OPEN ACCESS
VOL. 8, NO. 2, PP. 105–120, MAY 2026

ISSN : 2622-8106 (ONLINE)
DOI: 10.20895/INISTA.V8I2.2070

Journal of Informatics, Information System, Software Engineering and Applications (INISTA)

Evaluation of Security Vulnerabilities in the Single Submission Pengangkut System Using OWASP Top 10

Adi Gilang Wahyu Aji[1], Ika Kurniawati[2,*]

[1]Department of Information Systems, Nusa Mandiri University, Jakarta, Indonesia
[2]Department of Information Systems, Bina Sarana Informatika University, Jakarta, Indonesia

*Corresponding email: ika.iki@bsi.ac.id

Received: 18-11-2025; Revised: 17-05-2026; Accepted: 10-06-2026.

Abstract

International trade in the era of globalization has rapidly grown thanks to information and communication technology (ICT), but this also presents new challenges related to data security and user information protection. In Indonesia, the National Single Window (LNSW) utilizes the Single Submission Pengangkut web application to support international trade. Although this application plays an important role, potential security vulnerabilities could lead to data breaches and financial losses. This study aims to test the security vulnerabilities of the application using Penetration Testing methods based on the OWASP Top 10 standard. Testing was conducted using tools such as Nmap, Nessus, Kali Linux, and Burp Suite to identify and exploit vulnerabilities. The results of the testing revealed three vulnerabilities that did not pass the security test: Insecure Design, Vulnerable and Outdated Components, and Identification and Authentication Failures. Based on assessments using the Common Vulnerability Scoring System (CVSS), it was found that Insecure Design has a Medium vulnerability rating, while Vulnerable and Outdated Components and Identification and Authentication Failures fall under the info category, meaning they do not directly impact the application's security. To address these vulnerabilities, it is recommended to implement restrictions in the document input process, perform regular software updates, and implement multi-factor authentication (MFA). This study shows that applying the OWASP Top 10 as a guideline in penetration testing is effective for identifying and evaluating security vulnerabilities in the Single Submission Pengangkut web application.

Keywords: *OWASP Top 10, Pentest, Security Analysis, Vulnerability Testing, Web Security*

Corresponding Author:
*Ika Kurniawati
Department of Information Systems, Bina Sarana Informatika University
Jakarta, Indonesia
Email: ika.iki@bsi.ac.id

I. INTRODUCTION

In the current era of globalization, the economic sector particularly international trade has experienced rapid growth. This development is evident in the increasing speed of cross-border exchanges of goods and services, which rely heavily on advances in technology to streamline these processes. These technological advancements create significant opportunities for progress in the international trade sector export–import [1]. The growth of international trade (export–import) cannot be separated from the role of information and communication technology (ICT). ICT encompasses all aspects involving technology, engineering, and data management techniques used for controlling, processing, and utilizing information. This adaptation also requires a cultural shift toward multiculturalism [2]. Therefore, mastery of ICT is essential to support the technology-driven operations within this sector.

The rapid advancement of ICT, however, also brings new challenges, particularly related to data security and user information protection. Various organizations, both governmental and private, now rely on web-based systems to support operations and manage transactions. As the use of technology expands, the risks of data breaches and cyberattacks have also increased significantly. Numerous cases of data leaks across platforms indicate that personal information security still demands serious attention from all institutions, including government bodies. Web servers are frequently targeted by cyberattacks from irresponsible individuals—commonly known as hackers—who exploit vulnerabilities to obtain confidential information from organizations or companies, often resulting in significant harm.

In Indonesia, the National Single Window (LNSW) has taken initiatives to facilitate export and import activities by utilizing information technology, including the Single Submission Pengangkut web application. This application is designed to simplify the management of cross-border cargo operations. However, the high potential for cyber threats targeting web-based systems makes security a critical aspect of this application. Security vulnerabilities may lead to substantial losses for service providers and users, including reputational damage, data leaks, and financial losses due to delays in vessel departures, which increase operational costs. Disruptions in export–import shipping schedules can delay the delivery of goods to consumers, ultimately impacting business stakeholders [3]. Therefore, it is essential to conduct security testing to evaluate the system's security level.

System security disruptions have occurred in several government-managed systems. One example is the cyberattack on the temporary National Data Center 2, which caused system failures that could not be restored [4]. Other incidents include attacks on government websites such as the National Civil Service Agency (BKN), which resulted in the leakage of civil servant data [5]. Although evaluations of these incidents have been conducted, comprehensive security testing has not yet been performed [6]. While no data breach has been reported in the LNSW system, proactive security testing is necessary to prevent potential cyberattacks caused by unaddressed vulnerabilities. Regular security assessments are therefore required to ensure that all vulnerabilities can be detected and mitigated, enabling the application to operate securely and protecting user data while minimizing possible losses.

One widely used method for testing security vulnerabilities in web-based applications is the Open Web Application Security Project (OWASP) approach. OWASP is a non-profit organization established in the United States on April 21, 2004, dedicated to developing open and freely accessible security testing frameworks. One of its most recognized methodologies is the OWASP Top 10, which outlines the ten most critical security risks that may threaten website security [7].

Previous studies using the OWASP method have shown promising results. Studied by [8] demonstrated that applied the OWASP Top 10 as a reference standard in penetration testing is effective in identifying and evaluating significant vulnerabilities in a university's academic information system. The scan detected 23 vulnerabilities, 20 of which fell into categories identified by the OWASP Top 10. The findings provide valuable insights for improving the system's overall security. Another study by [9] analyzed the security of the website akprind.ac.id using the OWASP methodology to assess its security level. Through penetration testing based on OWASP Top 10 guidelines, 13 vulnerabilities were identified, including one high-severity risk, ten medium-severity risks, and two low-severity risks [10]. Based on these previous studies, OWASP remains an effective method for conducting penetration testing. The findings indicate that OWASP techniques are capable of identifying security weaknesses in web-based applications.

In this study, the author evaluates security vulnerabilities in the Single Submission Pengangkut application using penetration testing based on the OWASP Top 10 security standards. The results of this assessment are expected to serve as a reference for strengthening the system's security and minimizing potential entry points for cyberattacks.

II. RESEARCH METHOD

In this section, the steps of the method used in this study will be explained. The method used consists of 4 steps, namely problem identification, data collection, security testing, security analysis, and conclusion. The research stages can be seen in Figure 1.

Fig. 1. Research Method.

1. Problem identification, which involves analyzing potential security vulnerabilities within the system.
2. Data collection through observations at the LNSW Information Technology Directorate and interviews with system administrators.
3. Security testing using a white-box approach with tools such as Nmap, Nessus, Kali Linux, and Burp Suite to identify and exploit vulnerabilities.

A. Planning and Information Gathering

In this stage, the researcher conducted an assessment of the SSm Pengangkut website using a white-box approach, where prior access to system information and internal control was provided. The researcher gathered information related to user access levels and the operational use of the application to establish an understanding of how the system functions and to identify potential security weaknesses.

B. Scanning

The researcher performed scanning on the SSm Pengangkut web application using Nmap and Nessus.

C. Vulnerability Exploitation

The researcher conducted security vulnerability testing on the SSm Pengangkut web application using Kali Linux and Burp Suite, referring to the ten categories of vulnerabilities listed in the OWASP Top 10. Open Web Application Security Project (OWASP) is a non-profit organization focused on improving information security, particularly web application security. OWASP provides a framework used to secure websites. One of its key projects is the OWASP Top Ten, which is a list of security vulnerabilities that could be exploited in cyberattacks against websites. The list is regularly updated to reflect technological developments [11].

The OWASP Top 10 vulnerabilities for 2021 are as follows [12]:

1. **Broken Access Control:** Refers to poor implementation of access control, allowing users to access data or features beyond their assigned role.
2. **Cryptographic Failures:** Failures in protecting sensitive data—such as passwords or personal information—through proper encryption techniques.
3. **Injection:** Occurs when attackers insert malicious code, such as SQL or scripts, that can be executed by the application and potentially damage the system.
4. **Insecure Design:** Refers to design flaws within the application that make it vulnerable to attacks.
5. **Security Misconfiguration:** Involves incorrect or default configuration settings that can be exploited by attackers.
6. **Vulnerable and Outdated Components:** The use of unsupported or outdated software components that increase the risk of attacks.
7. **Identification and Authentication Failures:** Vulnerabilities in identifying or verifying users, often due to weak password practices or faulty authentication mechanisms.
8. **Software and Data Integrity Failures:** Failures in ensuring the integrity of software and data that are not properly verified, enabling potential manipulation.
9. **Security Logging and Monitoring Failures:** Failures arising from insufficient logging or monitoring of application activities, preventing detection of potential threats.
10. **Server-Side Request Forgery (SSRF):** A manipulation technique where attackers force the server to send requests to internal systems that should not be externally accessible, potentially leading to data theft.

D. Analysis and Report

After the testing process was completed, We conducted an analysis using the CVSS method. Security analysis was conducted using the Common Vulnerability Scoring System (CVSS) to assess the level of risk. CVSS is a framework used to evaluate the severity of security vulnerabilities in a system or application. Through CVSS, organizations or individuals can measure and classify vulnerabilities based on standardized scoring criteria [13].

CVSS serves as the industry standard for determining the security risk level of an application. It can be used to assess system vulnerabilities by both individuals and institutions. CVSS assessment consists of seven major components: (1) attack vector, (2) attack complexity, (3) privileges required, (4) user interaction, (5) confidentiality, (6) integrity, and (7) availability. CVSS scores range from 0.0 to 10.0 and are categorized into four levels of severity: Low (0.0–3.9), Medium (4.0–6.9), High (7.0–8.9), and Critical (9.0–10.0).

The seven components of CVSS are explained:

1. **Attack Vector (AV):** Describes how an attack can be initiated, whether through a network, remote access, or local access.
2. **Attack Complexity (AC):** Indicates the level of difficulty involved in exploiting the vulnerability, including whether special conditions are required.
3. **Privileges Required (PR):** Describes the level of access an attacker must possess to exploit the vulnerability.
4. **User Interaction (UI):** Indicates whether user interaction is necessary for successful exploitation.
5. **Confidentiality (C), Integrity (I), and Availability (A):** Describe the impact on data confidentiality, integrity, and availability when a vulnerability is exploited [14].

CVSS vulnerability score ranges from 0.0 to 10.0 and is categorized into five severity levels. The explanation of these five levels based on CVSS results is as follows [12]:

1. **Info:** A vulnerability with a score of 0.0 is considered informational and does not pose a direct security impact. However, it may still be used by attackers as supporting information to conduct further attacks.
2. **Low:** Vulnerabilities with scores ranging from 0.1 to 3.9 are classified as low severity. These vulnerabilities have minimal impact on the system, and their effect on confidentiality, integrity, and availability (CIA Triad) is generally limited.
3. **Medium:** Vulnerabilities with scores between 4.0 and 6.9 are categorized as medium severity. These vulnerabilities pose a moderate level of risk and may have a significant impact if left unaddressed, although they usually require additional time or steps to exploit.
4. **High:** Vulnerabilities with scores ranging from 7.0 to 8.9 require prompt remediation due to their serious impact on system or data security. These vulnerabilities may lead to data breaches or system damage.
5. **Critical:** Vulnerabilities with scores between 9.0 and 10.0 are considered critical and represent severe risk with substantial impacts on confidentiality, integrity, and availability. Immediate action is required to address these vulnerabilities.

E. Conclusion and Recommendations

In this stage, We summarizes the overall findings of the security assessment conducted on the SSm Pengangkut application. Based on the results, recommendations are provided to the relevant stakeholders to improve the system's security and minimize potential attack vectors.

III. RESULTS AND DISCUSSION

A. Security Test

1) Scanning Process

1. **Network Mapper (NMAP)**
   We performed a scanning process using the Nmap tool. The scanning was conducted to identify open ports and determine the version of the SSm Pengangkut web application. The results showed that only the HTTPS port (port 443) was open, and no application version information was detected. The scanning process using Nmap was performed with the command "`nmap demo-pengangkut.insw.go.id -sV`", which was used to identify the version of the SSm Pengangkut application. The Nmap scanning results are presented in Figure 2.

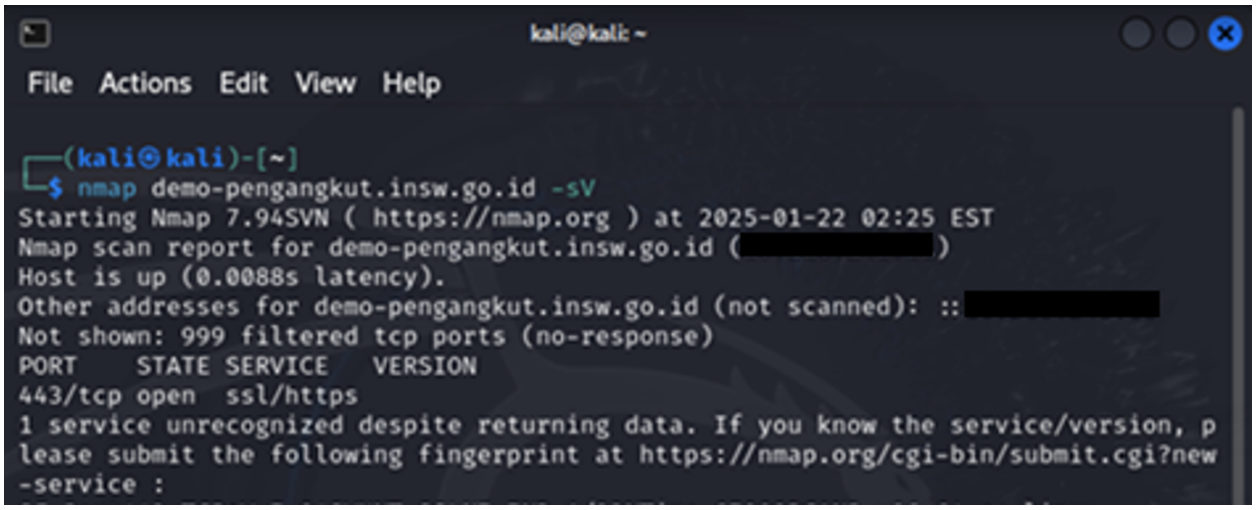

Fig. 2. Nmap scanning results.

2. **NESSUS**
We performed scanned using the Nessus tool to identify potential security vulnerabilities in the SSm Pengangkut web application. The Nessus scan results detected eight vulnerabilities categorized with Info severity. Info severity indicates vulnerabilities that exist within an application but do not have an immediate impact on its security. The scan results using Nessus are presented in Figure 3.

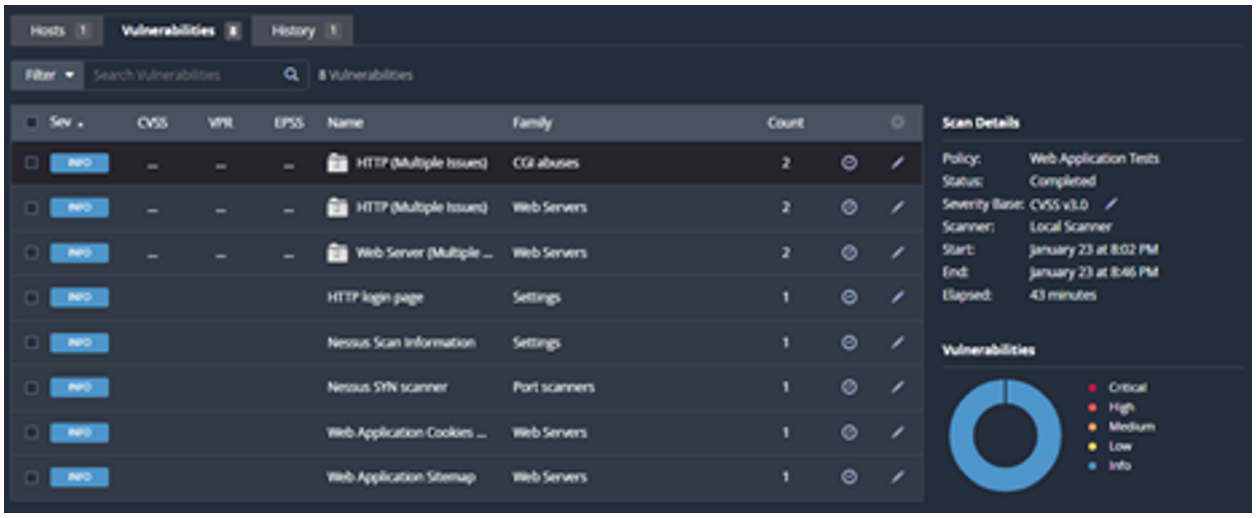

Fig. 3. The scan results.

2) Security Test Based on OWASP TOP 10

1. **Broken Access Control**
Testing for this category was conducted using the Burp Suite application tool. This test required two user accounts. The author used two accounts—one staff user account and one admin account—obtained through interviews with the Directorate of Information Technology team. We captured the request using the Burp Suite tool and attempted to modify the authorization header by replacing the admin account's authorization token with that of the staff user. After performing the test by changing the Authentication Bearer token to the staff user's token, the system returned an "access denied" response. The test results, shown in Figure 4, demonstrate that modifying the Authentication Bearer token using the staff user's credentials does not allow unauthorized access. Therefore, it can be concluded that the SSm Pengangkut website passes the Broken Access Control vulnerability test.

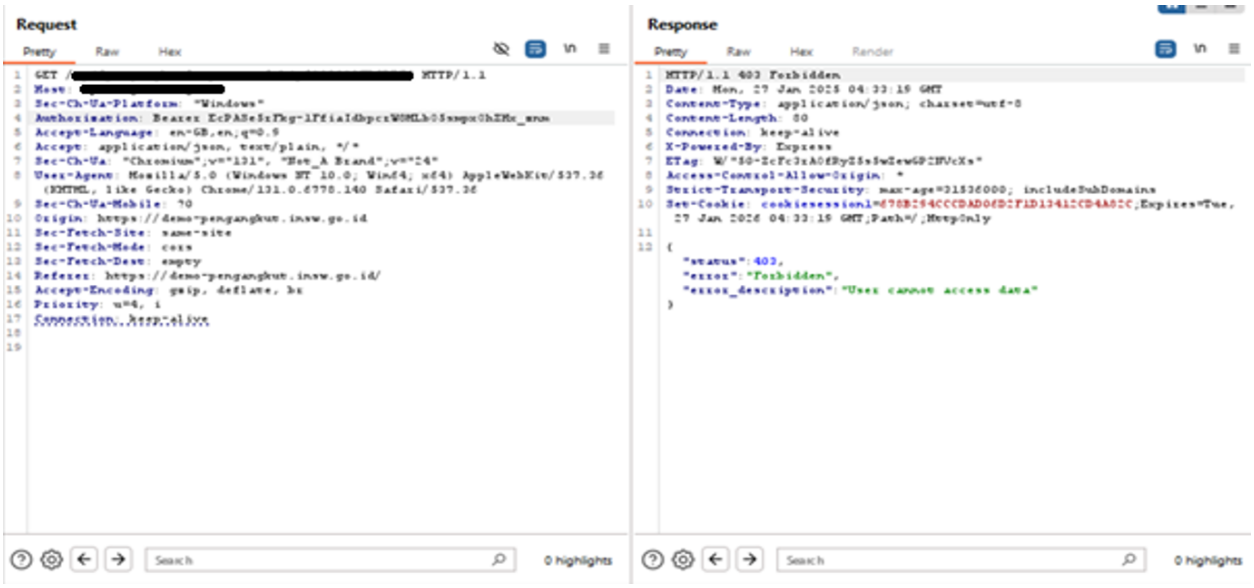

Fig. 4. Test Results Broken Access Control.

2. **Cryptography Failures**
   In this stage, we conducted testing by examining the agent ID. The results showed that the agent ID was already encrypted, indicating that the system passed the cryptographic vulnerability test. The results of the Cryptographic Failures test are shown in Figure 5.

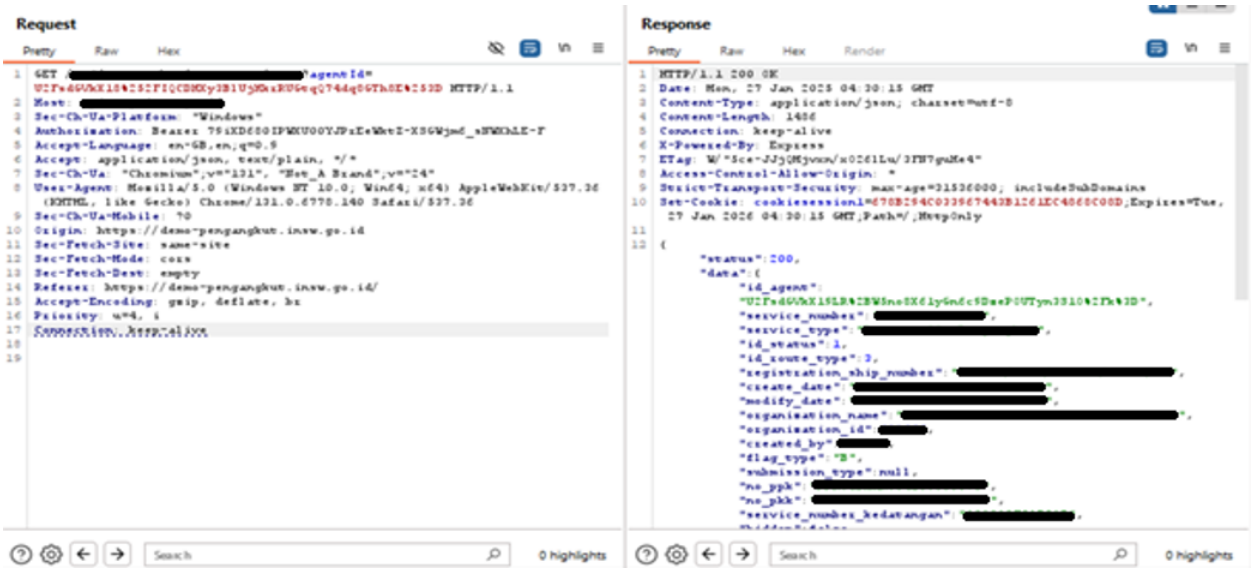

Fig. 5. Test Results Cryptography Failures.

3. **Injection**
   In this stage, we used the BurpSuite tool to perform an SQL injection attempt using the script "`+SELECT+VERSION()-`" to determine the database technology version. After performing the injection, the server response indicated that the system is protected by a Web Application Firewall (WAF), which prevents such attacks. Based on the test results, the Injection vulnerability on the demo-pengangkut website is considered to have passed the security assessment. The results of the Injection test are shown in Figure 6.
4. **Insecure Design**

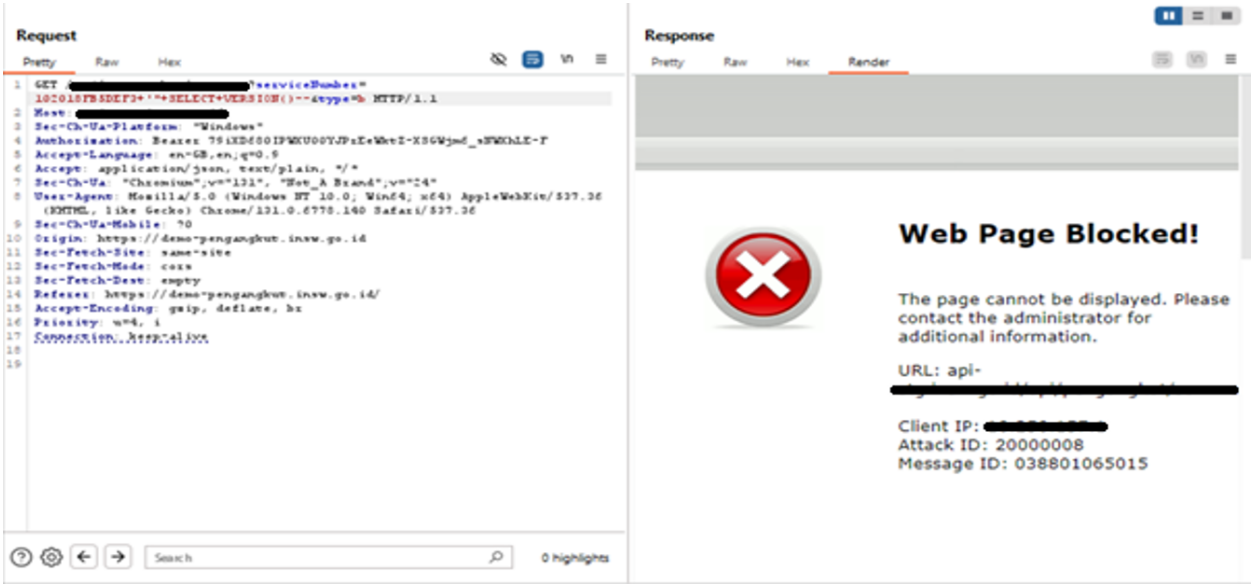

Fig. 6. Test Results Injection.

Testing was conducted by examining a menu that potentially contains an Insecure Design vulnerability. The document upload form, shown in Figure 7, was evaluated to check for signs of insecure design by attempting to upload a file as part of the assessment.

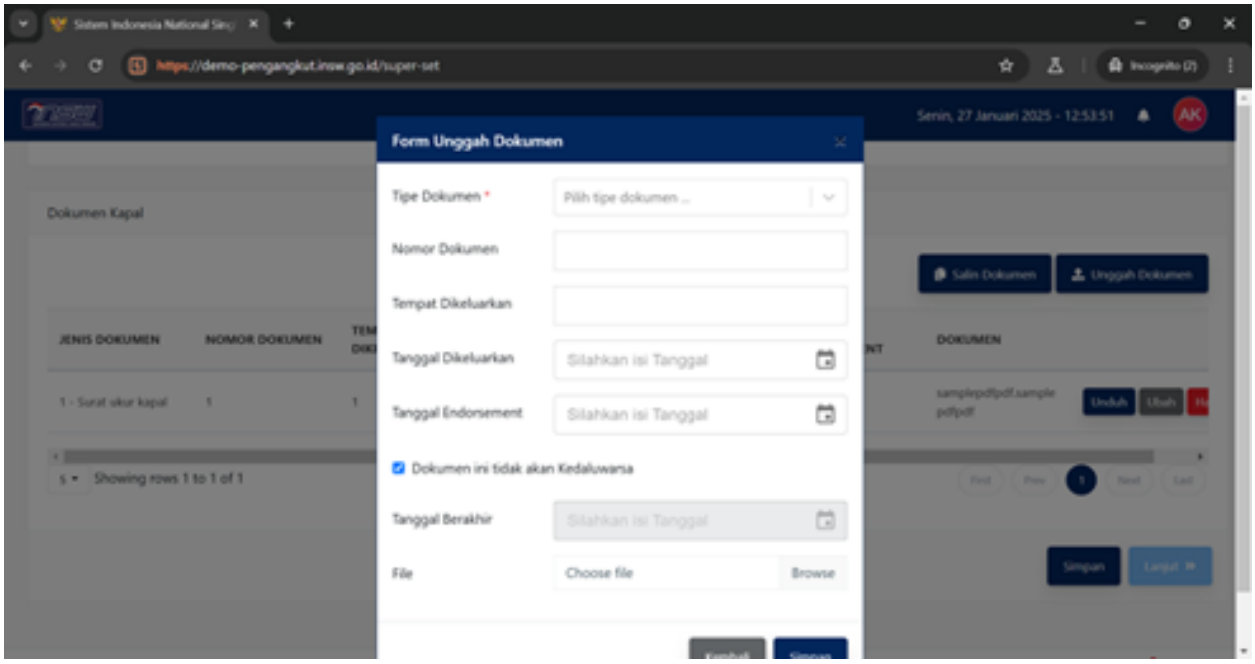

Fig. 7. Document upload form.

The testing was carried out using the Burp Suite tool. After intercepting the request, it was inserted into the Intruder module with the payload type set to `NULL PAYLOAD`, executed for 100 attempts. These 100 attempts were performed in accordance with the request limit defined by LNSW. The results showed that the server returned a 200 response code, indicating that the server accepted the input. The outcome of this test is shown in Figure 8.

Based on the results of the vulnerability testing, the server returned a 200 response code, indicating that the server continued to accept inputs due to the absence of limitations in the document creation

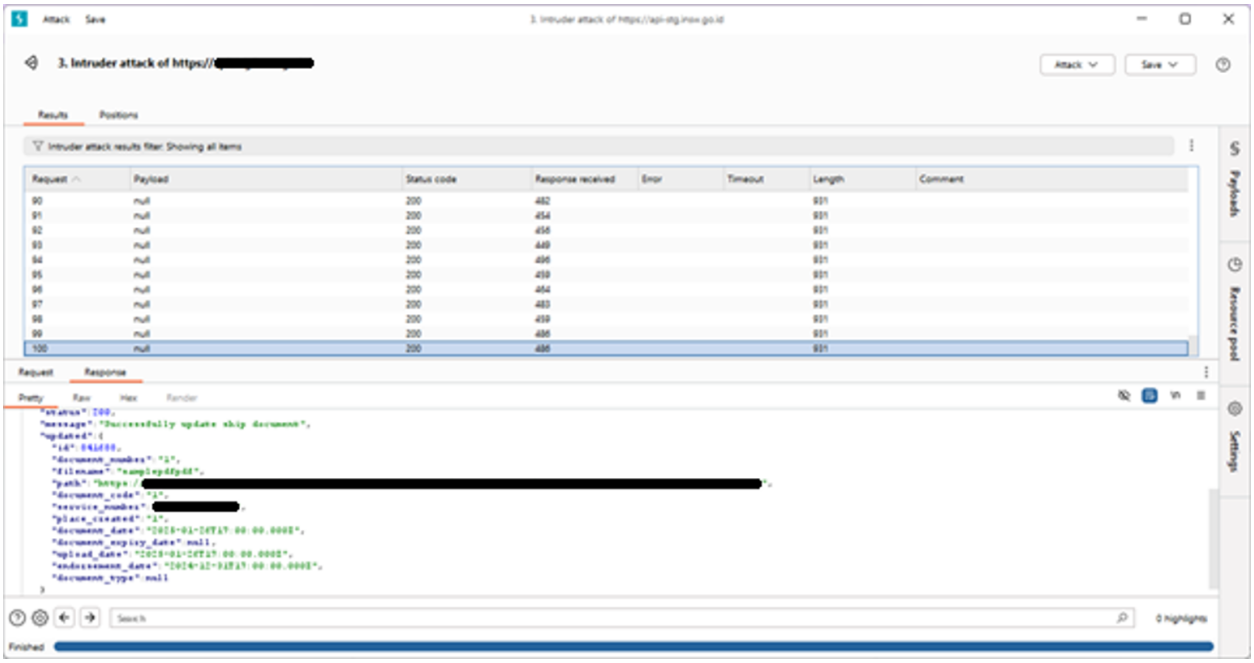

Fig. 8. Test Results Insecure Design.

process. Therefore, it can be concluded that an Insecure Design vulnerability exists. The Insecure Design finding was then evaluated using the CVSS scoring method, and the results are presented on Table I.

TABLE I. CVSS INSECURE DESIGN.

Severity	**Overall Score: 4.1**			
Medium	Attack Vector	Adjacent	Scope	Changed
	Attack Complexity	Low	Confidentiality	None
	Privileges Required	Low	Integrity	None
	User Interaction	None	Availability	Low

Based on the CVSS assessment results, it can be concluded that the security vulnerability under the Insecure Design category has a vulnerability score of 4.1 with a Medium severity level.

5. **Security Misconfiguration**
   Tested was carried out on Security Misconfiguration. The author performed an intercept using BurpSuite and found one request that could be executed. In this step, the tester added the script "`../../../../../../etc/passwd`" to check for directory traversal. The script insertion process can be seen in Figure 9. After the request was sent, the server was already protected by a WAF. The results of the Security Misconfiguration testing can be seen in Figure 10.
   Based on the security misconfiguration test, it can be concluded that the SSm Pengangkut web application passed the vulnerability assessment because it is protected by a WAF.
6. **Vulnerable and Outdated Component**
   At this stage, the testing was conducted using the Nuclei tool available in the Kali Linux operating system. The test was performed on the SSm Pengangkut web application, and the results showed no outdated components. The testing process results can be seen in Figure 11. However, when the author conducted testing using the ZAP tool, outdated components were detected. The testing process results can be seen in Figure 12. Therefore, it can be concluded that the application did not pass the vulnerability assessment under the Vulnerable and Outdated Components category.
   A CVSS evaluation was then carried out for this category, with the results shown in Table II. Based on the CVSS assessment, it can be concluded that the security issue under the Vulnerable and Outdated Components category has a vulnerability score of 0.0, with an informational severity level.

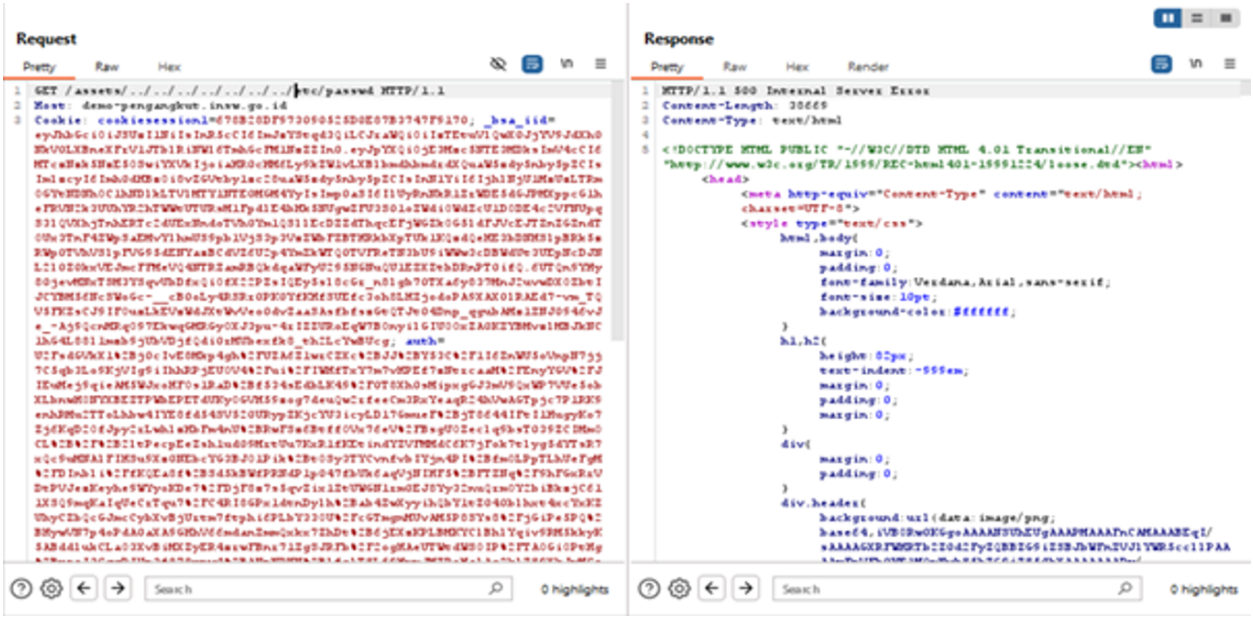

Fig. 9. Input Script Security Misconfiguration.

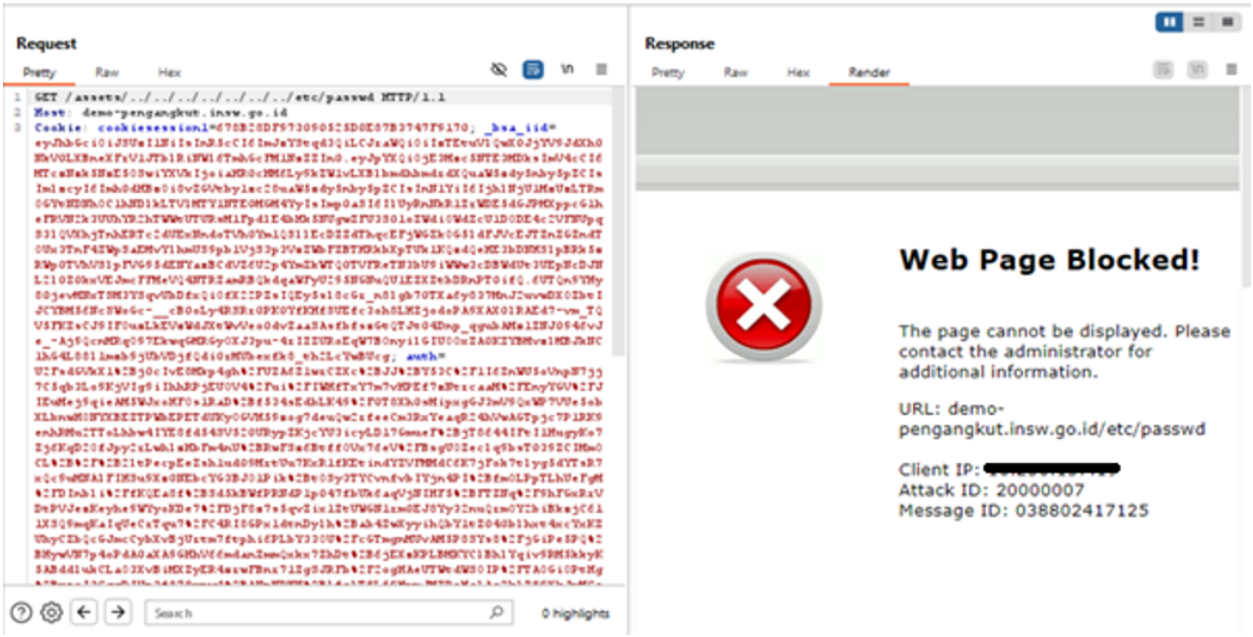

Fig. 10. Result Security Misconfiguration.

Fig. 11. Testing Process Vulnerable and Outdated Component Tools Nuclei.

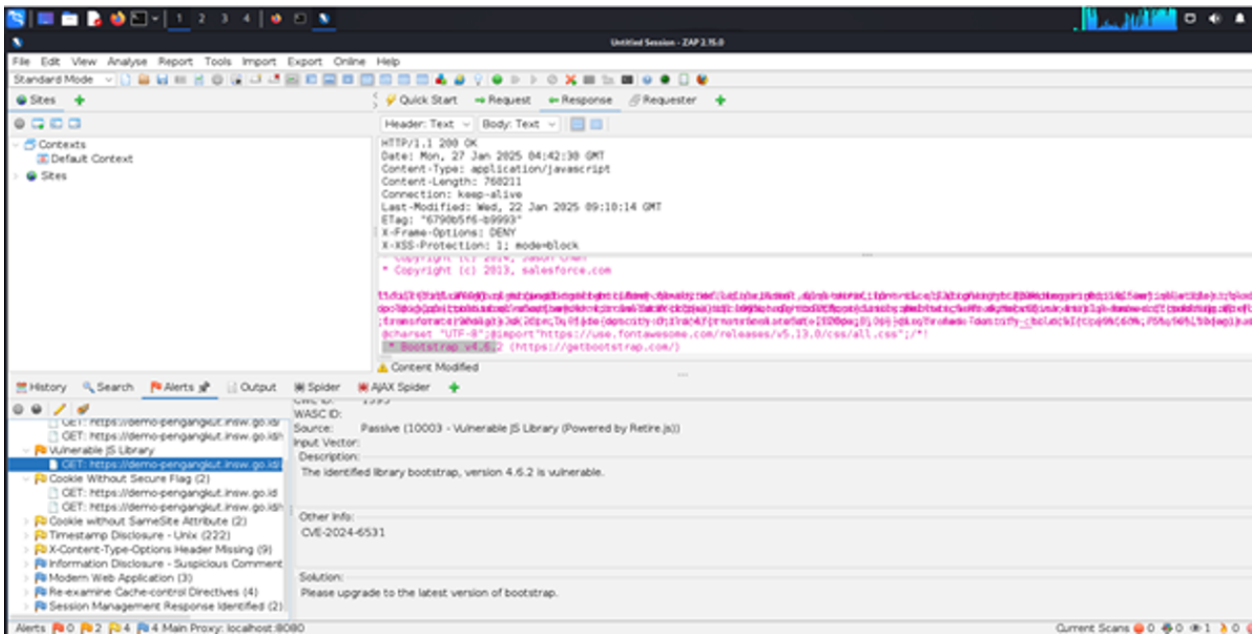

Fig. 12. Result Vulnerable and Outdated Component Tools ZAP.

TABLE II. CVSS VULNERABLE AND OUTDATED COMPONENT.

Severity Level	**Overall Score 0.0**			
Info	Attack Vector	None	Scope	Changed
	Attack Complexity	Low	Confidentiality	None
	Privileges Required	Low	Integrity	None
	User Interaction	None	Availability	None

7. **Identification and Authentication Failures**
   In this stage, the testing was carried out using the ZAP tool, and the results showed that the cookie did not meet the required standards. The testing results can be seen in Figure 13. It can therefore be concluded that the application did not pass the vulnerability assessment under the Identification and Authentication Failures category.

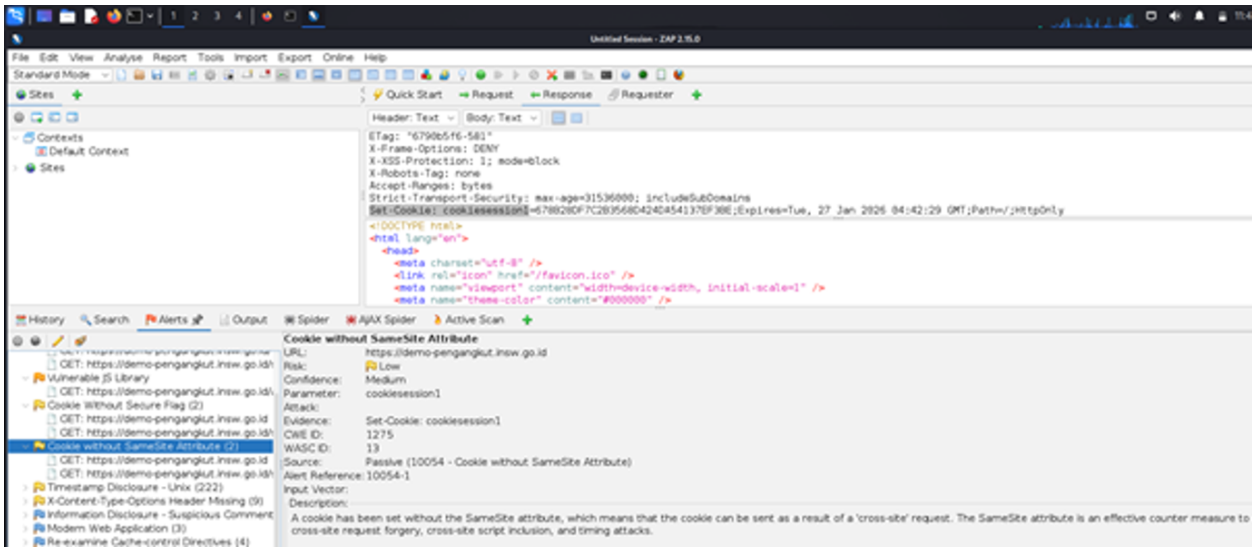

Fig. 13. Result Identification and Authentication Failures.

A CVSS evaluation was then performed for this category, with the results shown in Table III. Based on the CVSS assessment, it can be concluded that the security issue under the Identification and Authentication Failures category has a vulnerability score of 0.0, with an informational severity level.

TABLE III. CVSS IDENTIFICATION AND AUTHENTICATION FAILURES.

Severity Level	**Overall Score 0.0**			
Info	Attack Vector	None	Scope	Changed
	Attack Complexity	Low	Confidentiality	None
	Privileges Required	Low	Integrity	None
	User Interaction	None	Availability	None

8. **Software and Data Integrity Failures**
   In this stage, the testing was carried out using a script to perform a distributed denial-of-service (DDoS) attack, which involves sending a series of requests to the server with the aim of overwhelming it and causing it to become unavailable. After running the script, several requests were successfully received by the server. At the 798th response, the server reached its limit; however, it continued to operate without any operational issues. The results of the testing at this stage show that the server is secure and has passed the vulnerability assessment against DDoS attacks. The testing results can be seen in Figure 14.
9. **Security Logging and Monitoring Failures**

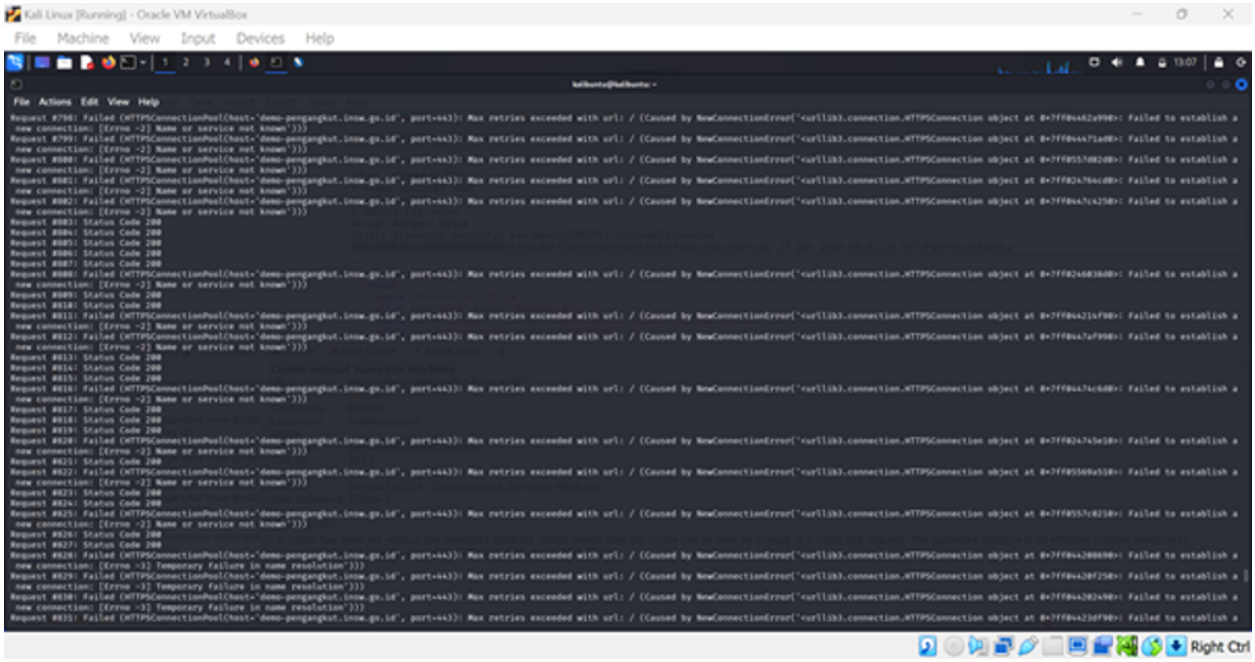

Fig. 14. Result Software and Data Integrity Failures.

In this section, the author is unable to present supporting data in the form of images required as evidence, because the data is highly sensitive to the organization. However, it can be stated that based on the results of the security logging and monitoring failures assessment, the system managed by LNSW already utilizes a log management application as well as a Security Information and Event Management (SIEM) solution.

10. **Server-Side Request Forgery**
In this section, the testing was carried out using the BurpSuite tool, where the author identified a request that was potentially injectable. The author attempted an XXE injection using the script "file:///etc/passwd," as shown in Figure 15. The results of the Server-Side Request Forgery testing can be seen in Figure 16, and it can be concluded that the SSm Pengangkut web application passed the vulnerability assessment because it is already protected by a WAF.

B. Security Testing Results

The security testing results for the Single Submission (SSm) Pengangkut website, based on the OWASP Top 10, show that there are three security vulnerabilities that did not pass the vulnerability assessment. These vulnerabilities are:

1. **Insecure Design:** Based on the CVSS assessment, this vulnerability falls under the Medium category with a vulnerability score of 4.1. The recommended action is to close the remaining security gaps by implementing input restrictions within the application.
2. **Vulnerable and Outdated Components:** Based on the CVSS assessment, this vulnerability falls under the Informational (Info) category. A vulnerability categorized as "Info" indicates that an issue was found in the application but it does not have a direct impact on security. The recommendation is to perform updates on supporting software components to ensure they do not become exploitable in the future.
3. **Identification and Authentication Failures:** Based on the CVSS assessment, this vulnerability also falls under the Informational (Info) category. To address this issue, it is recommended to apply standardization of the cookies used and implement two-factor authentication, such as Multi-Factor Authentication (MFA).

IV. CONCLUSION

The results of the vulnerability assessment conducted on the Single Submission (SSm) Pengangkut web application, referring to the OWASP Top 10, using tools such as Kali Linux, Nmap, Nessus, and Burp Suite,

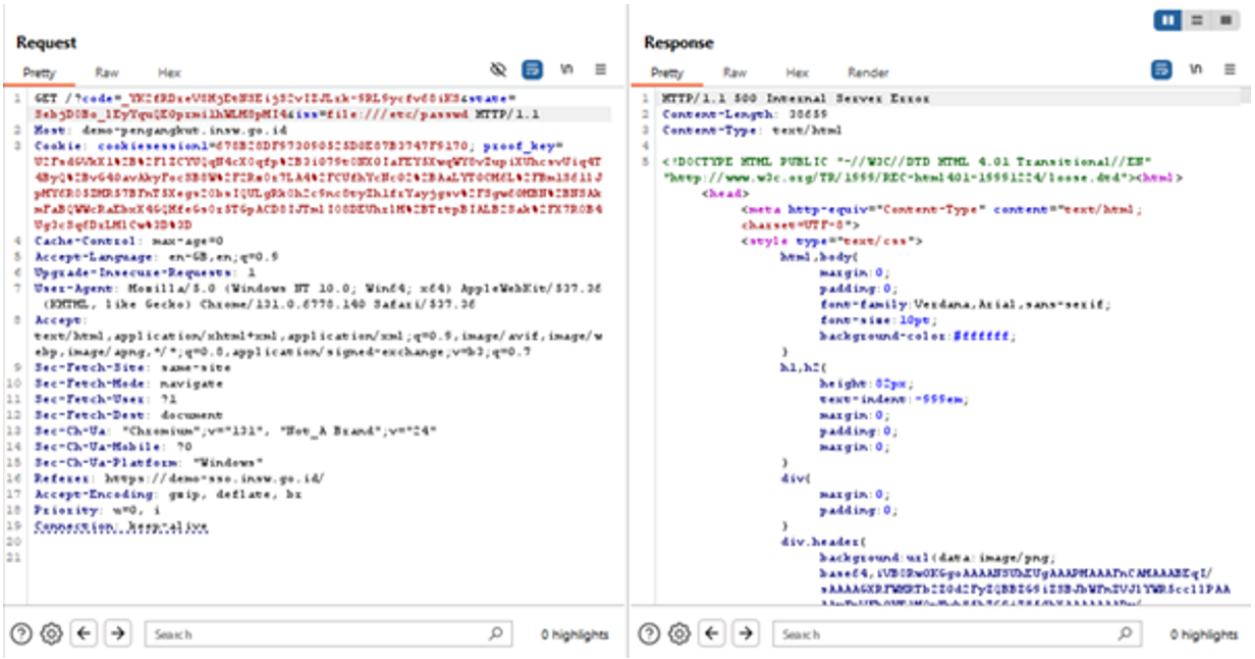

Fig. 15. Input Script Pengujian Server-Side Request Forgery.

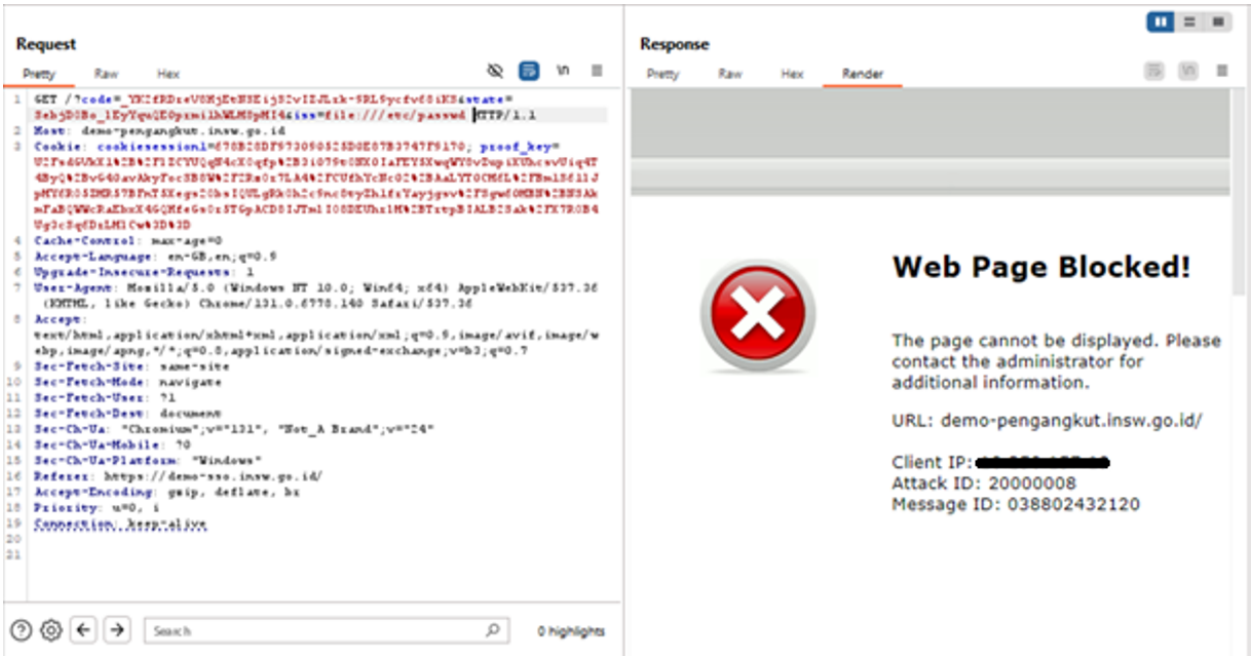

Fig. 16. Result Server-Side Request Forgery.

identified seven vulnerabilities that passed the assessment and three categories that did not pass, namely Insecure Design, Vulnerable and Outdated Components, and Identification and Authentication Failures. Based on the CVSS evaluation, Insecure Design falls under the Medium category with a vulnerability score of 4.1. Meanwhile, Vulnerable and Outdated Components and Identification and Authentication Failures fall under the Informational category, meaning the issues detected do not have a direct impact on the application's security.

These findings provide significant insight, showing that the detected vulnerabilities align with the major vulnerability categories defined by the OWASP Top 10 standard. From the results of this study, it can be concluded that applying OWASP Top 10 as a security standard reference in penetration testing is proven effective in identifying and evaluating significant vulnerabilities within the SSm Pengangkut web application.

Following the security testing of the Single Submission (SSm) Pengangkut web application, several recommendations can be implemented in future development processes:

1. Implement limitations when creating documents or performing input to the server.
2. Conduct regular reviews and documentation of components and versions used, and apply routine update policies to all software components to ensure the use of stable and secure versions.
3. Implement two-factor authentication, such as Multi-Factor Authentication (MFA), to enhance security and prevent unauthorized access to the system.

REFERENCES

[1] L. Saputri, S. W. Hamidah, and N. S. Husna, "Peluang dan tantangan ekspor impor di era globalisasi," *Jurnal Ekonomi Sakti*, vol. 13, no. 2, p. 163, 2024. doi:10.36272/jes.v13i2.340.

[2] D. Wiryany, S. Natasha, and R. Kurniawan, "Perkembangan teknologi informasi dan komunikasi terhadap perubahan sistem komunikasi indonesia," *Jurnal Nomosleca*, vol. 8, no. 2, pp. 242–252, 2022. doi:10.26905/nomosleca.v8i2.8821.

[3] L. Hawari, "Pengaruh keterlambatan kapal bagi kegiatan ekspor impor sub divisi hapag lloyd di pt. samudera agencies indonesia semarang," 2022.

[4] Tempo, "Pdns lumpuh karena serangan ransomware, data terdampak tidak bisa dipulihkan," 2024. [Online]. Available: https://www.tempo.co/hukum/pdns-lumpuh-karena-serangan-ransomware-data-terdampak-tidak-bisa-dipulihkan--45597.

[5] Tempo, "Kaleidoskop 2024: 6 serangan siber besar di indonesia," 2024. [Online]. Available: https://www.tempo.co/hukum/kaleidoskop-2024-6-serangan-siber-besar-di-indonesia-1188275.

[6] Kementerian Sekretariat Negara Republik Indonesia, "Evaluasi peretasan pdns, presiden: Semua data nasional harus direkam cadang," 2024. [Online]. Available: https://setneg.go.id/baca/index/evaluasi_peretasan_pdns_presiden_semua_data_nasional_harus_direkam_cadang.

[7] A. Elanda and R. L. Buana, "Analisis keamanan sistem informasi berbasis website dengan metode open web application security project (owasp) versi 4: Systematic review," *CESS (Journal of Computer Engineering, System and Science)*, vol. 5, no. 2, p. 185, 2020. doi:10.24114/cess.v5i2.17149.

[8] F. Tinambunan, A. Junaidi, and A. M. Rizki, "Pengujian sistem informasi akademik universitas x melalui pendekatan penetration testing berdasarkan owasp top 10," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 1, pp. 1062–1069, 2024. doi:10.36040/jati.v8i1.8920.

[9] N. Herawati, V. Budiyanto, and Uminingsih, "Analisis keamanan sebuah domain menggunakan open web application security project (owasp) zap," *Jurnal Teknologi Technoscientia*, vol. 15, no. 2, pp. 27–36, 2023. doi:10.34151/technoscientia.v15i2.4013.

[10] H. Setiawan, L. E. Erlangga, S. Siddiq, and Y. A. Gunawan, "Analisis kerawanan pada aplikasi website menggunakan standar owasp top 10 untuk penilaian risk rating," *Info Kripto*, vol. 17, no. 1, pp. 15–21, 2023. doi:10.56706/ik.v17i1.64.

[11] OWASP, "Owasp top ten," 2021. [Online]. Available: https://owasp.org/www-project-top-ten/.

[12] S. Margareth *et al.*, "Uji penetration testing web server xyz, menggunakan metode owasp top 10 dan cvss," 2024.

[13] H. S. Albab, "Pemanfaatan chatbot whatsapp sebagai uji analisis statis kerentanan sistem informasi akademik perguruan tinggi di indonesia," 2023.

[14] P. Rizkika, D. Juardi, and A. S. Y. Irawan, "Analisis keamanan pada aplikasi himfo berbasis android menggunakan mobsf," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 4, pp. 5945–5952, 2024. doi:10.36040/jati.v8i4.10051.