Topic 6 - Intrusion Detection Systems (IDS)

Master's in Segurança Informática

3rd November 2024

Diogo Coelho, up202107596

Leandro Roque Costa, up202408816

Rafael Castilho Silva, up202409620

Grup 7 - TP2

Contents

Introduction

This report provides an overview of Intrusion Detection Systems (IDS), focusing on their crucial role in identifying and mitigating cyber threats. It examines key IDS methods, including signature-based, anomaly-based, and machine learning approaches, along with trade-offs such as accuracy versus scalability. Real-world examples illustrate IDS effectiveness in practice.

The research relied on a targeted review of relevant studies from IEEE Xplore, limited to peer-reviewed articles published between 2006 and 2024, using specific search terms for IDS methodologies, performance metrics, and emerging technologies.

With today's rapidly expanding network environments, cyber threats and unauthorized access risks are rising, necessitating sophisticated security measures to safeguard data integrity and confidentiality. Traditional security methods like firewalls and antivirus software provide essential protection but often fall short in detecting complex threats like Denial of Service (DoS) attacks, which require more advanced defenses.

IDS enhance security by continuously monitoring system and network activity for suspicious behavior, alerting administrators to potential threats. According to the National Institute of Standards and Technology (NIST), IDS monitor for violations of security policies like firewall settings and authentication protocols. They can be deployed as Host-based (HIDS), Network-based (NIDS), or Hybrid systems, each offering a unique perspective on monitoring host-level and network-level threats.

This report analyzes IDS techniques, covering methodologies like signature-based and anomaly-based detection, and recent advancements including artificial intelligence, machine learning, and extended Berkeley Packet Filter (eBPF) technologies. Emerging trends, such as autonomous AI-driven detection and federated learning, are also discussed as they reshape real-time threat detection and response. Key design challenges, such as balancing detection accuracy with scalability and managing false positives, are explored to provide insight into practical IDS deployment.

1 Introduction to Intrusion Detection Systems (IDS)

1.1 Purpose and Importance of IDS

Intrusion Detection Systems (IDS) are essential in network security, continuously monitoring and analyzing network traffic to detect unauthorized access, policy violations, and other threats. By **identifying malicious patterns**, signatures, and anomalies, IDS help protect digital assets and maintain information confidentiality, integrity, and availability.

As cyber threats become more complex, IDS have become critical for alerting security teams early, minimizing damage, and preserving system trust. They play a vital role in a defense-in-depth strategy, enhancing the effectiveness of firewalls, antivirus solutions, and other security measures by proactively identifying potential threats.

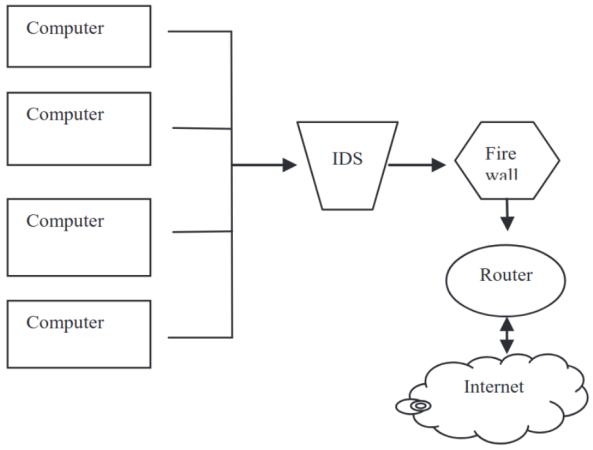

Figure 1: Intrusion Detection Systems

1.2 Types of IDS

Intrusion Detection Systems come in various forms, each designed to monitor specific activities across networked environments. The three primary types include Network-based IDS, Host-based IDS, and Hybrid/Distributed IDS.

1.2.1 Network-based IDS (NIDS)

A Network-based IDS (NIDS) monitors network traffic to detect unauthorized access, malware, and data leaks in real-time.

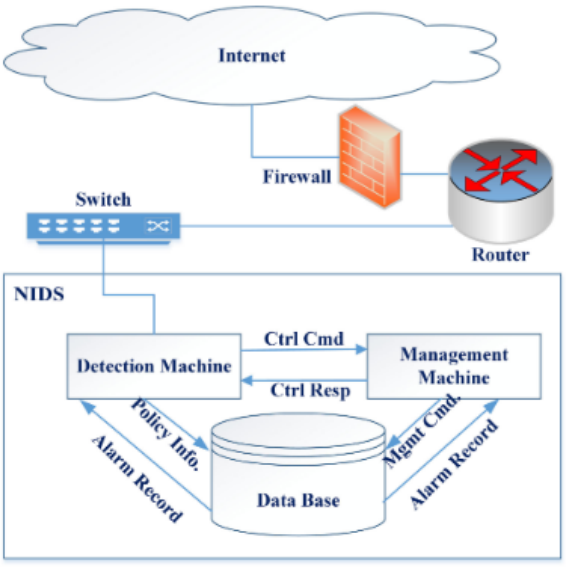

Figure 2: Network-based Intrusion Detection System (NIDS) components

According to *Research Trends in Network-Based Intrusion Detection Systems: A Review* [9], NIDS includes three components: the Detection Machine for identifying intrusions, the Management Machine for policy control, and the Database for logging.

NIDS faces fidelity, resource, and reliability challenges [6]. Figure 1 shows NIDS components and phases.

Common intrusion causes include high false positives from undetected bad packets, encrypted packets evading detection, weak authentication, and protocol-based attacks [1, 4].

1.2.2 Host-based IDS (HIDS)

A Host-based IDS (HIDS) is installed on individual systems to monitor local activities like file access, user logins, and processes, making it essential for device-level protection.

According to *A Systematic Literature Review on Host-Based Intrusion Detection Systems* [15], HIDS focuses on a single host to detect malicious activity, unlike Network-based IDS (NIDS) that monitors network-wide traffic.

HIDS functions by monitoring system calls, DLLs, logs, and registry keys, using either signature-based or anomaly-based detection. Signature-based HIDS match known attack patterns, while anomaly-based HIDS use statistical or machine learning models trained on normal behavior to identify deviations, making them effective against unknown threats.

Anomaly-based HIDS is particularly strong against zero-day attacks by identifying

behaviors outside typical profiles and alerting administrators.

1.2.3 Hybrid/Distributed IDS

A **Hybrid or Distributed IDS** integrates Network-based Intrusion Detection Systems (NIDS) and Host-based Intrusion Detection Systems (HIDS) to provide comprehensive network security by monitoring both network traffic and host activities. NIDS captures broader network data, while HIDS offers detailed insights into specific host behaviors such as file access and system calls.

In a Hybrid IDS, components are distributed across various nodes, servers, and devices, enabling collaboration and data aggregation to present a unified view of security incidents. This system is particularly effective in complex environments like corporate networks, cloud infrastructures, and industrial control systems, as it can detect sophisticated multi-stage attacks, including lateral movement and data exfiltration.

Additionally, Hybrid IDS solutions often employ real-time data analysis and alerting through centralized management, enhancing scalability and resilience across diverse architectures, including remote and IoT environments. The collaborative nature of Hybrid IDS allows for faster incident response and threat mitigation, as alerts from different sensors can be cross-referenced for a more accurate threat landscape.

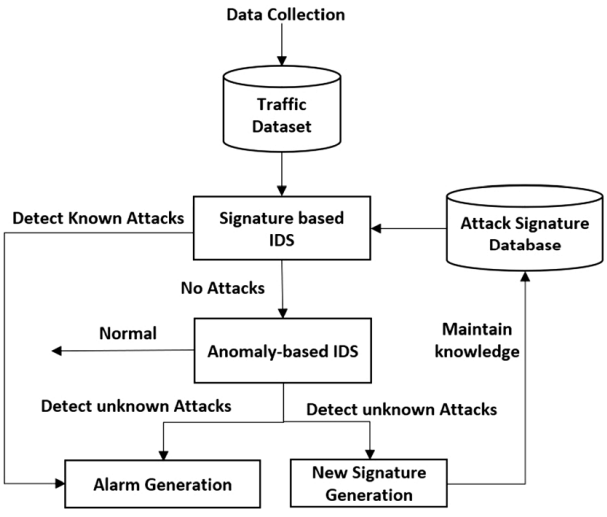

Figure 3: Hybrid-based intrusion detection system

Hybrid/Distributed IDS represents an evolution towards more **adaptive and proactive security postures**, capable of addressing advanced threats in real time and reducing

the number of false positives by providing context-rich analysis of both network and host data. This adaptability is increasingly essential in modern security environments where sophisticated attack techniques continually evolve, underscoring the importance of hybrid solutions that align with the complexity and scale of contemporary digital infrastructures.

2 Related Work - IDS Techniques and Approaches

2.1 Signature-Based Detection

As described in *Review on Signature-Based Detection for Network Threats* [15], a **signature** is a short code snippet representing a unique feature of a virus. There are three types of **feature codes**: **single**, **multiple**, and **compound** feature codes. The signature-based detection process includes two main stages:

1. **Preparations**: - **Sample collection**: Gather malicious code samples, including different file types (e.g., EXE, COM). - **Feature extraction**: Extract unique codes from each sample to identify malware accurately. - **Database storage**: Store extracted signatures efficiently to support fast lookups despite the large volume of data.

2. **Scanning and Decision-Making**: - **Code Preprocessing**: Prepare the code for scanning through type analysis, decompression, unpacking, and syntax analysis. - **Scanning**: Sequentially match feature codes from the target program with those in the database; if a match is found, the program is flagged as infected.

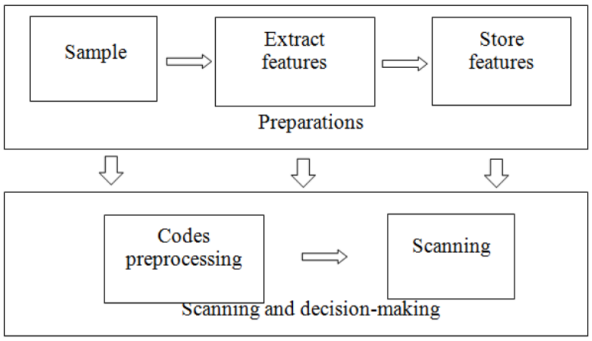

Figure 4: Procedure of signature-based detection

In summary, signature-based detection (see Fig. 3) begins with sample collection and signature extraction, followed by database storage and scanning, where matching algorithms play a key role in efficient detection.

2.2 Anomaly-Based Detection

According to *An overview of neural networks use in anomaly Intrusion Detection Systems* [14], anomaly-based detection monitors typical user or system behaviors, flagging deviations as potentially suspicious. Common techniques include threshold detection (e.g., flagging high CPU usage), statistical analysis (establishing norms from historical data), rule-based methods, and adaptive approaches like neural networks and genetic algorithms. These techniques, however, face the challenge of defining "normal" behavior accurately; insufficient differentiation between normal and abnormal activities can reduce effectiveness.

To address this, Sani et al. developed an offline anomaly detection system using a Multi-Layer Perceptron (MLP) neural network to profile user behaviors, identifying intrusions at the end of each session. Another approach focuses on program behavior profiles, analyzing standard operations of specific programs and detecting deviations to identify anomalies.

2.3 Specification-Based Detection

In the study titled *Specification-Based Intrusion Detection in WLANs*[5], the authors developed an IDS based on network protocol state models and site-specific security policies, enabling detection of deviations from expected behavior. This framework was shown to outperform other IDS techniques, effectively identifying both known and novel attacks.

The IDS's reliance on predetermined specifications helps reduce false positives and improves accuracy, especially in dynamic environments like WLANs, where anomaly-based systems may falter due to user variability.

The authors emphasize the need for continuous updates to adapt to new attack methods, highlighting the IDS's potential in proactively securing WLANs through precise behavior modeling.

2.4 Machine Learning and AI-Based Detection

According to *Comparison of Machine Learning and Deep Learning Models for Detecting Cyberbullying* [10], Artificial Intelligence (AI) algorithms analyze large datasets to detect

threats by identifying patterns and anomalies. Supervised learning uses labeled data to classify normal and malicious activities, while unsupervised learning detects deviations without predefined labels.

AI enhances traditional threat detection by enabling real-time monitoring and response through Intrusion Detection and Prevention Systems. Anomaly detection techniques in AI flag unusual behavior as potential threats, helping reduce false positives and improve response times.

AI's ability to learn from past data allows it to anticipate future threats, making it valuable for proactive threat-hunting in rapidly evolving cyber environments [10]. Effective AI-based threat detection requires seamless integration with existing cybersecurity frameworks.

2.5 Emerging Technologies

2.5.1 eBPF (Extended Berkeley Packet Filter)

According to *Enhancing Container Security with Per-Process Per-Container Egress Packet Filtering Using eBPF* [13]. The extended Berkeley Packet Filter (eBPF) acts as an in-kernel virtual machine, enabling developers to attach custom code to various points within the Linux kernel. This real-time data processing and analysis capability has made eBPF highly valuable for applications in networking, security, and performance monitoring. Programs written with eBPF integrate into different kernel subsystems, allowing for versatile network traffic monitoring and filtering.[9]

Bertrone et al. (2018) showcased eBPF's efficiency by demonstrating how it can replicate the filtering functionality of Linux's iptables, enabling more efficient network traffic management. Companies like Netflix have also adopted eBPF to monitor network flows, highlighting its adaptability for detailed traffic analysis. In cloud-native environments, Cilium leverages eBPF to filter traffic by pod or container, enhancing both security and performance. Expanding its applications, Fournier (2020) explored process-level filtering through eBPF for more granular packet control, particularly on outgoing traffic, effectively mitigating risks like ICMP tunneling.[9]

2.5.2 Blockchain-Based IDS

The integration of blockchain technology into Intrusion Detection Systems (IDS) is an emerging trend that significantly enhances security measures in network environments. By utilizing a decentralized ledger, blockchain ensures the integrity and immutability of data exchanged between nodes, which is crucial for collaborative IDS frameworks. [18].

In these systems, each node can securely share threat intelligence, reducing response times and improving the accuracy of intrusion detection. Research indicates that blockchain enhances transparency and accountability, allowing for better tracking of data access and modifications.

Moreover, employing blockchain helps to mitigate risks associated with single points of failure, which are common in traditional IDS setups. The collaborative nature of blockchain-based IDS promotes resilience against attacks, making them more robust in the face of sophisticated threats [18].

2.5.3 Federated Learning

Federated Learning is an iterative approach for improving a machine learning (ML) or deep learning (DL) model by distributing training across multiple clients. At the start of each round, an FL server sends its global model to selected clients, who then train it locally using their data. Clients send back the updated parameters, which the server aggregates to form a refined global model. This cycle repeats until optimal performance is reached, enabling data sharing without centralizing personally identifiable information.

In the study *Federated Learning Approach for Tracking Malicious Activities in Cyber-Physical Systems* [11] using an auto-encoder provides continuous on-device training for anomaly detection. This low-cost method tailors local learners for each normal pattern, effectively improving anomaly detection accuracy across varied data sources.

3 Tradeoffs and Challenges

3.1 Performance Metrics

In evaluating Intrusion Detection Systems, key performance metrics include True Positives, True Negatives, False Positives, and False Negatives. These values are essential for

assessing the system's accuracy in distinguishing between attack and non-attack records.

- **True Positive (TP)**: True Positive values represent instances where the IDS correctly identifies attack records as attacks. A high TP rate indicates effective detection capabilities, essential for minimizing security risks. The True Positive Rate (TPR), also known as *Recall* or *Sensitivity*, can be calculated as:

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

- **True Negative (TN)**: True Negative values occur when the IDS accurately identifies non-attack records as benign. This measure helps minimize unnecessary alerts, which can reduce the workload for security teams. The True Negative Rate (TNR), also known as *Specificity*, is calculated as:

$$\text{TNR} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

- **False Positive (FP)**: False Positive values indicate non-attack records that are incorrectly flagged as attacks. High FP rates can lead to "alert fatigue," where security personnel may overlook actual threats due to frequent false alarms. The False Positive Rate (FPR) is calculated as:

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

- **False Negative (FN)**: False Negative values represent instances where the IDS fails to detect actual attacks, misclassifying them as non-attacks. High FN rates are particularly concerning as they allow true threats to go undetected. The False Negative Rate (FNR) can be calculated as:

$$\text{FNR} = \frac{\text{FN}}{\text{TP} + \text{FN}}$$

These metrics form the basis of IDS performance evaluation, with additional metrics such as *Accuracy*, *Precision*, and the *F1 Score* providing further insights:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$
$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$
$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

These metrics allow for a comprehensive assessment of IDS performance, particularly in distinguishing between legitimate and malicious activities.

The balance between performance and accuracy is a key challenge in Intrusion Detection Systems (IDS). High accuracy often requires complex, resource-intensive algorithms, like deep learning models, which can capture subtle patterns in network traffic and user behavior. However, these complex models tend to slow down system performance, especially in real-time applications where rapid response is critical. On the other hand, simpler algorithms may run faster but sacrifice the depth needed to detect sophisticated threats, which compromises accuracy. Therefore, IDS designers often need to balance between lightweight, faster models and complex, accurate models to meet both real-time detection and resource efficiency needs[12].

3.2 False Positives vs. False Negatives

According to *Two-stage process based on data mining and optimization to identify false positives and false negatives generated by intrusion detection systems*[3], to secure information systems and prevent hackers from compromising them, efficient security technologies are essential. Intrusion Detection Systems (IDS) serve as critical defense layers, with their reliability hinging on accurately detecting threats. However, IDS often generate numerous false positives (FPs) and false negatives (FNs).

FPs are benign network traffic incorrectly flagged as threats, wasting analysts' time, while FNs indicate real intrusions that go undetected, risking network security. We propose a two-stage process that utilizes alerts from multiple IDSs. The first stage employs a k-means clustering algorithm to reduce alert volume.

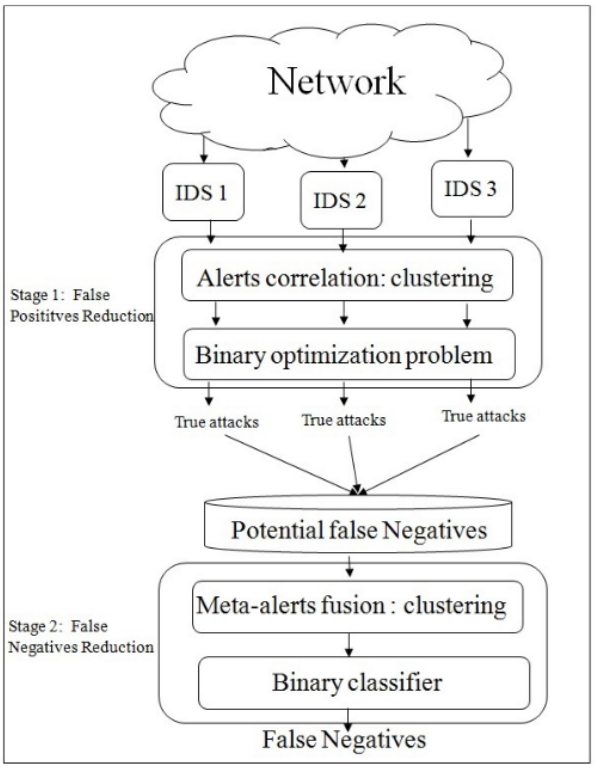

Figure 5: Proposed method for FP and FN reduction

To effectively identify false positives (FPs) and false negatives (FNs) in Intrusion Detection Systems (IDS), a prediction model is essential for classifying alerts.

Given the expertise needed to distinguish alerts accurately, a labeled training dataset is used to construct this model. This dataset is divided into two clusters: one for false alerts and the other for real attacks.

Within each cluster, FPs related to the same event are grouped via the k-means algorithm, as one event may produce multiple signatures. IP addresses identify the attacker in the network, while timestamps define the alert's occurrence window.

3.3 Scalability and Adaptability

As network sizes and data volumes grow, IDS scalability becomes essential to maintain performance across expansive infrastructures. Scalable IDSs often distribute detection across multiple nodes or use hybrid architectures, combining network-based and host-based monitoring to allocate resources where they are most needed. However, achieving this scalability introduces challenges in maintaining consistency and accuracy across a distributed system, especially in real-time.

Adaptability is also crucial for IDS to keep pace with evolving cyber threats. Adaptive systems adjust detection thresholds and incorporate new data patterns to identify novel attacks. However, frequent updates can strain system resources, increase costs, and, if not managed carefully, introduce vulnerabilities. Adaptive models, especially those using AI, require ongoing training and fine-tuning to ensure effective and stable performance[12].

3.4 Privacy Concerns

Privacy is a growing concern in IDS, particularly with the use of machine learning and data-intensive approaches. Collecting and analyzing network data often involves monitoring potentially sensitive information, raising questions about data protection and user privacy. To address this, techniques like federated learning are increasingly used to analyze data without centralized data storage, preserving privacy while still enabling accurate intrusion detection. Nevertheless, implementing privacy-preserving techniques can reduce the IDS's overall efficacy if crucial data for threat detection is restricted[12].

4 Evaluation of IDS Tools and Frameworks

4.1 Open-Source and Commercial IDS

4.1.1 Open-Source

Research highlights open-source IDS options (e.g., Snort[17] and Zeek) are highly valued for their flexibility, low-cost access, and active community support. Open-source IDSs often use a mix of signature-based detection (matching known attack patterns) and anomaly-based detection (identifying deviations from normal behavior), though their performance can vary based on system configurations and network size[9].

Advantages: Open-source IDS tools are generally highly configurable, adaptable, and scalable within large, diverse network environments. They are free to use and allow for extensive customization, making them ideal for organizations that need tailored security solutions or have dedicated IT teams.

Challenges: Open-source systems may lack comprehensive customer support, require manual setup, and have a steeper learning curve. For example, while Snort is powerful,

it can produce high false-positive rates if not fine-tuned properly

4.1.2 Commercial IDS

Commercial IDS tools, such as Palo Alto Networks IDS and Cisco's Firepower, often integrate machine learning (ML) and behavioral analysis to dynamically adjust to evolving threats. Commercial systems emphasize ease of use, integration, and maintenance, often offering robust support and a user-friendly interface that simplifies deployment.

Advantages: These systems come with vendor support, streamlined updates, and enhanced detection capabilities that include ML algorithms to improve accuracy. Commercial IDS tools are optimized for enterprises requiring low maintenance and high performance, especially for compliance purposes.

Challenges: The main tradeoff is cost, as commercial IDS solutions typically have high licensing fees. Moreover, these tools may offer limited customization options compared to open-source solutions and can sometimes consume more resources due to their extensive feature set [9].

Evaluating Intrusion Detection Systems (IDS) do not rely only on advantages and challenges, but also on several key criteria, including detection capability, configurability, resource consumption, scalability, and ease of integration.

4.1.3 Key factors

After researching open-source IDS options like Snort [7] [17] and commercial solutions like Cisco Firepower [16], a comparative analysis highlights distinct strengths and limitations of detection, flexibility, efficiency, scalability, and integration.

Snort, an open-source IDS, excels in signature-based detection, identifying known threats using **customizable** detection rules. Its flexibility makes it ideal for tailored environments, though it requires technical expertise and is mainly effective against recognized threats. Cisco Firepower, on the other hand, employs multi-layered detection, combining signatures, anomaly detection, and machine learning to identify sophisticated attacks, including unknown threats. The Firepower Management Center (FMC) simplifies configuration, making it accessible for teams with limited IDS experience.

In terms of **efficiency**, Snort's lightweight design suits small deployments with limited hardware, but scaling Snort across large networks can be challenging. Cisco Firepower is designed for scalability, with centralized control across physical and virtual environments, making it effective for high-data environments and large-scale deployments.

For **integration**, Snort allows flexible integration with various tools, though it may require custom configurations. Cisco Firepower, however, integrates seamlessly within the Cisco ecosystem, including Advanced Malware Protection, providing easy interoperability and centralized management.

In summary, Snort is an ideal, cost-effective choice for small-to-medium deployments or educational settings, such as the next assignment, due to its flexibility and low resource demands. Cisco Firepower is better suited for enterprise environments that require advanced detection, scalability, and integration within a Cisco infrastructure.

5 Future Trends and Recommendations

5.1 Predictions

Enhanced Machine Learning Integration: Advanced IDSs are expected to further leverage machine learning to improve real-time threat analysis. Techniques such as deep learning and reinforcement learning enable IDSs to detect increasingly sophisticated threats, including zero-day attacks, by analyzing behavioral patterns that deviate from established baselines [2] [8]

Behavioral and Anomaly-Based Detection: Transitioning from traditional signature-based detection, future IDS solutions will adopt anomaly-based and behavioral analysis models. These models provide the flexibility to detect novel threats by identifying deviations in user or system behavior, a crucial capability given the rise in complex, evasive cyber-attacks

Edge Computing and IoT Integration: As edge computing and IoT continue to grow, IDS will evolve to monitor data closer to the source. This trend is particularly relevant for securing distributed environments like industrial IoT or smart cities, where decentralized, real-time analysis is essential for timely response

Automation in Threat Response: Automation in IDS will become a priority, enabling rapid response to threats without the need for manual intervention. Automated incident response, using playbooks and predefined workflows, will help organizations address vulnerabilities swiftly, reducing time-to-remediation in high-risk scenarios. [8]

5.2 Best Practices

Based on the *Research Trends in Network-Based Intrusion Detection Systems: A Review*[9] is recommended to:

Adopt a Multi-Layered Detection Approach: Combining signature-based and anomaly-based detection techniques within IDS can maximize threat coverage, ensuring both known and unknown threats are more effectively addressed.

Continuous Updates and Tuning of Detection Rules: For IDS effectiveness, it is crucial to update detection signatures and refine rules regularly. These updates ensure that IDSs remain resilient against the latest attack patterns and minimize false positives that could impact network performance.

Leverage Threat Intelligence Feeds: Integrating external threat intelligence enables IDS to stay informed of emerging global threat trends, which is essential for proactive threat detection and response.

Centralized Management and Monitoring: Utilizing centralized dashboards or Security Information and Event Management **SIEM!** (**SIEM!**) systems can simplify IDS management across large networks, providing unified visibility and faster, more coordinated threat response.

Prioritize Scalability and Future-Readiness: With network traffic and cyber threats growing, choosing scalable IDS solutions—such as cloud-native or modular architectures—ensures adaptability and performance continuity as organizational needs expand.

6 Conclusion

In conclusion, this report examines the landscape of Intrusion Detection Systems (IDS), highlighting various approaches to combat the increasing complexity of network threats. The primary methods—signature-based, anomaly-based, specification-based, and machine learning-driven detection—each present unique benefits and trade-offs concerning detection accuracy, resource consumption, and adaptability to new threats. While signature-based systems effectively handle known threats, they struggle with novel attacks, a challenge addressed by anomaly-based and AI-driven techniques, which may, however, face higher false positive rates and greater computational demands.

Emerging technologies like federated learning, eBPF, and blockchain-enhanced IDS are advancing distributed detection, scalability, and data privacy. These innovations aim to balance detection precision and scalability. Future trends indicate a growing need for hybrid, multi-layered IDS architectures that integrate multiple detection methods, especially as organizations adopt Zero Trust frameworks and edge computing.

Overall, IDS technology is evolving to meet the demands of complex, high-volume network environments, emphasizing continuous rule updates, multi-layered detection, and centralized management for effective threat monitoring and response.

References

[1] A. Anchugam and N. Thangadurai. Title of the article. *Journal Name*, XX:123–456, 2020.

[2] Raad Sadi Aziz, Dhakaa Mohsin Kareem, and Sabiha F. Jawad. Intrusion detection systems: Status, challenges and future trends-a survey. In *2021 4th International Iraqi Conference on Engineering Technology and Their Applications (IICETA)*, pages 303–307, 2021.

[3] Hachmi Fatma and Mohamed Limam. A two-stage process based on data mining and optimization to identify false positives and false negatives generated by intrusion detection systems. In *2015 11th International Conference on Computational Intelligence and Security (CIS)*, pages 308–311, 2015.

[4] A. Ghorbani et al. Title of the article. *Journal Name*, XX:123–456, 2015.

[5] Rupinder Gill, Jason Smith, and Andrew Clark. Specification-based intrusion detection in wlans. In *2006 22nd Annual Computer Security Applications Conference (ACSAC'06)*, pages 141–152, 2006.

[6] M. Sazzadul Hoque, M. A. Mukit, and M. A. N. Bikas. An implementation of intrusion detection system using genetic algorithm. *arXiv*, 2012.

[7] Nattawat Khamphakdee, Nunnapus Benjamas, and Saiyan Saiyod. Improving intrusion detection system based on snort rules for network probe attack detection. In *2014 2nd International Conference on Information and Communication Technology (ICoICT)*, pages 69–74, 2014.

[8] Jitender Kumar, Rainu Nandal, Kamaldeep, and Omdev Dahiya. Accuracy and performance enhancement of machine learning system for ids. In *2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, pages 1–6, 2022.

[9] Satish Kumar, Sunanda Gupta, and Sakshi Arora. Research trends in network-based intrusion detection systems: A review. *IEEE Access*, 9:157761–157779, 2021.

[10] Kevin Alexander Lo, Cornelius Briant Joe, Samuel Philip, and Hidayaturrahman. Comparison of machine learning and deep learning models for detecting cyberbullying. In *2024 International Visualization, Informatics and Technology Conference (IVIT)*, pages 138–144, 2024.

[11] Chandu Jagan Sekhar Madala, G. Hemanth Kumar Yadav, S. Sivakumar, R. Nithya, Manjunatha K M, and M. Deivakani. Federated learning approach for tracking malicious activities in cyber-physical systems. In *2022 International Conference on Edge Computing and Applications (ICECAA)*, pages 494–499, 2022.

[12] MIT Sloan School of Management. Machine learning, explained. `https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained`, 2023. Accessed: 2024-10-28.

[13] Ruturaj Mohite and Balasubramanian Thangaraju. Enhancing container security with per-process per-container egress packet filtering using ebpf. In *2024 International Conference on Electrical, Computer and Energy Technologies (ICECET*, pages 1–8, 2024.

[14] Yusuf Sani, Ahmed Mohamedou, Khalid Ali, Anahita Farjamfar, Mohamed Azman, and Solahuddin Shamsuddin. An overview of neural networks use in anomaly intrusion detection systems. In *2009 IEEE Student Conference on Research and Development (SCOReD)*, pages 89–92, 2009.

[15] Hami Satilmiş, Sedat Akleylek, and Zaliha Yüce Tok. A systematic literature review on host-based intrusion detection systems. *IEEE Access*, 12:27237–27266, 2024.

[16] Cisco Systems. Introduction to the cisco firepower system, 2024. Accessed: Oct. 30, 2024.

[17] Cisco Systems. Snort: Open source intrusion prevention system, 2024. Accessed: Oct. 30, 2024.

[18] Raja Majid Ali Ujjan, Zeeshan Pervez, and Keshav Dahal. Snort based collaborative intrusion detection system using blockchain in sdn. In *2019 13th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*, pages 1–8, 2019.