Embedded Self Organizing Systems (Vol 12. No 14. 2025) (pp.04-09)

Embedded Selforganizing Systems

Issue Topic: "AI Applications in Engineering Science "

A Comparative Study of AI Models in Open-Source Intrusion Detection and Prevention Systems

Enkh-Od Erdene
Mongolian University of Science and Technology
E-mail: odko1988@gmail.com

Uranchimeg Tudevdagva[1,2]
[1]Mongolian University of Science and Technology
[2]Citi University
E-mail: uranchimeg@must.edu.mn

Dashdorj Yamkhin
Mongolian University of Science and Technology
E-mail: dashdorj@must.edu.mn

Abstract[1]— The rapid advancement of information technology, along with the continuous growth in the volume and diversity of network traffic, has led to a sharp increase in the number of cyber attackers, making the implementation of intrusion detection and intrusion prevention systems (IDS/IPS) essential for both public and private sector organizations. However, budget limitations often present a significant obstacle, rendering commercial IDS/IPS solutions inaccessible for many organizations. In response to this issue, this study undertakes a comparative analysis of two open-source systems, namely Snort and Suricata. This research seeks to evaluate their effectiveness in real-world scenarios and provide insights into optimal system configuration. The comparative results are intended to inform system selection decisions and guide practical implementation strategies [1]. Moreover, the research integrates the use of artificial intelligence (AI)-based models—specifically Random Forest, Decision Tree, and Logistic Regression—to analyze the log files generated during system testing. This approach demonstrates significant advantages, including reduced analysis time and improved operational efficiency. This study is expected to provide network security professionals and academic researchers with practical value, empirical evidence, and a solid technical foundation, thereby contributing to the advancement of cybersecurity.

Keywords— Artificial Intelligence, AI in Cybersecurity, Open-source IDS/IPS, Denial-of-Service attack, Hping3 tool, Suricata, Snort.

I. INTRODUCTION

The internet continues to transform how we connect with others, organize the flow of information, and share opinions. With its growing influence on individual consumers and large economies alike, the internet has become a vital part of our day-to-day lives. As reported by Statista, in 2025, the number of internet users worldwide stood at 5.56 billion, which means that around two-thirds of the global population is currently connected to the world wide web [2].

The cybercrime industry has grown into a colossal force. According to estimates, cybercrime will cost 10.29 trillion U.S. dollars worldwide in 2025, and it is projected to increase to approximately 16 trillion U.S. dollars by 2029 [3]. This anticipated figure underscores the potential for significant financial losses and emphasizes the critical importance of organizational implementation of proactive measures. In particular, it emphasizes the necessity for organizations to adopt advanced security solutions—such as Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS)—to mitigate and prevent such threats.

Research indicates that Intrusion Detection System (IDS) solutions exhibit detection rates ranging from 80% to 98%. Furthermore, organizations that have implemented Intrusion Prevention Systems (IPS) experience a reduction of 50% to 75% in response time to security incidents. By automating the processes of attack detection and defense, IPS significantly shortens the time between initial detection and the implementation of countermeasures.

Verizon's Data Breach Investigations Report (DBIR) indicates that 44% of all breaches analyzed showed ransomware was present, marking a notable rise from last year's report [4]. The report underscores the necessity of training artificial intelligence models through machine learning and deploying them in conjunction with intrusion detection and prevention systems.

Accordingly, this article seeks to investigate the benefits of employing open-source systems for network intrusion detection

and prevention, and to explore their integration with artificial intelligence (AI) models—specifically Random Forest, Decision Tree, and Logistic Regression—for enhanced analytical performance.

II. RELATED WORKS

The deployment of open-source intrusion detection and prevention systems (IDS/IPS) has gained significant traction due to increasing cybersecurity threats and the high cost of commercial solutions. Among these, Snort and Suricata are the most commonly used intrusion detection and prevention systems. Both utilize rule-based mechanisms for detecting known attack signatures, yet they differ in performance characteristics and scalability. Several studies have explored their comparative effectiveness. Liu et al. (2019) [5] highlighted the limitations of signature-based detection in handling zero-day exploits, advocating for enhanced rule flexibility and integration with intelligent systems.

To overcome the constraints of traditional IDS, researchers have increasingly focused on artificial intelligence (AI) techniques, particularly machine learning (ML) and deep learning (DL), to enhance detection capabilities. Shone et al. (2018) [6] introduced a stacked autoencoder-based deep learning model that achieved high accuracy in identifying both known and unknown attacks without relying on handcrafted features.

Beyond traffic analysis, the application of AI to IDS log data analysis has also shown promise. Ashiku and Dagli (2021) [7] This paper presents a deep learning-based IDS that adapts to evolving cyber threats by detecting both known and zero-day attacks. Using the UNSW-NB15 dataset, the model shows strong potential for enhancing network security and resilience.

The combination of rule-based detection and AI has been a focal point in the development of hybrid intrusion detection systems. Sommer and Paxson (2010) [8] emphasized the trade-offs between anomaly-based and signature-based detection, suggesting that a hybrid approach could mitigate false positives while improving adaptability. Garcia-Teodoro et al. (2016) [9] proposed a hybrid framework that integrates Snort rules with machine learning-based anomaly detection, utilizing flow-level features for real-time evaluation. Porambage et al. (2018) [10] mentioned that the integration of Artificial Intelligence (AI) algorithms and machine learning at the edge of the networks will further assist the data-intensive requirements of the IoT applications.

In summary, previous studies have laid a strong foundation for integrating network intrusion detection and prevention systems with AI-based analysis. However, there remains a need for practical implementations of such integrations, particularly in open-source IDS/IPS environments. This study examines the real-world application of Snort and Suricata in a controlled experimental setup and integrates AI-based models for log analysis to enhance efficiency and improve detection accuracy.

III. COMMERCIAL AND OPEN-SOURCE IDS/IPS

A. The Main Structure of Open-Source Systems

Free and open-source software (FOSS) denotes software with source code that is openly accessible to the public for viewing, studying, evaluating, and modifying. Because the source code is openly available, individuals are free to participate in improving the software's design voluntarily. This distinguishes it from proprietary or closed-source software, which typically restricts duplication rights, conceals source code from users, and is protected by patents. The advantages of using open-source software include reducing development costs, enhancing security and stability, and providing users with greater control over their hardware and systems.

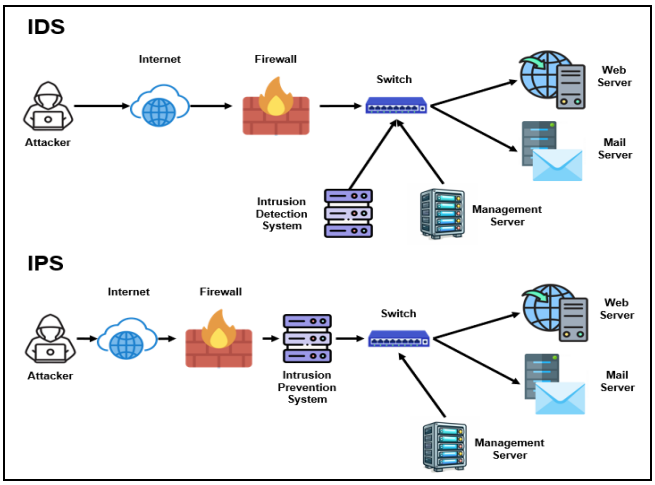

Figure 1: Operational Principles of IDS/IPS.

In the ever-evolving landscape of cybersecurity, protecting users' sensitive information and confidential data has become a paramount objective for both organizations and individuals. A critical component of this security infrastructure is the Intrusion Detection System (IDS) and Intrusion Prevention System (IPS). IDS and IPS both identify potential threats within a network; however, only IPS has the capability to actively manage and respond to them.

TABLE I. RATING LIST OF OPEN-SOURCE IDS/IPS [11]

No.	*IDS/IPS*	*Rating*
1	Snort	9.1
2	Suricata	9
3	Zeek (Bro)	8.9
4	Maltrail	8.5
5	Security Onion	7.6
6	Kismet	7
7	Psad	6.8

No.	IDS/IPS	Rating
8	Sagan	6.3

The above list of open-source systems is based on criteria such as core architecture, performance, rule configuration, packet reading, network security monitoring, and the ability to perform deep packet inspection.

Discussion. The comparative ratings in Table 1 align closely with prior evaluations of open-source IDS/IPS platforms, underscoring a clear stratification by architectural design, detection methodology, and extensibility. Snort's top score (9.1) reflects its long-standing prominence as a signature-based engine, corroborating Roesch's (1999) [12] findings on its high detection accuracy and extensive rule set library. Suricata's near-equal rating (9.0) concurs with Park and Ahn (2014) [13] to cover massive number of packets which are caused by digital convergence and ubiquitous IT system Suricata's have the availability to process packets in multithreading environment. Zeek (formerly Bro) scores marginally lower at 8.9, consistent with Paxson's (1999) [14] exposition of its event-driven architecture: although its signature-based speed lags behind, Zeek's powerful protocol analysis and scripting capabilities deliver unparalleled flexibility for complex, custom detections in network forensics.

Mid-tier systems—Maltrail (8.5) and Security Onion (7.6)—demonstrate the trade-offs inherent in combining anomaly-based detection with integrated dashboards.

The lower rated tools occupy specialized niches: Kismet (7.0) excels in wireless intrusion detection, but its focus on 802.11 renders it less applicable for general network traffic. Psad (6.8), which analyzes iptables logs, and Sagan (6.3), a multi-threaded log-analysis engine using Snort-compatible syntax, both reflect lower scores due to limited payload inspection and heavier reliance on host-based logs—characteristics that Sommer and Paxson (2010) [8] warn may increase false negatives in payload-rich threat scenarios. Collectively, these evaluations suggest that organizations should prioritize Snort, Suricata, or Zeek for core network defense, while supplementing with Maltrail or Psad in layered deployments and reserving niche tools like Kismet or Sagan for specialized environments.

B. *Snort Open-Source IDS/IPS*

Snort is one of the most widely used open-source intrusion detection systems (IDS), originally developed by Martin Roesch (1999) [16] and now maintained by Cisco Systems. It operates primarily as a network-based IDS (NIDS) and utilizes a signature-based detection approach to identify known attack patterns. Snort's widespread adoption is attributed to its flexibility, community-driven rule base, and compatibility with various network configurations.

C. *Suricata Open-Source IDS/IPS*

Suricata is an open-source intrusion detection and prevention system developed by the Open Information Security Foundation (OISF) [17]. The beta version was released in December 2009, and the first stable version was launched in July 2010. Suricata was designed with the goal of introducing new ideas and technol-ogies in the field of intrusion detection. The Open Information Security Foundation (OISF) provides Suricata with a set of rules for detecting and preventing attacks, while the Suricata engine streamlines the process of maintaining an optimized security level.

The Fig. 2 illustrates the constant growth of the global market for commercial Intrusion Detection and Prevention Systems (IDPS).

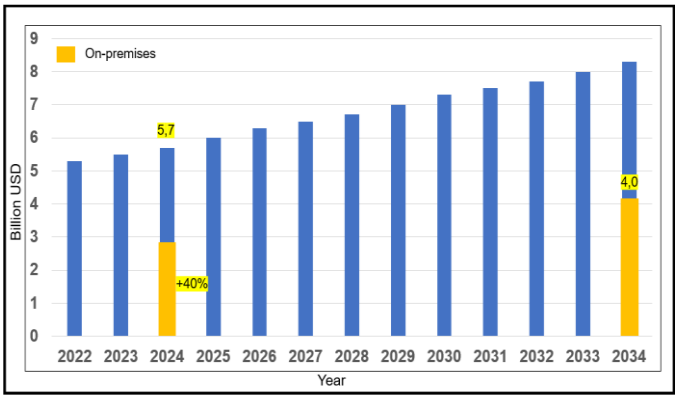

Figure 2: Market Growth Indicators for Commercial IDS/IPS, 2022-2034, in Billion USD.

Discussion. The global intrusion detection and prevention system (IDS/IPS) market was valued at USD 5.7 billion in 2024 and is estimated to register a CAGR of 7.3% between 2025 and 2034. Based on deployment model, the market is divided into on-premises, cloud and hybrid. In 2024, on-premises segment held a market share of over 40% and is expected to cross USD 4 billion by 2034 [15].

IV. EXPERIMENTAL WORK

A. *The General Structure of the Experimental Work*

The purpose of this experimental work is to simulate network attacks and analyze the resulting log files using AI-based models to enhance the accuracy and effectiveness of open-source intrusion detection and prevention systems.

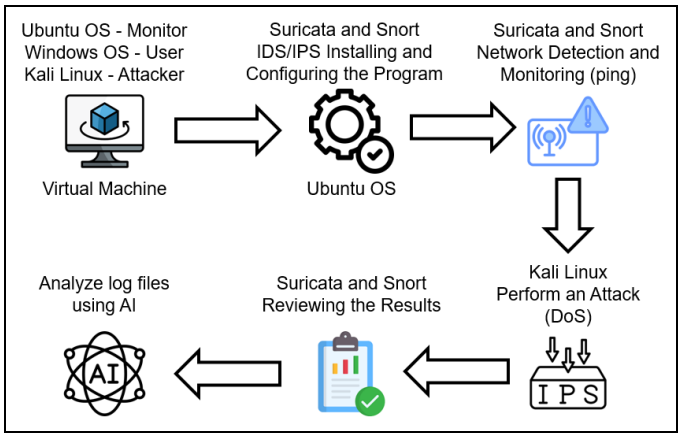

Figure 3: General Topology Diagram of the Experimental Work.

To conduct the experimental work, the following devices, operating systems, and attack tools were utilized:

a) Laptop – Dell G15 (Intel Core i5-13450, 16 GB RAM, 512 GB SSD)

b) Virtual Operating Systems – Ubuntu 64, Windows 10 Pro, Kali Linux 2025.2 deployed using VirtualBox

c) Open-source Systems – Snort and Suricata

d) Attack Tool – Hping3

e) Type of Attack – Denial-of-Service (DoS)

B. Preparing the Experimental Environment and Installation of Intrusion Detection and Prevention Systems (Snort, Suricata):

We created three virtual machines for this experiment. All are isolated from external networks and connected only via the host-only adapter, enabling direct communication between them. We will use the monitoring VM as a man-in-the-middle. The first VM, used for monitoring, runs on Ubuntu 24.04.2 LTS (64-bit) (Monitor); the second VM, used for protection, runs on Windows 10 Pro (64-bit) (User); and the third VM, used for attacks, runs on Kali Linux 2025.2 (Attacker).

We obtained the protected user's network address using the Windows Command Prompt. To organize network monitoring, we installed the open-source intrusion detection and prevention systems Snort and Suricata on the Ubuntu monitoring machine.

For Snort, we used the following commands:

Commands: [16]
apt-get update – Update the Operating System
apt-get install snort – Install Snort
nano /etc/snort/snort.conf – Configuration of the Network and Rule Database
nano /etc/snort/rules/local.rules – Write an Own Rule

The Snort rules were written as follows:

alert icmp any any -> $HOME_NET any (msg:"ICMP Detection Rule Snort"; sid:100001; rev:1:)

alert tcp any any -> $HOME_NET any (msg:"SYN Flood Detection Snort"; flags;S; flow:stateless; detection_filter: track bt_src, count 20, seconds 10; sid:100002; rev:1;)

Points to Note. You cannot set the interface to listen on in the static configuration file. When you start monitoring, you must specify the interface on the command line. Default rules must be deactivated and use only local rules for any experiment. After making any configuration changes, you should always verify that the system is operating correctly. You can use the following command to do so:

sudo snort -T -i enp0s3 -c /etc/snort/snort.conf – Verify the Configuration

For Suricata, we used the following commands:

Commands: [17]
add-apt-repository ppa: oisf/suricata-stable – Install Suricata Archive
apt update – Update the Operating System
apt-get install suricata – Install Suricata
cd /etc/suricata – Access the Suricata Directory
nano suricata.yaml – Configuration of the Network, Interface, NFQ for IPS and Rule Database
cd /var/lib/suricata/rules – Access the Rules Directory
nano custom.rules – Write an Own Rule

The Suricata rules were written as follows:

alert icmp any any -> $HOME_NET any (msg:"ICMP Detection Rule Suri-cata"; sid:123; rev:1:)

alert tcp any any -> $HOME_NET any (msg:"SYN Flood Detection Suricata"; flags;S; flow:stateless; detection_filter: track bt_src, count 20, seconds 10; sid:124; rev:1;)

Points to Note. You must configure the listening interface (For example, enp0s3) and the user's network address range (For example, 192.168.56.0/24) in the settings. You must enable NFQ to use IPS. Also, after making any configuration changes, you should always verify that everything is working correctly. You can use the following commands:
systemctl status suricata – Check the General Operation of the System
suricata -T – Verify Configuration

Configuration Status of Snort:

```
ipvar  HOME_NET 192.168.56.0/24
ipvar EXTERNAL_NET !$HOME_NET
include $RULE_PATH/local.rules
```

Configuration Status of Suricata:

```
address-groups:
        HOME_NET: "[192.168.56.0/24]"
af-packet:
        interface: enp0s3
nfq:
        mode: accept
        repeat-mark: 1
        repeat-mask: 1
        route-queue: 2
rule-files:
        custom.rules
```

For the network intrusion detection system, when checking the network traffic (Ping) from the attacker to the user

machine created in the virtual environment, it appears as follows:

Snort Detection Rule: 08/26-22:24:40.226534 [**] [1:100001:1] ICMP Detection Rule Snort [**] [Priority: 0] {ICMP} 192.168.56.103 -> 192.168.56.102

08/26-23:22:07.733589 [**] [1:100002:1] SYN Flood Detection Rule Snort [**] [Priority: 0] {TCP} 192.168.56.103:7298 -> 192.168.56.102:139.

Suricata Detection Rule: 08/26/2025-22:23:34.085485 [**] [1:123:1] ICMP De-tection Rule Suricata [**] [Classification: (null)] [Priority: 3] {ICMP} 192.168.56.103:8 -> 192.168.56.102:0

08/26/2025-22:21:48.047811 [**] [1:124:1] SYN Flood Detection Suricata [**] [Classification: (null)] [Priority: 3] {TCP} 192.168.56.103:52297 -> 192.168.56.102:139

C. *Performance Evalution of Intrusion Prevention System:*

The attack tool we selected (Hping3) not only spoofs the source IP address but also sends a large volume of TCP traffic to the target user, making it appear as though the traffic originates from a random or specifically defined source address as designated by the user.

Commands: [18]

```
nmap -sV 192.168.56.102
hping3 -S -p 139 --flood 192.168.56.102
```

For the network intrusion prevention system, it appears as follows:

Suricata Prevention Rule: 08/28/2025-00:41:48.742781 [Drop] [**] [1:125:1] ICMP Drop Suricata [**] [Classification: (null)] [Priority: 3] {ICMP} 192.168.56.103:8 -> 192.168.56.102:0

08/28/2025-00:43:41.568581 [Drop] [**] [1:126:1] SYN Flood Drop Suricata [**] [Classification: (null)] [Priority: 3] {TCP} 192.168.56.103:48729 -> 192.168.56.102:139

V. Experimental Result

The results of the experiment illustrate that open-source network intrusion detection and prevention systems offer the flexibility to define custom rules and implement tailored monitoring mechanisms suited to specific requirements.

A total of 50 experiments were conducted, each lasting between 1 and 3 minutes, with an average of 2 million attacks per experiment. The extracted log files (.CSV) were processed to generate 16,000 instances, comprising both normal and attack cases, and subsequently modified and tested in various configurations. When evaluated using machine learning algorithms, the Random Forest Classifier and Decision Tree Classifier achieved accuracies of 90–95% and delivered results within a short timeframe. In comparison, the Logistic Regression Classifier attained an accuracy of 93% but required longer processing time, indicating lower efficiency for large-scale log analysis.

In the experiment, when an attack was launched against the test user from a single attacker, the CPU load increased from 5% to 40%, and network traffic rose from 0 Mbps to 22.4 Mbps, resulting in an overall load index of approximately 31%.

This suggests that if attacks were launched simultaneously from multiple hosts, it could potentially render the target user or server completely unavailable.

In a network environment, there are a total of 65,535 ports, of which 12 are commonly used. For example, ports 20 and 21 are used for FTP to transfer files between the server and client, port 80 is used for HTTP to access the Internet, and port 443 is the secure, encrypted version of port 80. Proper port management can reduce the risk of attacks; specifically, unused ports should be closed on network management devices such as routers or firewalls.

VI. Conclusion

This study presents a comparative and integrative analysis of open-source Intrusion Detection and Prevention Systems (IDS/IPS) combined with artificial intelligence (AI)-based models—specifically Random Forest, Decision Tree, and Logistic Regression—to enhance log file analysis and threat detection accuracy. The experimental findings demonstrate that open-source platforms such as Snort and Suricata provide significant flexibility for customizing security rules, tailoring detection mechanisms, and optimizing performance in diverse network environments.

Through rigorous experimentation involving 50 controlled attack simulations and extensive log analysis, the research confirms that AI-assisted processing can substantially improve detection efficiency and reduce response time. In particular, the Random Forest and Decision Tree classifiers exhibited superior accuracy and speed, while Logistic Regression, though accurate, was less efficient for large-scale data handling. These insights highlight the value of applying suitable AI techniques to augment traditional IDS/IPS capabilities.

The originality of this research lies in its integration of open-source IDS/IPS frameworks with machine learning-based log analysis, bridging practical implementation with analytical intelligence. This hybrid approach provides a cost-effective and adaptive security model particularly beneficial for organizations with limited resources.

From a practical perspective, the results underscore several recommendations: institutions should leverage open-source systems for flexible, scalable intrusion detection; apply AI models such as Random Forest or Decision Tree for rapid and reliable log analysis; and ensure proactive network management by closing unused ports and optimizing system configurations. Collectively, these practices strengthen overall cybersecurity posture and pave the way toward intelligent, self-organizing defense systems capable of responding dynamically to emerging network threats.

ACKNOWLEDGMENT

This research was supported and funded by the Saxon State Ministry of Science, Culture and Tourism under the Saxon Student Mobility Program. We would like to express our sincere gratitude for the funding and support provided.

This research was supported by KOICA (Korea International Cooperation Agency) through the "Capacity Building Project for the School of Infromation and Communication Technology at Mongolian University of Science and Technology" (Contract No. P2019-00124).

REFERENCES

[1] E.Enkh-Od, Ya.Dashdorj, Kh.Uyanga, "English translation: A Study on the Use of Open-Source Intrusion Detection and Prevention Systems," *12th National Proceedings on Mongolian Information Technology (MIT), Ulaanbaatar, Mongolia, 2025,* pp. 58-63.

[2] Statista, "Digital Population Worldwide", *Accessed: September 14. 2025. Available online at https://www.statista.com/statista/617136/digital-population-worldwide.*

[3] Statista, "Cybercrime Growth Worldwide" *Accessed: September 14. 2025. Available online at https://www.statista.com/topics/13546/cybercrime-worldwide/#topicOverview/.*

[4] Verizon, "Data Breach Investigation Report" *Accessed: September 14. 2025. Available online at https://www.verizon.com/business/resources/reports/dbir/.*

[5] H.Liu, B.Lang, M.Liu, H.Yan, "CNN and RNN based payload classification methods for attack detection", *Elsevier, vol.163, 2019,* pp. 332-341.

[6] N.Shone, T.N.Ngoc, V.D.Phai, Q.Shi, "A Deep Learning Approach to Network Intrusion Detection", *IEEE Transactions on Emerging Topics in Computational Intelligence, vol.2, no.1, 2018,* pp. 41-50.

[7] L.Ashiku, C.Dagli, "Network Intrusion Detection System using Deep Learning", *Elsevier, vol.185, 2021,* pp. 239-247.

[8] R.Sommer, V.Paxson, "Outside the Closed World: On Using Machine Learning for Network Intrusion Detection", *IEEE Symposium on Security and Privacy, Oakland, CA, USA, 2010,* pp. 305-316.

[9] P.Garcia-Teodoro, J.Diaz-Verdejo, G.Macia-Fernandez, E.Vasquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges", *Elsevier, vol.28, no.1-2, 2009,* pp. 18-28.

[10] P.Porambage, J.Okwuibe, M.Liyanage, M.Ylianttila, T.Taleb, "Survey on Multi-Access Edge Computing for Internet of Things Realization", *IEEE Communications Surveys & Tutorials, vol.20, no.4, 2018,* pp. 2961-2991.

[11] LinuxLinks, "Best Free Open-Source NIDSs" *Accessed: September 14. 2025. Available online at https://www.linuxlinks.com/best-free-open-source-network-intrusion-detection-systems.*

[12] M.Roesch, "Snort-Lightweight Intrusion Detection for Networks", *13th Systems Admnistration Conference, Seattle, Washington, USA, 1999.*

[13] W.Park, S.Ahn, "Performance Comparison and Detection Analysis in Snort and Suricata Environment", *Wireless Personal Communications, vol.94, 2017,* pp. 241-252.

[14] V.Paxson, "Bro: A System for Detecting Network Intruders in Real-Time", *Computer Networks, vol.31, no.23-24, 1999,* pp. 2435-2463.

[15] Gminsights, "IDS and IPS Market" *Accessed: September 14. 2025. Available online at https://www.gminsights.com/industry-analysis/intrusion-detection-prevention-system-ids-ips-market.*

[16] Snort, "Official site" *Accessed: September 14. 2025. Available online at https://snort.org/*

[17] Suricata, "Official site" *Accessed: September 14. 2025. Available online at https://suricata.io/*

[18] Kali Linux, "Hping3 Tool" *Accessed: September 14. 2025. Available online at https://www.kali.org/tools/hping3/*