Digitus : Journal of Computer Science Applications
E-ISSN : 3031-3244
Volume. 2, Issue 1, January 2024
Page No: 10-21

Real-Time Threat Detection and Forensic Readiness in Wireless LANs: A Case Study Using Snort and HoneyPy

Samroh
STMIK Mercusuar, Indonesia
Correspondent: samroh74@gmail.com

Received : December 15, 2023
Accepted : January 10, 2024
Published : January 31, 2024

Citation: Samroh. (2024). Real-Time Threat Detection and Forensic Readiness in Wireless LANs: A Case Study Using Snort and HoneyPy. Digitus : Journal of Computer Science Applications, 2 (1), 10-21.

ABSTRACT: Wireless Local Area Networks (WLANs), especially in public sector infrastructures, face escalating security challenges due to their open architecture and exposure to various cyber threats. This study aims to evaluate the effectiveness of integrating Snort, an intrusion detection system (IDS), with HoneyPy, a low-interaction honeypot, to enhance real-time monitoring and forensic capabilities in WLAN environments. The methodology involved deploying Snort and HoneyPy within a simulated public network setup, using Ubuntu Server as the operating platform. Network attacks were emulated using tools such as Nmap, Hydra, and Metasploit to simulate various threat scenarios. Key metrics such as detection rate, false positive rate, and system responsiveness were used to evaluate performance. Visualization and log analysis tools including Kibana and Snorby were also incorporated to interpret intrusion data effectively. Results demonstrated that Snort successfully identified common scanning techniques and DDoS patterns using rule-based detection. HoneyPy effectively captured brute-force attack behaviors and provided rich interaction logs. The integrated setup facilitated enhanced incident correlation and provided valuable insights for forensic investigation. Visualization dashboards improved threat analysis and supported adaptive response strategies. In conclusion, the combined use of Snort and HoneyPy offers a scalable and cost-effective solution for public WLAN security. It enhances detection accuracy, supports forensic readiness, and provides actionable intelligence on attack behaviors. The findings highlight the practical relevance of layered defense models, offering concrete guidance for public institutions in strengthening WLAN security and forensic readiness.

Keywords: WLAN Security, Intrusion Detection, Honeypot, Snort, Honeypy, Forensic Readiness, Port Scanning, Brute-Force Attacks, Ddos Mitigation.
.

INTRODUCTION

Wireless local area networks (WLANs) have become an essential infrastructure in public sector organizations due to their ease of deployment and accessibility. However, these very characteristics expose WLANs to numerous security risks. Unlike wired networks that rely on physical barriers to protect data, WLANs operate through radio frequencies, making them particularly susceptible to eavesdropping, unauthorized access, and a range of cyberattacks, including man-in-the-middle (MITM) and denial-of-service (DoS) attacks (Alodat, 2022; Suroto, 2018). These vulnerabilities are further compounded by flaws in wireless security protocols, such as WEP, WPA, and even WPA2, which have been consistently shown to contain exploitable weaknesses (Kejiou & Bekaroo, 2022; Singh & Sharma, 2014). In light of these issues, maintaining the integrity and confidentiality of WLANs has become a pressing concern for public institutions, especially those responsible for managing sensitive data and public services.

To address these challenges, most public networks traditionally rely on firewalls to control and monitor traffic. Firewalls enforce security policies by filtering inbound and outbound data based on predetermined rules. However, their capacity to detect sophisticated intrusion attempts is limited, particularly in dynamic environments like WLANs. Firewalls often struggle to process encrypted traffic effectively and may fail to identify advanced persistent threats (APTs) without continuous rule updates and accurate configurations (Suhaimi et al., 2020; Kejiou & Bekaroo, 2022). This gap in detection capacity creates a security void, especially when public sector WLANs are increasingly targeted by well-coordinated and stealthy attacks. In such contexts, traditional perimeter defenses require reinforcement through supplementary intrusion detection tools that provide deeper visibility and real-time response capabilities.

Among the emerging solutions for enhancing WLAN security is the deployment of honeypots. These systems function as decoys that imitate legitimate services or hosts, intentionally designed to attract attackers. By doing so, they facilitate the observation and analysis of malicious behavior without exposing actual assets to danger (Suroto, 2018). The theoretical premise of honeypots lies in their ability to gather critical data on attacker intentions, tools, and techniques. Such insights prove invaluable in refining network defense strategies and improving incident response protocols (Suhaimi et al., 2020; Cheng et al., 2014). Although honeypots are not preventive in nature, they offer a high degree of situational awareness and serve as effective components in a layered security framework.

To improve the responsiveness of network defense, honeypots are often integrated with Intrusion Detection Systems (IDS), which monitor traffic patterns to identify anomalies or known threat signatures. One such IDS, Snort, has gained widespread use since its development by Martin Roesch in 1998. Initially conceived as a lightweight traffic analyzer, Snort evolved into a robust open-source IDS capable of detecting a wide range of network attacks. Its adaptability and rule-based detection mechanisms have made it a popular choice in both academic and enterprise environments. The Snort community continually contributes updated rule sets, enhancing its ability to identify emerging threats.

Complementing Snort's analytical capabilities is HoneyPy, a low-interaction honeypot framework tailored for lightweight deployment. Designed in Python, HoneyPy simulates vulnerable services to attract malicious activity in controlled settings. Its simplicity and flexibility make it suitable for academic research and operational network security tasks alike. HoneyPy has enabled security researchers to examine attacker behavior firsthand, providing empirical data that helps shape intrusion detection models and defense strategies.

Despite the promise of integrating IDS with honeypots, several challenges persist. Notably, the combination of Snort and HoneyPy can generate extensive log data, requiring careful filtering and interpretation to distinguish between genuine threats and false positives. Interoperability issues between IDS and honeypot frameworks also pose operational challenges, as seamless data sharing and correlation remain technically demanding. Moreover, the low-interaction nature of honeypots like HoneyPy may limit their realism, potentially affecting the accuracy of attack simulations. Additionally, discrepancies between the attack types observed in honeypots and those occurring in live networks can introduce blind spots in threat analysis.

In summary, while WLANs offer operational convenience, they are inherently vulnerable to a range of cyber threats, particularly in public sector environments. The limitations of traditional firewalls underscore the need for enhanced security measures, such as the integration of honeypots and intrusion detection systems. By deploying tools like Snort and HoneyPy, organizations can augment their defensive posture, gain critical insight into attacker behavior, and move toward a more adaptive and resilient network security strategy.

METHOD

This study adopts an applied research approach, combining network simulation and experimental testing to evaluate the integration of Snort and HoneyPy in a wireless LAN (WLAN) environment. The main objective is to assess the effectiveness of these tools in detecting and logging suspicious or malicious activity. The study takes place within the WLAN infrastructure of the Library and Archives Department of Pekanbaru City, a representative setting for public sector WLAN deployments.

System Environment

-Hardware Setup

- Server: Intel Xeon CPU E5-1607 v2, 16 GB RAM, 1 TB HDD.
- Router: MikroTik RB1100AHX2.
- Switch: D-Link DES-1008A.
- Client and Attacker Machines: Intel Core i5-6200U, 4 GB RAM.

-Software Stack

- Operating System: Ubuntu Server 14.04 LTS.
- IDS Tool: Snort (latest stable version).
- Honeypot Tool: HoneyPy.
- Attack Simulation Tools: Nmap, Hydra (for brute force), Metasploit.

Snort Configuration

-Installation
Snort is installed using the official Ubuntu package manager to ensure stability and compatibility. Dependencies like libpcap and libpcre are resolved prior to installation (Alodat, 2022).

-Network Configuration
Key configurations in snort.conf include:

- Setting the HOME_NET variable to match the internal network.
- Assigning correct network interface.
- Activating output logging for packet capture. Snort is connected to a dedicated network interface to ensure it captures all incoming traffic in promiscuous mode.

-Rules Management
The study integrates rule sets from Emerging Threats and the Snort community to enhance detection of modern attacks. These are updated weekly to reflect emerging threat signatures (Alodat, 2022).

- Performance Validation
Traffic is captured using tcpdump to ensure Snort processes packets correctly. Ubuntu monitoring tools assess system performance, including CPU load and memory utilization (Suroto, 2018).

2.4 HoneyPy Deployment

-Service Emulation
HoneyPy's configuration file (honeyproxy.py) is modified to simulate services such as SSH, FTP, and HTTP on designated ports. This diversification increases the potential for capturing varied attack types (Kejiou & Bekaroo, 2022).

- Plugin Integration
Plugins are deployed to simulate real-world vulnerabilities for each emulated service, such as invalid command processing or buffer overflow simulations in FTP or HTTP (Singh & Sharma, 2014).

- Network Placement
HoneyPy instances are deployed across various WLAN segments, including isolated VLANs, to cover broader network topologies and increase visibility of attempted intrusions (Kejiou & Bekaroo, 2022).

Attack Simulation and Benchmarking

-Nmap for Port Scanning

Nmap simulates reconnaissance attacks, generating traffic that targets open, closed, and filtered ports. This validates Snort's detection accuracy against standard probing techniques (Suhaimi et al., 2020).

-Brute Force Testing
Hydra scripts generate login attempts on emulated SSH and HTTP services. Detection latency and logging behavior by Snort and HoneyPy are observed and compared.

-Complex Attack Scenarios
Metasploit is employed to generate sophisticated payloads, including DoS and known exploit vectors, to test the system's robustness under advanced threat conditions (Cheng et al., 2014).

Evaluation Strategy

Performance is assessed based on:

- Accuracy of attack detection.
- Speed of alert generation.
- Volume and quality of logged data.
- System stability during high-traffic events.

Each scenario is repeated three times to ensure consistency and statistical reliability. Logs are reviewed for false positives/negatives, and cross-referenced between Snort and HoneyPy for validation.

RESULT AND DISCUSSION

Port Scanning Detection

Port scanning is a critical reconnaissance technique used by attackers to identify vulnerabilities in networked systems. In this study, Snort demonstrated high effectiveness in detecting such scans, particularly those initiated using Nmap. By applying signature-based detection, Snort identified various scanning techniques, including SYN, FIN, and Xmas scans. These were detected through analysis of abnormal TCP flag combinations and deviations from standard packet behaviors. Snort's real-time packet inspection capabilities enabled the detection of stealthy scans, which often evade traditional perimeter defenses.

HoneyPy, configured to emulate multiple vulnerable services, provided complementary support by logging unsolicited access attempts. It recorded connection attempts to simulated services like HTTP, FTP, and SSH, differentiating malicious traffic from legitimate queries based on frequency, source IP behavior, and malformed packet structures. This dual-layer approach provided a holistic view of scanning activity in the WLAN environment.

Among the most commonly detected scanning methods were SYN scans targeting open ports and UDP scans directed at emulated services. Active scanning via Nmap, when directed at HoneyPy ports, generated logs that clearly showed attacker probing patterns. HoneyPy's heuristic-based differentiation allowed for precise categorization of malicious probes.

Brute Force Attack Logging

Brute-force attack simulations were conducted to evaluate HoneyPy's capacity for logging repeated credential attempts. Tools like Hydra were used to launch high-volume authentication attempts on simulated web and SSH services. HoneyPy effectively recorded key metrics, including timestamp, IP source, and attempted credentials.

Behavioral patterns observed during testing included consistent attempts using dictionary-based lists and incremental timing strategies to bypass detection. HoneyPy's built-in logging and alerting functions flagged excessive failures from single IP addresses, prompting further investigation.

Data visualization was achieved using Kibana, which presented attack patterns in dynamic dashboards, highlighting peaks in login attempts and visual correlations between IPs and services targeted. These visual outputs provided security analysts with actionable insights into ongoing attack trends.

Snort complemented this process by identifying traffic anomalies associated with brute-force behavior. Specifically, rules designed to detect repetitive protocol-level attempts were triggered when attackers targeted honeypot-emulated services. This synergy between Snort and HoneyPy allowed for deeper forensics and faster administrative response.

DDoS Traffic Identification

To assess the system's resilience against volumetric threats, a simulated DDoS environment was created. Snort's detection rules successfully identified SYN flood attempts and application-layer attacks utilizing repetitive HTTP GET and POST requests. Detection was facilitated by baseline thresholds that flagged excessive traffic to a single destination over short periods.

Threshold tuning was performed using historical traffic logs, enabling dynamic adjustment based on time-of-day activity levels. This adaptive strategy minimized false positives while maintaining high sensitivity to genuine threats.

Snort logs were parsed using Barnyard2 and analyzed through Snorby, which provided graphical representations of attack intensity, origin, and duration. These tools facilitated comprehensive incident analysis and reporting.

While HoneyPy, due to its low-interaction design, was not fully equipped to emulate extensive DDoS environments, it did capture preliminary flood attempts. Logs revealed origin patterns and initial packet structures, which proved valuable in refining Snort's rule accuracy for future scenarios.

In summary, the integrated use of Snort and HoneyPy significantly enhanced the WLAN's ability to detect port scanning, brute-force, and DDoS attacks. The dual-layered setup provided detailed logs, real-time alerts, and visualization tools that improved threat awareness and response capabilities.

The implementation of Snort and HoneyPy as an integrated security system for wireless LANs demonstrates a significant enhancement in both detection accuracy and forensic preparedness. Passive network defense mechanisms, such as honeypots, play a crucial role in augmenting traditional security measures. Their strength lies in their capacity to attract, log, and analyze unauthorized access attempts without jeopardizing operational systems. Honeypots enable the extraction of detailed insights into attack vectors, attacker behavior, and evolving threat methodologies (Raman & Varadharajan, 2021; Althobaiti, 2019). These capabilities support proactive threat mitigation and the continual refinement of defensive protocols.

However, honeypots are not without limitations. They only register activity explicitly directed at them, thereby providing a partial picture of broader network threats. Overreliance on honeypots can lead to a skewed understanding of actual risk exposure (Fan & Fernández, 2017). Their deployment also demands ongoing maintenance, accurate emulation of real systems, and sophisticated configuration to avoid detection by adversaries (Alotaibi et al., 2022). Furthermore, integrating honeypot outputs with existing monitoring infrastructures may be technically challenging and resource-intensive (Wang et al., 2024).

The study affirms that coupling intrusion detection systems with honeypots significantly improves forensic readiness. IDS tools like Snort monitor network traffic in real time, identifying anomalies and suspicious behavior, while honeypots capture in-depth interaction data during attempted exploits. This dual mechanism enhances incident correlation and evidentiary value for forensic investigations (Li et al., 2019). In practice, cross-referencing IDS alerts with honeypot logs provides clearer attack narratives and strengthens the accuracy of post-incident analysis (Trajanovski & Zhang, 2021). By generating comprehensive threat profiles, integrated systems enable institutions to better prepare for future incidents and optimize threat intelligence workflows (Abbas-Escribano & Debar, 2023).

The use of tools such as the ELK Stack, Splunk, Grafana, and Snorby further bolsters analysis capabilities. These platforms support real-time data visualization, log aggregation, and attack pattern discovery, particularly in simulations involving DDoS vectors (Karthigha et al., 2024; Gao et al., 2024). Despite HoneyPy's limitation as a low-interaction honeypot, its ability to capture preliminary stages of DDoS attempts remains useful. It offers discrete insights into common attack patterns and attacker IP origin, although it may fall short in analyzing more sophisticated or multi-vector DDoS strategies (Ceron et al., 2020; Sibe & Muller, 2022).

Evaluating the effectiveness of real-time detection systems requires clear metrics. Detection rate and false positive rate are primary indicators, where high detection rates reflect a system's capability to correctly identify threats, and low false positives reduce administrative burden (Baykara & Daş, 2019; Nawrocki et al., 2023). Response time is critical for minimizing damage during attacks, while resource consumption metrics ensure operational efficiency without overwhelming network infrastructure (AlFraih & Chen, 2014).

Ethical and legal concerns are also central to honeypot deployment. The potential for privacy violations necessitates strict compliance with data protection laws such as GDPR (Huang et al., 2019). Furthermore, legal ambiguity regarding the use of deception in cybersecurity and the potential liability for unauthorized data capture from attackers must be carefully addressed (Faldi et al., 2023; Wang et al., 2016). Transparent policies and informed consent from network users may enhance ethical standing, though they must be balanced against operational security requirements (Veluchamy & Kathavarayan, 2021).

In conclusion, integrating Snort and HoneyPy into a public WLAN environment strengthens detection and monitoring capabilities. However, careful consideration of ethical, legal, and technical challenges is essential for responsible deployment. When appropriately managed, such systems offer valuable support to public institutions seeking to safeguard digital assets against evolving cyber threats.

CONCLUSION

This study found that integrating Snort and HoneyPy significantly improved WLAN security by enhancing detection accuracy and forensic readiness in public sector environments.

The deployment of Snort proved effective in identifying real-time network anomalies and detecting signature-based attack patterns. Simultaneously, HoneyPy functioned as a low-interaction honeypot capable of emulating vulnerable services and capturing malicious interactions for analysis. The combination of these tools provided a comprehensive detection framework that enhanced threat visibility and supported forensic investigations. Notably, the research found that integration between IDS and honeypot systems facilitated better incident correlation, enriched log analysis, and enabled more informed threat response strategies.

The most notable contribution lies in demonstrating that IDS-honeypot integration not only detects attacks in real time but also strengthens forensic value, providing richer insights for institutional cybersecurity strategies.

The primary contribution of this research lies in its demonstration of a practical, scalable, and cost-effective framework for enhancing WLAN security using open-source tools. It underscores the importance of layered defense mechanisms and highlights the value of combining proactive (IDS) and passive (honeypot) approaches for robust cybersecurity. Future research may integrate machine learning-based anomaly detection, while practitioners should prioritize periodic rule updates, ethical compliance, and resource allocation to ensure sustainable deployment.

REFERENCE

Abbas-Escribano, M., & Debar, H. (2023). An improved honeypot model for attack detection and analysis. https://doi.org/10.1145/3600160.3604993

Abe, S., Tanaka, Y., Uchida, Y., & Horata, S. (2018). Developing deception network system with traceback honeypot in ICS network. SICE Journal of Control Measurement and System Integration, 11(4), 372–379. https://doi.org/10.9746/jcmsi.11.372

AlFraih, A., & Chen, W. (2014). Design of a worm isolation and unknown worm monitoring system based on honeypot. https://doi.org/10.2991/lemcs-14.2014.150

Alodat, I. (2022). Examining wireless networks encryption by simulation of attacks. https://doi.org/10.21203/rs.3.rs-1361227/v1

Alotaibi, F., Al-Dhaqm, A., & Al-Otaibi, Y. (2022). A novel forensic readiness framework applicable to the drone forensics field. Computational Intelligence and Neuroscience, 2022, 1–13. https://doi.org/10.1155/2022/8002963

Althobaiti, A. (2019). An extensive study of honeypot technique. International Journal of Advanced Trends in Computer Science and Engineering, 8(6), 3318–3326. https://doi.org/10.30534/ijatcse/2019/103862019

Bakar, R., et al. (2020). An effective mechanism to mitigate real-time DDoS attack. IEEE Access, 8, 126215–126227. https://doi.org/10.1109/access.2020.2995820

Baykara, M., & Daş, R. (2019). Softswitch: A centralized honeypot-based security approach using software-defined switching for secure management of VLAN networks. Turkish Journal of Electrical Engineering & Computer Sciences, 27(5), 3309–3325. https://doi.org/10.3906/elk-1812-86

Ceron, J., Scholten, C., Pras, A., & Santanna, J. (2020). Mikrotik devices landscape, realistic honeypots, and automated attack classification. https://doi.org/10.1109/noms47738.2020.9110336

Chawda, K., & Patel, A. (2014). Dynamic & hybrid honeypot model for scalable network monitoring. https://doi.org/10.1109/icices.2014.7033844

Cheng, J., Hu, L., Liu, J., Zhang, Q., & Yan, C. (2014). A new mechanism for network monitoring and shielding in wireless LAN. Mathematical Problems in Engineering, 2014(1). https://doi.org/10.1155/2014/620926

Cheng, K., Wu, Z., Li, D., Li, X., & Ren, M. (2022). The TaintDroid based honeypot monitoring system for embedded device. Journal of Physics Conference Series, 2203(1), 012077. https://doi.org/10.1088/1742-6596/2203/1/012077

Dogra, A. (2024). Enhancing DDoS attack detection and network resilience through ensemble-based packet processing and bandwidth optimization. Int Res J Adv Engg Hub, 2(04), 930–937. https://doi.org/10.47392/irjaeh.2024.0130

Faldi, F., Romadoni, D., & SUMADI, M. (2023). The implementation of network server security system using honeypot. JIKO (Jurnal Informatika dan Komputer), 6(2). https://doi.org/10.33387/jiko.v6i2.6385

Fan, W., & Fernández, D. (2017). A novel SDN based stealthy TCP connection handover mechanism for hybrid honeypot systems. https://doi.org/10.1109/netsoft.2017.8004194

Gao, D., et al. (2024). Risk-aware SDN defense framework against anti-honeypot attacks using safe reinforcement learning. International Journal of Network Management, 34(6). https://doi.org/10.1002/nem.2297

Huang, C., Han, J., Zhang, X., & Liu, J. (2019). Automatic identification of honeypot server using machine learning techniques. Security and Communication Networks, 2019, 1–8. https://doi.org/10.1155/2019/2627608

Husák, M., Bartoš, V., Sokol, P., & Gajdoš, A. (2021). Predictive methods in cyber defense: Current experience and research challenges. Future Generation Computer Systems, 115, 517–530. https://doi.org/10.1016/j.future.2020.10.006

Ismail, I., Nor, S., & Marsono, M. (2014). Stateless malware packet detection by incorporating Naive Bayes with known malware signatures. Applied Computational Intelligence and Soft Computing, 2014, 1–8. https://doi.org/10.1155/2014/197961

Karim, I., Vien, Q., Le, T., & Mapp, G. (2017). A comparative experimental design and performance analysis of Snort-based intrusion detection system in practical computer networks. Computers, 6(1), 6. https://doi.org/10.3390/computers6010006

Katakwar, H., et al. (2020). Influence of network size on adversarial decisions in a deception game involving honeypots. Frontiers in Psychology, 11. https://doi.org/10.3389/fpsyg.2020.535803

Kejiou, A., & Bekaroo, G. (2022). A review and comparative analysis of vulnerability scanning tools for wireless LANs. https://doi.org/10.1109/nextcomp55567.2022.9932245

Li, Y., Shi, L., & Feng, H. (2019). A game-theoretic analysis for distributed honeypots. Future Internet, 11(3), 65. https://doi.org/10.3390/fi11030065

Meng, W., & Kwok, L. (2014). Enhancing the performance of signature-based network intrusion detection systems: An engineering approach. HKIE Transactions, 21(4), 209–222. https://doi.org/10.1080/1023697x.2014.970750

Nawrocki, M., et al. (2023). SOK: A data-driven view on methods to detect reflective amplification DDoS attacks using honeypots. https://doi.org/10.48550/arxiv.2302.04614

Pei, J., Chen, Y., & Ji, W. (2019). A DDoS attack detection method based on machine learning. Journal of Physics Conference Series, 1237(3), 032040. https://doi.org/10.1088/1742-6596/1237/3/032040

Raman, J., & Varadharajan, V. (2021). HoneynetCloud investigation model, a preventive process model for IoT forensics. Ingénierie Des Systèmes D'Information, 26(3), 319–327. https://doi.org/10.18280/isi.260309

Rambaldelli, G., et al. (2025). Characterization of small nucleolar RNA retaining transcripts in human normal and cancer cells. Non-Coding RNA Research, 13, 153–161. https://doi.org/10.1016/j.ncrna.2025.05.004

Satria, E., et al. (2021). The investigation on Cowrie honeypot logs in establishing rule signature Snort. IOP Conference Series Earth and Environmental Science, 644(1), 012031. https://doi.org/10.1088/1755-1315/644/1/012031

Singh, R., & Sharma, T. (2014). On the IEEE 802.11i security: A denial-of-service perspective. Security and Communication Networks, 8(7), 1378–1407. https://doi.org/10.1002/sec.1079

Sibe, R., & Muller, S. (2022). Digital forensic readiness of cybercrime investigating institutions in Nigeria: A case study of EFCC and the Nigeria Police Force. 34, 53–57. https://doi.org/10.15439/2022m9438

Suhaimi, H., et al. (2020). Genetic algorithm for intrusion detection system in computer network. Indonesian Journal of Electrical Engineering and Computer Science, 19(3), 1670. https://doi.org/10.11591/ijeecs.v19.i3.pp1670-1676

Surber, J., & Zantua, M. (2022). Intelligent interaction honeypots for threat hunting within the Internet of Things. Journal of the Colloquium for Information Systems Security Education, 9(1), 5. https://doi.org/10.53735/cisse.v9i1.147

Suroto, S. (2018). WLAN security: Threats and countermeasures. JOIV International Journal on Informatics Visualization, 2(4), 232–238. https://doi.org/10.30630/joiv.2.4.133

TAŞÇI, H., et al. (2021). Password attack analysis over honeypot using machine learning. Turkish Journal of Mathematics and Computer Science, 13(2), 388–402. https://doi.org/10.47000/tjmcs.971141

Tian, W., et al. (2019). Honeypot game-theoretical model for defending against APT attacks with limited resources. ETRI Journal, 41(5), 585–598. https://doi.org/10.4218/etrij.2019-0152

Trajanovski, T., & Zhang, N. (2021). An automated and comprehensive framework for IoT botnet detection and analysis. IEEE Access, 9, 124360–124383. https://doi.org/10.1109/access.2021.3110188

Ujjan, R., et al. (2021). Entropy based features distribution for anti-DDoS model in SDN. Sustainability, 13(3), 1522. https://doi.org/10.3390/su13031522

Veluchamy, S., & Kathavarayan, R. (2021). Deep reinforcement learning for building honeypots against runtime DoS attack. International Journal of Intelligent Systems, 37(7), 3981–4007. https://doi.org/10.1002/int.22708

Waili, A. (2023). Analysis of traffic using the Snort tool for the detection of malware traffic. International Journal of Information Technology and Computer Engineering, 33, 30–37. https://doi.org/10.55529/ijitc.33.30.37

Wang, K., et al. (2016). Game-theory-based active defense for intrusion detection in cyber-physical embedded systems. ACM Transactions on Embedded Computing Systems, 16(1), 1–21. https://doi.org/10.1145/2886100

Wang, S., et al. (2023). AI-enabled blockchain and SDN-integrated IoT security architecture for cyber-physical systems. Advanced Control for Applications Engineering and Industrial Systems, 6(2). https://doi.org/10.1002/adc2.131

Wang, L., et al. (2024). AARF: Autonomous attack response framework for honeypots to enhance interaction based on multi-agent dynamic game. Mathematics, 12(10), 1508. https://doi.org/10.3390/math12101508

Xiao, P., Qu, W., Qi, H., & Li, Z. (2015). Detecting DDoS attacks against data center with correlation analysis. Computer Communications, 67, 66–74. https://doi.org/10.1016/j.comcom.2015.06.012

Yao, J., & Chen, J. (2016). The design of website security defense system based on honeypot technology. https://doi.org/10.2991/wartia-16.2016.305