Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

15th June 2025. Vol. 4. No. 3; e-ISSN: 2808-4519

Web Security Vulnerability Analysis and Mitigation Based on OWASP TOP 10

M.Syarifudin[1]*, Lilik Widyawati[2], Ondi Asroni[3]

[1,2,3]*Fakultas Teknik, Universitas Bumigora*
syarifudinahmad457@gmail.com[1*]

Abstract

Information security is present as one of the main pillars in the challenges of the current era of technological development, especially on websites used by XYZ institutions. This study aims to test system security using penetration testing techniques with the latest standards, namely using OWASP TOP 10 in evaluating its security. The methods used in this research include scope, information gathering, vulnerability analysis, exploit, report and remediation, and testing is carried out based on the vulnerabilities obtained during vulnerability analysis according to the list of 10 types of vulnerabilities found in the OWASP Top 10 2021. The results showed that the system still has several security gaps consisting of security misconfiguration, vulnerable and outdated components, and identification and authentication failures. With appropriate improvements, the system can be more secure in the face of cyberattacks and maintain the confidentiality of mustahik data (zakat distributors). This research is expected to be a reference for system developers in improving the security of web-based applications, especially in the context of data protection.

Keywords: *Information Security, Website, OWASP TOP 10, Penetration Testing*

1. Introduction

Advances in information and communication technology (ICT) are the main reason government agencies and businesses are able to adapt; the development of ICT requires administrators to implement a security system that involves web-based applications to thwart potential cyberattacks[1]. Based on the State Cyber Cryptography Agency report, the total cyber anomaly traffic in Indonesia during 2023 is 403,990,813 anomalies. With MySQL database account anomaly activity and brute force guesses ranked 6th with a total of 14,789,313 anomalies[2]. This traffic anomaly activity can have an impact on disruption of public services or decrease in device and network performance and even theft of sensitive data[3].

This XYZ institution has adopted a web-based digital system to facilitate muzaki (zakat distributors) in channeling their zakat, but with the increasing dependence on technology, cybersecurity has become a serious concern due to increasingly massive and troubling cyberattacks, thus becoming a significant challenge for data protection and lawt[4]. Therefore, it is necessary to conduct security testing using penetration testing, which is a systematic series of steps to evaluate the security of a network or website by simulating cyberattacks in an ethical manner in order to find out where the vulnerabilities in the system are so that the gap can be repaired[5]. Where penetration testing consists of scope, reconnaissance, vulnerability detection, information analysis, planning, and penetration testing[6]. And OWASP TOP 10 is a standard published by the OWASP community that contains a top ten list of vulnerabilities that can compromise web security [7]. As for the list of 10 vulnerabilities based on the OWASP Top 10 2021 guide compiled on its website, shown in Figure 1 below[9].

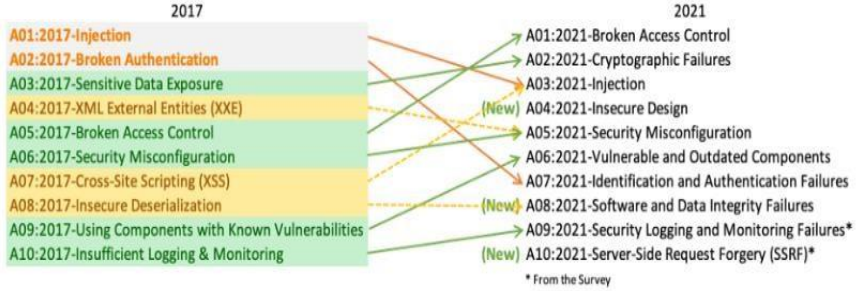

Fig. 1: OWASP TOP 10 List 2021

Figure 1 above shows ten vulnerabilities from the OWASP Top 10 which are updated regularly including releases in 2017 and 2021, in this study the reference for researchers is the list of the latest 2021 releases seen in the figure above from Broken Access Control to Server-Side Request Forgery (SSRF).

2. Research Methods

In this study, researchers used the penetration testing method, which is a test conducted to increase the security of a website so that it is not exposed to external access attacks using the OWASP Top 10 guide, as for the research flowchart in Figure 2 below.

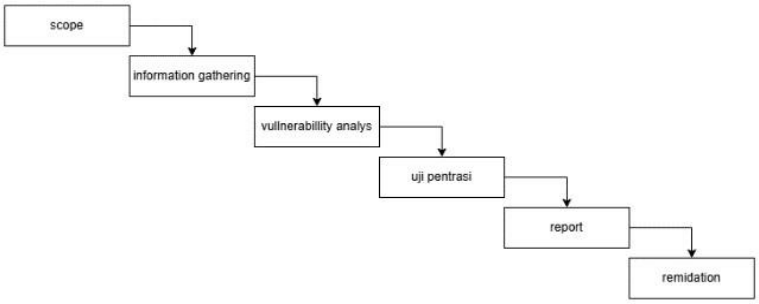

Fig. 2: Penetration testing flowchart

It can be seen that the research method in Figure 2 above include scope, reconnaissance, vulnerability analysis, exploit, reporting, and mitigation.

2.1. Scope

Researchers will identify the problems to be investigated, formulate research questions, and decide on the methods to be used. In this way, the research is focused and aligned with thed [8]. In this study, the scope of research is the XYZ agency, with the system being tested being the XYZ agency website, https://cmsdemo.lembagaxyz.id, and problem boundaries only use OWASP Top 10.
Researchers received approval from the XYZ institution to conduct research in its place and get permission to conduct vulnerability analysis & mitigation of the zakat web owned by the institution.

Table 1: Tools used

Category	Details
Lenovo laptop Core i3 Gen 7	Computer Specification
4 GB DDR4 (3.84 GB usable)	Device Model and Manufacturer
120 GB SSD	Storage Type and Capacity
Microsoft Windows 10 Pro	Host Operating System
VirtualBox with Kali Linux	Virtualization and Guest OS Platform
OWASP ZAP, Acunetix, Nmap, Hydra, Burp Suite, and Metasploit	Tools for Penetration Testing
Windows Defender	Host System Antivirus

Test plan: At this stage the researcher only tests the XYZ institution website, namely https://cmsdemo.lembagaxyz.id/, according to the OWASP TOP 10 guidelines. As for the steps the author takes in carrying out the penetration testings, they will be described through the flowchart below.

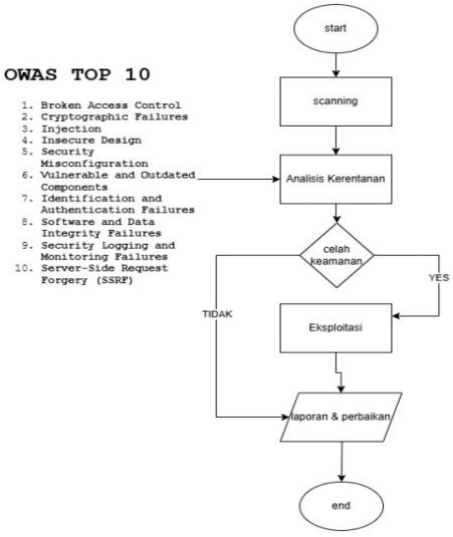

Fig. 3: Pentesting flowchart

2.2. Reconnaissance

Collecting initial data about the target system, such as URLs, subdomains, and services used, in this study the authors used several tools to conduct reconnaissance, including Wappalyzer, WhatWeb, Dmitry, and others.

2.2.1. Wappalyzer

From the results of the analysis using Wappalyzer, information on website supporting technology is obtained, including Histats, which functions to analyze web visitor statistics; HSTS, which functions to increase security by forcing browsers to use HTTPS; web frameworks using CodeIgniter; and other supporting technologies, including CDN.

2.2.2. Dmitry

DMitry allows gathering information about the target host from simple WHOIS lookups on the target to uptime and TCP port portscan reports[9].

Table 2: Dmitry's reconnaissance results

Types of information	Detail
IP Address	153.92.xx.xx.
Hosting Name	cmsdemo.lembagaxyz.id
IP Range	153.92.8.8 - 153.92.15.255
Hosting Providers	Hostinger
Network status	LEGACY

From the results of table 2 above, it shows that the IP used by the web target is 153.92.xx.xx and some other supporting information related to the hosting service provider.

2.2.3 Nmap (Network Mapper)

Nmap is an information scanning tool that specifically looks for open ports on a network. Nmap is specifically designed to ping open ports and send information back to the hacker[10].

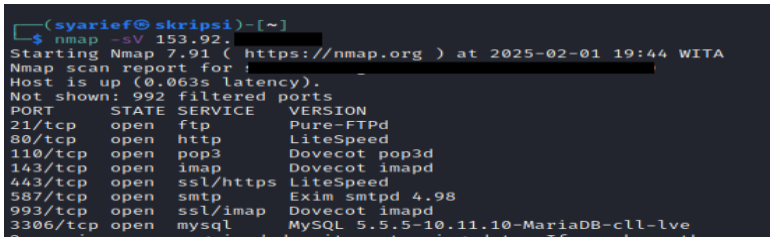

Fig. 4: Nmap reconnaissance result

In the table above, information related to open ports is obtained, consisting of ports 21,80,110,143,443,587,993,3306 and this can be a potential target for attackers.

2.3. Vulnerability Analysis / Scanning

Perform scans to identify potential weak points, such as gaps in ports or open services. This vulnerability detection effort is an important part after the identification stage, while validation aims to reduce the number of identified vulnerabilities to valids[11]. At this stage the author uses tools including OWASP ZAP and Acunetix.

2.3.1. Acunetix

From the scan results using Acunetix, three types of vulnerabilities were found on the site, namely Subresource Integrity (SRI) not implemented with a medium vulnerability level, Content Security Policy (CSP) not implemented with a medium vulnerability level, and HTML form without CSRF protection with a low vulnerability level.

2.3.2. OWASP ZAP (Zed Attack Proxy)

From the results of the analysis using ZAP, there are several vulnerabilities obtained, including X-Frame-Options Not Set with a medium vulnerability level, Absence of Anti-CSRF Tokens, Cross-Domain JavaScript Source File Inclusion, and Server Leaks Information via "X-Powered-By" HTTP Header, which each have a low vulnerability level, and then Information Disclosure - Suspicious Comments and Timestamp Disclosure - Unix, which are included in the informational category.

2.4. Exploitation

At this stage the author exploits the results of vulnerability analyis by using tools including Metasploit, SQLmap, Hydra, and Burpsuite. This stage is an important stage in penetration testing for testers to find out the extent of the target web vulnerability that has been analyzed

for vulnerabilities in the previous stage. In this exploit, the tester refers to the OWASP top 10 standards for analyzing the vulnerabilities found, which are summarized below based on a combination of several tools used, namely Acunetix, OWASP ZAP, and Nmap.

Table 3: OWASPZAP, Acunetix and Nmap exploitation results

OWASP TOP 10	Alerts	Severity Level
A05 – Security Misconfiguration	Content Security Policy (CSP) not implemented	Medium
	X-Frame-Options Header Not Set	Medium
	Server Leaks Information via "X-Powered-By" HTTP Header	Low
	Open Ports Exposing Unnecessary Services(Nmap	Medium
A06 - Vulnerable and Outdated Components	Subresource Integrity (SRI) not implemented	Medium
	Cross-Domain JavaScript Source File Inclusion	Low
A07 - Identification and Authentication Failures	Absence of Anti-CSRF Tokens	Low
	HTML form without CSRF protection	Low

2.4.1 Security misconfiguration

a. Content Security policy

At this stage the tester uses a website, namely https://securityheaders.com/, to evaluate the presence and configuration of HTTP Security Headers on a website.

The exploit results show that there are several headers that have not been applied to the website, including Content-Security-Policy, XFrame-Options, Referrer-Policy and permission-policy which are part of the HTTP security header even the rating given from the test is D which indicates that the security configuration is very weak, because many headers are missing.

b. X-Frame-Options Header Not Set

Serves as a security mechanism to prevent clickjacking by disabling iframes on a website.

In testing these vulnerabilities, the tester utilizes the BurpClickBandit feature on Burp Suite to automatically test the web vulnerability to clickjacking attacks. The target site is indeed vulnerable to clickjacking attacks; this triggers the attacker to carry out exploits to trap victims into clicking on something they don't want.

c. Server Leaks Information via "X-Powered-By" HTTP Header

From the exploit results using the CLI cmd with the curl -I command, information related to PHP used by the target website, and this triggers the attacker to find out the vulnerability of the PHP used through X-Powered-By information: PHP/7.4.33.

d. Open Ports Exposing Unnecessary Services (3306 mysql)

At this stage, the testers tried to perform more in-depth exploitation to obtain vulnerabilities from this port, using the Metasploit Framework.

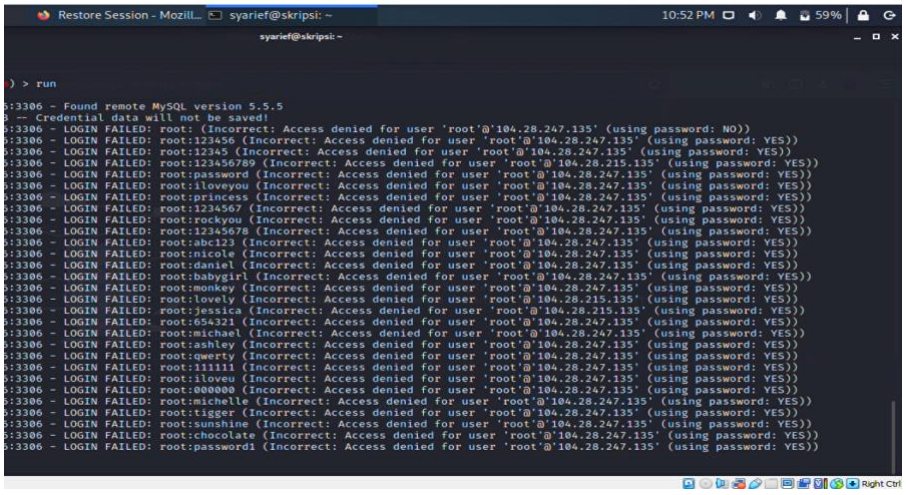

Fig. 5: Msfconsole exploit

From the exploitation results, Figure 4 above shows that the server is running MySQL version 5.5.5, then shows a warning that no *database* is active or found in this session and *username* and *password* attempts in the *database*, and there is no *rate* limiting in limiting the *brute force* access of this test, which results in a massive and effective brute force attack so that the CPU and memory of the server may be affected in the *brute force* attack because the server has to process a large number of login attempts without any *filter*.

3. Result and Discussion

3.1. Reporting and mitigation

The reporting phase is a testing phase that serves to document all test findings and results, including what vulnerabilities, attacks that can occur, and mitigations for improvement[12].

Table 4: Reporting and mitigation results

OWASP TOP 10	Vulnerability	Risk Assessment	Mitigation
A05 - Security Misconfiguration	*Content Security Policy (CSP) not implemented*	No implementation of Content-Security-Policy, X-Frame-Options, ReferrerPolicy, permission-policy	Header set Content-Security-Policy-Report-Only "default-src 'self'; script-src 'self' cdnjs.cloudflare.com cdn.jsdelivr.net unpkg.com ajax.googleapis.com maps.googleapis.com maps.gstatic.com code.jquery.com 'sha256-1+XWn73CATkcg5nsO9vP/hFDUnyn5juMFkGJ8nVXM5o='; style-src 'self' fonts.googleapis.com cdnjs.cloudflare.com cdn.jsdelivr.net unpkg.com 'unsafe-inline'; img-src 'self' data: maps.gstatic.com maps.googleapis.com *.tile.openstreetmap.org unpkg.com; font-src 'self' fonts.gstatic.com cdnjs.cloudflare.com cdn.jsdelivr.net unpkg.com; connect-src 'self' maps.googleapis.com maps.gstatic.com; frame-src 'self' www.google.com www.youtube.com; object-src 'none';" The application of the csp above is to set a security policy where the loading of scripts, css, fonts and content only from certain parties is written above, thus reducing the attack of unauthorized parties in damaging the appearance of a system.
	X-Frame-Options Header Not Set	Vulnerable to clicjacking attacks, where attackers can use iframes	The header always sets X-Frame-Options "SAMEORIGIN" to prevent clickjacking embedded in iframes from other domains.
	Server Leaks Information via "X-Powered-By" HTTP Header	The site reveals PHP/7.4.33 server information, increasing the risk of attack.	The unset X-Powered-By header serves to remove the display of the PHP version used for security reasons.
	Open Ports Exposing Unnecessary Services (Nmap)	MySQL port 3306 is open without firewall/rate limiting, vulnerable to brute force and performance degradation.	Enable ModSecurity and immunify360 as a firewalls to protect against various threats, including brute force and xss attacks.
A06 - Vulnerable and Outdated Components	*Subresource Integrity (SRI) not implemented*	Found the use of external resources such as maps.google.com and unpkg.com without SRI, increasing the risk of XSS, MITM, and supply chain attacks.	The same as the recommendation on the previous vulnerability, namely on CSP, only added a security mechanism for CDN maps and unpkg.
	Cross-Domain JavaScript Source File Inclusion	The application does not validate external sources, allowing malicious code injection through third-party JavaScript files.	Likewise, this vulnerability, where external files need to be validated to avoid injection by adding CSP security, can be considered as a vulnerability.
A07 - Identification and Authentication Failures	*Absence of Anti-CSRF Tokens & HTML Form without CSRF Protection*	The login form does not implement CSRF protection, making it vulnerable to request forgery attacks.	Because the target web uses the Codeigniter framework in web development, there is a CSRF feature that needs to be activated in config.php.

4. Conclusion

The conclusion of this test shows that with various combinations of vulnerability analysis tools, the XYZ institution website still has various weaknesses that need to be improved to reduce the risk of exploitation, including the implementation of stricter security policies, such as Content Security Policy (CSP), X-Frame-Options, Referrer-Policy, and Permission Policy, to prevent injection-based attacks and clickjacking. Optimized configuration by hiding unnecessary information, restricting access to unused ports, and implementing firewalls and rate-limiting mechanisms to prevent brute-force attacks. Enhanced authentication protection by adding an anti-CSRF token, limiting the number of login attempts, and implementing two-factor authentication (2FA) to improve user account security. Validation and security of external resources by implementing Subresource Integrity (SRI) and ensuring that all software dependencies used are always updated and come from trusted sources. Suggestions for further research can implement security from existing vulnerabilities.

References

[1] S. Sabariman, H. Haeruddin, and D. Lee, "Analisis Kerentanan Aplikasi Akademik Berbasis Website Xyz Menggunakan Owasp," *J. Khatulistiwa Inform.*, vol. 11, no. 2, pp. 92–102, 2024, doi: 10.31294/jki.v11i2.20194.

[2] BSSN, "Lanskap Keamanan Siber Indonesia," no. 70, 2024, [Online]. Available: https://www.bssn.go.id/wp-content/uploads/2024/03/Lanskap-Keamanan-Siber-Indonesia-2023.pdf

[3] P. Studi *et al.*, "ANALISIS PERKEMBANGAN KEAMANAN SIBER DAMPAK DARI KEBOCORAN DATA PUSAT DATA NASIONAL SEMENTARA 2 SURABAYA ASSESSING AND UNDERSTANDING THE CURRENT SITUATION : ANALYSIS OF CYBER SECURITY DEVELOPMENTS THE IMPACT OF THE," vol. 2, no. June, 2024.

[4] F. Rahman Najwa, "Analisis Hukum Terhadap Tantangan Keamanan Siber: Studi Kasus Penegakan Hukum Siber di Indonesia," *AL-BAHTS J. Ilmu Sos. Polit. dah Huk.*, vol. 2, no. 1, pp. 8–16, 2024, doi: 10.32520/albahts.v2i1.3044.

[5] S. Hidayatulloh and D. Saptadiaji, "Penetration Testing pada Website Universitas ARS Menggunakan Open Web Application Security Project (OWASP)," *J. Algoritm.*, vol. 18, no. 1, pp. 77–86, 2021, doi: 10.33364/algoritma/v.18-1.827.

[6] B. Wicaksono, Y. R. Kusumaningsih, and C. Iswahyudi, "Pengujian Celah Keamanan Aplikasi Berbasis Web Menggunakan Teknik Penetration Testing Dan DAST (Dynamic Application Security Testing)," *J. Jarkom*, vol. 8, no. 1, pp. 1–9, 2020, [Online]. Available: https://journal.akprind.ac.id/index.php/jarkom/article/view/2755/2103

[7] J. J. B. H. Yum Thurfah Afifa Rosaliah, "Pengujian Celah Keamanan Website Menggunakan Teknik Penetration Testing dan Metode OWASP TOP

10 pada Website SIM," *Senamika*, vol. 2, no. September, pp. 752–761, 2021.
[8] S. Andriansyah and Nurhasanah, "Seminar Nasional Industri dan Teknologi (SNIT), Politeknik Negeri Bengkalis," *Konsep Desain Menentukan Hull Type, Mater. Dan Propulsi Unmanned Surf. Veh. Untuk Patroli Di Wil. Rokan Hiir Dengan Metod. Desicion Tree*, no. Lcm, pp. 478–486, 2020.
[9] J. Greig, "Dmitry - Deepmagic Information Gathering Tool." [Online]. Available: https://github.com/jaygreig86/dmitry
[10] I. Abdurrohim, "Penetration Testing Sistem Keamanan Aplikasi Web Berbasis e-Commerce Pada Perusahaan Hptasik," *J. Ilmu Komput.*, vol. 1, no. March, pp. 125–131, 2019.
[11] S. Utoro, B. A. Nugroho, M. Meinawati, and S. R. Widianto, "Analisis Keamanan Website E-Learning SMKN 1 Cibatu Menggunakan Metode Penetration Testing Execution Standard," *Multinetics*, vol. 6, no. 2, pp. 169–178, 2020, doi: 10.32722/multinetics.v6i2.3432.
[12] M. Noval, R. Darmawan, Y. Muhyidin, and D. Singasatia, "Analisis Keamanan Web Sman 1 Wanayasa Menggunakan Sqlmapdengan Metode Penetration Testing Execution Standard (Ptes)," vol. 2, pp. 110–121, 2024, [Online]. Available: https://jurnal.kolibi.org/index.php/scientica/article/view/2748/2658