JURNAL MANAJEMEN TEKNOLOGI INFORMATIKA
ISSN 2988-0645 (*print*) dan 2987-8691 (*online*)
Jl. Veteran No.26B, Purus, Kec. Padang Barat, Kota Padang, Sumatera Barat 25115 Website: https://www.jentik.org | E-mail: admin@jentik.org

Multi-source log-based security monitoring for JDIH Kota Padang website

Kevin Maulana Firdaus[1], Rahadian Bisma[2], Agus Salim[3]
kevin.23100@mhs.unesa.ac.id, rahadianbisma@unesa.ac.id, gest.java@gmail.com
[1,2] information System, Surabaya State University, Surabaya, Indonesia
[3] Padang City Department of Communications and Informatics, Padang, Indonesia

Informasi Artikel
Diterima : 15-07-2026
Direview : 03-08-2026
Disetujui : 31-08-2026
Keyword
server log, website security, JDIH, security monitoring, indicator prioritization

Abstrack
Government websites play an essential role in supporting digital public services and therefore require continuous security monitoring to maintain service availability, integrity, and reliability. This study proposes a lightweight security monitoring approach based on the correlation of multi-source server logs for the JDIH Kota Padang website. A descriptive quantitative method was employed using three categories of server logs, namely access logs, authentication logs, and error logs. The analytical process consisted of log preprocessing, event categorization, frequency and percentage analysis, visualization, security indicator mapping, and the calculation of the Security Event Priority Score (SEPS) to prioritize security monitoring activities. The analysis identified 213,356 web requests originating from 20,630 unique IP addresses, including 24,315 HTTP 404 responses, 14,222 HTTP 403 responses, 80,525 failed password events, 61,393 invalid user events, and dominant server errors related to directory access restrictions and missing resources. These findings indicate that server logs provide valuable operational evidence for identifying abnormal access patterns, authentication anomalies, and server configuration issues. The proposed SEPS framework enables administrators to prioritize monitoring efforts and implement appropriate mitigation strategies based on observed security indicators. The study contributes a practical, evidence-based monitoring framework that integrates multiple server log sources without requiring intrusive testing or complex machine learning models, making it suitable for periodic security monitoring in local government digital services.

A. Introduction

Digital transformation has increased the dependence of public institutions on web-based information systems. Government websites are no longer used only as publication media, but also as digital service channels that support transparency, public information access, administrative efficiency, and public service delivery. In the context of e-government, public digital services must be maintained to remain accessible, stable, secure, and accountable. Previous research on e-government website vulnerability assessment shows that government websites require systematic security evaluation to support service reliability and information security [1]. In addition, the NIST Cybersecurity Framework emphasizes that cybersecurity management should include the ability to identify, protect, detect, respond to, and recover from cybersecurity risks in digital systems [2]. Therefore, security monitoring becomes an important activity in maintaining the reliability of government digital services.

Jaringan Dokumentasi dan Informasi Hukum (JDIH) Kota Padang, or the Legal Documentation and Information Network of Padang City, is a digital service that provides legal documentation and regulatory information to the public. As a publicly accessible website, the JDIH Kota Padang website receives various types of requests from legitimate users, search engine crawlers, automated bots, and other sources that require further monitoring. Because the website is connected to the internet and functions as a public legal information service, its operational activities must be evaluated periodically to identify abnormal access patterns, authentication problems, server errors, and potential service disruptions. Without continuous monitoring, technical problems and suspicious activities may remain undetected until they affect service availability, system reliability, or public trust.

One source of technical evidence that can be used for security monitoring is the server log. Server logs are records generated by a system to document activities, events, and errors. Access logs record web requests, HTTP status codes, HTTP methods, user agents, and requested paths. Authentication logs record login activities, failed login attempts, invalid users, and session activities. Error logs record system and application errors, including missing files, forbidden directories, SSL/TLS problems, upstream timeout, and backend-related errors. Log management is an important component of information security because logs can support monitoring, detection, analysis, and incident response [3]. Information security testing and assessment also emphasizes the importance of collecting technical evidence to evaluate system weaknesses and determine appropriate recommendations [4]. In intrusion detection and prevention, system and network event records can support early identification of abnormal activities [5]. Thus, server log analysis is relevant for government digital services that require continuous monitoring, accountability, and evidence-based security evaluation.

Cybersecurity monitoring is not only a technical activity, but also part of cybersecurity governance. Previous work by Firdaus emphasized that security management should not rely solely on technical controls, but must integrate human factors, governance processes, policies, standard operating procedures, logging, monitoring mechanisms, and technical controls into a unified framework [6]. This perspective is relevant to local government website security because log-based monitoring requires not only technical data analysis, but also administrative procedures, access control, incident response, and continuous evaluation by system administrators. In this context, the effectiveness of security monitoring depends on how server log findings are interpreted and transformed into practical follow-up actions.

Several previous studies have examined system log analysis and anomaly detection using various computational approaches. DeepLog introduced deep learning-based anomaly detection and diagnosis from system logs [7]. He et al. reported system log analysis for anomaly detection based on operational experience [8]. Drain proposed an online log parsing approach using a fixed-depth tree structure [9]. LogAnomaly developed an

unsupervised approach for detecting sequential and quantitative anomalies in unstructured logs [10]. Other studies have discussed robust log-based anomaly detection on unstable log data [11], automated log parsing tools and benchmarks [12], and transformer-based log anomaly detection using BERT architecture [13]. These studies show that logs are valuable data sources for identifying abnormal behavior in information systems.

Broader cybersecurity studies also provide important foundations for security monitoring and anomaly detection. Data mining and machine learning have been widely used for cybersecurity intrusion detection [14], while network anomaly detection has been discussed as an important approach for identifying unusual traffic behavior [15]. Studies on intrusion detection systems describe various techniques, datasets, and implementation challenges [16], including systematic reviews of machine learning and deep learning approaches in network intrusion detection [17]. Other research has discussed cybersecurity data science [18], deep learning for cybersecurity intrusion detection [19], general anomaly detection concepts [20], anomaly-based network intrusion detection [21], and the limitations of applying machine learning to intrusion detection in open environments [22]. Related intrusion detection literature also discusses benchmark datasets and system foundations, including UNSW-NB15 [23], intrusion traffic characterization [24][24], KDD CUP 99 [25], real-time network intrusion detection [26], intrusion detection taxonomy [27], base-rate fallacy [28], comprehensive IDS review [29], and network-based intrusion detection dataset surveys [30].

Although previous studies provide strong foundations for vulnerability assessment, log analysis, anomaly detection, and intrusion detection, several gaps remain in the context of local government digital services. Many studies focus on vulnerability scanning, penetration testing, advanced machine learning, or single-source log analysis. These approaches are useful, but they may require specific tools, technical expertise, or complex infrastructure. In local government website operations, administrators often need a practical and lightweight monitoring approach that can use server data already available in the system. Therefore, there is a need for a security monitoring approach that integrates multiple server log sources and maps the findings into practical indicators and recommendations.

The problem addressed in this study is the limited discussion of lightweight security monitoring based on integrated server logs in local government digital services. Vulnerability assessment and penetration testing can identify weaknesses at a specific time, but log-based monitoring can provide operational evidence from actual system activities. This study therefore focuses on access logs, authentication logs, and error logs as complementary sources for identifying request patterns, authentication-related events, error categories, and practical security recommendations for the JDIH Kota Padang website.

The novelty of this study lies in the development of a multi-source log-based security monitoring approach for a local government digital service by integrating access logs, authentication logs, and error logs. Unlike studies that focus on a single log source, vulnerability scanning, or advanced anomaly detection models, this study maps multi-source log findings into security indicators and actionable recommendations for website administrators. The proposed approach is lightweight because it uses log data already generated by the server and does not require intrusive testing in its initial implementation.

This study aims to analyze access logs, authentication logs, and error logs of the JDIH Kota Padang website to identify system activity patterns, potential abnormal activities, and security monitoring recommendations. The main contribution of this study is a practical log-based indicator mapping that connects server log findings with security implications and actionable recommendations. Practically, this study can help local government website administrators conduct periodic log monitoring, prioritize security improvements, and strengthen website security through evidence-based evaluation.

B. Methods

1. Research design

This study used a descriptive quantitative approach based on multi-source web server log analysis. The method was designed to identify security monitoring indicators from three types of server logs, namely access logs, authentication logs, and error logs. This approach was selected because the study did not aim to build a predictive machine learning model, but to extract measurable evidence from actual server activities and map the findings into practical security recommendations for the JDIH Kota Padang website.

The proposed method consists of log data collection, preprocessing, event categorization, metric calculation, visualization, abnormal activity indication, and security recommendation mapping. The main contribution of the method is the integration of three different log sources into a lightweight monitoring workflow. This integration allows the analysis to capture web request patterns, authentication-related activities, and server or application error patterns in one structured monitoring approach. The complete workflow of the proposed method is illustrated in Figure 1.

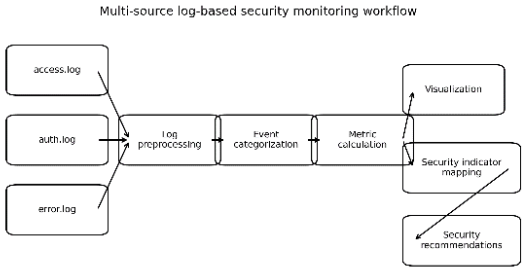

Figure 1. Multi-source log-based security monitoring workflow

2. Data source and preprocessing

The research object was the JDIH Kota Padang website, which functions as a local government digital service for public legal documentation and regulatory information. The dataset used in this study consisted of three server log files generated from the operational environment of the website. The access log covered the period from 10 to 24 February 2026, the authentication log covered February 2026, and the error log covered the period from 11 to 25 February 2026. Table 1 summarizes the log data sources used in this study. The table presents each log source, observation period, and main analytical focus.

Table 1. Log data sources

Log source	Period	Main focus
access.log	10–24 February 2026	HTTP status codes and methods
auth.log	February 2026	Login and SSH authentication events
error.log	11–25 February 2026	Server and backend error categories

Because this study used descriptive log analysis rather than supervised learning, the dataset was not divided into training, validation, and testing sets. Instead, all available log records were used as observational data for categorization, frequency calculation, percentage distribution, visualization, and security indicator mapping. Sensitive information such as IP addresses, usernames, internal paths, and other identifiable technical details was not displayed in the manuscript to reduce security and privacy risks.

The preprocessing stage was conducted to transform raw log records into structured analytical variables. In the access log, relevant fields were extracted from each request record, including timestamp, HTTP method, requested path, HTTP status code, and user agent. In the authentication log, each record was parsed based on

authentication event patterns, such as failed password, invalid user, successful login, session opened, session closed, connection closed, and connection reset. In the error log, records were categorized based on error messages and severity, including directory forbidden, file not found, SSL/TLS error, upstream timeout, PHP/FastCGI error, and other errors.

3. Metric calculation and visualization

The main metrics used in this study were event frequency, percentage distribution, dominant category, and security indication. Event frequency was used to count how many times a specific event appeared in the log. Percentage distribution was used to compare the proportion of each category within the total number of records. Dominant category was used to identify the most frequent event in each log source. Security indication was used to interpret whether a specific finding required further monitoring or administrative follow-up.

The percentage of each event category was calculated using Equation (1).

$$P_c = \frac{N_c}{N} \times 100\%$$

In Equation (1), (P_c) represents the percentage of category (c), (N_c) represents the number of records in category (c), and (N) represents the total number of records in the corresponding log source. This equation was used to calculate the percentage distribution of HTTP status codes, HTTP methods, and error categories.

The visualization stage was conducted to present the distribution of key findings from each log source. The access log was visualized through HTTP status code distribution and HTTP method distribution. The authentication log was visualized through authentication event distribution. The error log was visualized through error category distribution. These visualizations were used to support interpretation and to help identify dominant patterns that require further monitoring.

4. Security indicator mapping

Security indicator mapping was conducted by connecting log findings with possible security or operational issues. A high number of 404 status codes was interpreted as an indication of invalid path access, broken links, unavailable resources, or scanning activity. A high number of 403 status codes was interpreted as an indication of restricted resource access attempts. The appearance of PUT, CONNECT, and PROPFIND methods was interpreted as unusual HTTP method usage that may indicate probing against server configuration.

Authentication-related events were also mapped into security indicators. Failed password and invalid user events were interpreted as possible unauthorized login attempts, username enumeration, or brute force SSH activity. However, these findings were not treated as evidence of a successful attack because further validation by the system administrator is required. Error log categories were mapped into operational and configuration indicators. Directory forbidden and file not found events were linked to restricted directory access, unavailable paths, or repeated probing, while SSL/TLS errors and upstream timeout were linked to secure connection issues and backend performance problems. The evaluation in this study was performed by examining whether the proposed workflow could produce measurable findings from each log source and map them into practical security recommendations.

C. Results and Discussions

1. Web request patterns from access logs

The access log analysis was conducted to identify request patterns received by the JDIH Kota Padang website during the observation period. The access log covered

the period from 10 to 24 February 2026 and was generated from an Ubuntu 22.04 server running Nginx. The analysis identified 213,356 requests from 20,630 unique IP addresses. These findings indicate that the website received a high volume of traffic from various sources. Table 2 summarizes the main access log findings, including total requests, unique IP addresses, dominant status codes, and dominant HTTP methods.

Table 2. Access log summary

Variable	Result
Total requests	213,356
Unique IP addresses	20,630
Status code 200	126,442
Status code 301	36,703
Status code 404	24,315
Status code 403	14,222
GET method	190,862
POST method	20,785
Unusual methods	PUT, CONNECT, PROPFIND

To provide a clearer visual comparison, the HTTP status code distribution is presented in Figure 2.

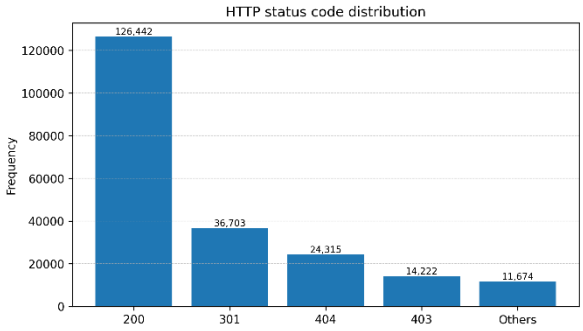

Figure 2. HTTP status code distribution

In addition to status code analysis, HTTP method analysis was also conducted to identify the types of requests received by the server, as shown in Figure 3.

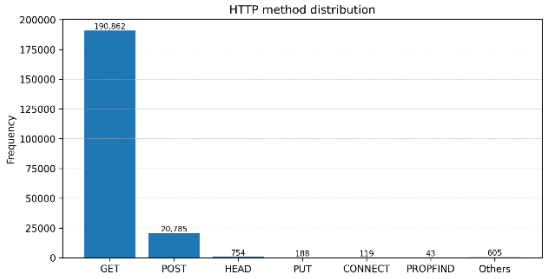

Figure 3. HTTP method distribution

Figure 2 confirms that successful requests were dominant, while 404 and 403 responses remained important indicators for further monitoring. The 200 status code was the most dominant response with 126,442 requests or 59.26% of the total access log records. This indicates that most requests were successfully served. The 301 status code appeared in 36,703 requests or 17.20%, indicating redirected requests. However, the presence of 24,315 requests with 404 status codes and 14,222 requests with 403 status codes should be monitored because these responses may indicate invalid paths, unavailable resources, broken links, scanning activity, or restricted resource access attempts.

The relatively high number of 404 responses is important because repeated access to unavailable paths can reflect broken internal links, outdated indexed pages, bot requests, or scanning indications. The 403 responses are also relevant because they may represent attempts to access restricted files or directories. These findings support previous studies that emphasize the need for continuous monitoring and security evaluation on government websites [1]. They also align with the log management perspective, which states that log records can provide useful evidence for monitoring, detection, analysis, and incident response [3].

Figure 3 shows that GET and POST were dominant, while PUT, CONNECT, and PROPFIND appeared in smaller numbers and should be reviewed by the administrator. GET was the dominant method with 190,862 requests or 89.46%, followed by POST with 20,785 requests or 9.74%. This pattern is normal for a public website because users generally access pages and submit data through standard web interactions. However, PUT, CONNECT, and PROPFIND are not commonly required for ordinary public website access and therefore need administrative review.

The presence of unusual HTTP methods does not automatically indicate a successful attack. However, from a monitoring perspective, these requests are important because they may indicate probing against the web server configuration. PUT may relate to resource modification attempts, CONNECT may relate to proxy or tunneling behavior, and PROPFIND is commonly associated with WebDAV-related requests. The implication of this finding is that administrators should restrict unused HTTP methods at the web server level. This result supports the broader cybersecurity monitoring principle that abnormal technical events should be identified early and followed by preventive control measures [2], [4].

2. Authentication activity and server error categories

The authentication log analysis was conducted to identify login-related activities and possible unauthorized access attempts. The authentication log covered February 2026 and contained 450,196 lines. Table 3 summarizes the authentication log findings. The authentication event distribution is visualized in Figure 4 to highlight the most frequent login-related activities.

Table 3. Authentication log summary

Event	Frequency
Failed password	80,525
Invalid user	61,393
Successful login	179
Session opened	31,162
Session closed	31,076
Connection closed	61,394
Connection reset	9,058

The authentication event distribution is visualized in Figure 4 to highlight the most frequent login-related activities.

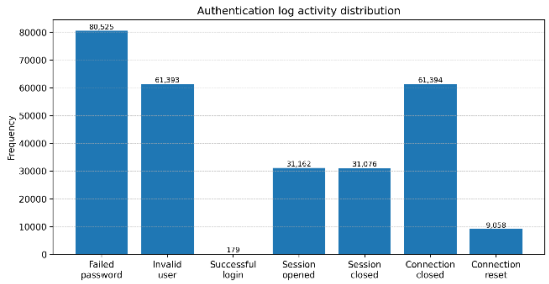

Figure 4. Authentication log activity distribution

Figure 4 shows that failed password and invalid user events were the most prominent authentication-related findings, indicating the need for SSH access monitoring. The analysis identified 80,525 failed password events and 61,393 invalid user events. These two categories were the most important authentication-related findings because they may indicate possible unauthorized login attempts, username enumeration, or brute force SSH activity.

The high number of failed password and invalid user events should be treated as an important security monitoring signal. However, this finding must be interpreted carefully because the log does not provide sufficient evidence to conclude that a successful attack or compromise occurred. Further validation by the system administrator is required to confirm whether the events were caused by automated brute force attempts, misconfigured services, repeated administrative errors, or other causes. This interpretation is consistent with intrusion detection literature, which emphasizes that abnormal events require validation and contextual analysis before being treated as confirmed incidents [5], [14], [16].

The error log analysis was conducted to identify [14] server, configuration, and backend-related problems. The error log covered the period from 11 to 25 February 2026 and was generated from an Ubuntu 22.04 server running Nginx and PHP 8.1-FPM. Table 4 summarizes the error log findings. The distribution of error categories is visualized in Figure 5 to compare the dominant server and configuration-related errors.

Table 4. Error log summary

Error category	Frequency
Directory forbidden	182
File not found	164
SSL/TLS error	115
Upstream timeout	19
PHP/FastCGI error	4
Others	8

The distribution of error categories is visualized in Figure 5 to compare the dominant server and configuration-related errors.

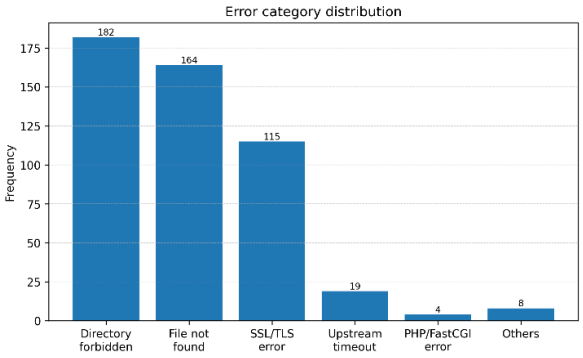

Figure 5. Error category distribution

Figure 5 shows that directory forbidden and file not found were the dominant error categories, indicating the need for permission review and link auditing. Directory forbidden was the most frequent error category with 182 events or 36.99%, followed by file not found with 164 events or 33.33%. SSL/TLS error appeared in 115 events or 23.37%, while upstream timeout appeared in 19 events or 3.86%. These findings indicate that access-related and configuration-related errors were dominant in the error log.

The dominance of directory forbidden and file not found indicates the need to review directory access rules, permission settings, unavailable paths, and broken

links. SSL/TLS errors should also be evaluated because secure connection issues may affect user trust and service reliability. Upstream timeout events indicate that backend responsiveness should be monitored, especially when the website depends on PHP-FPM or database services. These findings show that error logs can provide operational insight that complements access log and authentication log analysis. Compared with previous log analysis studies that focus on anomaly detection models [7]-[13], this study emphasizes practical interpretation and administrative follow-up for local government website monitoring.

3. Security recommendation mapping, limitations, and future work

The main contribution of this study is the mapping of multi-source log findings into security indicators and practical recommendations. Table 5 presents the security recommendation mapping derived from access log, authentication log, and error log findings. The mapping connects raw log evidence with security indications and practical follow-up actions for website administrators.

Table 5. Security recommendation mapping

Main finding	Security indication	Recommendation
High 404 and 403 responses	Invalid or restricted access attempts	Link audit and permission review
PUT, CONNECT, and PROPFIND requests	Unusual HTTP method usage	Restrict unused HTTP methods
Failed password and invalid user events	Possible brute force SSH activity	Apply SSH hardening and fail2ban
Directory forbidden and file not found	Access or path configuration issue	Review directory rules and missing paths
SSL/TLS error and upstream timeout	Connection or backend issue	Review SSL/TLS and backend performance

The results indicate that the JDIH Kota Padang website requires periodic monitoring of abnormal access patterns, restricted access attempts, unusual HTTP methods, authentication failures, and backend errors. The high number of failed password and invalid user events should be followed up with SSH hardening, key-based authentication, root login restriction, access policy review, and fail2ban implementation. The 404 and 403 responses should be followed up through link auditing, repeated invalid path monitoring, permission checking, and directory listing protection. The unusual HTTP methods should be restricted if they are not required by the application, while SSL/TLS errors and upstream timeout events should be reviewed to maintain secure connection quality and backend performance.

Compared with vulnerability assessment studies that focus on scanning and testing weaknesses in web applications [1], this study focuses on log-based monitoring using operational evidence generated by the server. This approach is useful for local government digital services because server logs are already available and can be analyzed periodically without requiring intrusive testing. The proposed approach also complements previous research on cybersecurity governance, which emphasizes that security management requires the integration of people, processes, and technology [6]. In this study, log monitoring is not only treated as a technical task, but also as a governance activity that requires administrator interpretation, standard operating procedures, and continuous follow-up actions.

Theoretically, this study contributes to the field of security monitoring by demonstrating how access logs, authentication logs, and error logs can be integrated into a single lightweight monitoring framework. Practically, the findings can help local government website administrators prioritize security improvements based on evidence from server logs. This is especially relevant for public digital services that require continuous availability, public trust, and accountable security practices.

This study has several limitations. The analysis was descriptive and did not apply automatic anomaly detection, machine learning, or predictive modeling. The study also did not perform penetration testing, exploitation, or forensic validation, so

the findings should be interpreted as security indications rather than confirmed incidents. In addition, IP addresses, usernames, and sensitive paths were not disclosed in the manuscript to reduce security and privacy risks. These limitations affect the depth of attribution and incident confirmation, but they do not reduce the usefulness of the results as an initial monitoring framework.

Future research may extend this study by developing an automated log monitoring dashboard, applying anomaly detection algorithms, comparing log patterns across longer observation periods, and evaluating multiple local government websites. Future work may also integrate alert prioritization, correlation rules, and administrator feedback to improve the accuracy and usefulness of log-based monitoring. Such development would allow the proposed approach to evolve from descriptive monitoring into more proactive and intelligent security monitoring for public digital services.

D. Conclusions

This study demonstrates that multi-source server log analysis can be used as a practical and lightweight approach for monitoring the security of the JDIH Kota Padang website as a local government digital service. By integrating access logs, authentication logs, and error logs, this study identified web request patterns, authentication-related events, and server error categories that provide early indicators for security monitoring. The access log analysis showed that most requests were successfully processed, but the presence of high 404 and 403 responses indicates the need to monitor invalid paths, broken links, scanning indications, and restricted resource access attempts. The HTTP method analysis showed that GET and POST were dominant, while the appearance of PUT, CONNECT, and PROPFIND should be reviewed because these methods are not commonly required for ordinary public web access. The authentication log analysis indicated possible unauthorized login attempts or brute force SSH activity through high failed password and invalid user events, although these findings require administrator validation before being concluded as successful attacks. The error log analysis showed that directory forbidden and file not found were the most frequent categories, followed by SSL/TLS errors and upstream timeout, indicating the need for configuration review and backend performance monitoring. These findings answer the research objective by showing that access logs, authentication logs, and error logs can complement each other in identifying security monitoring indicators for a government website. The original contribution of this study is the development of a multi-source log-based security monitoring approach that maps raw server log findings into security indicators and actionable recommendations. Practically, the findings can support periodic log monitoring, SSH hardening, HTTP method restriction, link auditing, permission checking, directory listing protection, SSL/TLS review, rate limiting, user-agent monitoring, and backend performance evaluation. Future research may extend this approach by developing an automated log monitoring dashboard, applying anomaly detection algorithms, and comparing log patterns across longer observation periods or multiple local government websites.

E. Acknowledgements

The authors declare no conflict of interest. This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

F. References

E. Z. Darojat, E. Sediyono, and I. Sembiring, "Vulnerability Assessment Website E-Government dengan NIST SP 800-115 dan OWASP Menggunakan Web Vulnerability Scanner," *JURNAL SISTEM INFORMASI BISNIS*, vol. 12, no. 1, pp. 36–44, Sep. 2022, doi: 10.21456/vol12iss1pp36-44.

"The NIST Cybersecurity Framework (CSF) 2.0," Feb. 2024. doi: 10.6028/NIST.CSWP.29.

K. Kent and M. Souppaya, "Special Publication 800-92 Guide to Computer Security Log Management Recommendations of the National Institute of Standards and Technology."

K. A. Scarfone, M. P. Souppaya, A. Cody, and A. D. Orebaugh, "Technical guide to information security testing and assessment.," Gaithersburg, MD, 2008. doi: 10.6028/NIST.SP.800-115.

K. A. Scarfone and P. M. Mell, "Guide to Intrusion Detection and Prevention Systems (IDPS)," Gaithersburg, MD, 2007. doi: 10.6028/NIST.SP.800-94.

K. M. Firdaus and U. N. Surabaya, "JURNAL MANAJEMEN TEKNOLOGI INFORMATIKA Integrasi Faktor Manusia dalam Tata Kelola Keamanan Siber Berbasis Cloud: Studi Pengembangan Framework," *Jl. Veteran No.26B*, p. 25115.

M. Du, F. Li, G. Zheng, and V. Srikumar, "DeepLog: Anomaly detection and diagnosis from system logs through deep learning," in *Proceedings of the ACM Conference on Computer and Communications Security*, Association for Computing Machinery, Oct. 2017, pp. 1285–1298. doi: 10.1145/3133956.3134015.

S. He, J. Zhu, P. He, and M. R. Lyu, "Experience Report: System Log Analysis for Anomaly Detection," in *Proceedings - International Symposium on Software Reliability Engineering, ISSRE*, IEEE Computer Society, Dec. 2016, pp. 207–218. doi: 10.1109/ISSRE.2016.21.

P. He, J. Zhu, Z. Zheng, and M. R. Lyu, "Drain: An Online Log Parsing Approach with Fixed Depth Tree," in *Proceedings - 2017 IEEE 24th International Conference on Web Services, ICWS 2017*, Institute of Electrical and Electronics Engineers Inc., Sep. 2017, pp. 33–40. doi: 10.1109/ICWS.2017.13.

W. Meng *et al.*, "LogAnomaly: Unsupervised Detection of Sequential and Quantitative Anomalies in Unstructured Logs," 2019.

X. Zhang *et al.*, "Robust log-based anomaly detection on unstable log data," in *ESEC/FSE 2019 - Proceedings of the 2019 27th ACM Joint Meeting European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, Association for Computing Machinery, Inc, Aug. 2019, pp. 807–817. doi: 10.1145/3338906.3338931.

J. Zhu *et al.*, "Tools and Benchmarks for Automated Log Parsing," Jan. 2019, [Online]. Available: http://arxiv.org/abs/1811.03509

U. DigitalCommons, U. All Graduate Theses, and H. Guo, "LogBERT: Log Anomaly Detection via BERT," 2021.

A. L. Buczak and E. Guven, "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection," *IEEE Communications Surveys and Tutorials*, vol. 18, no. 2, pp. 1153–1176, Apr. 2016, doi: 10.1109/COMST.2015.2494502.

M. Ahmed, A. Naser Mahmood, and J. Hu, "A survey of network anomaly detection techniques," Jan. 01, 2016, *Academic Press*. doi: 10.1016/j.jnca.2015.11.016.

A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, Dec. 2019, doi: 10.1186/s42400-019-0038-7.

Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 1, Jan. 2021, doi: 10.1002/ett.4150.

I. H. Sarker, A. S. M. Kayes, S. Badsha, H. Alqahtani, P. Watters, and A. Ng, "Cybersecurity data science: an overview from machine learning perspective," *J. Big Data*, vol. 7, no. 1, Dec. 2020, doi: 10.1186/s40537-020-00318-5.

M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, “Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study,” *Journal of Information Security and Applications*, vol. 50, Feb. 2020, doi: 10.1016/j.jisa.2019.102419.
V. Chandola, “Anomaly Detection : A Survey,” 2009.
P. García-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, “Anomaly-based network intrusion detection: Techniques, systems and challenges,” *Comput. Secur.*, vol. 28, no. 1–2, pp. 18–28, Feb. 2009, doi: 10.1016/j.cose.2008.08.003.
R. Sommer and V. Paxson, “Outside the Closed World: On Using Machine Learning For Network Intrusion Detection.”
N. Moustafa and J. Slay, “UNSW-NB15: A Comprehensive Data set for Network Intrusion Detection systems (UNSW-NB15 Network Data Set).” [Online]. Available: https://cve.mitre.org/
Harry Setya Hadi and Nicodemus Rahanra, “Explainable Artificial Intelligence Methods for Autonomous Robot Decision Making: A Multi Agent Framework with Safety Assurance and Ethical Constraint Optimization ”, *ISR*, vol. 1, no. 1, pp. 65–81, Jan. 2026.
I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, “Toward generating a new intrusion detection dataset and intrusion traffic characterization,” in *ICISSP 2018 - Proceedings of the 4th International Conference on Information Systems Security and Privacy*, SciTePress, 2018, pp. 108–116. doi: 10.5220/0006639801080116.
IEEE Staff, *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*. I E E E, 2009.
V. Paxson, “Bro: A System for Detecting Network Intruders in Real-Time.”
H. Debar, M. Dacier, and A. Wespi, “Towards a Taxonomy o f I n trusion-Detection Systems.”
S. Axelsson, “The Base-Rate Fallacy and the Difficulty of Intrusion Detection,” 1094. [Online]. Available: http://www.ce.chalm-
H. J. Liao, C. H. Richard Lin, Y. C. Lin, and K. Y. Tung, “Intrusion detection system: A comprehensive review,” Jan. 2013. doi: 10.1016/j.jnca.2012.09.004.
M. Ring, S. Wunderlich, D. Scheuring, D. Landes, and A. Hotho, “A Survey of Network-based Intrusion Detection Data Sets,” 2019.
Setya Hadi, R. Rauf, Agus Salim, and Kevin Maulana Firdaus, “Smart Seismic Intelligence Machine Learning for Spatial Clustering and Earthquake Magnitude Prediction in Indonesia”, *ZTR*, vol. 8, no. 1, pp. 65–72, Mar. 2026.