Journal of Information Technology and Computer Science (INTECOMS)
Volume 9 Nomor 2, Tahun 2026
e-ISSN : 2614-1574
p-ISSN : 2621-3249

ANALISIS EFEKTIVITAS IMPLEMENTASI MODEL KEAMANAN BERBASIS PENIPUAN DUA TINGKAT DALAM SISTEM KEAMANAN SIBER

ANALYSIS OF THE EFFECTIVENESS VALUE OF IMPLEMENTING THE TWO-TIER DECEPTION-DRIVEN SECURITY MODEL IN CYBER SECURITY SYSTEMS

Sahrul Ramadhan[1*], Agung Budi Sutanto[2], Arya Adhyaksa Waskita[3]
Teknik Informatika S2, Universitas Pamulang, Indonesia[1,2,3]
Sahrul.rin@gmail.com[1]

ABSTRACT

The increasing complexity of cyber attacks, especially Brute Force and SQL Injection, poses a significant risk to production environments. Conventional reactive security measures are often unable to provide sufficient understanding regarding the behavior of attackers. This study designs and analyzes a "Two-Tier Deception Architecture" aimed at improving early warning capabilities without sacrificing the integrity of the production system. This architecture physically and logically separates the production environment as Tier 1 and the deception-based laboratory environment as Tier 2. By utilizing a combination of Fail2Ban and NFTables, the system stealthily redirects traffic from detected malicious actors to a separate environment hosting the Cowrie and DVWA honeypots. All security logs are collected and analyzed using a centralized ELK Stack SIEM. Evaluation using a curated dataset of 100 samples (consisting of 60 legitimate activities and 40 malicious activities) achieved a detection and redirection accuracy of 95%. The system demonstrates minimal resource usage on the production server while providing precise threat intelligence. This research shows that the inclusion of a deception tier within standard infrastructure substantially strengthens proactive defense and incident response effectiveness.

Keywords: *Cybersecurity, Deception Technology, ELK Stack, Honeypot, SIEM, Threat Intelligence*

ABSTRAK

Meningkatnya kompleksitas serangan siber, terutama Brute Force dan SQL Injection, menimbulkan risiko signifikan bagi lingkungan produksi. Langkah-langkah keamanan reaktif konvensional seringkali tidak mampu memberikan pemahaman yang cukup mengenai perilaku penyerang. Studi ini merancang dan menganalisis "Arsitektur Penipuan Dua Tingkat" yang bertujuan untuk meningkatkan kemampuan peringatan dini tanpa mengorbankan integritas sistem produksi. Arsitektur ini secara fisik dan logis memisahkan lingkungan produksi sebagai Tingkat 1 dan lingkungan laboratorium berbasis penipuan sebagai Tingkat 2. Dengan memanfaatkan kombinasi Fail2Ban dan NFTables, sistem secara diam-diam mengalihkan lalu lintas dari pelaku jahat yang terdeteksi ke lingkungan terpisah yang menampung honeypot Cowrie dan DVWA. Semua log keamanan dikumpulkan dan dianalisis menggunakan SIEM ELK Stack terpusat. Evaluasi menggunakan kumpulan data yang dikurasi sebanyak 100 sampel (terdiri dari 60 aktivitas sah dan 40 aktivitas jahat) mencapai akurasi deteksi dan pengalihan sebesar 95%. Sistem ini menunjukkan penggunaan sumber daya minimal pada server produksi sambil memberikan intelijen ancaman yang tepat. Penelitian ini menunjukkan bahwa penyertaan lapisan penipuan dalam infrastruktur standar secara substansial memperkuat pertahanan proaktif dan efektivitas respons insiden.

Kata Kunci: Keamanan Siber, Teknologi Penipuan, ELK Stack, Honeypot, SIEM, Intelijen Ancaman

INTRODUCTION

he cybersecurity landscape is now shifting from perimeter-based defense to a proactive approach in threat hunting [1]. Contemporary enterprises face persistent automated attacks in the form of scanning and exploitation. Conventional Intrusion Detection Systems (IDS) often generate a multitude of alerts, the majority of which are false positives, causing analytical fatigue for security professionals. Furthermore, once an attacker is blocked by a conventional firewall, organizations typically gain little or no intelligence regarding the attacker's intent, tools, or geographical origin [2].

Deception technology, particularly honeypots, provides a distinct advantage by turning the tables on the attackers. By offering decoy targets, organizations can divert attackers from true IT assets and collect valuable behavioral data [3]. Honeypots serve as an early warning tool to significantly reduce the detection time for emerging threats [4], [5]. However, directly integrating honeypots into

production networks often poses serious risks, such as lateral movement if the honeypot is compromised [2], [6].

Several recent studies have emphasized the critical role of deception technology in mitigating these exact vulnerabilities within contemporary network security. Susanto and Romli emphasized that relying solely on conventional security creates blind spots, whereas deploying decoy servers or honeypots allows defenders to securely monitor and analyze cyber threats without compromising the integrity of the primary infrastructure [7]. Expanding on this concept, a comprehensive analysis by Morić et al. evaluated various honeypot solutions, identifying specific deployments like Cowrie as essential tools for capturing attackers' tactics, techniques, and procedures (TTPs). Their research highlighted the necessity of seamlessly integrating these honeypots with active security protocols, such as firewalls, to optimize proactive threat detection [3].

To enhance the resilience of these decoy systems, researchers have begun proposing multi-layered architectures and advanced configuration strategies. Gamilla et al. explored a two-tier deception-driven security model, demonstrating that combining honeypots with deceptive network segments provides a proactive defense mechanism that successfully confuses potential threats and significantly improves overall network resilience [6]. However, the effectiveness of these deceptive segments relies heavily on their psychological impact on adversaries. Aggarwal et al. investigated this through a two-sided deception strategy by manipulating the observable configurations of both real machines and honeypots. Their findings revealed that concealing the identities of honeypots, or making real machines mimic decoys, successfully creates uncertainty, leading human attackers to attempt more exploits and exfiltrate data directly from the honeypot environments [8]. Building upon these foundational studies, there remains a critical need for an architecture that not only deploys honeypots in a separate tier but also actively and dynamically routes live malicious traffic into them without disrupting legitimate users.

This study addresses that challenge by proposing a Two-Tier Deception Architecture. This architecture implements a Redirect-on-Detection mechanism where the production server acts as a gatekeeper. Once a threat is confirmed based on behavioral thresholds, the malicious session is redirected to a secondary deception tier. This method ensures the production server remains focused on serving valid users while the deception tier manages the analysis of malicious actors.

RESEARCH METHODOLOGY

Network Architecture

This architecture is implemented on two Virtual Private Servers (VPS) to guarantee complete physical and logical isolation [9]. At Tier 1, the system operates an Ubuntu-based Nginx web server that reflects the true corporate assets. Security monitoring is managed through Fail2Ban, with routing configuration controlled by NFTables. Meanwhile, at Tier 2, the system, managed via Docker Compose, hosts the deception and analysis engines. This tier includes low to medium-interaction Honeypots utilizing Cowrie, which is used to intercept SSH and Telnet protocols, recording brute force attempts as well as malicious shell interactions [10], [11], and DVWA (Damn Vulnerable Web Application), which functions as a web-based decoy to securely attract and analyze HTTP-related attacks such as SQL Injection [12], [7].

To ensure seamless data collection, Filebeat agents are deployed on both tiers to continuously forward raw logs to the centralized Logstash on Tier 2.

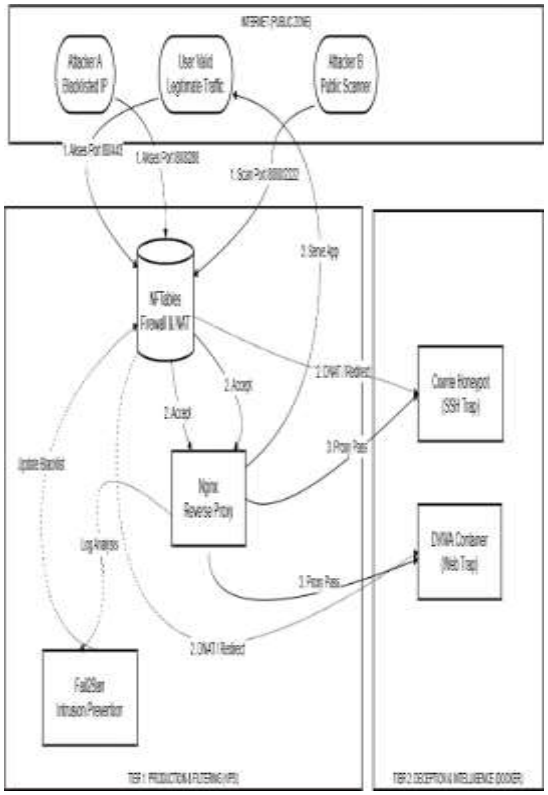

Fig 1. Two-Tier Network Architecture

Topology Security Flow Logic

The security mechanism operates based on a "Redirect-on-Detection" logic. As incoming traffic from an external IP address enters Tier 1, it is continuously monitored by Fail2Ban. If the activity is legitimate, the user is allowed to access the production web or SSH services normally. However, if the traffic exhibits malicious behavior and reaches the defined authentication failure threshold (maxretry = 3), Fail2Ban automatically triggers an NFTables rule. This rule seamlessly redirects the attacker's connection to the isolated honeypots (Cowrie or DVWA) in Tier 2. Once the attacker is trapped, all malicious interactions are recorded, enriched with GeoIP coordinates using Logstash Grok filters, and indexed into Elasticsearch [13]. Finally, the data is visualized through the Kibana dashboard to support real-time threat profile analysis [1], [14].

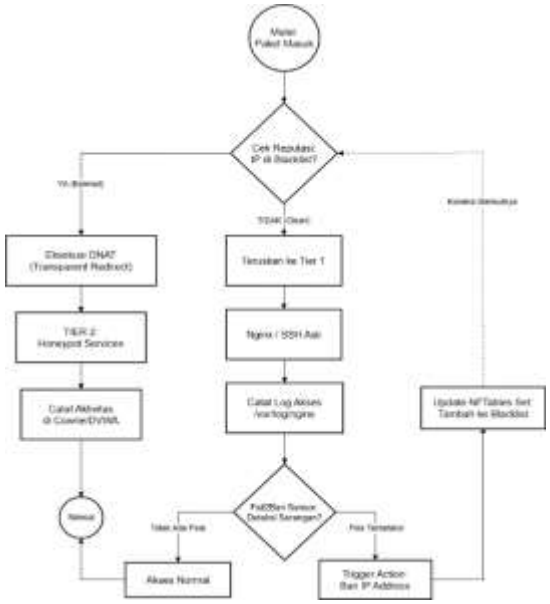

Fig 2. Security Redirection Flowchart

System Components

Tier 1 architecture relies on Nginx as the primary production web service, mirroring actual corporate digital assets to provide a realistic target for external actors. Security monitoring at this entry point is managed by Fail2Ban, which functions as a threshold-based intrusion prevention agent that dynamically interacts with the NFTables framework. NFTables is utilized for its advanced traffic management capabilities, offering superior performance over legacy IPTables in executing dynamic Destination Network Address Translation (DNAT) rules. This configuration allows the system to stealthily redirect identified malicious sessions to the deception tier without interrupting the connection state or alerting the attacker of the transition.

The deception environment in Tier 2 is orchestrated using Docker Compose to host a suite of low to medium-interaction honeypots, primarily Cowrie and DVWA. Cowrie is deployed to intercept and record unauthorized SSH and Telnet interactions, capturing critical data such as brute force credentials and terminal commands executed by the intruder. Complementing this, DVWA serves as a vulnerable web decoy designed to attract and analyze application-layer exploits, specifically SQL Injection and cross-site scripting, within a secure and isolated network segment. The

physical and logical isolation of these services ensures that any potential compromise within Tier 2 does not impact the integrity of the production environment in Tier 1.

Centralized logging and analysis are facilitated by the ELK Stack, which manages the entire threat intelligence pipeline from data ingestion to visualization. Filebeat agents are installed across both tiers to forward raw log data to Logstash, where the information is parsed and enriched using Grok filters and GeoIP databases. The structured data is then indexed into Elasticsearch for high-performance retrieval and long-term storage. Finally, Kibana provides the visualization interface, enabling security administrators to build real-time dashboards for monitoring attack trends, geographical origins, and adversary behavior. This integrated stack transforms raw security events into actionable intelligence, supporting proactive incident response and system fortification.

Implementation and Configuration Fail2Ban and Transparent Redirection Configuration

The core of this system relies on transitioning from the classic IPTables to the more advanced NFTables framework for dynamic traffic management [15]. The redirection is not executed by a static firewall rule, but dynamically orchestrated by Fail2Ban.

To achieve this, a custom action script named action.d/nftables-nat.conf was created within the Fail2Ban directory. When an IP address breaches the threshold, Fail2Ban triggers the following NFTables command:

```
nft add rule ip nat PREROUTING ip saddr <attacker_ip> tcp dport {80, 443, 8288} counter dnat to <Tier2_IP>
```

This Destination Network Address Translation (DNAT) rule ensures a seamless and transparent experience for the attacker. From the attacker's perspective, the TCP handshake completes successfully, and they receive the expected SSH banner or HTTP response. They believe they have successfully accessed the target production server, unaware that their traffic is being transparently routed across the network to the isolated Tier 2 honeypot [16].

Authentication Threshold and Time Window Argumentation

A crucial parameter in this implementation is the configuration within the jail.local file of Fail2Ban. The system relies on three primary variables: maxretry, findtime, and bantime.

A value of maxretry = 3 is determined by considering the principle of balancing security and usability. This setting is based on the CIS (Center for Internet Security) Benchmarks for Linux environments, which recommends an account lockout threshold of between 3 to 5 failed attempts [15]. Setting the lower bound of 3 establishes a robust security posture to combat automated botnets while still permitting minor typing errors from legitimate administrators.

Furthermore, the findtime was configured to 60 seconds. This means an attacker must commit 3 failed attempts within a 1-minute window to trigger the redirection. Once triggered, the bantime (redirection duration) is set to 86400 seconds (24 hours), ensuring that persistent threats are isolated in the deception tier long enough to gather substantial behavioral intelligence [16].

Centralized SIEM ELK Stack Implementation

The implementation of the ELK Stack SIEM in this architecture serves as a centralized command center that transforms raw log data from both tiers into structured threat intelligence. This process begins with the deployment of Filebeat agents across Tier 1 and Tier 2 to ensure real-time data collection without imposing significant overhead on production server resources. The gathered logs encompass Nginx access logs and Fail2Ban alerts from the production server, as well as login activities and command executions from the Cowrie and DVWA containers within the deception laboratory environment [10], [12]. This centralized data transmission is crucial for maintaining log integrity; even if an attacker attempts to erase their footprint on a compromised server, the original log

copies remain securely stored within the ELK data center.

Data processing is managed by Logstash through a sophisticated pipeline mechanism utilizing Grok filters. These filters are responsible for parsing unstructured log strings into specific data fields, such as source IP addresses, usernames, passwords, and the specific attack vectors employed. A primary advantage of this implementation is the integration of GeoIP databases, which enables Logstash to perform automated data enrichment. Every incoming attacker IP address is mapped to geographical coordinates, country names, and Autonomous System Numbers (ASN), providing a profound understanding of the origin and profile of the adversary groups [1], [3].

The structured data is subsequently indexed into Elasticsearch to facilitate high-performance search and large-scale analytics. As the final interface, Kibana is utilized to construct visualization dashboards that serve as a comprehensive Threat Intelligence panel [13]. Through these dashboards, the security team can monitor the geographical distribution of attacks, identify trends in frequent brute force attempts, and analyze SQL Injection payloads captured by the deception system [13], [14]. This ELK Stack integration ensures that every adversary interaction within Tier 2 provides significant learning value to proactively fortify Tier 1 defenses [1], [16]

RESULT AND DISCUSSION

Testing Scenario and Attack Simulation

To empirically evaluate the efficacy of the Two-Tier Deception Architecture, a controlled attack simulation was conducted utilizing a dataset of 100 interaction samples, which were categorized into legitimate and malicious traffic. The legitimate traffic consisted of 60 samples generated using automated browser scripts and basic curl commands to request static assets and HTML pages from the Tier 1 Nginx server at normal browsing speeds. The malicious traffic comprised 40 samples, evenly split between SSH Brute Force and SQL Injection attacks. The 20 SSH Brute Force attempts were executed using the THC-Hydra utility with a standard password dictionary targeted at the Tier 1 IP address. Concurrently, the 20 SQL Injection samples were generated using Sqlmap targeting specific URL parameters. This SQLi dataset was further divided into 15 aggressive payloads and 5 slow-paced evasion payloads, the latter utilizing the --delay parameter to deliberately attempt to bypass rate-limiting mechanisms.

The efficacy of the architecture was tested empirically using a curated dataset of 100 interaction samples. The evaluation is summarized in Table 1.

Performance Evaluation (Confusion Matrix

The evaluation of the redirection accuracy is detailed in Table I.

TABLE I. EVALUATION RESULT OF 100 SAMPLES

Category	Scenario	Count	Result
True Negative	Legitimate Web Access	60	Successfully Allowed
True Positive	SSH Brute Force	20	Successfully Redirected
True Positive	SQL Injection (Standard)	15	Successfully Redirected
False Negative	SQL Injection (Evasion)	5	Failed to Redirect

The system achieved an overall accuracy of 95%. The architecture successfully distinguished between normal users and aggressive scanners, immediately routing the 20 SSH Brute Force attempts and 15 Standard SQLi attempts to the Cowrie and DVWA containers in Tier 2.

Fig 3. Dashboard Visualization of Detection Accuracy and Traffic Distribution

Analysis of False Negatives (Evasion Techniques)

A critical finding in this research is the False Negative rate of 5%, which was observed exclusively during slow-paced "SQLi evasion" attacks. Because the detection mechanism in Tier 1 relies on Fail2Ban—which operates on a rate-limiting and threshold logic (maxretry = 3 within findtime = 60s)—attackers utilizing evasion techniques successfully bypassed the trigger.

By using the --delay 25 parameter in Sqlmap, the attacker only sent 2 requests per minute. This rate falls below the defined threshold, causing Fail2Ban to perceive the traffic as legitimate. Consequently, these 5 malicious payloads bypassed the redirection mechanism and were recorded directly in the production Nginx access logs. This highlights a limitation of threshold-based redirection, emphasizing the need for supplementary signature-based detection (such as a Web Application Firewall) in Tier 1 for highly stealthy attacks.

Resource Impact and Overhead Analysis

A common concern with implementing deception and SIEM agents on production servers is computational overhead. To quantify this, the resource utilization of Tier 1 (Production) was monitored using the htop utility across three states: Idle, Normal Traffic, and Under Attack. The results are presented in Table 2.

TABLE 2. RESOURCE UTILIZATION ON PRODUCTION VPS

System State	CPU Usage (%)	RAM Usage (MB)	Active Security Service
Idle / Baseline	0.2%	550 MB	Nginx, Fail2Ban, Filebeat
Normal Traffic	0.5%	610 MB	Nginx (serving), Filebeat
Under Attack (Brute Force)	1.2%	704 MB	Nginx, Fail2Ban (Parsing), NFTables (Routing)

Note: Total Server Capacity is 4 CPUs and 4096 MB (4GB) RAM.

The data in Table II clearly indicates that the security layer is highly optimal. Even during a full-scale Brute Force attack where Fail2Ban actively parses logs and triggers NFTables, the CPU usage peaked at merely 1.2%, and memory consumption reached only 704 MB. This proves that offloading the heavy processing and honeypot hosting to Tier 2 ensures that the Two-Tier architecture imposes virtually zero latency on the production environment.

Threat Intelligence Visualization and Profiling

By redirecting attackers to Tier 2, the ELK Stack SIEM successfully generated actionable, high-fidelity threat intelligence without risking actual corporate data. The Kibana dashboard visualizations provided several key insights, primarily through geographical profiling, where Logstash successfully extracted GeoIP coordinates from the attacker IPs. This visualization revealed that the majority of automated SSH Brute Force campaigns originated from specific foreign botnet groups, particularly from Autonomous System Numbers (ASNs) known for bulletproof hosting.

Fig 4. Geographical mapping of threat origins and legitimate traffic using Kibana Maps

CONCLUSION

This research successfully implemented and verified a Two-Tier Deception Architecture utilizing Honeypots and the ELK Stack SIEM. With a 95% accuracy rate, this system has proven effective in identifying malicious actors seamlessly without disrupting legitimate production traffic. The physical isolation of the deception tier creates a secure environment for in-depth analysis of attacks. The resulting threat intelligence enables faster and proactive incident response. Future development will be directed toward automating the "Feedback Loop" by utilizing the Elasticsearch REST API to automatically inject threat indicators back into the primary firewall, moving towards a fully autonomous defense ecosystem

DAFTAR PUSTAKA

[1] A. B. Ajmal, M. Alam, A. A. Khaliq, S. Khan, Z. Qadir, and M. A. P. Mahmud, "Last Line of Defense: Reliability through Inducing Cyber Threat Hunting with Deception in SCADA Networks," *IEEE Access*, vol. 9, pp. 126789–126800, 2021, doi: 10.1109/ACCESS.2021.3111420.

[2] A. P. Gamilla, T. D. Palaoag, and M. A. Naagas, "Enhancing reconnaissance security: a 2-tier deception-driven model approach (2TDDSM)," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 34, no. 3, pp. 1999–2006, 2024, doi: 10.11591/ijeecs.v34.i3.pp1999-2006.

[3] Z. Morić, V. Dakić, and D. Regvart, "Advancing Cybersecurity with Honeypots and Deception Strategies," *Informatics*, vol. 12, no. 1, 2025, doi: 10.3390/informatics12010014.

[4] M. A. R. Al Amin, S. Shetty, L. Njilla, D. K. Tosh, and C. Kamhoua, "Hidden markov model and cyber deception for the prevention of adversarial lateral movement," *IEEE Access*, vol. 9, pp. 49662–49682, 2021, doi: 10.1109/ACCESS.2021.3069105.

[5] A. Javadpour, F. Ja'fari, T. Taleb, M. Shojafar, and C. Benzaïd, "A comprehensive survey on cyber deception techniques to improve honeypot performance," *Comput. Secur.*, vol. 140, no. June 2023, p. 103792, 2024, doi: 10.1016/j.cose.2024.103792.

[6] A. P. Gamilla, T. D. Palaoag, and M. A. Naagas, "Probing the depths: assessing the efficacy of the two-tier deception-driven security model," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 36, no. 3, pp. 1631–1639, 2024, doi: 10.11591/ijeecs.v36.i3.pp1631-1639.

[7] C. Susanto and M. A. Romli, "Application of Honeypot in Network Security for Detecting Cyber Attacks on it Infrastructure," *J-INTECH (Journal Inf. Technol.*, no. 10, pp. 24–32, 2025.

[8] P. Aggarwal, Y. Du, K. Singh, and C. Gonzalez, "Decoys in Cybersecurity : An Exploratory Study to Test the Effectiveness of," *IJCAI-21 1st Int. Work. Adapt. Cyber Def. arXiv2108.11037v1*, 2020.

[9] T. Yu, Y. Xin, and C. Zhang, "HoneyFactory: Container-Based Comprehensive Cyber Deception Honeynet Architecture," *Electron.*, vol. 13, no. 2, 2024, doi: 10.3390/electronics13020361.

[10] M. Baçer, E. Y. Güven, and M. A. Aydin, "SSH and Telnet Protocols Attack Analysis Using Honeypot Technique," *Proc. - 6th Int. Conf. Comput. Sci. Eng. UBMK 2021*, vol. 7, pp. 806–811, 2021, doi: 10.1109/UBMK52708.2021.9558948.

[11] N. Ilg, P. Duplys, D. Sisejkovic, and M. Menth, "A survey of contemporary open-source honeypots, frameworks, and tools," *J. Netw. Comput. Appl.*, vol. 220, no.

August, p. 103737, 2023, doi: 10.1016/j.jnca.2023.103737.

[12] H. Fan, Q. Tan, R. Tan, and B. Nie, "HoneyDecoy: A Comprehensive Web-Based Parasitic Honeypot System for Enhanced Cybersecurity," *Proc. - 2023 IEEE SmartWorld, Ubiquitous Intell. Comput. Auton. Trust. Veh. Scalable Comput. Commun. Digit. Twin, Priv. Comput. Data Secur. Metaverse, SmartWorld/UIC/ATC/ScalCom/DigitalTwin/PCDS/Me*, pp. 1–8, 2023, doi: 10.1109/SWC57546.2023.10448731.

[13] I. G. Adnyana, A. M. Dirgayusari, and K. J. Atmaja, "Data Visualization for Building a Cyber Attack Monitoring Dashboard Based on Honeypot," *Sink. J. dan Penelit. Tek. Inform.*, vol. 8, no. October, pp. 2510–2518, 2024.

[14] U. Bartwal, S. Mukhopadhyay, R. Negi, and S. Shukla, "Security Orchestration, Automation, and Response Engine for Deployment of Behavioural Honeypots," *5th IEEE Conf. Dependable Secur. Comput. DSC 2022 SECSOC 2022 Work. PASS4IoT 2022 Work. SICSA Int. Pap. Compet. Cybersecurity*, pp. 1–8, 2022, doi: 10.1109/DSC54232.2022.9888808.

[15] A. Dermawan, Yuhandri, and Sumijan, "Analisis Perbandingan Optimalisasi Port Knocking Dan Honeypot dengan Iptables Pada Server Untuk Keamanan Jaringan," *KESATRIA J. Penerapan Sist. Inf. (Komputer Manajemen)*, vol. 5, no. 2, pp. 543–556, 2024, [Online]. Available: https://pkm.tunasbangsa.ac.id/index.php/kesatria/article/view/364.

[16] M. Farrag, S. G. Sayed, and M. Zamzam, "Bluffing the Hackers: Automated Decoy Creation and Real-Time Cyber Deception," *2024 7th Int. Conf. Signal Process. Inf. Secur. ICSPIS 2024*, no. 1, pp. 1–6, 2024, doi: 10.1109/ICSPIS63676.2024.10812608.