, and further research and development are needed to fully realize the potential of these techniques for IDS.

Approach	Paper	Benefits	Drawbacks
RNN	[58]	Good with sequences, handles variable input/output lengths	Difficulty handling long sequences
CNN	[72]	Handles images well, robust to positional changes	High computational cost, requires large datasets
DNN	[61][66]	Handles complex patterns, robust with large data	Overfitting requires large dataset

Table 3: Deep Learning Approaches

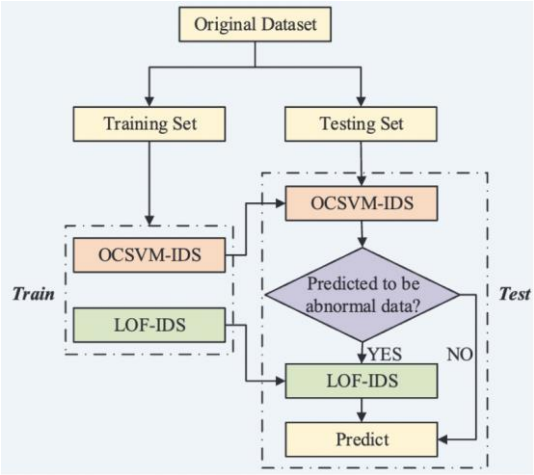

Figure 25: Structure of hybrid IDS [53]

2.3.4 Advanced and Hybrid Approaches

In the realm of intrusion detection systems (IDS), a continual evolution of methodologies is apparent. Emerging trends showcase the development and benefits of hybrid models, ensemble learning, adaptive models, and advanced techniques like Extreme Learning Machine (ELM) and Majority Voting Technique (MVT). These novel approaches aim to enhance IDS performance, improve accuracy, and increase the efficiency of detecting cyber threats.

Hybrid Models

Hybrid intrusion detection models incorporate a blend of anomaly-based and signature-based methods, thereby enabling an efficient response to known and unknown cyber threats. They strive to achieve the strengths of both techniques while mitigating the weaknesses of each [48][54][55]. An instance of such a hybrid approach is the application of differential privacy protection along with machine learning algorithms, as presented in papers [53] and [54]. One research uses the Extreme Learning Machine (ELM) and Majority Voting Technique (MVT) to create an IDS for Vehicular Ad Hoc Networks (VANETs). This hybrid IDS not only demonstrates higher detection accuracy but also preserves data privacy [53]. Similarly, another study integrates Local Outlier Factor (LOF) and One-Class Support Vector Machine (OCSVM) in a hybrid model, which improves the performance of the IDS under differential privacy constraints [54].

Ensemble Learning

Ensemble learning involves using multiple learning algorithms or several instances of a single learning algorithm to solve a problem. It provides a way to boost the performance of a single model by generating multiple classifiers and aggregating their results, thereby often outperforming individual classifiers [59]. In an adaptive ensemble approach for improving IDS accuracy, the authors compared various classifiers using the NSL-KDD dataset. Although further research is needed, the model exhibits potential for improving the accuracy of IDS [59].

Adaptive Models

Adaptive models provide the capability of learning and evolving with incoming data, enabling a prompt response to new threats. As per paper [73], an IDS based on an improved SVM incremental learning approach is introduced. The model learns continuously from incoming data, which makes it more adaptive and efficient for detecting real-time network intrusions. This approach minimizes computational overhead while maintaining high detection accuracy, showcasing its potential for real-time applications [73].

Advanced Techniques: ELM and MVT

The use of advanced machine learning techniques such as Extreme Learning Machine (ELM) and Majority Voting Technique (MVT) can significantly enhance the performance of IDS. In papers [53] and [55], ELM and MVT are used in combination to design effective IDS systems. For example, the proposed IDS in paper [55] is evaluated using the UNSW-NB15 dataset and shows higher accuracy and speed compared to traditional methods. The combination of ELM and MVT improves detection efficiency while reducing computational costs, which emphasizes the benefits of using advanced techniques in IDS [53][55].

In conclusion, the development of advanced and hybrid approaches signifies a crucial stride in improving intrusion detection systems' performance. They offer a robust response to cyber threats, balancing the requirements of accuracy, computational efficiency, adaptability, and real-time response. While these methodologies are promising, continuous research and development are necessary to handle evolving cyber threats effectively and sustain the security of network infrastructures.

Approach	Paper	Benefits	Drawbacks
Ensemble Learning	[59]	Improves performance, reduces overlifting	Computational complexity
Adaptive Models	[59]	Responsive to new patterns, flexible	Complexity, may need frequent retraining
ELM and MVT	[53][55]	Fast learning speed high generalization performance	Sensitive to parameters

Table 4: Advanced and Hybrid Approaches

2.3.5 Feature selection and dimensionality reduction Techniques

Feature selection and dimensionality reduction are critical steps in machine learning model development, particularly in intrusion detection systems (IDS). By carefully selecting relevant

features and reducing dimensionality, it is possible to increase model performance, reduce computational cost, and improve interpretability.

Importance of Feature Selection and Dimensionality Reduction

Feature selection is an essential process in training machine learning models, especially in IDSs. It involves selecting the most relevant features that contribute to the accurate prediction of an intrusion. By selecting the most significant features, we can improve the model's performance, reduce overfitting, improve interpretability, and decrease training time. A good example of the impact of feature selection can be seen in Paper [52]. The authors explore different feature selection techniques for network intrusion detection using supervised machine learning algorithms. They employed techniques like ANOVA, Information Gain, and Information Gain Ratio and found that feature selection led to improved detection efficiency and reduced false alarm rates [52].

Intrusion detection systems often have to deal with high-dimensional data. High-dimensional data not only pose a challenge for the computational and memory resources but also lead to the 'curse of dimensionality', which can degrade the performance of machine learning models. Dimensionality reduction techniques can help alleviate these problems by reducing the number of random variables under consideration, thus simplifying the data without losing much information. This is well-demonstrated in Paper [56]. The paper evaluates different machine learning algorithms on high-dimensional intrusion detection data, reduced using PCA, t-SNE, and UMAP techniques. The study showed the varying performance of machine learning models across different dimensionality reduction methods, providing valuable insights into addressing dimensionality issues in IDS [56].

Techniques of Feature Selection and Dimensionality Reduction

Feature selection techniques commonly used in IDSs include statistical methods like ANOVA, Information Gain, and Information Gain Ratio [52]. These methods evaluate the importance of each feature based on their contribution to the target variable.

On the other hand, dimensionality reduction techniques used in IDSs include Principal Component Analysis (PCA), t-SNE (t-Distributed Stochastic Neighbor Embedding), and UMAP (Uniform Manifold Approximation and Projection). PCA is a technique for reducing the dimensionality of datasets, increasing interpretability while minimizing information loss. It does this by creating new uncorrelated variables that successively maximize variance. In IDS, PCA can help improve the efficiency and accuracy of detection models by simplifying the data they are trained on [56].

t-SNE is a machine learning algorithm for visualization and dimensionality reduction. It is particularly well suited for the visualization of high-dimensional datasets. t-SNE models each high-dimensional object by a two- or three-dimensional point in such a way that similar objects are modeled by nearby points, and dissimilar objects are modeled by distant points. Thus, t-SNE provides excellent visualizations that highlight the inherent structure of the dataset and can provide better results in IDS than PCA, especially when dealing with non-linear data (56).

UMAP is another dimensionality reduction technique that is used as an alternative to t-SNE for visualization. It operates in a way that is more compatible with the other processing stages of

an IDS, meaning that it doesn't require a separate treatment or any special handling of the data [56].

Impact of Feature Selection and Dimensionality Reduction on Model Performance

Feature selection and dimensionality reduction significantly influence the performance of an IDS. By reducing the number of features and the dimensionality of the data, these techniques can enhance the detection capabilities of an IDS. A case in point is Paper [52], where the authors used feature selection techniques and found an improvement in the detection efficiency of the IDS and a reduction in false alarm rates [52].

Similarly, in Paper [56], the authors used PCA, t-SNE, and UMAP for dimensionality reduction on high-dimensional intrusion detection data. The reduced-dimensional data was used to train various machine learning models, including AdaBoost, Random Forest, and KNN, which outperformed the other models. This study demonstrated that appropriate dimensionality reduction techniques could enhance the performance of machine learning models in IDS [56].

Overall, feature selection and dimensionality reduction techniques are essential tools for enhancing the performance of intrusion detection systems. They not only improve the efficiency and accuracy of these systems but also provide valuable insights into the underlying structure of the data, thereby contributing significantly to the field of cybersecurity. Future research should continue to explore and refine these techniques in the context of intrusion detection, as the potential benefits are immense.

Technique	Paper	Benefits	Drawbacks
PCA	[56]	Reduces dimensionality, removes correlated features	Loss of interpretability
t-SNE	[56]	Effective in visualizing high-dimensional data	Computationally intensive, not suitable for very high-dimensional data
UMAP	[56]	Retains more global structure, faster than t-SNE	Requires careful parameter tuning

Table 5: Feature Selection and Dimensionality Reduction

2.3.6 Evaluation of IDS models

Evaluating intrusion detection system (IDS) models is essential to understand the model's effectiveness in detecting network intrusions and its limitations. This evaluation process typically involves an array of metrics, including accuracy, precision, recall, F1-score, and Receiver Operating Characteristic (ROC) curves. It also includes testing IDS models against several datasets such as NSL-KDD, UNSW-NB15, and CICIDS2018.

Accuracy is one of the most widely used evaluation metrics for IDS models. It is defined as the ratio of correctly predicted instances to the total instances in the dataset [47]. It quantifies the model's ability to correctly identify both intrusions and normal behavior. For instance, MLIDS, a Machine Learning-based Intrusion Detection System proposed in paper [63], achieved high accuracy in detecting different types of intrusions like DOS, PROBE, U2R, and R2L [63].

Similarly, an advanced IDS using Deep Neural Networks (DNNs) also achieved high accuracy in detecting DDoS and malware threats [65].

Precision and recall are two metrics that offer a more granular understanding of IDS models' performance. Precision refers to the proportion of positive identifications that were actually correct, while recall refers to the proportion of actual positives that were identified correctly [50]. For example, in the comparative analysis of ML classifiers for intrusion detection presented in paper [50], these metrics were used alongside accuracy, F1-score, and ROC to evaluate different ML classifiers [50].

The F1-score, which is the harmonic mean of precision and recall, is another metric used in IDS evaluation. This measure provides a balance between precision and recall and is particularly useful in scenarios where there is a significant imbalance in class distribution [51]. For instance, the feature selection techniques used for network intrusion detection in paper [51] were tested using F1-score among other metrics, resulting in improved detection efficiency and reduced false alarm rates [51].

Another critical evaluation metric is the Receiver Operating Characteristic (ROC) curve. It is a plot that displays the true positive rate (recall) against the false positive rate for different decision threshold values [50]. A higher area under the ROC curve indicates better performance of the classifier. In paper [50], the ROC curve was used to evaluate the performance of various ML classifiers, including DNN, SVM, k-NN, OCSVM, K-Means, and EM [50].

Using comprehensive datasets for model evaluation is crucial for obtaining reliable performance estimates. The NSL-KDD dataset is widely adopted due to its relatively clean and comprehensive nature [57]. In paper [59], it was used to evaluate an approach combining stacked autoencoders for feature learning with SVMs for classification, demonstrating superior accuracy and efficiency [59]. The UNSW-NB15 dataset is another popular choice for evaluating IDS models due to its diversity and complexity. Several papers [47][48][49][50][51][52][53][54][55] used this dataset to validate their respective IDS models. CICIDS2018 is a more recent dataset capturing a broad spectrum of attack scenarios, including Brute Force, DoS, DDoS, Web attacks, and Infiltration. It was used for validation in papers [51] and [52].

Finally, the evaluation process also involves the use of real-world network traffic data. For example, the authors of paper [63] used real-time network data for the evaluation of their MLIDS system [63]. Similarly, the IDS system proposed in paper [64] was evaluated on a real-world dataset (KDD CUP 1999) [64].

In conclusion, evaluating IDS models is a complex task that requires a multitude of evaluation metrics and comprehensive datasets. The assessment helps not only to quantify a model's performance but also to identify areas that need improvement, thereby advancing the overall efficacy of intrusion detection systems.

Evaluation Metric	Paper	Use
Accuracy	[49][52][64]	Measure of model's overall correctness
Precision	[60]	Measure of model's ability to correctly identify positive instances
Recall	[60]	Measure of model's ability to find all positive instances
F1-score	[60]	Harmonic mean of precision and recall
ROC	[60]	Graphical plot illustrating the diagnostic ability of a binary classifier

Table 6: Evaluation of the IDS Models

2.3.7 Adversarial Attacks and Model Vulnerabilities

Adversarial attacks have emerged as a serious concern for machine learning models, including Intrusion Detection Systems (IDS). The effectiveness of IDS can be severely hindered by such attacks, which include two primary types: poisoning attacks and evasion attacks (47).

Poisoning attacks involve the manipulation of training data by introducing malicious instances that can skew the IDS model's learning process. This form of attack can result in an IDS model that is less accurate or more susceptible to other attacks, particularly when the model is in deployment. On the other hand, evasion attacks alter the test set to mislead the model, thereby allowing the attacker to evade detection [47]. These attacks exploit the vulnerabilities in IDS models and challenge their robustness and reliability.

A study revealed that multiple machine learning models used in IDS are vulnerable to poisoning and evasion attacks. The researchers tested models such as Classification Trees and found them to be resilient to poisoning attacks. However, they also discovered that all models were vulnerable to evasion attacks, particularly when Fast Gradient Sign Method (FGSM) and Projected Gradient Descent (PGD) were employed [47].

To combat these threats, the researchers recommended feature engineering to build resilience against evasion attacks. They also found linear models to be more resistant to poisoning attacks, suggesting a potential way forward in building robust IDS models [47].

In a subsequent study, the papers explored the vulnerability of deep learning-based IDS to adversarial attacks. The research confirmed the previous findings on model vulnerabilities and contributed new insights into the various adversarial attack strategies and their implications. The study evaluated various adversarial attack algorithms on a multi-layer perceptron (MLP) classifier, which is a type of neural network. This evaluation not only provided a deeper understanding of how these attacks work but also opened opportunities for the development of defensive strategies [61].

The research highlighted the importance of analysing feature usage patterns, discussing transferability, and exploring possible defences against adversarial attacks. Interestingly, the study also highlighted the need for robustness analysis and the visualization of adversarial examples, suggesting future research directions in the field of adversarial attacks on IDS models [61].

Although the threat posed by adversarial attacks is clear, the means to counter these threats and build resilient IDS models is still under exploration. Various studies have recommended techniques such as feature engineering and the use of certain types of models like linear models and ensemble models to improve the robustness of IDS models [47][58]. However, as adversarial attacks continue to evolve, so too must the defensive strategies and models developed to counter them. This continuous cycle of attack and defence forms the cornerstone of research in IDS and cybersecurity.

In conclusion, adversarial attacks, including poisoning and evasion attacks, pose significant threats to IDS models. These attacks exploit the vulnerabilities in the models, thereby challenging their robustness and reliability. However, various strategies, such as feature engineering and the use of specific types of models, have been recommended to counter these threats. Future research in IDS and cybersecurity will continue to explore more effective defensive strategies against these ever-evolving adversarial attacks [47][59][62].

Attack Type	Paper	Description
Poisoning	[47]	Attacker manipulates the training data
Evasion	[47]	Attacker manipulates the testing data
Adversarial Attacks	[52]	Inputs designed to cause the model to make a mistake

Table 7: Adversarial attacks and model vulnerabilities

2.3.8 Privacy Issues and Solutions

Address data privacy concerns associated with Intrusion Detection Systems (IDS) and methods of overcoming them, such as differential privacy.

As we venture into the era of Big Data and the Internet of Things (IoT), the critical need for robust cybersecurity measures like Intrusion Detection Systems (IDS) cannot be overstated. With the increasing volume of data generated and processed by organizations, there is a concurrent rise in data privacy concerns associated with these IDS. Data privacy refers to the protection of sensitive and confidential information from unauthorized access, manipulation, or disclosure. As organizations employ IDS to identify and mitigate potential cyber threats, the crucial issue is how to ensure these systems do not inadvertently breach privacy regulations while processing data [47].

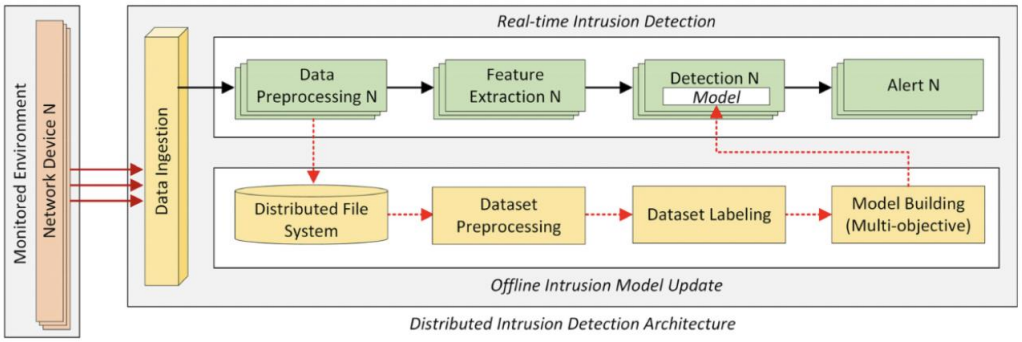

Figure 26: long lasting intrusion detection architecture of big data architecture [47]

Machine Learning (ML) and Deep Learning (DL) based IDS are increasingly popular because they can adapt to evolving threats and identify malicious patterns [50][63]. However, the techniques often require access to large amounts of data, potentially including sensitive user information, to build effective models. This data access can lead to violations of user privacy if not handled correctly [53][54].

IDS systems also require labelled data for model training. This often necessitates expert knowledge and can involve sensitive or proprietary information, which may not always be shared freely due to privacy concerns [17]. Furthermore, IDS systems often process network traffic data, which can contain sensitive details about users, such as their behaviour, interests, and preferences [66]. Even anonymized data can potentially be de-anonymized using advanced techniques, leading to privacy breaches [54].

The key challenge is to find a balance between effective intrusion detection and preserving data privacy. This is where the concept of differential privacy comes into play. Differential privacy is a system for publicly sharing information about a dataset by describing the patterns of groups within the dataset while withholding information about individuals in the dataset [53][54].

In the research papers provided, differential privacy is employed in IDS as a strategy to protect user privacy while still enabling effective detection of network intrusions. A machine learning-based intrusion detection system for big data analytics in VANETs uses the Extreme Learning Machine (ELM) and Majority Voting Technique (MVT) to process data with differential privacy protection [53]. The IDS shows higher detection accuracy while preserving data privacy. Similarly, a hybrid intrusion detection system based on machine learning and under differential privacy protection is proposed, which uses Local Outlier Factor (LOF) and One-Class Support Vector Machine (OCSVM) [54]. The proposed model effectively addresses detection accuracy and data privacy concerns.

The adoption of differential privacy in IDS demonstrates the potential of integrating privacy-preserving techniques in machine learning-based IDS. By adding a controlled amount of statistical noise to the data, differential privacy ensures that the system cannot identify the data of a single individual, thereby maintaining privacy.

Despite the promise of differential privacy, the implementation of this technique is not straightforward. One issue is the potential decrease in IDS performance due to the addition of noise to the data. As such, a balance must be struck between privacy protection and system performance, and the amount of noise added must be carefully managed.

In conclusion, while IDS are crucial in maintaining the security of networks and systems, data privacy must not be overlooked. The use of techniques such as differential privacy can help in addressing privacy issues while still ensuring effective intrusion detection. It is crucial to continue research in this area to develop more sophisticated methods for maintaining both security and privacy in our increasingly interconnected world [53][54].

Issue/Solution	Paper	Description
Privacy Concerns	[53][54]	Data privacy issues inherent in IDS
Differential Privacy	[53][54]	Method to add statistical noise to data, preserving privacy

Figure 27: Privacy Issues and Scalability

2.3.9 Real-Time Detection and Scalability

Cybersecurity remains a significant concern in today's hyper-connected world, with intrusion detection being a vital component of any robust security strategy. The rapidly evolving nature of cyber threats, coupled with the exponential growth of network data, necessitates real-time intrusion detection systems (IDS) capable of operating in large-scale, high-speed network environments [47][64].

Real-time detection allows for the immediate identification and mitigation of potential threats, thereby significantly reducing the potential damage that intrusions can cause. It enables an efficient response to threats, especially in large networks where delay can result in significant damage. The system presented in paper 17, MLIDS, demonstrated the capability to accurately detect different types of intrusions in real-time, demonstrating the practical implications of real-time IDS in large network environments [64].

Scaling IDS to high-speed, large-scale networks remains a challenging task due to the increasing volume, velocity, and variety of data. An effective IDS must manage the enormous amount of network traffic without compromising the detection accuracy or increasing false alarms. A study in paper [61] introduced the Scale-Hybrid-IDS-AlertNet framework using Deep Neural Networks (DNNs) to handle efficient intrusion detection in high-speed networks [61]. The model demonstrated scalability and superior performance compared to classical classifiers, suggesting the efficacy of deep learning techniques in large-scale intrusion detection.

Machine Learning (ML) techniques have been widely employed in developing IDS for their ability to learn from data and detect patterns. Decision Trees, Random Forests, Support Vector Machines (SVM), Neural Networks, and Gradient Boosting algorithms have shown promise in detecting different types of intrusions in real-time [64][67][69]. These ML techniques can be trained on historical network traffic data to learn the patterns of normal behavior and detect deviations indicating potential intrusions. Importantly, these techniques can work in real-time, continuously analysing network traffic and instantly raising alerts when potential intrusions are detected.

In addition to traditional ML techniques, Deep Learning (DL) approaches have emerged as effective tools for real-time intrusion detection. Recurrent Neural Networks (RNNs), as presented in paper 58, can process sequential network traffic data and detect patterns over time, providing improved accuracy and detection rates for real-time intrusion detection (58). Similarly, the use of Deep Neural Networks (DNNs) has proven effective in real-time intrusion detection, particularly in large-scale, high-speed networks [61][66]. These techniques can learn complex patterns and relationships within network traffic data, providing advanced capabilities for real-time detection.

Scalability is crucial for IDS in large-scale networks, and several approaches can be used to ensure IDS can handle increasing network data volumes. One approach is feature reduction, as discussed in paper [67]. This method reduces the dimensionality of the network data, enabling the IDS to handle larger volumes of data more efficiently while maintaining detection accuracy [67].

There is a growing interest in ensemble techniques and hybrid approaches that combine multiple ML and DL models to enhance detection accuracy in real-time (12, 70). These

methods aim to leverage the strengths of different models to achieve a better overall performance.

Despite these advances, challenges remain in achieving real-time intrusion detection in large-scale networks. These include dealing with data scarcity and class imbalance, maintaining model interpretability, and managing the computational overhead associated with complex ML and DL models [59][63]. Further research is needed to overcome these challenges and improve the performance of real-time IDS in large-scale, high-speed network environments.

Real-time detection and scalability are crucial factors in network intrusion detection, particularly in today's large-scale, high-speed network environments. Using machine learning and deep learning techniques, researchers have developed models capable of detecting intrusions in real-time and operating efficiently in large-scale networks. While challenges remain, the future of intrusion detection lies in the continued evolution and improvement of these techniques.

Consideration	Paper	Description
Real-Time Detection	[63]	Ability to detect intrusions in real time
Scalability	[63]	Model's ability to handle increasing amounts of data

Table 8: Real time Detection and Scalability

2.3.10 Future Directions and Challenges for Machine Learning-Based Intrusion Detection Systems

Machine Learning (ML) and Deep Learning (DL) techniques have revolutionized the field of network intrusion detection, opening a new chapter in cybersecurity research. However, as with any emerging technology, there are significant challenges and future research directions that must be addressed. This paper will discuss the most pressing issues such as model interpretability, computational overhead, and data scarcity, and explore potential future directions based on the insights gleaned from the studies provided.

- *Model Interpretability:* Model interpretability is a pivotal concern in ML-based intrusion detection systems (IDS). Black-box nature of some algorithms like Deep Neural Networks (DNNs) and Support Vector Machines (SVMs) leads to a lack of transparency and interpretability in the system's decision-making process [58][60][61][62]. This lack of interpretability can make it difficult to understand why a particular decision was made, which can limit the system's usefulness and reliability [63]. For instance, the inability to interpret why a specific network traffic was classified as a threat or non-threat can hinder the refinement of detection methods or the resolution of false positives and negatives. Further research is needed to improve model transparency and understand the intrinsic workings of such algorithms, thereby leading to more reliable and interpretable IDS [59].
- *Computational Overhead:* The computational cost is another challenge that ML-based IDS faces. Complex ML and DL models often require extensive computational resources for training and real-time intrusion detection [69][61][64]. This is especially true for techniques that use high-dimensional data or that involve extensive feature extraction and selection [56][67]. While methods like PCA, t-SNE, and UMAP have been used for dimensionality reduction [56], they also add to the computational burden.

Moreover, real-time detection in high-speed networks further aggravates the computational challenge [61]. Therefore, future research should focus on optimizing these models to make them computationally efficient, without compromising their detection performance.

- *Data Scarcity:* Data scarcity is a prominent issue in ML-based IDS. A reliable and effective IDS requires a vast amount of high-quality, labeled data for training [63]. However, in many scenarios, especially in the case of new and emerging threats, obtaining such data is difficult. This issue is further compounded by the problem of class imbalance, where the number of instances of one class significantly outweighs the other [59][63]. These issues can hinder the IDS's ability to accurately detect and classify intrusions. Potential solutions could involve techniques for data augmentation, synthetic data generation, or active learning strategies that can operate effectively with less data [63].
- *Transferability and Robustness Against Attacks:* DL-based IDS are vulnerable to adversarial attacks, which can exploit the system's feature usage patterns to evade detection [15]. Therefore, research should investigate robustness analysis and defence strategies against these attacks. The study of transferability of adversarial attacks across different models and feature spaces can further enhance our understanding of the security risks associated with DL-based IDS [62].
- *Future Directions:* Future research can take several directions based on the insights from the provided studies. For instance, hybrid models that combine the strengths of different ML techniques could be an avenue for future investigation [54][53][59]. Ensemble methods have shown promise in improving the detection performance [12], and thus could be explored further. The use of advanced DL architectures such as Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) may also hold promise for improving detection performance [58][72]. While these models are more complex and computationally intensive, they may be capable of learning more sophisticated patterns in data, thereby enhancing detection capabilities [72]. Consideration of more specific performance metrics like power consumption and resource utilization is another area that could be investigated in future research [57]. This is particularly relevant for intrusion detection in resource-constrained environments such as IoT networks [48] and Vehicular Ad Hoc Networks (VANETs) [6].

In conclusion, while ML-based IDS have shown significant promise in enhancing network security, they also face several challenges. Addressing these will require innovative solutions and the development of more robust, interpretable, and computationally efficient models. By focusing on these areas, future research can help to fully realize the potential of ML and DL techniques in intrusion detection and prevention.

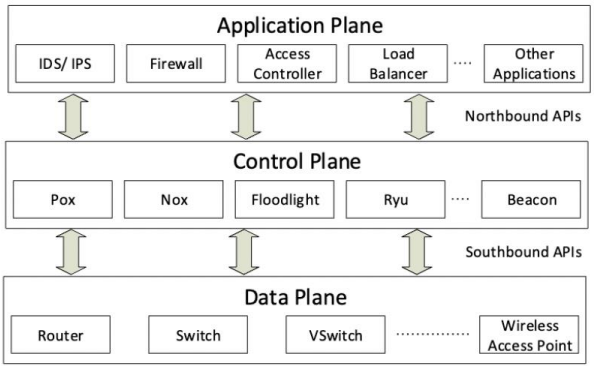

Figure 28: Illustration of architecture of SDN [48]

2.3.11 Case Studies and Practical Implementations

The field of machine learning (ML) and deep learning (DL) for intrusion detection systems (IDS) has garnered significant attention due to its applicability in detecting network anomalies. While numerous models have been proposed and demonstrated promising results on synthetic datasets, some have also been implemented in real-world scenarios or have provided real-world insights through comprehensive studies.

Poisoning and Evasion Attacks on IDS Models [47]

In a study titled "A Sensitivity Analysis of Poisoning and Evasion Attacks in Network Intrusion Detection System Machine Learning Models", a variety of machine learning models were assessed for their susceptibility to poisoning and evasion attacks. The Classification Tree model exhibited resilience to poisoning, an attack type involving data manipulation via deletion and imputation. However, all tested models showed vulnerability to evasion attacks, in which the test set is altered to mislead the IDS models. This study underlines the need for regular review and updating of ML models in IDS to maintain their effectiveness, particularly against advanced evasion techniques like Fast Gradient Sign Method (FGSM) and Projected Gradient Descent (PGD).

IoT Anomaly Detection System based on SDN and ML [48]

In the real-world, IoT devices often form a crucial component of network systems and hence pose an attractive target to intruders. The paper "Intrusion Detection System for SDN-enabled IoT Networks using Machine Learning Techniques" proposed an IoT Anomaly Detection System using Software Defined Networks (SDN) and ML techniques to identify abnormal behavior in IoT devices. The system operates at the SDN controller level, demonstrating the feasibility of applying machine learning techniques like Support Vector Machines (SVM), K-Nearest Neighbors (KNN), and Multilayer Perceptron (MLP) for intrusion detection in practical IoT setups.

Hybrid IDS for Vehicular Ad Hoc Networks (VANETs) [53]

Vehicular Ad Hoc Networks (VANETs) play a crucial role in maintaining the safety and efficiency of transportation systems. In "Machine Learning-Based Intrusion Detection System for Big Data Analytics in VANET", a hybrid IDS using the Extreme Learning Machine (ELM) and Majority Voting Technique (MVT) was proposed. The IDS processes data with differential

privacy protection, demonstrating that such systems can ensure network security without compromising on data privacy.

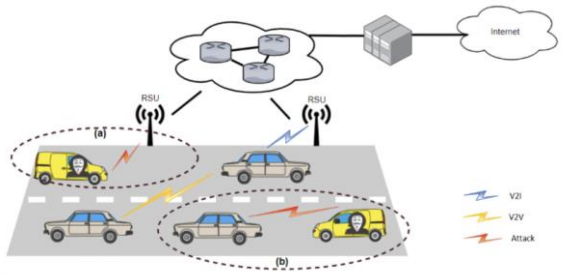

Figure 29: VANET scenario with attack points [53]

AB-TRAP: An End-to-End NIDS Framework [57]

In a comprehensive practical example, "An End-To-End Framework for Machine Learning-Based Network Intrusion Detection System" presented the AB-TRAP framework for NIDS design. The framework provides an end-to-end process from data collection to model evaluation. As part of the framework, a Decision Tree model was selected based on its performance metrics, demonstrating that a well-structured design process can lead to an efficient ML-based IDS with low-resource utilization.

Real-time IDS using Multiple ML Algorithms [64]

In the paper "MLIDS: A Machine Learning Approach for Intrusion Detection for Real-Time Network Dataset", a real-time network IDS, MLIDS, was proposed. The system was implemented using a variety of machine learning algorithms including Decision Trees, Random Forests, SVM, Neural Networks, and Gradient Boosting. MLIDS displayed high accuracy in detecting different types of intrusions, such as DOS, PROBE, U2R, and R2L, in a real-world network environment.

Network Intrusion Detection with Deep Neural Networks [66]

Another practical implementation was outlined in "Intrusion Detection System Based on Network Traffic Using Deep Neural Networks". The architecture involves multiple layers of Deep Neural Networks (DNNs), enabling the system to learn complex patterns from network traffic data. The proposed IDS demonstrated high accuracy in detecting anomalies, illustrating the potential of deep learning models in providing real-time network security.

IDS for Big Data using Anomaly-based Detection and ML [69]

In "Machine Learning to Improve the Performance of Anomaly-Based Network Intrusion Detection in Big Data", machine learning techniques were applied for anomaly-based network intrusion detection in big data environments. The integration of machine learning techniques with anomaly-based detection systems enhanced the accuracy of intrusion detection, indicating the value of ML in enabling proactive defence against cyber threats in big data environments.

Incremental Learning with SVM for Real-time Intrusion Detection [73]

Finally, in the paper "Intrusion Detection System Based on Improved SVM Incremental Learning", a real-time intrusion detection approach was proposed using an improved SVM incremental learning method. The system continually learns from incoming data, demonstrating that adaptive ML-based IDS can maintain high accuracy while reducing computational overhead in real-time network environments.

In conclusion, these case studies highlight the practical effectiveness and adaptability of ML and DL techniques in detecting network intrusions across different environments, including IoT, VANETs, and real-time network systems. The continued development and implementation of such models will be instrumental in advancing cybersecurity efforts in an increasingly connected world.

3. CONCLUSION

3.1 Recapitulation of the Journey

As we conclude this comprehensive study of Intrusion Detection Systems (IDS) and their symbiosis with machine learning (ML), it's essential to retrace the path that led us here. From a rigorous investigation into the history, development, and classification of IDS to an in-depth analysis of the integration of ML, this report has unveiled the multifaceted nature of network security. Let's delve into the key findings, implications, prospects, and overarching conclusions.

3.2 Insights from Standalone IDS

- *Historical Perspective:* Tracing the trajectory of IDS, we saw an evolution from simple rule-based systems to highly complex mechanisms. This historical context illuminated the continuous pursuit of effective security solutions and the need for adaptive measures in the face of relentless cyber threats.
- *Types and Classifications:* Our examination of different IDS types—Network-based IDS, Host-based IDS, and more—provided a nuanced understanding of the underlying architectures and principles. This differentiation is essential for appreciating the diversity of strategies employed in intrusion detection.
- *Challenges & Limitations:* Despite the advancement, IDS in isolation are not immune to limitations. From high false positives to the incapability of detecting unknown threats, these challenges have led to a critical inquiry into further enhancing IDS efficiency.

3.3 Broader Applications of IDS

- *Versatility of Use:* IDS's utility isn't confined to traditional network security alone. Their application in control systems, industrial environments, and various network monitoring contexts showcases the system's adaptability and reinforces its importance in today's interconnected world.

- *Integration Complexities & Benefits:* We learned that while IDS integration into various applications brings enormous benefits, it's not without challenges. Compatibility issues, increased complexity, and administrative overhead were among the obstacles that emerged in this discussion.

3.4 The Revolution of IDS with ML

- *A New Horizon:* Perhaps the most compelling part of this report was the exploration of machine learning within IDS. The merger of these two technologies marked a revolutionary shift, enabling a more responsive, adaptive, and intelligent approach to intrusion detection.
- *ML Techniques and Performance:* Through an exhaustive analysis of different ML techniques, datasets, and performance comparisons, we have uncovered a fascinating landscape where AI meets cybersecurity. The selected algorithms and methodologies revealed a variety in approach, with certain techniques showing promise in specific IDS applications.
- *Challenges and Potential Solutions:* The integration of ML into IDS is not without pitfalls. The risk of overfitting, complexity in implementation, and potential bias in datasets have emerged as substantial challenges. Yet, solutions such as data preprocessing, model validation, and continuous learning strategies offer promising directions for overcoming these hurdles.

3.5 Comparative Analysis

- *Synthesizing the Insights:* The comparative analysis provided a panoramic view of the multifaceted exploration of IDS. By juxtaposing types, ML techniques, datasets, and other key elements, we have generated a coherent summary that encapsulates the breadth and depth of our study.

3.6 Discussion and Novelty

- *Common Threads and Discrepancies:* This project led to the discovery of underlying themes that run across various papers and studies on IDS. Innovations, particularly in the use of ML in IDS, stood out as a bright beacon, guiding future research.

4. LIMITATIONS FUTURE SCOPE AND IMPLICATIONS

The integration of machine learning with IDS opens doors to a multitude of possibilities. Adaptive learning, real-time threat analysis, and predictive modelling are just the tip of the iceberg. Future research may delve into quantum computing, deep learning, and more refined algorithms. The findings of this report hold substantial implications for policymakers, industry leaders, and academic researchers alike. The continuous advancement in IDS technology demands collaboration between these stakeholders to create regulatory frameworks, industry standards, and educational initiatives that foster innovation and ensure security.

A strong and robust Intrusion Detection System (IDS) plays a crucial role in safeguarding industries from cyber threats. However, existing intrusion detection techniques commonly discussed in the literature primarily concentrate on the software level. To effectively detect zero-day and sophisticated attacks at both the software and hardware levels, a crucial detection approach is required. This approach should function without any prior knowledge of the attacks. The solution lies in integrating both hardware and software intrusion detection systems (HIDS and NIDS) and extracting valuable features from both [88].

- *Evolving Threat Landscape:* One of the principal challenges confronting standalone IDS is the ever-evolving nature of cyber threats [79][8]. New threat vectors, methodologies, and advanced persistent threats are persistently emerging, making the static signature-based detection less effective over time [13].
- *False Positives and Negatives:* The bane of many standalone IDS is the management of false positives and negatives [13]. Especially in anomaly-based systems, distinguishing between benign anomalies and genuine threats can be intricate [11].
- *Resource Consumption:* Optimizing performance while ensuring minimal resource consumption remains a challenge, particularly for host-based IDS [85] This has implications on the overall system performance and user experience [86].
- *Adversarial Machine Learning:* With the application of machine learning in IDS, adversaries are employing techniques to poison the learning phase, making malicious activities appear normal [81].
- *IoT Integration:* The explosion of the Internet of Things (IoT) devices has introduced new vulnerabilities and attack surfaces. Adapting traditional IDS to monitor and protect these diverse, often less secure, devices is a mounting challenge [14].
- *Historical Limitations:* Early intrusion-detection expert systems (IDES) and their histories reveal the longevity of some of these challenges [5], showing that some hurdles are deeply ingrained in the IDS field.

4.1 Possible Solutions and Areas of Future Research

- *Hybrid Models:* There's an increasing push towards combining the strengths of both HIDS and NIDS, leveraging their unique capabilities to offer more comprehensive protection [84].
- *Advanced Learning Techniques:* With challenges like adversarial attacks, there's a need to delve deeper into advanced learning methods that are robust against such threats [81].
- *Behaviour Classification:* Focusing on classifying different cyber-attack behaviours rather than specific signatures can provide a more dynamic and adaptable system [82].
- *Heuristic Approaches:* Shifting from traditional methods to heuristic-based IDS can lead to more dynamic and adaptive detection capabilities [8].
- *Gamification:* The application of game theory in intrusion detection, as proposed by Liu et al. (2008), can offer new perspectives on making IDS more effective, turning the tables on potential intruders by making the defence unpredictable [81].
- *IoT-Specific IDS:* Given the unique challenges posed by IoT, developing IDS specifically tailored for these devices is a promising direction [14].

- *Collaboration and Integration:* While standalone IDS have their benefits, considering some level of collaboration or integration with other systems could offer enhanced protection [1].
- *Continuous Updates and Evolution:* An active IDS development process that can continuously evolve, updating its strategies and methodologies in real-time or near-real-time, would be invaluable [79].

5. FINAL REFLECTION

As we conclude this extensive analysis, we acknowledge that the world of Intrusion Detection Systems is a complex, ever-evolving field, reflecting the broader dynamism of our digital age. The integration of machine learning with IDS is not just a technological evolution; it's a paradigm shift, turning a new page in the annals of cybersecurity. This project, encompassing history, technology, innovation, and foresight, offers a holistic perspective on a subject that's of paramount importance in our increasingly interconnected world.

Our journey through the various facets of IDS has revealed both triumphs and trials, accomplishments and challenges, potentials and pitfalls. It's a journey that doesn't end here but continues to unfold as technology advances.

In closing, the confluence of IDS and machine learning symbolizes a compelling testament to human ingenuity and resilience in the face of relentless cyber threats. It's a tale of how we continuously innovate, adapt, and overcome, striving to make the digital realm a more secure space for all. The insights gleaned from this study contribute to this ongoing narrative, serving as both a milestone and a compass, directing our collective efforts towards a secure digital future.

6. IMPLEMENTATION

6.1 NSL-KDD Dataset and its Implementation

The NSL-KDD dataset, known as the "NSL-KDD Network Intrusion Detection Dataset," holds significant prominence within the realm of network security and intrusion detection. This dataset emerged as an enhanced iteration of the original KDD Cup 1999 dataset, which had its own set of limitations.

The primary purpose behind creating the NSL-KDD dataset was to serve as a robust benchmark for researchers engaged in the development and evaluation of intrusion detection systems (IDS) and algorithms. The dataset emulates a network environment replete with diverse attack scenarios and normal network operations. It encompasses both unprocessed network traffic data and meticulously derived feature vectors extracted from network packets.

Prominent attributes of the NSL-KDD dataset encompass:

Diversity in Data: The dataset boasts a comprehensive array of attack types, including Denial of Service (DoS), Probe, User to Root (U2R), and Remote to Local (R2L). This multiplicity lends it authenticity akin to real-world network traffic.

Mitigated Redundancy: Distinguishing itself from the original KDD Cup 1999 dataset, the NSL-KDD variant has been meticulously purged of redundant and duplicated records. This augmentation contributes to a more authentic and heterogeneous dataset, thus elevating its utility in intrusion detection research.

Annotated Data: Each network connection incorporated in the NSL-KDD dataset is thoughtfully categorized as either a standard connection or an instance of an attack. This categorization facilitates the application of supervised machine learning techniques.

Feature Elicitation: The dataset furnishes researchers with preprocessed feature vectors. These vectors are extrapolated from network traffic attributes like source and destination IP addresses, port numbers, protocol types, and other relevant characteristics, rendering them conducive to the creation of intrusion detection models.

Researchers harness the NSL-KDD dataset to navigate through the iterative phases of training, testing, and validating intrusion detection models, and algorithms. The overarching aim is to devise methodologies that accurately discern and classify network attacks while concurrently minimizing instances of both false positives and false negatives.

In summary, the NSL-KDD dataset assumes a pivotal role in the evolution of intrusion detection methodologies, providing a versatile platform for researchers to innovate and enhance the security paradigm within network environments.

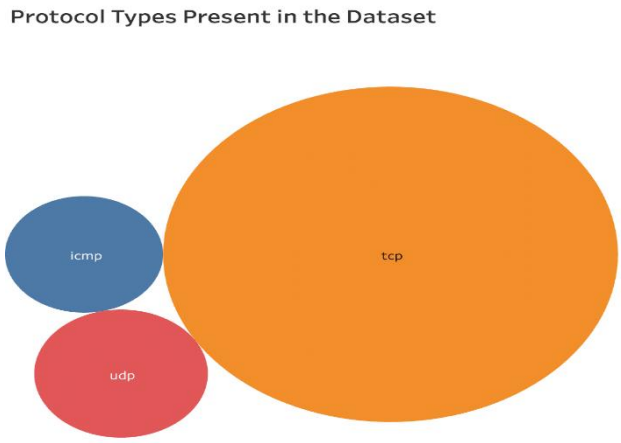

Figure 30: Figure indicating the protocols.

Chart Description

The chart titled " Protocol Types present in the NSL-KDD Dataset" offers a visual breakdown of the prevalence of TCP, ICMP, and UDP within the dataset. The chart is divided into three distinct segments, each representing one of the protocol types, with the segment size proportional to the count or frequency of each protocol within the dataset.

TCP (Transmission Control Protocol): The segment representing TCP might be the largest, indicating that this connection-oriented protocol is the most prevalent in the dataset. TCP ensures that data

packets are received in order and without errors, making it a popular choice for many applications such as web browsing and email.

ICMP (Internet Control Message Protocol): The ICMP segment may be smaller compared to TCP. This protocol is mainly used by network devices like routers to send error messages and operational information. Its presence in the dataset might be related to various network diagnostics and error reporting.

UDP (User Datagram Protocol): The UDP segment could be the smallest among the three, signifying that this connectionless protocol is less common within the dataset. UDP is known for its speed and efficiency but lacks the error-checking capabilities of TCP. It's often used for applications where speed is preferred over reliability, such as video streaming or online gaming.

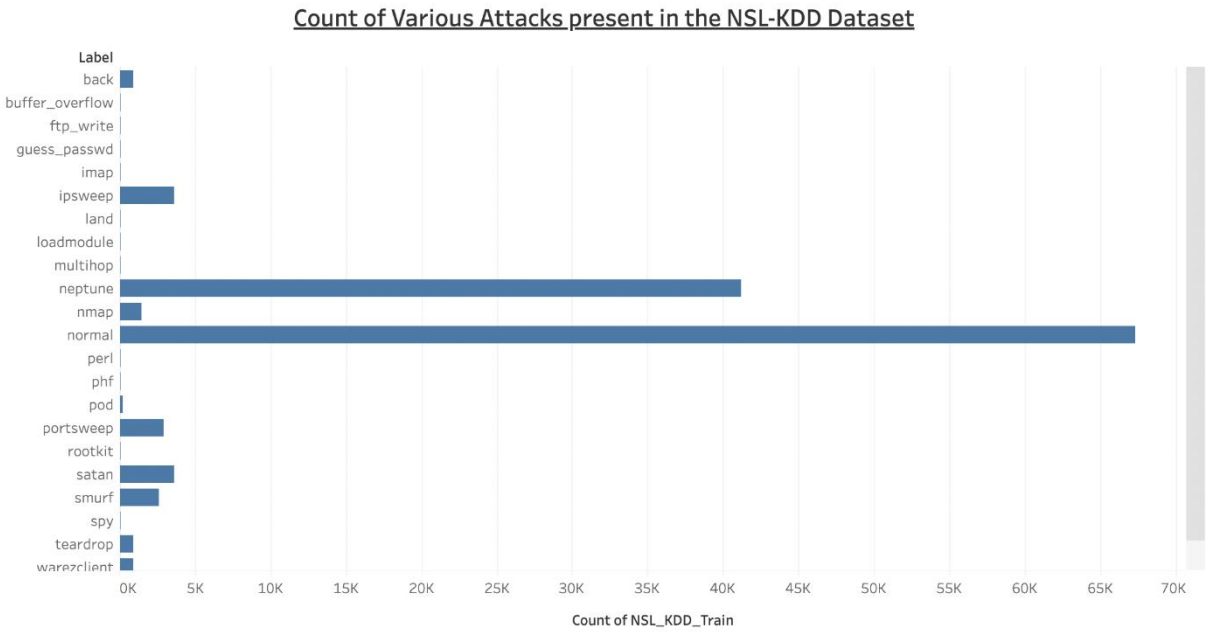

Figure 31: Count of various attacks present in the NSL-KDD Dataset

Graph Description:

The NSL-KDD dataset is a renowned data collection used extensively in network intrusion detection studies. Comprising various network connections, each entry is categorized as either an ordinary connection or an attack. Attacks are subdivided into several types including DoS (Denial of Service), R2L (Unauthorized access from a remote machine), U2R (Unauthorized access to local superuser privileges), and Probe (Surveillance and other probing). This section presents an analysis of these various attack types within the dataset and explores their distribution.

The graph titled "Count of Various Attacks Present in the NSL-KDD Dataset vs. Counts of NSL-KDD Dataset" provides an insightful visual representation of the dataset's composition. The different categories of connections, including various attack types (DoS, R2L, U2R, Probe) and normal connections, are plotted on the X-Axis. The Y-Axis depicts the count or frequency of each category within the dataset.

Attack category	Attack name
DoS	**Apache2**, Smurf, Neptune, Back, Teardrop, Pod, Land, **Mailbomb**, **Processtable**, **UDPstorm**
remote to local (R2L)	WarezClient, Guess_Password, WarezMaster, Imap, Ftp_Write, **Named**, MultiHop, Phf, Spy, **Sendmail**, **SnmpGetAttack**, **SnmpGuess**, **Worm**, **Xsnoop**, **Xlock**
user to root (U2R)	Buffer_Overflow, **Httptuneel**, Rootkit, LoadModule, Perl, **Xterm**, **Ps**, **SQLattack**
probe	Satan, **Saint**, Ipsweep, Portsweep, Nmap, **Mscan**

Figure 32: List of attacks presented in NSL-KDD dataset.[88]

Denial of Service (DoS) Attacks:

- Smurf: Spoofs victim's IP and sends ICMP echo requests to a broadcast address, causing a flood of responses to overwhelm the target system.
- Neptune: Sends a flood of TCP packets with random source addresses to exhaust the target system's resources.
- Back: Floods the target system with TCP ACK packets, disrupting its communication with other devices.
- Teardrop: Exploits IP fragmentation vulnerabilities by sending overlapping and malformed packets, causing system crashes.
- Pod (Ping of Death): Sends oversized or malformed ICMP ping packets, leading to system crashes.
- Land: Spoofs source IP and sends TCP SYN packets with the target's IP as source and destination, confusing the system and causing crashes.
- Mailbomb: Floods the victim's email inbox with a massive volume of emails, overwhelming the email server.

Remote to Local (R2L) Attacks:

- WarezClient: Software used to download copyrighted material illegally from the internet.
- Guess_Password: Repeatedly attempts various password combinations to gain unauthorized access.
- WarezMaster: Operates or manages a network distributing copyrighted material without authorization.
- Imap: Attempts to exploit vulnerabilities in the IMAP service to gain unauthorized email access.
- Ftp_Write: Gains unauthorized write access to an FTP server, enabling file manipulation.
- Named: Exploits DNS service vulnerabilities to gain unauthorized access.
- MultiHop: Uses multiple compromised systems in sequence to hide the attacker's identity.
- Phf: Exploits vulnerabilities in the "Personal Home Page Form Interpreter" to execute arbitrary commands.

- Spy: Unauthorized access to monitor and gather information without the system owner's knowledge.
- Sendmail: Exploits vulnerabilities in the Sendmail service to gain unauthorized access.
- SnmpGetAttack: Exploits SNMP vulnerabilities to gather information from the target system.
- SnmpGuess: Attempts to guess SNMP community strings to control SNMP-enabled devices.
- Worm: Self-replicating malware that spreads across systems without human intervention.
- Xsnoop: Intercepts and views network traffic, including sensitive information from other users.
- Xlock: Exploits X Window System vulnerabilities to gain unauthorized access or control.

User to Root (U2R) Attacks:

- Buffer_Overflow: Exploits software vulnerabilities to overwrite memory locations and execute arbitrary code.
- Httptuneel: Bypasses security controls to gain unauthorized access to web applications.
- Rootkit: Malicious software enabling privileged access and hiding its presence from regular security measures.
- LoadModule: Exploits the ability to load dynamic modules in a server application to execute arbitrary code.
- Perl: Uses malicious Perl scripts to exploit system vulnerabilities.
- Xterm: Exploits vulnerabilities in the Xterm terminal emulator to escalate privileges.
- Ps: Gathers information about running processes to identify potential vulnerabilities.
- SQLattack: Exploits web application input validation to manipulate the underlying database and gain unauthorized access.

Probe Attacks:

- Satan: Network vulnerability scanner used for probing systems for weaknesses.
- Saint: Network vulnerability scanner that identifies potential security issues in target systems.
- Ipsweep: Scans a range of IP addresses to identify active and responsive systems.
- Portsweep: Focuses on scanning for open ports on target systems to find potential entry points.
- Nmap: Powerful network scanning tool to discover hosts and services on a computer network.
- Mscan: Network scanner probing for open ports and vulnerabilities.

F. #	Feature name.	F. #	Feature name.	F. #	Feature name.
F1	duration	F15	Su attempted	F29	Same srv rate
F2	protocol type	F16	Num root	F30	Diff srv rate
F3	service	F17	Num file creations	F31	Srv diff host rate
F4	flag	F18	Num shells	F32	Dst host count
F5	source bytes	F19	Num access files	F33	Dst host srv count
F6	destination bytes	F20	Num outbound cmds	F34	Dst host same srv rate
F7	land	F21	Is host login	F35	Dst host diff srv rate
F8	wrong fragment	F22	Is guest login	F36	Dst host same src port rate
F9	urgent	F23	Count	F37	Dst host srv diff host rate
F10	hot	F24	Srv count	F38	Dst host serror rate
F11	number failed logins	F25	Serror rate	F39	Dst host srv serror rate
F12	logged in	F26	Srv serror rate	F40	Dst host rerror rate
F13	num compromised	F27	Rerror rate	F41	Dst host srv rerror rate
F14	root shell	F28	Srv rerror rate	F42	Class label

Figure 33: List of features of NSL-KDD Dataset.[88]

- F1 - Duration: The duration of the network connection in seconds.
- F2 - Protocol type: The type of network protocol used for the connection (e.g., TCP, UDP, ICMP).
- F3 - Service: The network service on the destination machine that was accessed (e.g., http, ftp, smtp).
- F4 - Flag: Various flags indicating the status of the connection (e.g., FIN, SYN, RST).
- F5 - Source bytes: The number of data bytes sent from the source to the destination.
- F6 - Destination bytes: The number of data bytes sent from the destination to the source.
- F7 - Land: A binary feature indicating if the connection is from/to the same host/port (1 if yes, 0 otherwise).
- F8 - Wrong fragment: A binary feature indicating the presence of a wrong fragment in the connection (1 if yes, 0 otherwise).
- F9 - Urgent: A binary feature indicating if the urgent flag is set in the connection (1 if yes, 0 otherwise).
- F10 - Hot: A binary feature indicating if the connection is flagged as "hot" (1 if yes, 0 otherwise).
- F11 - Number failed logins: The number of failed login attempts.
- F12 - Logged in: A binary feature indicating if the user is logged in (1 if yes, 0 otherwise).
- F13 - Num compromised: The number of compromised conditions.
- F14 - Root shell: A binary feature indicating if a root shell is obtained (1 if yes, 0 otherwise).
- F15 - Su attempted: A binary feature indicating if the "su" command (substitute user) was attempted (1 if yes, 0 otherwise).
- F16 - Num root: The number of root accesses.
- F17 - Num file creations: The number of file creation operations.
- F18 - Num shells: The number of shell prompts requested.
- F19 - Num access files: The number of accesses to files.

- F20 - Num outbound cmds: The number of outbound commands in an ftp session.
- F21 - Is host login: A binary feature indicating if the login belongs to the "host" category (1 if yes, 0 otherwise).
- F22 - Is guest login: A binary feature indicating if the login belongs to the "guest" category (1 if yes, 0 otherwise).
- F23 - Count: The number of connections to the same host as the current connection.
- F24 - Srv count: The number of connections to the same service as the current connection.
- F25 - Serror rate: The percentage of connections that have "SYN" errors.
- F26 - Srv serror rate: The percentage of connections to the same service that have "SYN" errors.
- F27 - Rerror rate: The percentage of connections that have "REJ" errors.
- F28 - Srv rerror rate: The percentage of connections to the same service that have "REJ" errors.
- F29 - Same srv rate: The percentage of connections to the same service.
- F30 - Diff srv rate: The percentage of connections to different services.
- F31 - Srv diff host rate: The percentage of connections to different hosts.
- F32 - Dst host count: The number of connections to the same destination host.
- F33 - Dst host srv count: The number of connections to the same service on the destination host.
- F34 - Dst host same srv rate: The percentage of connections to the same service on the destination host.
- F35 - Dst host diff srv rate: The percentage of connections to different services on the destination host.
- F36 - Dst host same src port rate: The percentage of connections from the same source port to the destination host.
- F37 - Dst host srv diff host rate: The percentage of connections to different hosts for the same service on the destination host.
- F38 - Dst host serror rate: The percentage of connections to the destination host that have "SYN" errors.
- F39 - Dst host srv serror rate: The percentage of connections to the destination host's service that have "SYN" errors.
- F40 - Dst host rerror rate: The percentage of connections to the destination host that have "REJ" errors.
- F41 - Dst host srv rerror rate: The percentage of connections to the destination host's service that have "REJ" errors.
- F42 - Class label: The type of network attack associated with each connection record (e.g., normal, DoS, Probe, U2R, R2L).

6.2 Evaluation Metrics used in NSL-KDD Dataset

When evaluating intrusion detection models on the NSL-KDD dataset, you can use various evaluation metrics to assess their performance. Here are some commonly used evaluation metrics:

Accuracy (ACC): The proportion of correctly classified instances out of the total instances. While accuracy is important, it may not be the best metric when dealing with imbalanced datasets, as it can be misleading when the classes are not evenly distributed.

Precision: Also known as the positive predictive value, precision measures the proportion of true positive predictions (correctly identified attacks) out of all instances predicted as positive (both true positives and false positives). It helps assess the model's ability to avoid false alarms.

Recall (Sensitivity or True Positive Rate): Recall measures the proportion of true positive predictions out of all actual positive instances (true positives and false negatives). It helps assess the model's ability to identify all relevant instances.

F1-Score: The harmonic mean of precision and recall. It provides a balanced measure of a model's performance by considering both false positives and false negatives. F1-score is particularly useful when there's an uneven class distribution.

Specificity (True Negative Rate): Measures the proportion of true negative predictions (correctly identified non-attacks) out of all actual negative instances (true negatives and false positives). It complements recall by focusing on the true negative rate.

ROC Curve and AUC: The Receiver Operating Characteristic (ROC) curve is a graphical representation of the true positive rate (recall) against the false positive rate as the discrimination threshold changes. The Area Under the Curve (AUC) summarizes the ROC curve's performance in a single value, where higher AUC indicates better overall performance.

Confusion Matrix: A table that summarizes the performance of a classification model by showing the counts of true positive, true negative, false positive, and false negative predictions.

Matthews Correlation Coefficient (MCC): MCC considers all four elements of the confusion matrix and provides a balanced measure of classification performance. It ranges from -1 to +1, where +1 indicates perfect prediction, 0 indicates random prediction, and -1 indicates total disagreement between prediction and observation.

Area under Precision-Recall Curve (AUC-PR): Like ROC-AUC, AUC-PR plots precision against recall and calculates the area under the curve. It is especially useful for imbalanced datasets.

Balanced Accuracy: The average of sensitivity and specificity, providing a balanced measure of classification performance.

The above are the all-possible evaluation metrics for the NSL-KDD dataset available. Among them most widely used ones are Accuracy, Precision, Recall and F1-Score. Moreover, Confusion Matrix was used by us during the implementation process, and we were able to compare the performance of Machine Learning algorithms on NSL-KDD Dataset.

Model	Attack Category	Accuracy	Precision	Recall	F1 Score
SVC	DOS	0.8582	0.8837	0.8582	0.8527
SVC	PROBE	0.8999	0.9004	0.8999	0.8901
SVC	R2L	0.7710	0.8235	0.7710	0.6714
SVC	U2R	0.9936	0.9918	0.9936	0.9912
RFC	DOS	0.8327	0.8561	0.8327	0.8260
RFC	PROBE	0.8975	0.8940	0.8975	0.8904
RFC	R2L	0.7733	0.8248	0.7733	0.6769
RFC	U2R	0.9933	0.9933	0.9933	0.9900
KNN	DOS	0.9054	0.9170	0.9054	0.9035
KNN	PROBE	0.9007	0.8996	0.9007	0.8922
KNN	R2L	0.7709	0.6926	0.7709	0.6717
KNN	U2R	0.9941	0.9933	0.9941	0.9921

Figure 34:- Model Performance Evaluation Metrics

The performance of three machine learning models, namely Support Vector Classifier (SVC), Random Forest Classifier (RFC), and K-Nearest Neighbour's (KNN), was evaluated across four different attack categories: DOS, PROBE, R2L, and U2R.

DOS Attacks: **KNN** led with an **accuracy of 90.54%,** closely followed by **SVC** with **85.82%**, and **RFC** with **83.27%**. Precision, Recall, and F1 Scores followed similar patterns across the models.

PROBE Attacks: The accuracy for the three models was **fairly close**, with **SVC** and **KNN** scoring just **above 90%,** and **RFC slightly below at 89.75%**. Similar trends were observed in the other metrics.

R2L Attacks: The models showed a decrease in performance for R2L attacks, with accuracies hovering around the 77% mark. Notably, the precision for **KNN was significantly lower at 69.26%.**

U2R Attacks: **All three models demonstrated remarkable performance** in detecting U2R attacks, with **accuracies exceeding 99%**. **KNN** slightly **outperformed** the others with **an accuracy of 99.41%.**

The results demonstrate that while the models performed consistently across different attacks, there were some variations, especially in the R2L category. KNN generally exhibited the highest performance for DOS and U2R attacks, while SVC and RFC remained competitive across the board. The insights from this evaluation could guide further refinements and selection of appropriate models for specific attack detection tasks.

7. REFERENCES

1. I. Mukhopadhyay, K. S. Gupta, D. Sen and P. Gupta, "Heuristic Intrusion Detection and Prevention System," 2015 International Conference and Workshop on Computing and Communication (IEMCON), Vancouver, BC, Canada, 2015, pp. 1-7, doi: 10.1109/IEMCON.2015.7344479.

2. Jyothsna, V.V.R.P.V., Prasad, R. and Prasad, K.M., 2011. A review of anomaly based intrusion detection systems. International Journal of Computer Applications, 28(7), pp.26-35.

3. Wang, Z. and Li, X., 2013. Intrusion prevention system design. In Proceedings of the International Conference on Information Engineering and Applications (IEA) 2012: Volume 3 (pp. 375-382). Springer London.

4. Kunal and M. Dua, "Machine Learning Approach to IDS: A Comprehensive Review," 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2019, pp. 117-121, doi: 10.1109/ICECA.2019.8822120.

5. R. Lalduhsaka, A. Khan and A. K. Roy, "Issues and Challenges in Building a Model for Intrusion Detection System," 2021 5th International Conference on Information Systems and Computer Networks (ISCON), Mathura, India, 2021, pp. 1-5, doi: 10.1109/ISCON52037.2021.9702322.

6. Hubballi, Neminath & Suryanarayanan, Vinoth. (2014). False Alarm Minimization Techniques in Signature-Based Intrusion Detection Systems: A Survey. Computer Communications. 49. 10.1016/j.comcom.2014.04.012.

7. Otoum, Y., Nayak, A. AS-IDS: Anomaly and Signature Based IDS for the Internet of Things. J Netw Syst Manage 29, 23 (2021). https://doi.org/10.1007/s10922-021-09589-6

8. Liao, H. J., Richard Lin, C. H., Lin, Y. C., & Tung, K. Y. (2013, January). Intrusion detection system: A comprehensive review. Journal of Network and Computer Applications, 36(1), 16–24. https://doi.org/10.1016/j.jnca.2012.09.004

9. Mohamed, A. B., Idris, N. B., & Shanmugum, B. (n.d.). A Brief Introduction to Intrusion Detection System. A Brief Introduction to Intrusion Detection System | SpringerLink. https://doi.org/10.1007/978-3-642-35197-6_29

10. Thakare, S., Ingle, P., & Meshram, B. (2012). IDS : Intrusion Detection System the Survey of Information Security. International Journal of Emerging Technology and Advanced Engineering Website: Www.ijetae.com, 2(8), 2250–2459. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=6f575bdb4ce1ccff7cd48a89639c4de2cb61ed5f

11. J. R. Yost, "The March of IDES: Early History of Intrusion-Detection Expert Systems," in IEEE Annals of the History of Computing, vol. 38, no. 4, pp. 42-54, Oct.-Dec. 2016, doi: 10.1109/MAHC.2015.41.

12. Uppal, H. A. M., Javed, M., & Arshad, M. (2014). An overview of intrusion detection system (IDS) along with its commonly used techniques and classifications. International Journal of Computer Science and Telecommunications, 5(2), 20-24.

13. Ashoor, Asmaa Shaker, and Sharad Gore. "Importance of intrusion detection system (IDS)." International Journal of Scientific and Engineering Research 2, no. 1 (2011): 1-4.

14. Pradhan, M., Nayak, C. K., & Pradhan, S. K. (2020). Intrusion detection system (IDS) and their types. In Securing the internet of things: Concepts, methodologies, tools, and applications (pp. 481-497). IGI Global.

15. H. Saadat, A. Aboumadi, A. Mohamed, A. Erbad and M. Guizani, "Hierarchical Federated Learning for Collaborative IDS in IoT Applications," 2021 10th Mediterranean Conference on Embedded Computing (MECO), Budva, Montenegro, 2021, pp. 1-6, doi: 10.1109/MECO52532.2021.9460304.

16. L. Wirz, A. Ketphet, N. Chiewnawintawat, R.

Tanthanathewin and S. Fugkeaw, "OWADIS: Rapid Discovery of OWASP10 Vulnerability based on Hybrid IDS," 2023 15th International Conference on Knowledge and Smart Technology (KST), Phuket, Thailand, 2023, pp. 1-6, doi: 10.1109/KST57286.2023.10086878.

17. D. D. Priya, A. Kiran and P. Purushotham, "Lightweight Intrusion Detection System(L-IDS) for the Internet of Things," 2022 International Conference on Advancements in Smart, Secure and Intelligent Computing (ASSIC), Bhubaneswar, India, 2022, pp. 1-4, doi: 10.1109/ASSIC55218.2022.10088328.

18. Bela Shah and Kinjal Dave. 2016. SIP Based Intrusion Detection System for VoIP based Applications. In Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies (ICTCS '16). Association for Computing Machinery, New York, NY, USA, Article 28, 1–5. https://doi-org.lib-ezproxy.concordia.ca/10.1145/2905055.2905086

19. Robert Bronte, Hossain Shahriar, and Hisham M. Haddad. 2016. A Signature-Based Intrusion Detection System for Web Applications based on Genetic Algorithm. In Proceedings of the 9th International Conference on Security of Information and Networks (SIN '16). Association for Computing Machinery, New York, NY, USA, 32–39. https://doi-org.lib-ezproxy.concordia.ca/10.1145/2947626.2951964

20. Shengjie Li, Zhaopeng Liu, Yue Zhang, Xiaopeng Niu, Leye Wang, and Daqing Zhang. 2019. A real-time and robust intrusion detection system with commodity wi-fi. In Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers (UbiComp/ISWC '19 Adjunct). Association for Computing Machinery, New York, NY, USA, 316–319. https://doi-org.lib-ezproxy.concordia.ca/10.1145/3341162.3343789

21. Robert Mitchell and Ing-Ray Chen. 2012. Specification based intrusion detection for unmanned aircraft systems. In Proceedings of the first ACM MobiHoc workshop on Airborne Networks and Communications (Airborne '12). Association for Computing Machinery, New York, NY, USA, 31–36. https://doi-org.lib-ezproxy.concordia.ca/10.1145/2248326.2248334

22. Ming Liu, Zhi Xue, Xianghua Xu, Changmin Zhong, and Jinjun Chen. 2018. Host-Based Intrusion Detection System with System Calls: Review and Future Trends. ACM Comput. Surv. 51, 5, Article 98 (September 2019), 36 pages. https://doi-org.lib-ezproxy.concordia.ca/10.1145/3214304

23. Safa Otoum, Burak Kantarci, and Hussein Mouftah. 2021. A Comparative Study of AI-Based Intrusion Detection Techniques in Critical Infrastructures. ACM Trans. Internet Technol. 21, 4, Article 81 (November 2021), 22 pages. https://doi-org.lib-ezproxy.concordia.ca/10.1145/3406093

24. Salam Khanji and Asad Khattak. 2021. Towards a Novel Intrusion Detection Architecture using Artificial Intelligence. In Proceedings of the 9th International Conference on Software and Information Engineering (ICSIE '20). Association for Computing Machinery, New York, NY, USA, 185–189. https://doi-org.lib-ezproxy.concordia.ca/10.1145/3436829.3436842

25. Lata, S. and Singh, D. (2022) 'Intrusion detection system in cloud environment: Literature survey & future research directions', International Journal of Information Management Data Insights, 2(2), p. 100134. doi:10.1016/j.jjimei.2022.100134.

26. Jayakumar, Naveenkumar & Angral, Sheetal & Sharma, Rohan. (2014). Integrating Intrusion Detection System with Network monitoring. International Journal of Scientific and Research Publications,. 4. 1-4.

27.S. Kumar, S. Gupta and S. Arora, "Research Trends in Network-Based Intrusion Detection Systems: A Review," in IEEE Access, vol. 9, pp. 157761-157779, 2021, doi: 10.1109/ACCESS.2021.3129775.

28. Z. A. Haddad, M. Hanoune and A. Mamouni, "A collaborative framework for intrusion detection (C-NIDS) in Cloud computing," 2016

2nd International Conference on Cloud Computing Technologies and Applications (CloudTech), Marrakech, Morocco, 2016, pp. 261-265, doi: 10.1109/CloudTech.2016.7847708.

29. Y. Mehmood, M. A. Shibli, U. Habiba and R. Masood, "Intrusion Detection System in Cloud Computing: Challenges and opportunities," 2013 2nd National Conference on Information Assurance (NCIA), Rawalpindi, Pakistan, 2013, pp. 59-66, doi: 10.1109/NCIA.2013.6725325.

30. C. N. Modi and D. Patel, "A novel hybrid-network intrusion detection system (H-NIDS) in cloud computing," 2013 IEEE Symposium on Computational Intelligence in Cyber Security (CICS), Singapore, 2013, pp. 23-30, doi: 10.1109/CICYBS.2013.6597201.

31. Xinzhou Qin, Wenke Lee, L. Lewis and J. B. D. Cabrera, "Integrating intrusion detection and network management," NOMS 2002. IEEE/IFIP Network Operations and Management Symposium. 'Management Solutions for the New Communications World'(Cat. No.02CH37327), Florence, Italy, 2002, pp. 329-344, doi: 10.1109/NOMS.2002.1015591.

32. A. Abubakar and B. Pranggono, "Machine learning based intrusion detection system for software defined networks," 2017 Seventh International Conference on Emerging Security Technologies (EST), Canterbury, UK, 2017, pp. 138-143, doi: 10.1109/EST.2017.8090413.

33. S. Bose, S. Bharathimurugan and A. Kannan, "Multi-Layer Integrated Anomaly Intrusion Detection System for Mobile Adhoc Networks," 2007 International Conference on Signal Processing, Communications and Networking, Chennai, India, 2007, pp. 360-365, doi: 10.1109/ICSCN.2007.350763.

34. Xin-You Zhang, Cheng-Zhong Li and Qing-Gui Hu, "The network management design integrated with the intrusion detection system," Proceedings of 2004 International Conference on Machine Learning and Cybernetics (IEEE Cat. No.04EX826), Shanghai, China, 2004, pp. 257-262 vol.1, doi: 10.1109/ICMLC.2004.1380672.

35. H. Mohamed, L. Adil, T. Saida and M. Hicham, "A collaborative intrusion detection and Prevention System in Cloud Computing," 2013 Africon, Pointe aux Piments, Mauritius, 2013, pp. 1-5, doi: 10.1109/AFRCON.2013.6757727.

36. R. Zhao, Y. Mu, L. Zou and X. Wen, "A Hybrid Intrusion Detection System Based on Feature Selection and Weighted Stacking Classifier," in IEEE Access, vol. 10, pp. 71414-71426, 2022, doi: 10.1109/ACCESS.2022.3186975.

37.Kumar, Amit, Harish Chandra Maurya, and Rahul Misra. "A research paper on hybrid intrusion detection system." International Journal of Engineering and Advanced Technology (IJEAT) Vol 2 (2013).

38.Grimmer, Martin, et al. "A modern and sophisticated host based intrusion detection data set." IT-Sicherheit als Voraussetzung für eine erfolgreiche Digitalisierung 11 (2019): 135-145.

39. Wasicek, Armin, et al. "Context-aware intrusion detection in automotive control systems." Proc. 5th ESCAR USA Conf. 2017.

40. Hu, Yan, et al. "A survey of intrusion detection on industrial control systems." International Journal of Distributed Sensor Networks 14.8 (2018): 1550147718794615.

41. Yang, Kai, et al. "iFinger: Intrusion detection in industrial control systems via register-based fingerprinting." IEEE Journal on Selected Areas in Communications 38.5 (2020): 955-967.

42. Singh, Amrit Pal, and Manik Deep Singh. "Analysis of host-based and network-based intrusion detection system." International Journal of Computer Network and Information Security 6.8 (2014): 41-47.

43. Gao, Wei, and Thomas H. Morris. "On cyber attacks and signature based intrusion detection for modbus based industrial control systems." Journal of Digital Forensics, Security and Law 9.1 (2014): 3.

44. Zhu, Bonnie, and Shankar Sastry. "SCADA-specific intrusion detection/prevention systems: a survey and taxonomy." Proceedings of the 1st

workshop on secure control systems (SCS). Vol. 11. 2010.

45. Caselli, Marco, Emmanuele Zambon, and Frank Kargl. "Sequence-aware intrusion detection in industrial control systems." Proceedings of the 1st ACM Workshop on Cyber-Physical System Security. 2015.

46. Agarwal, Nancy, and Syed Zeeshan Hussain. "A closer look at intrusion detection system for web applications." Security and Communication Networks 2018 (2018).

47. K. Talty, J. Stockdale and N. D. Bastian, "A Sensitivity Analysis of Poisoning and Evasion Attacks in Network Intrusion Detection System Machine Learning Models," MILCOM 2021 - 2021 IEEE Military Communications Conference (MILCOM), San Diego, CA, USA, 2021, pp. 1011-1016, doi: 10.1109/MILCOM52596.2021.9652959.

48. J. Ashraf, N. Moustafa, A. D. Bukhshi and A. Javed, "Intrusion Detection System for SDN-enabled IoT Networks using Machine Learning Techniques," 2021 IEEE 25th International Enterprise Distributed Object Computing Workshop (EDOCW), Gold Coast, Australia, 2021, pp. 46-52, doi: 10.1109/EDOCW52865.2021.00031.

49. E. Viegas, A. O. Santin and V. Abreu Jr, "Machine Learning Intrusion Detection in Big Data Era: A Multi-Objective Approach for Longer Model Lifespans," in IEEE Transactions on Network Science and Engineering, vol. 8, no. 1, pp. 366-376, 1 Jan.-March 2021, doi: 10.1109/TNSE.2020.3038618.

50. S. Thirimanne, L. Jayawardana, P. Liyanaarachchi and L. Yasakethu, "Comparative Algorithm Analysis for Machine Learning Based Intrusion Detection System," 2021 10th International Conference on Information and Automation for Sustainability (ICIAfS), Negambo, Sri Lanka, 2021, pp. 191-196, doi: 10.1109/ICIAfS52090.2021.9605814.

51. Mendoza, Arjonel M., Rowell M. Hernandez, Ryndel V. Amorado, Myrna A. Coliat and Poul Isaac C. De Chavez. “Network Data Feature Selection in Detecting Network Intrusion using Supervised Machine Learning Techniques.” 2022 2nd Asian Conference on Innovation in Technology (ASIANCON) (2022): 1-6.

52. Zang, M., Yan, Y. and 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring) Helsinki, Finland 2021 April 25 - 2021 April 28 (2021) “2021 Ieee 93rd Vehicular Technology Conference (vtc2021-Spring),” in Machine Learning-Based Intrusion Detection System for Big Data Analytics in Vanet. IEEE, pp. 1–5. doi: 10.1109/VTC2021-Spring51267.2021.9448878.

53. Shi, J. et al. (2021) “2021 Ieee 94th Vehicular Technology Conference (vtc2021-Fall),” in A Hybrid Intrusion Detection System Based on Machine Learning Under Differential Privacy Protection. IEEE, pp. 1–6. doi: 10.1109/VTC2021-Fall52928.2021.9625540.

54. J. Gao, S. Chai, C. Zhang, B. Zhang and L. Cui, "A Novel Intrusion Detection System based on Extreme Machine Learning and Multi-Voting Technology," 2019 Chinese Control Conference (CCC), Guangzhou, China, 2019, pp. 8909-8914, doi: 10.23919/ChiCC.2019.8865258.

55. T. N. Varunram, M. B. Shivaprasad, K. H. Aishwarya, A. Balraj, S. V. Savish and S. Ullas, "Analysis of Different Dimensionality Reduction Techniques and Machine Learning Algorithms for an Intrusion Detection System," 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), Arad, Romania, 2021, pp. 237-242, doi: 10.1109/ICCCA52192.2021.9666265.

56. Gustavo De Carvalho Bertoli et al. (2021) “An End-To-End Framework for Machine Learning-Based Network Intrusion Detection System,” 9, pp. 106790–106805. doi: 10.1109/ACCESS.2021.3101188.

57. C. Yin, Y. Zhu, J. Fei and X. He, "A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks," in IEEE Access, vol. 5, pp. 21954-21961, 2017, doi: 10.1109/ACCESS.2017.2762418.

58. X. Gao, C. Shan, C. Hu, Z. Niu and Z. Liu, "An Adaptive Ensemble Machine Learning Model for Intrusion Detection," in IEEE Access, vol. 7, pp. 82512-82521, 2019, doi:

10.1109/ACCESS.2019.2923640.

59. R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat and S. Venkatraman, "Deep Learning Approach for Intelligent Intrusion Detection System," in IEEE Access, vol. 7, pp. 41525-41550, 2019, doi: 10.1109/ACCESS.2019.2895334.

60. Y. Xin et al., "Machine Learning and Deep Learning Methods for Cybersecurity," in IEEE Access, vol. 6, pp. 35365-35381, 2018, doi: 10.1109/ACCESS.2018.2836950.

61. Z. Wang, "Deep Learning-Based Intrusion Detection With Adversaries," in IEEE Access, vol. 6, pp. 38367-38384, 2018, doi: 10.1109/ACCESS.2018.2854599.

62. M. Al-Qatf, Y. Lasheng, M. Al-Habib and K. Al-Sabahi, "Deep Learning Approach Combining Sparse Autoencoder With SVM for Network Intrusion Detection," in IEEE Access, vol. 6, pp. 52843-52856, 2018, doi: 10.1109/ACCESS.2018.2869577.

63. M. D. Rokade and Y. K. Sharma, "MLIDS: A Machine Learning Approach for Intrusion Detection for Real-Time Network Dataset," 2021 International Conference on Emerging Smart Computing and Informatics (ESCI), Pune, India, 2021, pp. 533-536, doi: 10.1109/ESCI50559.2021.9396829.

64. Karuna S. Bhosale, Maria Nenova and Georgi Iliev, "Modified Naive Bayes Intrusion Detection System (MNBIDS)", 2018 International Conference on Computational Techniques Electronics and Mechanical Systems (CTEMS), 2018.

65. Dimitra Chamou et al., "Intrusion Detection System Based on Network Traffic Using Deep Neural Networks", 2019 IEEE 24th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), 2019.

66. Yogendra Kumar Jain and Upendra, “An Efficient Intrusion Detection Based on Decision Tree Classifier Using Feature Reduction”, International Journal of Engineering Research and Development e-ISSN: 2278-067X, p-ISSN: 2278-800X, www.ijerd.com Volume 8, Issue 2 (August 2013), PP. 23-38.

67. Siriporn Chimphlee and Witcha Chimphlee, "Machine learning to improve the performance of anomaly-based network intrusion detection in big data",Indonesian Journal of Electrical Engineering and Computer Science Vol. 30, No. 2, May 2023, pp. 1106~1119 ISSN: 2502-4752, DOI: 10.11591/ijeecs.v30.i2.pp1106-1119.

68. P. Raviteja et al., “Implementation Of Machine Learning Algorithms For Detection Of Network Intrusion,” vol. 8, no. 2, pp. 163–169, 2020.

69. J. A. Abraham and V. R. Bindu, "Intrusion Detection and Prevention in Networks Using Machine Learning and Deep Learning Approaches: A Review," 2021 International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA), Coimbatore, India, 2021, pp. 1-4, doi: 10.1109/ICAECA52838.2021.9675595.

70. Alex Shenfield, David Day, Aladdin Ayesh, Intelligent intrusion detection systems using artificial neural networks, ICT Express, Volume 4, Issue 2, 2018, Pages 95-99,ISSN 2405-9595, https://doi.org/10.1016/j.icte.2018.04.003.

71. Mukkamala S, Janoski G, Sung A. Intrusion detection using neural networks and support vector machines. Paper presented at: Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN’02 (Cat. No. 02CH37290). Honolulu, HI, USA: IEEE; vol. 2, 2002:1702-1707.

72. H. Du, S. Teng, M. Yang and Q. Zhu, "Intrusion Detection System Based on Improved SVM Incremental Learning," 2009 International Conference on Artificial Intelligence and Computational Intelligence, Shanghai, China, 2009, pp. 23-28, doi: 10.1109/AICI.2009.254.

73. Li Xiangmei Qin Zhi “The Application of Hybrid Neural Network Algorithms in Intrusion Detection System “978-1-4244-8694-6/11 ©2011 IEEE.

74. R. Patgiri, U. Varshney, T. Akutota and R. Kunde, "An Investigation on Intrusion Detection System Using Machine Learning," 2018 IEEE Symposium Series on Computational Intelligence (SSCI), Bangalore, India,

2018, pp. 1684-1691, doi: 10.1109/SSCI.2018.8628676.

75. U. S. K. Perera Miriya Thanthrige, J. Samarabandu and X. Wang, "Machine learning techniques for intrusion detection on public dataset," 2016 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), Vancouver, BC, Canada, 2016, pp. 1-4, doi: 10.1109/CCECE.2016.7726677.

76. K. Alrawashdeh and C. Purdy, "Toward an Online Anomaly Intrusion Detection System Based on Deep Learning," 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), Anaheim, CA, USA, 2016, pp. 195-200, doi: 10.1109/ICMLA.2016.0040.

77. U. S. K. P. M. Thanthrige, J. Samarabandu and X. Wang, "Machine learning techniques for intrusion detection on public dataset", 2016 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), pp. 1-4, May 2016.

78. M. J. Fadaeieslam, B. Minaei-Bidgoli, M. Fathy and M. Soryani, "Comparison of two feature selection methods in intrusion detection systems", Computer and Information Technology 2007. CIT 2007. 7th IEEE International Conference on, pp. 83-86, 2007.

79. E. Guillen, D. Padilla and Y. Colorado, "Weaknesses and strengths analysis over network-based intrusion detection and prevention systems," 2009 IEEE Latin-American Conference on Communications, Medellin, Colombia, 2009, pp. 1-5, doi: 10.1109/LATINCOM.2009.5305047.

80. M. Ahmed, R. Pal, M. M. Hossain, M. A. N. Bikas and M. K. Hasan, "NIDS: A Network Based Approach to Intrusion Detection and Prevention," 2009 International Association of Computer Science and Information Technology - Spring Conference, Singapore, 2009, pp. 141-144, doi: 10.1109/IACSIT-SC.2009.96.

81. S. Liu, D. Y. Zhang, X. Chu, H. Otrok and P. Bhattacharya, "A Game Theoretic Approach to Optimize the Performance of Host-Based IDS," 2008 IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, Avignon, France, 2008, pp. 448-453, doi: 10.1109/WiMob.2008.20.

82. O. Al-Jarrah and A. Arafat, "Network Intrusion Detection System using attack behavior classification," 2014 5th International Conference on Information and Communication Systems (ICICS), Irbid, Jordan, 2014, pp. 1-6, doi: 10.1109/IACS.2014.6841978.

83. A. Efe and İ. N. Abacı , "Comparison of the Host Based Intrusion Detection Systems and Network Based Intrusion Detection Systems", Celal Bayar University Journal of Science, vol. 18, no. 1, pp. 23-32, Mar. 2022, doi:10.18466/cbayarfbe.832533

84. Wang, Z.Q., Zhang, D.K., 2012. HIDS and NIDS Hybrid Intrusion Detection System Model Design. AEF 6–7, 991–994. https://doi.org/10.4028/www.scientific.net/aef.6-7.991 (https://doi.org/10.4028/www.scientific.net/aef.6-7.991)

85. Y. -j. Ou, Y. Lin, Y. Zhang and Y. -j. Ou, "The Design and Implementation of Host-Based Intrusion Detection System," 2010 Third International Symposium on Intelligent Information Technology and Security Informatics, Jian, China, 2010, pp. 595-598, doi: 10.1109/IITSI.2010.127.

86. Nguyen Thanh Van, Tran Ngoc Thinh and Le Thanh Sach, "An anomaly-based network intrusion detection system using Deep learning," 2017 International Conference on System Science and Engineering (ICSSE), Ho Chi Minh City, Vietnam, 2017, pp. 210-214, doi: 10.1109/ICSSE.2017.8030867.

87. Khraisat, Ansam & Gondal, Iqbal & Vamplew, Peter & Kamruzzaman, Joarder. (2019). Survey of intrusion detection systems: techniques, datasets and challenges. Cybersecurity. 2. 10.1186/s42400-019-0038-7.

88. Latah, Majd & Toker, Levent. (2018). Towards an Efficient Anomaly-Based Intrusion Detection for Software-Defined Networks. IET Networks. 7. 10.1049/iet-net.2018.5080