TECHNO
Vol.25, No.2, Oktober 2024, Hal. 73~86
P-ISSN: 1410-8607, E-ISSN: 2579-9096

IMPLEMENTASI PENETRATION TESTING PADA SISTEM INFORMASI TERPADU LAYANAN PRODI MENGGUNAKAN FRAMEWORK ISSAF

Anisa Sholawati[1], Hario Jati Setyadi[2], Amin Padmo Azam Masa[3]
Program Studi S1 Sistem Informasi, Universitas Mulawarman Samarinda
Fakultas Teknik, Universitas Mulawarman Samarinda

Informasi Makalah

Dikirim, 04 Februari 2024
Direvisi, 12 Maret 2024
Diterima, 02 April 2024

Kata Kunci*:*
Penetration Testing
Website SIPLO
Sistem Keamanan
Framework ISSAF
OWASP Top 10

INTISARI

Keamanan sistem informasi menjadi aspek kritis dalam lingkungan teknologi informasi yang terus berkembang. *Penetration testing* adalah salah satu metode yang efektif untuk mengidentifikasi dan menguji keamanan dalam sebuah simulasi serangan terhadap suatu sistem atau jaringan guna menemukan celah keamanan yang disebabkan oleh keamanan dari suatu sistem, konfigurasi yang tidak benar atau kelemahan operasional dalam proses teknik. Penelitian ini melakukan identifikasi kerentanan pada website SIPLO dengan metode *penetration testing* menggunakan *framework* ISSAF (*Information System Securiry Assesment Framework*) untuk mengukur keamanan sistem. *Penetration testing* dilakukan dengan menggunakan tiga jenis serangan teratas dari OWASP Top 10 yaitu, XSS, *Brute-force attack,* SQL *injection,* dan DDoS *attack*. Hasil identifikasi kerentanan pada *website* SIPLO menunjukkan bahwa terdapat 23 kerentanan pada SIPLO dengan dua kerentanan memiliki level tinggi, empat kerentanan memiliki level sedang, sebelas kerentanan memiliki level rendah, enam kerentanan memiliki level *informational*. Walaupun dengan kerentanan-kerentanan tersebut *website* SIPLO masih terlindungi dari serangan XSS dan SQL *injection*. Serangan *brute-force attack* hanya berhasil teridentifikasi satu *username* dan satu *password.* Sedangkan serangan DDoS tidak memiliki pengaruh yang signifikan terhadap SIPLO namun tercatat adanya perbedaan sebanyak 217,221 *millisecond* saat dilakukannya serangan DDoS tersebut.

Keyword*:*
Penetration Testing
SIPLO Website
Security System
Framework ISSAF
OWASP Top 10

ABSTRACT

Information system security is critical in the continuously developing information technology environment. Penetration testing is an effective method for identifying and testing security in a simulated attack on a system or network to find security gaps caused by system security, incorrect configuration, or operational weaknesses in technical processes. This research identifies vulnerabilities on the SIPLO website using penetration testing using the ISSAF (Information System Security Assessment Framework) framework to measure system security. Penetration testing uses the top three types of attacks from the OWASP Top 10: XSS, Brute-force attacks, SQL injection, and DDoS attacks. The results of vulnerability identification on the SIPLO website show that there are 23 vulnerabilities on SIPLO, with two vulnerabilities having a high level, four vulnerabilities having a medium level, 11 vulnerabilities having a low level, and six vulnerabilities having an informational level. Despite these vulnerabilities, the SIPLO website is still protected from XSS and SQL injection attacks. The brute-force attack only succeeded in identifying one username and password. Meanwhile, the DDoS attack did not significantly affect SIPLO, but a difference of 217.221 milliseconds was recorded when the DDoS attack was carried out.

Korespondensi Penulis:

Amin Padmo Azam Masa

Program Studi Sistem Informasi

Universitas Mulawarman Samarinda

JL. Kuaro, Gn. Kelua, Kota Samarinda, Kalimantan Timur

Email: aminpadmo@unmul.ac.id

1. PENDAHULUAN

Keamanan siber (*cybersecurity*) menjadi salah satu perhatian utama bagi organisasi, perusahaan, dan individu. Ancaman seperti serangan *malware*, peretasan, pencurian data dan lainnya dapat menyebabkan kerugian serius seperti kerugian finansial, kehilangan data penting, serta merusak citra dan reputasi [1]. Dalam penanggulangan risiko tersebut, uji penetrasi (*penetration testing*) menjadi alat penting dalam melindungi sistem dan jaringan dari ancaman siber. *Penetration testing* (pentest) adalah suatu kegiatan dimana seseorang mencoba mensimulasikan serangan yang bisa dilakukan terhadap jaringan organisasi/perusahaan tertentu untuk menemukan kelemahan yang ada pada sistem jaringan tersebut [2]. Dalam mengatasi risiko ini, *penetration testing* menjadi alat penting dalam melindungi sistem dan jaringan dari ancaman siber [3]. Dengan melakukan *penetration testing* organisasi atau perusahaan dapat meningkatkan keamanan sistem dan menjaga reputasi organisasi atau perusahaan tersebut [4].

Sistem Informasi Terpadu Layanan Prodi Sistem Informasi (SIPLO) Universitas Mulawarman merupakan sistem yang dirancang khusus untuk menyediakan berbagai layanan, informasi, dan interaksi yang mempermudah pengelolaan akademik dan kehidupan perkuliahan bagi mahasiswa Prodi Sistem Informasi. SIPLO bertujuan untuk meningkatkan efisiensi, aksesibilitas, dan transparansi dalam mengelola berbagai aspek kehidupan akademik mahasiswa. Terdapat beberapa data penting yang perlu dilindungi didalam SIPLO seperti, nilai, data presensi, data skripsi, data PKL dan sebagainya. Berdasarkan hasil observasi langsung pada website dan wawancara dengan pengembang *website* SIPLO diperoleh masalah yang terdapat pada website SIPLO, yaitu *website* SIPLO masih menggunakan HTTP (*Hyeprtext Transfer Protocol*) dimana penggunaan HTTP mempunyai arti bahwa data yang dikirimkan antara klien dan *server* tidak dienkripsi sehingga membuat SIPLO masih sangat rentan dengan ancaman siber. Selain itu, website SIPLO juga belum pernah dilakukan identifikasi kerentanan dengan metode *penetration testing* menggunakan *framework* ISSAF untuk mengukur tingkat keamanan sebuah sistem.

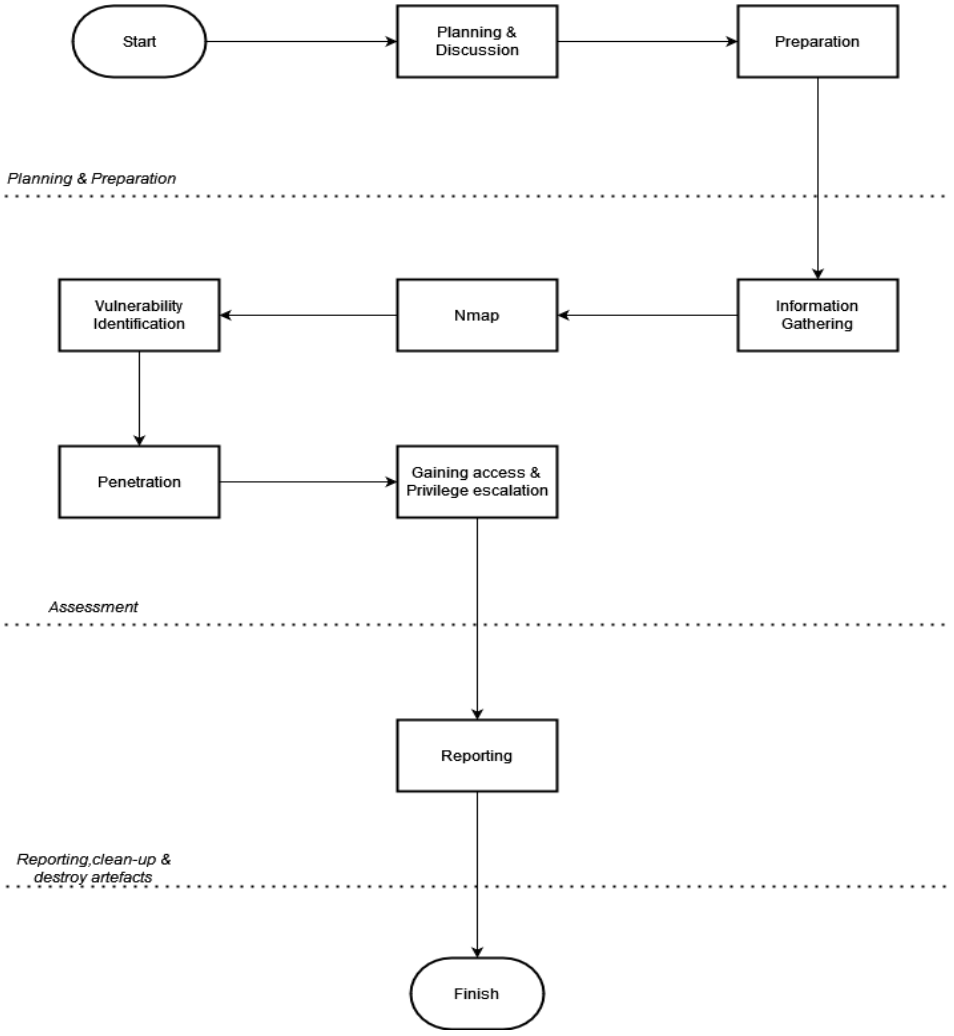
Berdasarkan masalah tersebut *framework* ISSAF (*Information System Security Assessment Framework*) menjadi salah satu *framework* yang disarankan dalam *penetration testing*. *Framework* ISSAF merupakan *framework* yang terarah dan terdiri dari langkah dalam pengelompokkan informasi, penilaian dan laporan hasil pengujian sistem keamanan terhadap domain yang diuji serta melakukan analisa terhadap hasil-hasilnya [5]. Penelitian ini mengimplementasikan serangan dari 3 serangan teratas dari OWASP Top 10 yaitu, XSS, *brute-force attack*, SQL *injection*, DDoS *attack* pada *website* SIPLO dengan tujuan untuk melakukan identifikasi kerentanan dengan melakukan pengujian keamanan *website* SIPLO sehingga diperoleh tingkat keamanan sebuah sistem. OWASP Top 10 adalah sebuah metode yang dirilis oleh komunitas OWASP yang berisikan daftar teratas celah keamanan yang dapat mengancam suatu *website*, daftar ini terus berkembang dan berubah mengikuti perkembangan teknologi *website* yang terus berkembang [6].

2. METODE

2.1. Metode Penelitian

Penelitian ini mengikuti tahapan-tahapan pada *framework* ISSAF, maka alur penelitian yang akan dilewati adalah tahapan *planning & preparation*, *assessment*, dan *reporting, clean-up & destroy artefacts.* Tahap pertama diawali dengan tahap *planning & preparation* dipersiapkan kebutuhan dasar penelitian baik dari sisi *hardware* maupun *software*, instalasi *software* seperti mempersiapkan *Virtual Box* Kali Linux, kemudian didalamnya melakukan instalasi *tools* yang akan digunakan seperti SQLmap, mempersiapkan *wordlists* untuk keperluan *brute-force attack*, LOIC untuk keperluan DDoS *attack*. Tahap kedua, melakukan

kajian pustaka berkaitan dengan penelitian yang dijalankan dan melakukan penjadwalan serangan yang dilakukan. Tahap ketiga yaitu tahap *assessment*, dimana *information gathering*, *network mapping*, dan *vulnerability identification* pada SIPLO dilakukan. Tahap keempat, setelah mendapatkan informasi mengenai SIPLO, maka dimulai proses *penetration* untuk menyerang *website* SIPLO dengan 3 serangan teratas dari OWASP Top 10 yaitu: serangan XSS, *brute-force attack*, SQL *injection*, dan DDoS *attack*. Tahap kelima, yaitu melaporkan apa saja yang ditemukan selama melakukan *penetration testing*, tahapan ini disebut sebagai tahapan *reporting, clean-up & destroy artefacts*.



Gambar 1. Alur Metode Penelitian

2.2. Penetration Testing

Penetration testing (Pentest) adalah penilaian yang dilakukan pada sebuah software atau sistem untuk mengidentifikasi kerentanan yang dapat dieksploitasi [7]. Pentest juga merupakan pendekatan proaktif yang mapan untuk mengevaluasi keamanan asset digital dengan secara aktif mengidentifikasi dan mengeksploitasi kerentanan yang ada [8]. Pentest adalah sebuah metode pengujian terhadap suatu sistem atau jaringan komputer yang bertujuan untuk mengevaluasi keamanan sistem atau jaringan komputer tersebut. Evaluasi dilakukan dengan cara melakukan sebuah simulasi serangan (*attack*) terhadap suatu sistem atau jaringan guna menemukan celah keamanan yang disebabkan oleh keamanan dari suatu sistem, konfigurasi yang tidak benar atau kelemahan yang disebabkan oleh kelemahan dari suatu sistem, konfigurasi yang tidak benar atau kelemahan operasional dalam proses teknik. Laporan hasil dari sebuah pentest akan memberikan masukan terhadap pemilik sistem, tentang celah keamanan terhadap sistem yang dapat digunakan sebagai

bahan evaluasi dari sistem keamanan komputer yang sedang berjalan guna melakukan penambalan kebocoran celah yang terdapat dalam sistem sehingga dapat segera dilakukan pencegahan lebih dini [9].

2.3. OWASP ZAP (*Open Web Application Security Project Zed Attack Proxy*)

OWASP ZAP merupakan *Web Vulnerability Scanner* (WVS) open source yang paling banyak digunakan dan dipelihara. OWASP ZAP menawarkan untuk *crawling* aplikasi web dengan spider tradisional dan AJAX spider dan melakukan serangan untuk menemukan kerentanan yang terdaftar di OWASP Top 10 *most common vulnerabilities* 2021, OWASP ZAP juga memiliki fitur skrip, yang memungkinkan pengguna untuk membuat skrip pada tahap otentikasi [10]. OWASP Zed Attack Proxy adalah alat berbasis Java yang hadir dengan antarmuka grafis intuitif, memungkinkan penguji keamanan aplikasi web untuk melakukan *fuzzing, scripting, spidering*, dan proxy untuk menyerang aplikasi web [11].

2.4. OWASP Top 10

OWASP Top 10 atau yang biasa disebut dengan OWASP 10 adalah sebuah metode yang dirilis oleh komunitas OWASP yang berisikan 10 daftar teratas celah keamanan yang dapat mengancam keamanan suatu website, daftar ini terus berkembang dan berubah-ubah mengikuti perkembangan teknologi website yang terus berkembang [6]. Berdasarkan standar yang dikeluarkan oleh OWASP terdapat beberapa langkah yang dapat dijadikan acuan untuk menilai dan menguji keamanan pada sebuah website. Pada Tabel 1 merupakan rincian perbandingan antara OWASP 10 tahun 2017 dan 2021.

Tabel 1. Perbandingan OWASP Top 10-2017 dan OWASP Top 10-2021

2017	2021
A01:2017-*Injection*	A01:2021-*Broken Access Control*
A02:2017-*Broken Authentication*	A02:2021-*Cryptographic Failures*
A03:2017-*Sensitive Data Exposure*	A03:2021-*Injection*
A04:2017-*XML External Entities (XXE)*	A04:2021-*Insecure Design*
A05:2017-*Broken Access Control*	A05:2021-*Security Misconfiguration*
A06:2017-*Security Misconfiguration*	A06:2021-*Vulnerable and Outdated Components*
A07:2017-*Cross-Site Scripting*	A07:2021-*Identification and Authentication Failures*
A08:2017-*Insecure Deserialization*	A08:2021-*Software and Data Integrity*
A09:2017-*Using Components with Known Vulnerabilities*	A09:2021-*Security Logging and Monitoring*
A10:2017-*Insufficient Logging & Monitoring*	A10:2021-*Server-Side Request Forgery* (SSRF)

2.5. ISSAF (*Information System Security Assessment Framework*)

Information System Security Assessment Framework (ISSAF) adalah kerangka kerja pengujian penetrasi yang memiliki beberapa keunggulan dalam kontrol keamanan, yang memiliki struktur yang jelas dan intuitif yang dapat memandu penguji melalui langkah-langkah yang kompleks. Metodologi ini menjelaskan proses pengujian penetrasi yang optimal untuk membantu penguji melakukan pengujian secara lengkap dan benar, menghindari kesalahan yang umumnya terkait dengan strategi serangan yang dipilih secara acak [12]. Framework ISSAF adalah metodologi yang didasarkan pada kerangka kerja yang dikembangkan untuk mengevaluasi keamanan sistem informasi, sehingga memiliki struktur analisis keamanan di beberapa domain [13].

3. HASIL DAN PEMBAHASAN

3.1. *Planning & Discussion*

Tahap ini merupakan tahap awal dari proses *penetration testing* dalam *framework* ISSAF. Pada tahap ini dilakukan wawancara dengan dosen sekaligus *developer* dari *website* SIPLO. Hasil wawancara tersebut dapat dilihat pada Tabel 2. Skenario serangan diperlukan untuk membandingkan hasil serangan yang

akan dilaksanakan, membantu mengidentifikasi potensi kerentanan, dan kelemahan dalam sistem. Skenario serangan yang dilakukan pada penelitian ini dijelaskan pada Tabel 3.

Tabel 2. Hasil Wawancara

No.	Jawaban
1.	SIPLO belum pernah dilakukan penetration testing
2.	Pernah terjadi serangan kecil-kecilan yang tidak memiliki dampak besar kepada SIPLO
3.	SIPLO tidak memiliki subdomain

Tabel 3. Skenario Serangan

Jenis-Jenis Serangan	Tanggal Serangan
Reflected XSS	4 November 2023, 11 November 2023, 18 November 2023
Brute-Force Attack	4 November 2023, 11 November 2023, 18 November 2023
SQL *Injection*	5 November 2023, 12 November 2023, 19 November 2023
DDoS *Attack*	5 November 2023, 12 November 2023, 19 November 2023

3.2. *Preparation*

Tahap *preparation* (persiapan) melibatkan persiapan sebelum melaksanakan *penetration testing* seperti mempersiapkan sistem operasi Kali Linux. Pada tahap *preparation* juga dilakukannya instalasi *tools* yang akan digunakan pada penelitian seperti *whois, nslookup, whatweb,* dan lain-lainnya untuk keperluan *information gathering*. Kemudian ada *tools* seperti nmap untuk keperluan *network mapping* dan *tool* OWASP ZAP untuk melakukan *vulnerability identification*. Daftar *tools* yang digunakan dalam penelitian ini dapat dilihat pada Tabel 4.

Tabel 4. Daftar *Tools*

Tools	Fungsi	Sumber
nslookup	Mendapatkan informasi dari *server* DNS	Kali Linux
whois	Mendapatkan informasi detail kontak	Kali Linux
whatweb	Mengidentifikasi CMS dari situs *web*	Kali Linux
RED_HAWK	*Information gathering*, *vulnerability scanning,* dan *crawling*	github.com/ Tuhinshubhra/ RED_HAWK
emailscarper.py	Mencari *email* dari sebuah *domain*	Kode Python
SSLscan	Mencari informasi keamanan protokol SSL/TLS di *server*	Kali Linux
nmap	Memindai dan mengidentifikasi *port*	Kali Linux
OWASP Zap	Memindai kerentanan pada *website*	Kali Linux
hydra	Melakukan pengujian *brute-force*	Kali Linux
sqlmap	Melakukan pengujian SQL *injection*	Kali Linux
LOIC	Melakukan pengujian DDoS *attack*	https://sourceforge.net/projects/loic/

3.3. *Information Gathering*

Proses *information gathering* (pengumpulan informasi) juga disebut dengan *reconnaissance* merupakan tahap awal dari fase *assessment* yang terdapat dalam *framework* ISSAF [14]. Proses ini melibatkan pengumpulan data target untuk memahami dan mengevaluasi tentang keamanan sistem atau jaringan yang akan diserang. Terdapat 2 jenis *information gathering/ reconnaissance* yaitu *active reconnaissance* dan *passive reconnaissance* [15]. Pada penelitian ini menggunakan jenis *active reconnaissance* dimana peretas melakukan interaksi langsung dengan target *server* yaitu SIPLO untuk mengumpulkan informasi menggunakan beberapa *tools* seperti nslookup, whois, whatweb, red_hawk, dan SSLscan. *Information gathering* membantu membangun dasar pemahaman yang kuat mengenai target yang akan dievaluasi keamanannya. Hasil dari pencarian IP *address* SIPLO menggunakan tool nslookup dapat dilihat pada Tabel 5. Pada Tabel 5 berhasil didapatkan alamat *server* dan IP *address* dari SIPLO. Kemudian hasil dari pencarian informasi target menggunakan *tool* whois dapat dilihat pada Tabel 6. Penggunaan *tool* whois berhasil didapatkan informasi umum yang dipublikasikan seputar SIPLO, seperti nama perusahaan

beserta alamat, negara, email dan juga nomor telepon. Selanjutnya hasil dari pencarian informasi target menggunakan *tool* whatweb dapat dilihat pada Tabel 7.

Tabel 5. Hasil nslookup

Pencarian	Hasil Pencarian
Server	1**.9*.*4.1**
IP *Address*	1**.1*7.8*.*4

Tabel 6. Hasil whois

Pencarian	Hasil Pencarian
Nama Perusahaan	UPT. TIK Mulawarman Hostmaster
Alamat	Jl. Sambaliung, Kampus Gunung Kelua, Samarinda, Kalimantan Timur
Negara	ID
Telepon	+62 5xx 7xxxx
Email	hostmaster@unmul.ac.id

Tabel 7. Hasil whatweb

Pencarian	Hasil Pencarian
Apache	2.4.6
Bootstrap, Cookie	XSRF_Token, Laravel_Session
Email	si@ft.unmul.ac.id
HTTPOnly	Laravel_Session
Meta Author	SI Unmul
OpenSSL	1.0.2k-fips
PHP	7.3.1, 7.4.33

Menggunakan *tool* whatweb berhasil didapatkan CMS (*Content Management System*) yang digunakan oleh SIPLO. Pengujian SSL (*Secure Socket Layer*) menggunakan *tool* sslscan untuk memindai *port* pada server dan mengidentifikasi protocol SSL/TLS yang didukung. Hasil dari sslscan dapat dilihat pada Tabel 8. Menggunakan *script python* dapat dilakukan *email scanning*. Hasil dari *email scanning* menggunakan *emailscarper.py* ditemukan 4 email yang dapat dilihat pada Tabel 9.

Tabel 8. Hasil sslscan

SSL/TLS *Protocols*	*Status*
SSLv2	*Disabled*
SSLv3	*Disabled*
TLSv1.0	*Disabled*
TLSv1.1	*Disabled*
TLSv1.2	*Enabled*
TLSv1.3	*Disabled*

Tabel 9. Hasil emailscarper

Hasil Pencarian
atasi.jurnal@gmail.com
select2-bootstrap4-theme@x.x.x
si@ft.unmul.ac.id
fteknik.unmul@ft.unmul.ac.id

3.4. *Network Mapping* (Nmap)

Pengujian *network mapping* (pemetaan jaringan) bertujuan untuk memahami infrastruktur jaringan organisasi dan menemukan informasi teknis yang dapat digunakan dalam proses *penetration testing*. Hasil pemindaian nmap dapat dilihat pada Tabel 10.

Tabel 10. Hasil nmap

Port	*State*	*Service*
80/tcp	*Open*	http

Berdasarkan Tabel 10, dapat diketahui bahwa hanya terdapat 1 *port* yang terbuka di SIPLO yaitu *port* 80 yang merupakan salah satu *port* komunikasi yang umum diguakan dalam protokol TCP/IP dan merupakan *port default* untuk protokol http. *Port* 80 merupakan *port* standar untuk layanan web, ketika *user* mengakses situs web menggunakan protokol http, *browser* biasanya secara otomatis menggunakan *port* 80. Namun komunikasi *port* 80 tidak dienkripsi, sehingga informasi yang dikirimkan antara *client* dan *server* tidak dilindungi dari pihak ketiga yang mungkin mencoba megawasi atau mencuri data. HTTP beroperasi menggunakan metode pertukaran *hypertext*, dimana *server* mengirimkan dokumen-halaman web dalam format *hypertext* ke *client* dan dokumen ini kemudian dapat ditampilkan oleh *browser* untuk pengguna.

3.5. *Vulnerability Identification*

Vulnerability Identification (identifikasi kerentanan) dalam *framework* ISSAF merupakan tahap yang penting dalam proses *penetration testing* dengan tujuan untuk mengidentifkasi dan mengevaluasi kerentanan yang ada dalam suatu sistem atau jaringan [16]. Dengan menggunakan *tool* OWASP telah berhasil diidentifikasi 23 kerentanan yang terdapat dalam SIPLO. Kerentanan tersebut dapat dilihat pada Tabel 11.

Tabel 11. *Alerts counts by alert type*

Alert type	*Risk*	*Count*
PII Disclosure	*High*	2 (8.7%)
SQL Injection	*High*	1 (4.3%)
Absence of Anti-CSRF Tokens	*Medium*	2 (8.7%)
Content Security Policy (CSP) Header Not Set	*Medium*	154 (669.6%)
Missing Anti-clickjacking Header	*Medium*	135 (587.0%)
Vulnerable JS Library	*Medium*	4 (17.4%)
Application Error Disclosure	*Low*	2 (8.7%)
Big Redirect Detected (Potential Sensitive Information Leak)	*Low*	3 (13.0%)
Cookie No HttpOnly Flag	*Low*	139 (604.3%)
kalinCookie Without Secure Flag	*Low*	1 (4.3%)
Cookie Without SameSite Attribute	*Low*	1 (4.3%)
Cross-Domain JavaScript Source File Inclusion	*Low*	407 (1,769.6%)
Server Leaks Version Information via "X-Powered-By" HTTP Response Header Field(s)	*Low*	157 (682.6%)
Server Leaks Version Information via "Server" HTTP Response Header Field	*Low*	407 (1,749.6%)
Strict-Transport-Security Header Not Set	*Low*	1 (4.3%)
Timestamp Disclosure - Unix	*Low*	3 (13.0%)
X-Content-Type-Options Header Missing	*Low*	385 (1,673.9%)
Authentication Request Identified	*Informational*	1 (4.3%)
Information Disclosure – Suspicious Comments	*Informational*	352 (1,530.4%)
Modern Web Application	*Informational*	135 (587.0%)
Session Management Response Identified	*Informational*	141 (613.0%)
User Agent Fuzzer	*Informational*	1130 (4,931.0%)
User Controllable HTML Element Attribute (Potential XSS)	*Informational*	4 (17.4%)

Berdasarkan Tabel 11, *tool* OWASP ZAP telah berhasil mengidentifikasi celah keamanan yang terdapat celah untuk SQL *injection* (OWASP_2021_A03) dengan level *high* (tinggi), halaman SIPLO berhasil dimanipulasi dengan menggunakan serangan [1 AND 1 = 1 -] dan [1 OR 1 =1 -] pada URL SIPLO. Dalam melakukan pencegahan serangan ini hendaknya admin tidak membuat kueri SQL dinamis yang menggunakan penggabungan string sederhana, lebih baik juga untuk menerapkan 'daftar izin' untuk karakter yang diizinkan, atau 'daftar tolak' untuk yang karakter yang tidak diizinkan dalam *input* pengguna.

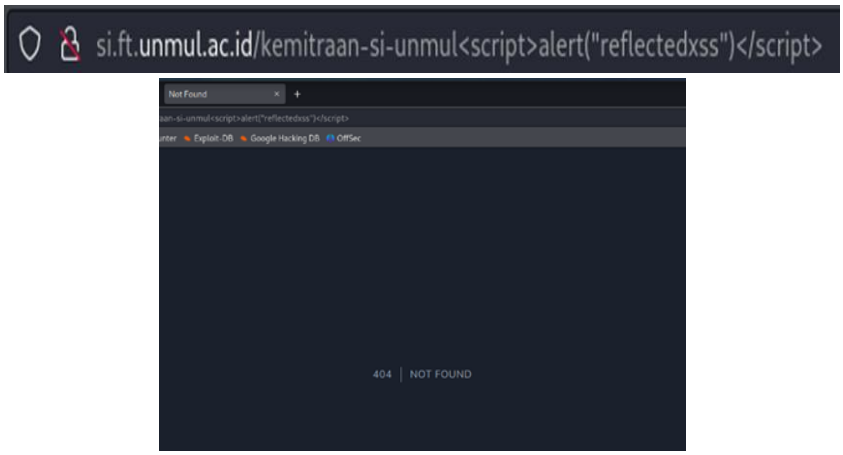
Kerentanan yang terdeteksi selanjutnya adalah tidak ditemukannya anti-CSRF token (OWASP_2021_A01). Serangan CSRF adalah serangan yang melibatkan pemaksaan korban untuk mengirim permintaan HTTP ke tujuan target tanpa sepengatahuan atau niat mereka untuk melakukan tindakan tersebut. Penyebab utamanya adalah fungsionalitas aplikasi yang menggunakan URL atau formulir yang dapat diprediksi dengan cara yang berulang-ulang. Sifat dari serangan ini adalah dengan mengeksploitasi kepercayaan yang dimiliki oleh sebuah situs web terhadap pengguna. Sebaliknya, XSS mengeksploitasi kepercayaan yang dimiliki pengguna terhadap situs web. Risiko pengungkapan informasi meningkat ketika situs target rentan terhadap XSS, karena XSS dapat digunakan sebagai *platform* untuk CSRF. Serangan ini dapat dicegah dengan memastikan situs web bebas dari masalah XSS, karena Sebagian besar pertahanan CSRF dapat dilewati menggunakan skrip yang dikendalikan oleh penyerang. Terdapat juga kerentanan *Cookie without SameSite attribute* dengan level *low*. *Cookie* yang ditetapkan tanpa atribut *SameSite* berarti bahwa *cookie* tersebut dapat dikirim sebagai hasil dari permintaan '*cross-site*'. *SameSite* atribut adalah Tindakan penanggulan yang efektif untuk CSRF, *cross-site script inclusion*, dan *timing attacks*. Pencegahan dapat dilakukan dengan memastikan bahwa atribut *SameSite* disetel ke '*lax*' atau idealnya '*strict*' untuk semua *cookie*. Kerentanan dengan level *low* selanjutnya adalah kerentanan *server leaks information via "X-Powered-By" HTTP Response Header Field(s)* (OWASP_2021_A01) dimana *server* web membocorkan informasi melalui satu atau beberapa *header* respons HTTP "*X-Powered-By*". Akses ke informasi tersebut dapat memudahkan penyerang mengidentifikasi kerangka kerja/ komponen lain yang menjadi sandaran web dan kerentanan yang mungkin dimiliki oleh komponen tersebut. Salah satu cara pencegahannya adalah pastikan server web dikonfigurasikan untuk menekan *header* "*X-Powered-By*".

3.6. *Penetration*

Pada Penelitian ini akan melakukan implementasi 4 jenis serangan (*penetration*) pada website SIPLO yang diuraikan sebagai berikut.

3.6.1 *Reflected XSS*

Reflected XSS juga dikenal sebagai serangan XSS yang tidak persisten atau menetap [17]. *Script* berbahaya dipantulkan ke situs web lain di browser pengguna. Ini terjadi ketika *input* pengguna dari URL atau data POST tercermin pada halaman tanpa disimpan, sehingga memungkinkan penyerang untuk menyuntikkan konten berbahaya [18]. Pada Gambar 3 dimasukkan skrip berbahaya didalam URL menggunakan bahasa Javascript. Jika berhasil maka *website* akan mengembalikan tulisan "*reflectedxss*". Berdasarkan hasil serangan XSS pada website SIPLO yang ditujukkan pada Gambar 2, *website* tidak mengembalikan tulisan tersebut dan menampilkan 404 *not found page*. Sehingga dapat disimpulkan bahwa serangan *reflected* XSS pada SIPLO adalah gagal atau tidak berhasil.

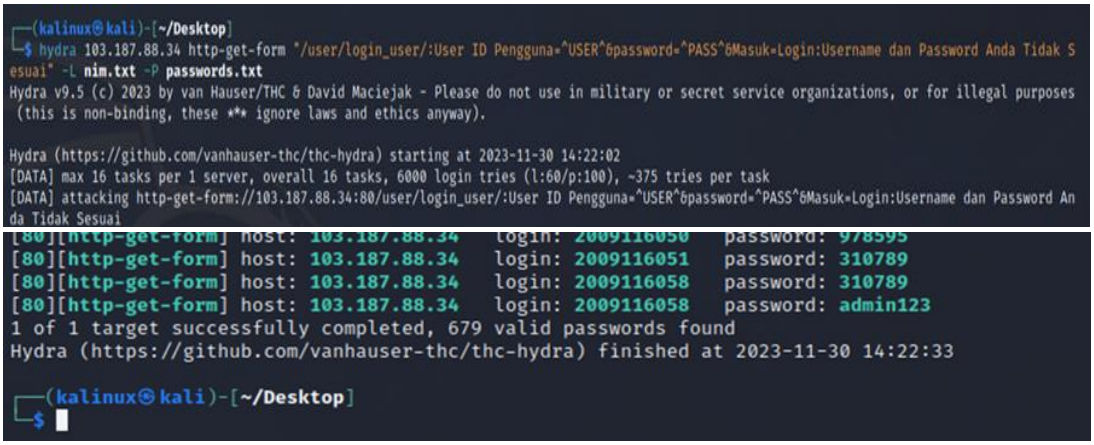


Gambar 2. Pengujian *Reflected* XSS dengan Hasil *Not Found Page*

3.6.2 *Hybrid Brute Force*

Hybrid Brute-Force merupakan serangan yang dilakukan dengan cara menggabungkan 2 metode sebelumnya yaitu metode serangan sederhana dan metode dictionary attack [19]. Umumnya, peretas sudah memiliki daftar password potensial, kemudian menambahkan kombinasi karakter, huruf, dan angka untuk menemukan password yang benar. Contoh password tersebut seperti "anisa1999" atau "siplo2020". *Command line* dari serangan ini diawali dengan menuliskan *tool*-nya yaitu **hydra**, kemudian menuliskan alamat IP, diikuti dengan menentukan *module* yang ingin diserang misalnya HTTP atau SSL, kemudian

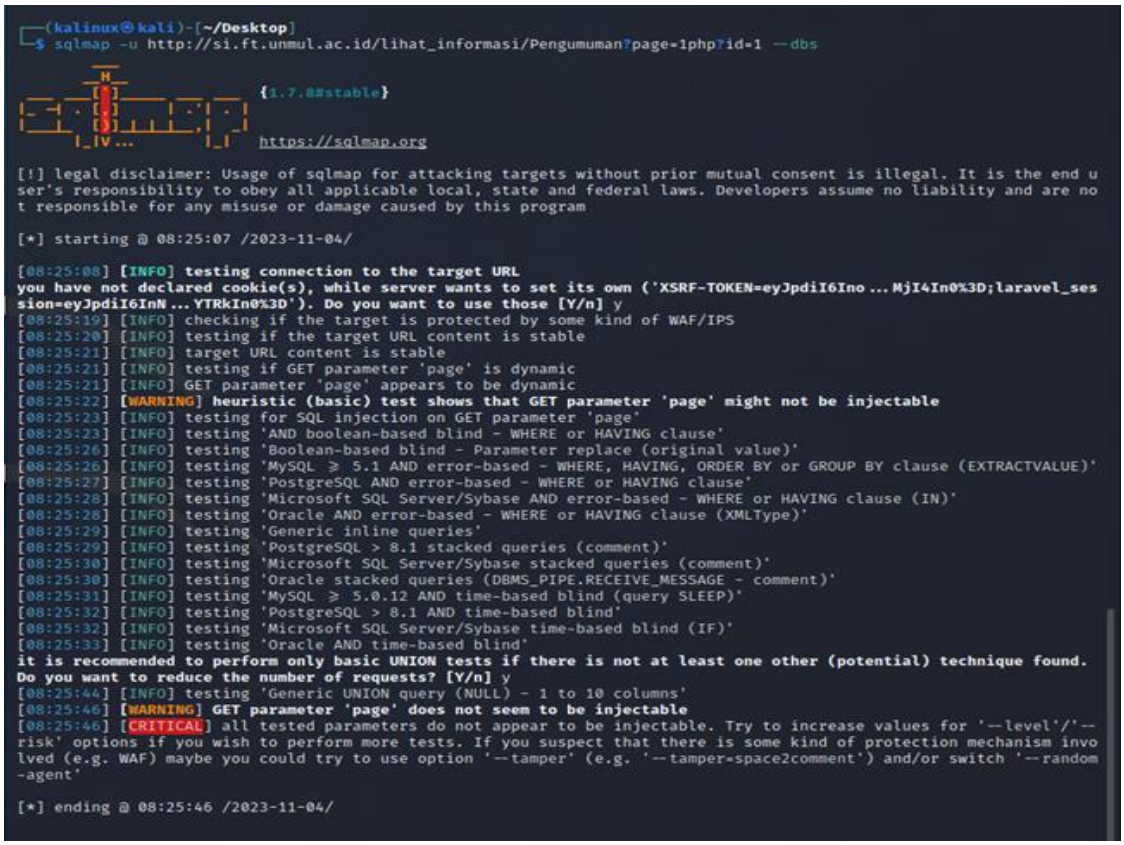
menuliskan *module setting* nya. *Module setting* sendiri bervariasi tergantung dari *website* target. **-L** digunakan untuk memberitahu hydra apa yang dimasukkan dalam *username*, kemudian seperti *command* sebelumnya **-P** digunakan untuk memberitahu hydra apa yang dimasukkan dalam *password*, *Command line* dari serangan ini dapat dilihat pada Gambar 3. Serangan *hybrid brute-force* menggunakan *tool* hydra hanya 1 *username* dan *password* yang berhasil ditemukan.



Gambar 3. *Command Line* Hydra dan Hasil Serangan *Hybrid Brute Force*

3.6.3 *SQL Injection*

SQL *injection* adalah sebuah aksi *hacking* yang dilakukan di aplikasi *client* dengan cara memodifikasi perintah SQL yang ada pada memori aplikasi *client* [20]. SQL *injection* merupakan teknik mengeksploitasi aplikasi berbasis yang di dalamnya menggunakan basis data untuk penyimpanan data [21]. Penyebab utama terjadinya SQL *injection* adalah tidak adanya penangan terhadap karakter (–) karakter tanda petik satu (‘) dan juga karakter *double minus* (- -) yang menyebabkan suatu aplikasi dapat disisipi dengan perintah SQL, sehingga seorang peretas menyisipkan perintah SQL kedalam suatu parameter maupun suatu form [22]. *Bug* SQL *injection* sangat berbahaya, karena teknik ini memungkinkan seseorang dapat *login* ke dalam sistem tanpa harus memiliki akun, selain itu SQL *injection* juga memungkinkan seseorang merubah, menghapus, maupun menambahkan data-data yang berada di dalam *database* tersebut. Bahkan yang lebih berbahaya lagi yaitu mematikan *database* itu sendiri, sehingga tidak bisa memberi layanan kepada *web server*. Berdasarkan Gambar 4, serangan ini gagal karena tidak adanya *cookie* dan semua parameter yang diujikan oleh sqlmap tidak berhasil diinjeksikan.



Gambar 4. SQL *Injection* Menggunakan sqlmap

3.6.4 *DDoS Attack*

Distributed Denial of Service (DDoS) adalah serangan yang menghabiskan sumber daya dengan mengirimkan paket berbahaya dalam jumlah besar, yang mengakibatkan keegagalan layanan jaringan [23]. Serangan ini memanfaatkan ratusan hingga ribuan komputer untuk menyerang suatu perangkat secara bersamaan. Komputer yang digunakan untuk melakukan serangan DDoS akan mengirimkan paket ke satu perangkat dalam jumlah yang sangat besar sehingga perangkat tersebut tidak dapat bekerja sebagaimana mestinya. Serangan ini juga mengakibatkan pengguna tidak dapat memperoleh informasi dan tanggapan dari perangkat target serangan. Berdasarkan hal pertama yang perlu dilakukan dalam LOIC adalah untuk menentukan target URL atau IP, kemudian memilih *port*, menentukan metode dan *threads*, setelah selesai menentukan maka dapat mengklik tombol yang berada di kanan atas LOIC dan secara otomatis LOIC akan langsung mengirimkan *requests* sebanyak-banyaknya pada *website* target.

Sebelum serangan dilakukan, rata-rata waktu akses adalah 10.835 ms, kemudian serangan dilakukan melalui *port* 80 dengan metode TCP menggunakan 900 *threads* selama 1 menit berhasil mengirimkan *requests* sebanyak 12.732.532 *requested* dan rata-rata waktu akses meningkat menjadi 228.056 ms. Meski rata-rata waktu akses meningkat selama serangan berlangsung itu tidak memberi efek yang signifikan terhadap SIPLO dan SIPLO masih dapat diakses seperti biasa hanya saja memerlukan waktu yang sedikit lama dari biasanya. Setelah serangan selesai, rata-rata waktu akses kembali menurun menjadi 11.359 ms.

3.7. *Gaining Access & Privilege Escalation*

Gaining Access & Privilege Escalation merupakan lanjutan dari tahap *penetration* dengan tujuan untuk mendapatkan akses yang lebih tinggi atau hak istimewa di dalam *website*/sistem [24]. Dengan melakukan hal tersebut, peretas dapat mengeksplorasi dan mengeksploitasi kerentanan yang telah diindetifikasi sebelumnya pada tahap *vulnerability identification*. Selanjutnya hasil dari tahap ini dan tahap *penetration* akan dilaporkan kepada pihak SIPLO beserta dengan rekomendasi perbaikan kerentanannya. Namun, karena percobaan penetrasi pada tahap *penetration* tidak berhasil maka peretas gagal untuk mendapatkan akses yang lebih tinggi di dalam SIPLO.

3.8. *Reporting and Result*

Tahap *reporting* melibatkan penyusunan laporan yang merinci mengenai hasil-hasil *penetration testing*, temuan kerentanan serta rekomendasi untuk meningkatkan keamanan sistem/*website*. Dalam penelitian yang telah dilakukan berhasil ditemukan 23 potensi kerentanan pada SIPLO yang mungkin dapat dieksploitasi. Berdasarkan 23 kerentanan tersebut terdapat 4 kerentanan yang berkaitan dengan 3 serangan teratas yang terdapat pada OWASP Top 10. Daftar kerentanan tersebut dapat dilihat di Tabel 12.

Tabel 12. Daftar Kerentanan

Temuan Kerentanan	*Level*	Deskripsi	OWASP No.
Kerentanan SQL *Injection*	*High*	Salah satu halaman SIPLO berhasil dimanipulasi dengan menggunakan [1 AND 1 = 1] dan [1 OR 1 = 1 -]	OWASP_2021_A03
Tidak ditemukannya anti-CSRF token	*Medium*	Kerentanan ini mengakibatkan risiko pengungkapan informasi meningkat karena situs rentan terhapad XSS.	OWASP_2021_A01
Cookie without SameSite attribute	*Low*	*Cookie* yang ditetapkan tanpa atribut *SameSite* berarti bahwa *cookie* tersebut dapat dikirim sebagai hasil dari permintaan '*cross-site*'	OWASP_2021_A01
Server leaks information via "X-Powered-By" HTTP Response Header Field(s)	*Low*	*Server web* membocorkan informasi melalui satu atau beberapa *header* respons HTTP "*X-Powered-By".* Ini dapat mengakibatkan informasi tersebut dapat memudahkan penyerang mengidentifikasi kerangka kerja/komponen yang menjadi sandaran dari *web* tersebut.	OWASP_2021_A01

Langkah-langkah yang dapat diambil untuk mencegah kerentanan yang ada pada Tabel 12 adalah :

1. Tidak membuat kueri SQL dinamis yang menggunakan penggabungan *string* sederhana, lebih baik juga untuk menerapkan 'daftar izin' untuk karakter yang diizinkan, atau 'daftar tolak' untuk karakter yang tidak diizinkan dalam *input* pengguna.
2. Memastikan situs web bebas dari masalah XSS, karena Sebagian besar pertahanan CSRF dapat dilewati menggunakan skrip yang dikendalikan oleh penyerang.
3. Memastikan bahwa atribut *SameSite* disetel ke '*lax*' atau idealnya '*strict*' untuk semua *cookie*.
4. Pastikan *server web* dikonfigurasikan untuk menekan *header* '*X-Powered-By*'.

Hasil dari *penetration testing* yang dilakukan menggunakan 4 jenis serangan berdasarkan 3 serangan teratas OWASP Top 10 disajikan pada Tabel 13. Jenis XSS yang digunakan adalah *reflected* XSS dengan hasil tidak berhasil karena saat skrip berbahaya diinjeksikan pada URL, SIPLO menampilkan 404 *not found page*. Serangan DDoS menggunakan *tool* LOIC (*Low Orbit Ion Cannon*), serangan ini tidak memberi efek yang signifikan terhadap SIPLO hanya saja memerlukan waktu akses yang sedikit lebih lama dari biasanya yaitu dari waktu akses 10.835 ms menjadi lebih lambat sebesar 228.056 ms. Jenis serangan *brute-force attack* yang digunakan adalah *hybrid brute-force* yang menggnakan *tool* hydra dengan hasil tidak berhasil karena hydra hanya berhasil menemukan satu *username* dan satu *password* . Serangan SQL *injection* menggunakan *tool* sqlmap dengan perolehan hasil penelitian "Gagal" atau tidak berhasil melakukan serangan terhadap website SIPLO karena semua parameter yang diujikan oleh sqlmap tidak berhasil diinjeksikan.

Tabel 13. Tabel Hasil Serangan Menggunakan Tiga Serangan teratas OWASP Top 10 Website SIPLO

Pengujian	Jenis Serangan	Hasil
A01:2021-*Broken Access Control*	XSS	Gagal
	DDoS	Gagal
A02:2021-*Crypthographic Failures*	*Brute-Force Attack*	Gagal
A03:2021-*Injection*	SQL *Injection*	Gagal

4. KESIMPULAN

Berdasarkan hasil penelitian menggunakan 3 jenis serangan teratas dari OWASP Top 10 yaitu *Broken Access Control* (Kontrol Akses Yang Rusak), *Crypographic Failures* (Kegagalan Kriptografi), *Injection* (Injeksi), dapat disimpulkan bahwa keamanan *website* SIPLO masih aman. Implementasi proses pertama serangan *hybrid brute-force* menggunakan *tool* hydra dengan hasil mendapatkan satu *username* dan satu *password* yang ditemukan. Proses serangan kedua, *reflected* XSS yang telah dilakukan tidak berhasil diinjeksikan skrip kedalam URL SIPLO. Proses kedua dengan serangan DDoS menggunakan *tool* LOIC menunjukkan tidak mempengaruhi SIPLO secara signifikan, namun tercatat terdapat perbedaan *millisecond* saat mengakses SIPLO ketika serangan DDoS dilakukan yaitu dari waktu akses 10.835 ms menjadi lebih lambat sebesar 228.056 ms. Proses ketiga serangan SQL *injection* menggunakan *tool* sqlmap dengan hasil "Gagal" melakukan serangan terhadap website SIPLO karena semua parameter yang diujikan oleh sqlmap tidak berhasil diinjeksikan.

DAFTAR PUSTAKA

[1] E. Susanto, Lady Antira, K. Kevin, E. Stanzah, and A. A. Majid, "Manajemen Keamanan Cyber di Era Digital," *J. Bus. Entrep.*, vol. 11, no. 1, pp. 23–33, 2023.

[2] A. Kholiq and D. Khoirunnisa, "Analisis Keamanan Wireless Local Area Network (WLAN) Dengan Metode Penetration Testing Execution Standard (PTES)(Studi Kasus: PT. Win Prima Logistik)," *J. Ilm. Fak. Tek. LIMIT'S Vol*, vol. 15, no. 1, 2019.

[3] S. Sahren, R. A. Dalimuthe, and M. Amin, "Penetration Testing Untuk Deteksi Vulnerability Sistem Informasi Kampus," in *Prosiding Seminar Nasional Riset Information Science (SENARIS)*, 2019, vol. 1, pp. 994–1001.

[4] M. Huda, *Keamanan Informasi*. Nulisbuku, 2020.

[5] S. Andriyani, M. F. Sidiq, and B. P. Zen, "Analisis Celah Keamanan Pada Website Dengan Menggunakan Metode Penetration Testing Dan Framework Issaf Pada Website SMK Al-Kautsar," *LEDGER J. Inform. Inf. Technol.*, vol. 2, no. 1, pp. 1–13, 2023.

[6] Y. Yudiana, A. Elanda, and R. L. Buana, "Analisis Kualitas Keamanan Sistem Informasi E-Office Berbasis Website Pada STMIK Rosma Dengan Menggunakan OWASP Top 10," *CESS (Journal Comput. Eng. Syst. Sci.*, vol. 6, no. 2, pp. 185–191, 2021.

[7] A. Rahman and L. Williams, "A bird's eye view of knowledge needs related to penetration testing," in *Proceedings of the 6th Annual Symposium on Hot Topics in the Science of Security*, 2019, pp. 1–2.

[8] I. O. Riandhanu, "Analisis Metode Open Web Application Security Project (OWASP) Menggunakan Penetration Testing pada Keamanan Website Absensi," *J. Inf. dan Teknol.*, pp. 160–165, 2022.

[9] A. Dharmawan, "Penetration Testing Menggunakan Owasp Top 10 Pada Domain Xyz. Ac. Id," *Electro Luceat*, vol. 8, no. 1, pp. 100–108, 2022.

[10] A. Jakobsson and I. Häggström, "Study of the techniques used by OWASP ZAP for analysis of vulnerabilities in web applications." 2022.

[11] R. V. Aditama and E. S. Negara, "Pemindai Kerentanan Terhadap Website Jago Masak Dengan Metode Pengujian Penetrasi OWASP ZAP," *J. Mantik*, vol. 6, no. 3, pp. 3406–3412, 2022.

[12] I. Sanjaya, G. M. A. Sasmita, and D. M. S. Arsa, "Information technology risk management using ISO 31000 based on issaf framework penetration testing (Case study: Election commission of x city)," *Int. J. Comput. Netw. Inf. Secur.*, vol. 12, no. 4, pp. 30–40, 2020.

[13] I. Sanjaya, G. M. A. Sasmita, and D. M. S. Arsa, "Evaluasi Keamanan Website Lembaga X Melalui Penetration Testing Menggunakan Framework ISSAF," *J. Ilm. Merpati*, vol. 8, no. 2, pp. 113–124, 2020.

[14] E. P. Silmina and R. A. A. Amanda, "Analisis Keamanan Jaringan Sistem Informasi Sekolah Menggunakan Penetration Test dan ISSAF," *J. Transm.*, vol. 24, no. 3, pp. 83–91.

[15] R. Maland, "Sudomy: Semi-automated Information Gathering Tools for Subdomain Enumeration and Analysis."

[16] R. Umar, I. Riadi, and M. I. A. Elfatiha, "Analisis Keamanan Sistem Informasi Akademik Berbasis Web Menggunakan Framework ISSAF," *Jutisi J. Ilm. Tek. Inform. dan Sist. Inf.*, vol. 12, no. 1, 2023.

[17] P. Chaudhary, B. B. Gupta, and A. K. Singh, "Securing heterogeneous embedded devices against XSS attack in intelligent IoT system," *Comput. Secur.*, vol. 118, p. 102710, 2022.

[18] S. Suroto and A. Asman, "Ancaman Terhadap Keamanan Informasi Oleh Serangan Cross-Site Scripting (Xss) Dan Metode Pencegahannya," *Zo. Komput. Progr. Stud. Sist. Inf. Univ. Batam*, vol. 11, no. 1, pp. 11–19, 2021.

[19] S. Zhang, X. Xie, and Y. Xu, "A brute-force black-box method to attack machine learning-based systems in cybersecurity," *IEEE Access*, vol. 8, pp. 128250–128263, 2020.

[20] A. Zirwan, "Pengujian dan Analisis Keamanan Website Menggunakan Acunetix Vulnerability Scanner," *J. Inf. dan Teknol.*, pp. 70–75, 2022.

[21] B. Wiguna, W. A. Prabowo, and R. Ananda, "Implementasi Web Application Firewall Dalam Mencegah Serangan SQL Injection Pada Website," *Digit. Zo. J. Teknol. Inf. dan Komun.*, vol. 11, no. 2, pp. 245–256, 2020.

[22] A. Bastian, H. Sujadi, and L. Abror, "Analisis Keamanan Aplikasi Data Pokok Pendidikan (Dapodik) Menggunakan Penetration Testing dan SQL Injection. Infotech Journal, 6 (2), 65â€"70." 2020.

[23] M. Arman and N. Rachmat, "Implementasi Sistem Keamanan Web Server Menggunakan Pfsense," *Jusikom J. Sist. Komput. Musirawas*, vol. 5, no. 1, pp. 13–23, 2020.

[24] A. W. Wardhana and H. B. Seta, "Analisis Keamanan Sistem Pembelajaran Online Menggunakan Metode ISSAF pada Website Universitas XYZ," *Inform. J. Ilmu Komput.*, vol. 17, no. 3, pp. 226–237, 2021.