Journal of Applied Informatics and Computing (JAIC)
Vol.10, No.2, April 2026, pp. 1872~1877
e-ISSN: 2548-6861

Implementation of a Security System using Snort and Honeypot for Network Attack Detection and Prevention

Muhammad Dimas [1]*, Agus Wijayanto [2]*, Dicky Satrio Ikhsan Utomo [3]*, Djumhadi [4]*
* Teknologi Informasi, Fakultas Ilmu Komputer, Universitas Mulia
dimasmuhammad@students.universitasmulia.ac.id [1], aguswijayanto@universitasmulia.ac.id [2], dicky@universitasmulia.ac.id [3], djumhadi@universitasmulia.ac.id[4]

Article Info

Article history:

Received 2026-01-30
Revised 2026-03-03
Accepted 2026-04-08

Keyword:

Snort,
Honeypot,
Network security,
Intrusion Detection System.

ABSTRACT

This research is based on the increasing incidence of cyberattacks on network infrastructure, especially in the telecommunications sector, which demands an effective and sustainable network security system. The purpose of this study is to implement Snort as an Intrusion Detection System (IDS) and Honeypot as a decoy system to improve the ability to detect, monitor, and mitigate attacks on server networks. The research method used is action research, which includes the stage of diagnosis, action planning, implementation of actions, and reflection. The system implementation was carried out in a simulation environment using Kali Linux and VirtualBox as the virtualization platform. The test was carried out through attack simulations in the form of port scanning using Nmap, brute force attack using Hydra, and Denial of Service (DoS) simulations. The results showed that Snort was able to detect all attacks tested with a recall rate of 100% while Honeypot managed to redirect attacks and record attacker activity in detail. The integration of Snort and Honeypot has been proven to increase threat visibility and provide additional protection to the main server, making it effective as an open-source-based network security solution.

I. INTRODUCTION

As information technology continues to advance—particularly in the field of computer network security and its associated services—it has significantly facilitated various aspects of daily human activities. However, alongside these advancements, serious challenges have emerged, especially concerning security issues. Modern society has become highly dependent on information systems, while, paradoxically, the number of security incidents has increased sharply. According to data published by the National Cyber and Crypto Agency of Indonesia (BSSN) in the Indonesian Cyber Security Landscape report, a total of 403,990,813 anomalous traffic incidents were recorded in Indonesia throughout 2023. The highest number of anomalies occurred in August, reaching 78,464,385 incidents, while the lowest was recorded in November with 19,296,439 incidents. Such high levels of anomalous activity pose significant risks, including network and system performance degradation, sensitive data breaches, reputational damage, and a decline in organizational trust.

These conditions are further exacerbated by the relatively low level of awareness regarding information system security. In many organizations, security incident reporting and handling mechanisms are still performed manually by system administrators. Consequently, system integrity and availability heavily depend on the presence and responsiveness of administrators, which may lead to delayed incident response and increased vulnerability.

In Indonesia, the telecommunications sector, which serves as the backbone of the national digital infrastructure, faces increasingly complex security challenges. Companies such as PT Telkom Indonesia, through its subsidiary PT Telkom Akses, are responsible for delivering reliable and secure network services. PT Telkom Akses Balikpapan, as part of the regional operational network, manages various operational support servers, including those supporting the KAWAN (Kawal Gangguan) application. The KAWAN application is a network disturbance monitoring and supervision system

designed to detect, report, and handle service disruption incidents in real time. The server acts as the primary data center for the KAWAN application, handling large volumes of data related to network status, disturbance logs, and technical team coordination. Given its critical role in maintaining the continuity of telecommunications services in the Balikpapan region and its surroundings, the server becomes a potential target for cyber threat actors. Therefore, the implementation of effective network security mechanisms, such as a Snort-based Intrusion Detection System (IDS) integrated with a Honeypot, is essential to mitigate these threats.

II. METHOD

The research carried out at PT. Telkom Access Balikpapan uses a quantitative approach by applying the action research method. The action research method is a research approach that aims to test and develop solutions to problems that occur through real and sustainable actions.

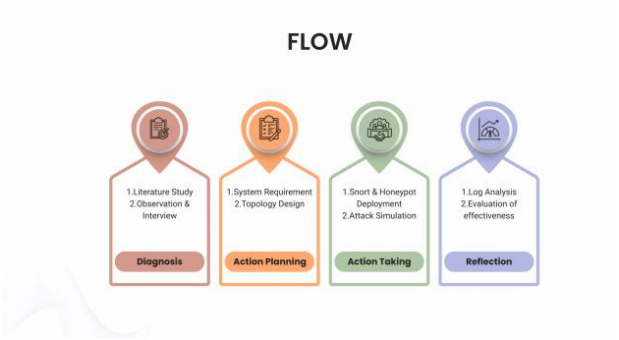

Figure 1. Research Flow

A. Diagnosis

The initial stage aims to identify the conditions and problems that occur in the research object. At this stage, system needs analysis, problem analysis that arises, user needs analysis, and analysis of the ongoing network topology are carried out.

B. Action Planning

At this stage, the researcher formulates a problem solving plan based on the results of the analysis at the diagnosis stage. The plan prepared includes the implementation of network security system configurations using Snort and Honeypot by utilizing the Kali Linux and Ubuntu Server operating systems. In addition, at this stage, the necessary hardware and software requirements are prepared, as well as the planning of attack testing scenarios against the systems that have been designed.

C. Action Taking

The implementation stage of the plan that has been prepared beforehand. At this stage, the installation and configuration of snort and honeypot is carried out on the Ubuntu Server operating system as a network security solution at PT. Back to Basics. Once the system is implemented, tests are carried out through attack simulations to evaluate the performance of the security system built.

D. Reflection

Which aims to review and evaluate all stages of research that have been carried out. At this stage, an analysis of the results of system implementation and testing is carried out, so that the effectiveness of the solution applied can be known and becomes the basis for further improvement and development.

III. RESULTS AND DISCUSSION

A. Diagnosis

The observation results showed that there was no real-time network traffic monitoring mechanism so that suspicious activities such as port scanning and login attempts were not detected. This condition shows that the network is vulnerable to attacks and requires a detection system that can monitor network traffic in real-time. The results ofthe observation of the old topology are shown in Figure 2.

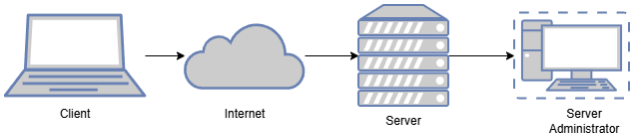

Figure 2. Existing topology

B. Action Planning

The action planning stage is focused on planning the implementation of a structured and integrated network security system. The planning began with the construction of a test network environment as a simulation medium, followed by the installation and configuration of Snort as an Intrusion Detection System to monitor network traffic. In addition, it is also planned to implement honeypots, such as Cowrie, which function as a bait target to attract and record attack activities. To improve detection capabilities, Snort rules were designed and adjusted to be able to identify various types of network attacks. Furthermore, the detection results from Snort are planned to be integrated with the logs generated by the honeypot so that attack data can be analyzed more comprehensively. This planning stage ends with the preparation of attack scenarios as a system testing method, with the aim of producing a network security system that is able to effectively detect, log, and analyze attacks.

The proposed system architecture consists of a monitored server, a Snort-based Intrusion Detection System positioned inline to observe network traffic, and a honeypot placed as a decoy service within the same network segment. Incoming traffic is first inspected by Snort to detect suspicious patterns based on predefined rules. Potential attack traffic is then redirected to the honeypot, which emulates vulnerable services and records attacker interactions. This architecture is designed to enhance threat visibility while protecting the main server from direct exploitation.

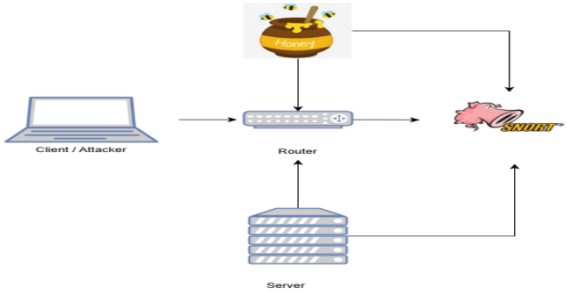

Figure 3. Plan new topology

C. Action Taking

The action taking stage focuses on implementing the proposed network security system and conducting attack simulations to evaluate its effectiveness. In this phase, Snort was deployed as an Intrusion Detection System (IDS) to monitor network traffic and generate alerts when suspicious activities were detected. Snort operates based on rule-driven mechanisms, enabling flexible configuration to identify various attack patterns.

In parallel, the honeypot deployed in this study is a low-interaction honeypot based on Cowrie, which emulates vulnerable SSH and Telnet services. Cowrie is designed to safely capture attacker interactions without exposing real system resources. The logging mechanism records detailed information, including source IP addresses, authentication attempts, executed commands, timestamps, and session duration. These logs provide valuable insights into attacker behavior and support further forensic analysis.

The analysis of honeypot logs revealed recurring attack patterns that can be utilized as threat intelligence. Observed patterns include repeated authentication attempts using common username and password combinations, sequential command execution following successful login attempts, and automated interaction behavior indicative of brute-force tools. Such findings demonstrate that honeypot-generated logs not only serve as attack records but also provide actionable insights that can support the development of more effective detection rules and enhance proactive security measures.

After the deployment process, several attack scenarios were conducted within a local network environment. The simulated attacks included network scanning using Nmap, Denial of Service (DoS) attacks generated with HPING3, and brute-force attacks performed using Hydra. These scenarios were designed to represent common threats encountered in real-world networks.

During the testing process, all attack activities were monitored simultaneously by Snort and the honeypot system. Snort generated alert logs for detected attacks, while Cowrie recorded detailed interaction logs from the attackers. The collected log data were then analyzed to evaluate detection performance and to support further assessment using confusion matrix metrics.

D. Reflection

Snort performance evaluation is carried out using a confusion matrix to determine the system's ability to classify network traffic between attacks and non-attacks.

During the experimental phase, the Snort IDS and honeypot system operated within acceptable resource limits in the simulated environment. No significant degradation in server responsiveness was observed during attack simulations. However, in large-scale deployments, extensive packet inspection and log generation may increase CPU utilization, memory consumption, and network latency. Therefore, performance monitoring and resource optimization are essential considerations for production implementation

Although the simulation environment based on Kali Linux and VirtualBox is adequate for controlled testing and initial validation of intrusion detection mechanisms, it does not fully represent real-world network conditions. In operational networks, traffic characteristics are more dynamic and heterogeneous, influenced by legitimate user behavior, background traffic, and varying workloads. Therefore, the results of this study should be interpreted as an initial evaluation. Further studies are recommended to implement the proposed system in a real production environment to assess its performance under actual network conditions.

Evaluation of Snort performance based on NMAP Scanning attacksBased on the data from the research results, the Confusion matrix resulting from the NMAP Scanning attack is shown in Table 1.

TABLE 1
NMAP ATTACK TESTING

Time	Identity	Nmap -sX	Nmap -sN	Nmap -sS	Nmap -sV	TP	TN	FP	FN
20.45	Log1-1	1	1	1	1	2011	0	15	0
21.00	Log2-1	1	1	1	1	2021	0	5	0
21.30	Log3-1	1	1	1	1	2025	0	9	0
21.35	Log4-1	1	1	1	1	2031	0	13	0
21.40	Log5-1	1	1	1	1	2036	0	10	0
	TOTAL	**5**	**5**	**5**	**5**	**10124**	**0**	**52**	**0**

Based on table 1 above, it shows that out of 10124 True Positives, there are 2011 True Positives in Log1-1, there are 2021 True Positives in Log2-1, there are 2025 True Positives in Log3-1, there are 2031 True Positives in Log4-1 and there are 2036 True Positives in Log5-1.

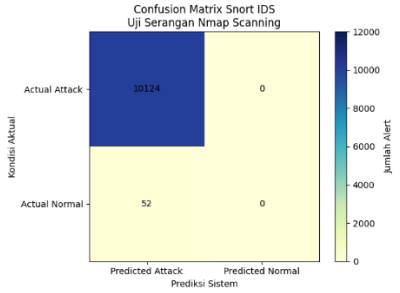

Figure 4. Spread of the Confusion Matrix of NMAP Attacks

Evaluation of Snort performance based on DoS attacks. Based on the data from the research results, the Confusion matrix resulting from the DoS attack is shown in the Table 2.

TABLE 2
DOS ATTACK TESTING

Time	Identity	Hping3	TP	TN	FP	FN
20.15	Log2-1	1	19	0	9	0
20.20	Log2-2	1	25	0	22	0
20.25	Log2-3	1	29	0	24	0
20.30	Log2-4	1	39	0	30	0
20.35	Log2-5	1	43	0	12	0
	TOTAL	**5**	**155**	**0**	**97**	**0**

Based on table 2 above, it shows that out of 155 True Positives, there are 19 True Positives in Log2-1, there are 25 True Positives in Log2-2, there are 29 True Positives in Log2-3, there are 39 True Positives in Log2-4 and there are 43 True Positives in Log2-5.

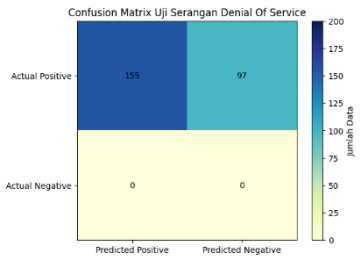

Figure 5. Spread of the Confusion Matrix of DoS Attacks

Evaluation of Snort performance based on Brute Force attacks. Based on the data from the results of the research, the Confusion matrix resulting from the Brute Force attack is shown in the Table 3.

TABLE 3
BRUTE FORCE ATTACK TESTING

Time	Identity	Hydra-Attack	TP	TN	FP	FN
20.55	Log3-1	1	12	0	2	0
21.00	Log3-2	1	14	0	0	0
21.05	Log3-3	1	18	0	0	0
21.10	Log3-4	1	22	0	0	0
21.15	Log3-5	1	26	0	0	0
	TOTAL	**5**	**92**	**0**	**2**	**0**

Based on table 3 above, it shows that out of the 92 True Positives, there are 12 True Positives in Log3-1, there are 14 True Positives in Log3-2, there are 18 True Positives in Log3-3, there are 22 True Positives in Log3-4 and there are 26 True Positives in Log3-5.

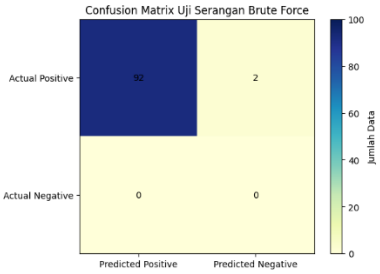

Figure 6. Spread of the Confusion Matrix of Brute Force Attacks

Evaluate Snort's performance based on the entire attack. Based on the data from the research results, the Confusion matrix resulting from all attacks is shown in the Table 4.

TABLE 4
TEST RESULTS OF THE ENTIRE ATTACK

No	Types of Attacks	TP	TN	FP	FN
1	Nmap Scanning	10124	0	52	0
2	DoS	155	0	97	0
3	Brute Force	92	0	2	0
	TOTAL	**10371**	**0**	**151**	**0**

Based on the results on the entire confusion matrix table, it can be seen that the Snort system successfully detected all attacks that occurred, which is shown by a True Positive (TP) value of 10,371 and a False Negative (FN) value of 0. This shows that no attack escapes without being detected by the system. Then a number of 151 False Positives (FP) were found indicating non-attack traffic.

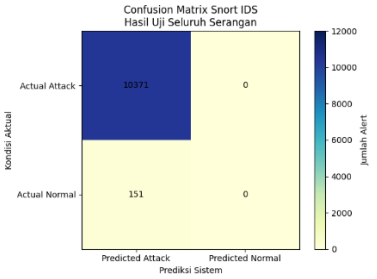

Figure 7. Spread of the Confusion Matrix of Throughout the Attack

The evaluation of Snort's performance in this study was carried out using a confusion matrix as a measuring tool to assess the accuracy of the system's detection of the attack being tested. In this test, the True Negative (TN) value is zero because the test scenario is focused on network traffic containing the attack.

Based on the results of the system test, the following evaluation data was obtained:

- True Positive (TP) = 10.371

- False Positive (FP) = 151
- False Negative (FN) = 0
- True Negative (TN) = 0

1. Accuracy

Accuracy indicates the level of accuracy of the system in classifying overall network traffic. Here is the formula for accuracy:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + TN}$$

Calculation results : $Accuracy = \frac{10371 + 0}{10371 + 0 + 151 + 0} = \frac{10371}{10522} =$ 0,9856 or 98,56% .

2. Precision

Precision shows the level of accuracy of the system in detecting an attack, i.e. how many alerts are actually an attack. Here is the formula of precision:

$$Precision = \frac{TP}{TP + FP}$$

Calculation results : $Precision = \frac{10371}{10371 + 151} = \frac{10371}{10522} =$ 0,9856 or 98,56%.

3. Recall

Recall shows the system's ability to detect all attacks that actually occur. Here is the formula for the recall:

$$Recall = \frac{TP}{TP + FN}$$

Calculation results : $Recall = \frac{10371}{10371 + 0} =$ 1 or 100%.

TABLE 5
FINAL CALCULATION RESULTS

Evaluation Matrix	Rumus	Result
Accuracy	(TP + TN) / (TP + TN + FP + FN)	98,56%
Precision	TP / (TP + FP)	98,56%
Recall	TP / (TP + FN)	100%

Based on the results of the calculation of the Accuracy, Precision, and Recall values, the IDS Snort system has an accuracy and precision rate of 98.56% and a recall value of 100%. This shows that all attack tests were successfully detected with a very low false positive rate.

The recall value of 100% indicates that all simulated attacks were successfully detected by the Snort system during the testing phase. This result was obtained because the evaluation scenario focused on predefined attack traffic within a controlled environment. However, the presence of 151 false positives indicates that some non-malicious traffic patterns were incorrectly classified as attacks. In real-world implementations, false positives may increase due to legitimate activities that resemble attack signatures. Consequently, continuous rule tuning and alert validation are necessary to reduce operational overhead and improve detection reliability.

In addition to detection success, system performance can also be assessed through accuracy, false alarm rate, and response time. The false alarm rate observed in this study is reflected by the number of false positives generated during testing. Although response time was not quantitatively measured, alert generation occurred in near real-time following attack initiation. These indicators demonstrate that the proposed system is capable of timely detection, although further optimization is required to minimize false alarms in operational environments.

E. Limitations

This study focuses on a limited set of attack scenarios, namely port scanning, brute force, and Denial of Service (DoS), as these attacks are widely recognized as common network-level threats and are frequently used as baseline scenarios in IDS evaluation studies due to their reproducibility and clear traffic patterns. This selection allows controlled assessment of detection performance while minimizing experimental complexity. However, the evaluation was conducted in a virtualized environment, which may not fully represent the diversity and dynamics of real-world network traffic. In addition, application-layer attacks such as SQL injection, malware-based traffic, and web exploitation were not included in the current testing scope. The system was also not integrated with real-time notification mechanisms or centralized monitoring dashboards, which limits its immediate operational usability. Furthermore, this study did not perform a direct comparative evaluation with other IDS/IPS solutions or traditional security approaches. These constraints indicate that further validation on real network infrastructures, expanded attack scenarios, system integration enhancements, and comparative analysis are necessary to strengthen the generalizability and practical applicability of the proposed security system.

IV. CONCLUSION

Based on the results of research and discussion on the Implementation of Security Systems Using Snort and Honeypot for Network Attack Detection and Prevention in the case study of PT. Telkom Access Balikpapan, then several conclusions can be drawn.

Snort has been successfully implemented as an Intrusion Detection System (IDS) in the server environment of PT. Telkom Access Balikpapan and is able to detect various suspicious activities on network traffic, such as port scanning, brute force attacks, and Denial of Service (DoS) simulations.

Honeypot (Cowrie) successfully serves as an effective decoy system in attracting attacks to dummy services, particularly on SSH services. Honeypot is capable of recording attacker activity in detail, including IP addresses, login attempts, attack patterns, and time of occurrence.

The integration of Snort and Honeypot increases the effectiveness of the network security system, where Snort plays a role in detecting attacks early, while Honeypot serves as a diversion of attacks and a means of collecting attack data. This approach reduces the risk of direct attacks on the main server and provides better visibility into cyber threats.

For production-scale implementation, several aspects must be considered, including system scalability, continuous rule updates, centralized log management, and administrator expertise. Regular maintenance is required to ensure that Snort rules remain effective against evolving attack techniques, while honeypot configurations should be periodically reviewed to prevent misuse. With proper maintenance and integration into organizational security policies, the proposed system can function as a sustainable network security solution.

The results of the study show that the system built can be used as a support for network security policies, especially in increasing awareness of unauthorized activities that have the potential to disrupt telecommunication service operations.

REFERENCES

[1] Achmad, R., Manullang, E. V., & Sanmas, E. R. (2020). Rancang Bangun Aplikasi Deteksi Dan Penanganan Serangan Ddos Dan Port Scanning Memanfaatkan Snort Pada Jaringan Komputer. Jurnal Teknologi Informasi, 8(1), 44–53.

[2] Azizah Zakiah., Ardhian Ekawijaya., & Eka Angga Laksana. (2019). Implementasi Metode Action Research Untuk Peningkatan Daya Saing UMKM Melalui E-Commerce. Jurnal Penelitian Komunikasi dan Opini Publik, 23(1), 54-62.

[3] Fadhil Raditya., & Jeckson Sidabutar. (2022). Analisis Rules Intrusion Detection Prevention System (IDPS) Suricata untuk Mendeteksi dan Menangkal Aktivitas Crypto Mining pada Jaringan. Jurnal Edukasi dan Penelitian Informatika, 8(2), 348-355.

[4] Februariyanti, H. (2006). Standar dan Manajemen Keamanan Komputer. Jurnal Teknologi Informasi Dinamik, XI(2), 134–142.

[5] Lanskap Keamanan Siber Indonesia. (n.d.).

[6] Laode Ikhwanul Uzlah., Rizal Adi Saputra., & Isnawaty. (2024). Deteksi Serangan Siber Pada Jaringan Komputer Menggunakan Metode Random Forest. Jurnal Mahasiswa Teknik Informatika, 8(3), 2787-2793.

[7] Maulana, A. N., Data, M., & Bakhtiar, F. A. (2025). Perancangan dan Implementasi Snort Rule Set untuk Deteksi Serangan SQL Injection. Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer, 9(9).

[8] Meilia Intan Sabila., Muhlis Tahir., Saskia Dwi Mardania., & Rizky Ilham Arifin. (2025). Implementasi Snort Sebagai IDS Dalam Mendeteksi Serangan Port Scanning NMAP Pada Simulasi Jaringan Virtual. Jurnal Mahasiswa Teknik Informatika, 9(2), 6944-6948.

[9] Nugraha, A., & Gustian, D. A. (2021). Deteksi Malware Dridex Menggunakan Signature-based Snort.

[10] Pratama, M. A., Setiawan, H., & Mair, Z. R. (2023). Implementasi Honeypot Sebagai Pendeteksi Serangan Pada Virtual Private Server (VPS). Jurnal Software Engineering and Computational Intelligence, 1(1), 26–39.

[11] Revanza Hafiz Erianto., Wahyu Adi Prabowo., & Trihastuti Yuniati. (2025). Analisis Sistem Keamanan Pada Software-Defined Network Dengan Hybrid

[12] Honeypot Menggunakan Quality Of Service, e-Proceeding of engineering, 12(2), 3350-3358.

[13] Toriyansa Natanegara., Yusuf Muhyidin., & Dayan Singasatia. (2023). Implementasi Honeypot Cowrie dan Snort Sebagai Alat Deteksi Serangan Pada Server. Jurnal Mahasiswa Teknik Informatika, 7(3), 1871-1877.

[14] V.Mohan Patro., & Manas Ranjan Patra. (2014). Augmenting Weighted Average with Confusion Matrix to Enhance Classification Accuracy. Transactions on Machine Learning and Artificial I nt elligence, 2(4), 77-91.

[15] Wijaya, B., & Pratama, A. (2020). Deteksi Penyusupan Pada Server Menggunakan Metode Intrusion Detection System (Ids) Berbasis Snort. Jurnal Sisfokom (Sistem Informasi Dan Komputer), 9(1), 97–101.

[16] Suci Sekar Sari., & Agus Teddyana. (2024). Analisis Efektifitas Rule Snort dalam Mendeteksi Serangan Jaringan. Publikasi Teknik Informatika dan Jaringan, 2(4), 01-15.