Digital Transformation Technology (Digitech) | e-ISSN: 2807-9000
Volume 5, **Number** 1, Maret 2025
https://doi.org/10.47709/digitech.v5i1.6881

Analisis Keamanan Website Pada Instansi XYZ Melalui Penetration Testing Menggunakan Framework ISSAF & OWASP

Abdullah[1], Muhammad Koprawi[2*]
[1,2]Universitas Amikom Yogyakarta, Indonesia
[1]abdullah.12@students.amikom.ac.id, [2]koprawi@amikom.ac.id

open access
Histori Artikel:
Diajukan: 22 Agustus 2025
Disetujui: 25 Agustus 2025
Dipublikasi: 26 Agustus 2025
Kata Kunci:
Vulnerability Assessment; ISSAF Framework; Cybersecurity; Penetration Testing; OWASP

Abstrak
Perkembangan teknologi digital telah mengubah cara organisasi beroperasi, termasuk dalam sektor kesehatan yang sangat bergantung pada pengelolaan informasi dan basis data. Informasi menjadi aset berharga yang harus dilindungi dari ancaman pihak tidak berwenang. Serangan siber terhadap institusi kesehatan berpotensi menimbulkan kerugian finansial, kebocoran data, serta gangguan layanan penting. Untuk memitigasi risiko tersebut, *penetration testing* memiliki peran krusial dalam mengidentifikasi kerentanan sistem sebelum dimanfaatkan oleh penyerang. Penelitian ini menggunakan kerangka kerja ISSAF serta OWASP *Web Security Testing Guide* (WSTG) sebagai acuan metodologis, dengan dukungan perangkat OWASP *Zed Attack Proxy* (ZAP) untuk simulasi dan deteksi kelemahan keamanan. Proses pengujian dilakukan melalui enam tahapan utama: *Gathering Publicly Available Information, Network Scanning, System Profiling, Service Profiling, Application Testing,* serta *Vulnerability Identification/Assessment*. Hasil pengujian menunjukkan adanya tiga kerentanan utama pada *website* institusi XYZ, yaitu *Identified Viruses, File Extensions That Have Sensitive Information,* dan *Default Credentials*. Berdasarkan hasil pengujian tambahan menggunakan OWASP ZAP, ditemukan 2 kerentanan dengan tingkat risiko tinggi, 4 berisiko sedang, 5 berisiko rendah, dan 3 bersifat informasional. Temuan ini menegaskan bahwa penerapan framework ISSAF dan OWASP efektif dijadikan pedoman *penetration testing*, terutama jika dikombinasikan dengan instrumen OWASP WSTG. Penelitian ini merekomendasikan 12 langkah perbaikan keamanan untuk memperkuat pertahanan *website*, serta menegaskan pentingnya asesmen keamanan yang sistematis demi menjamin integritas, kerahasiaan, dan ketersediaan data pada institusi kesehatan di era digital.

PENDAHULUAN

Perkembangan dan evolusi di era teknologi sangatlah pesat, internet dan teknologi tercatat telah melakukan banyak perubahan dan perkembangan sehingga masuk dalam segala lini kehidupan masyarakat, kini sebagian besar masyarakat bergantung pada layanan jaringan komputer melebihi masa sebelumnya dengan semakin bertambahnya pengguna layanan internet maka semakin banyak informasi yang dapat diperoleh dari internet. Informasi sendiri menjadi hal penting di era digital ini baik untuk organisasi bisnis maupun individu. Memberikan informasi tentang kepribadiannya seseorang di internet membuat semakin menipisnya privasi yang dimiliki, belakangan ini banyak individu yang mulai sadar dengan bagaimana informasi yang mereka berikan dapat dimanfaatkan dengan cara yang tidak baik (Dewanto, 2018).

Keamanan informasi saat ini sangatlah penting agar dapat menciptakan kenyamanan dalam menggunakan teknologi berbasis *website* tentunya kita perlu memperhatikan aspek keamanan yang menjadi salah satu faktor utama yang perlu diperhatikan. Salah satu indikator yang dapat terlihat adalah banyaknya serangan yang terjadi di indonesia, Badan Siber dan Sandi Negara (BSSN) mencatat lebih dari 1,6 miliar anomaly trafik atau serangan siber sepanjang tahun 2021 ia mengungkapkan data itu diperoleh hasil pemantauan dan identifikasi potensi serangan siber selama 24 jam penuh setiap hari (Prabowo, 2022).

Oleh karena itu, diperlukan proses *penetration testing* atau uji penetrasi untuk menganalisis tingkat keamanan *website* instansi XYZ. *Penetration testing* merupakan simulasi serangan yang dilakukan terhadap sistem dengan tujuan untuk menemukan celah keamanan yang dapat dimanfaatkan oleh penyerang (Palomo-Duarte et al. 2023). Penelitian ini menggunakan kerangka kerja ISSAF *(Information Systems Security Assessment Framework)* dan OWASP *Web Security Testing Guide* (WSTG) sebagai acuan metodologi pengujian, serta memanfaatkan alat OWASP *Zed Attack Proxy* (ZAP) untuk melakukan simulasi serangan dan mendeteksi potensi kerentanan pada sistem. Penelitian ini diharapkan dapat memberikan rekomendasi perbaikan sebagai upaya peningkatan keamanan *website* instansi XYZ sehingga mampu meminimalisir potensi serangan siber di masa depan.

STUDI LITERATUR

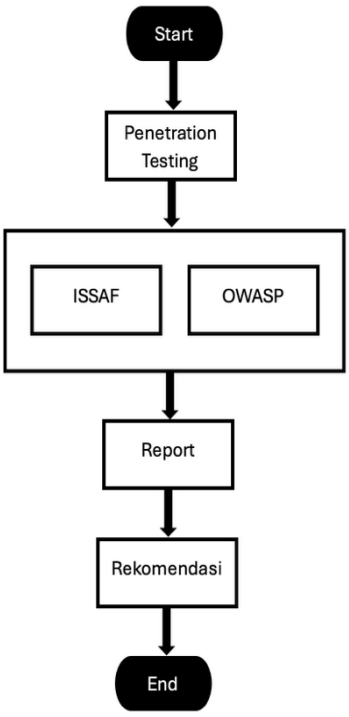
Penelitian yang terkait analisis keamanan melalui pendekatan *penetration testing* menggunakan Framework ISSAF sudah banyak dilakukan. Salah satunya penelitian yang dilakukan oleh (Mario, Tjiptabudi, and Ndaumanu 2024), Penelitian ini melakukan celah keamanan *website* Dapen Xmelalui *penetration testing* menggunakan *Framework* ISSAF, ditemukan 21 kerentanan, terdiri dari 6 kerentanan tingkat menengah dan 15 kerentanan tingkat rendah.

Penelitian selanjutnya (Saragih, Kurniati, and Hidayasari 2025) mengidentifikasi beberapa kerentanan, seperti ketiadaan pengaturan header keamanan, penggunaan versi PHP yang sudah usang, serta potensi terjadinya serangan *Denial of Service* (DoS) menggunakan *Framework* ISSAF. Kemudian penelitian lain (Umar, Riadi, and Wicaksono 2024) melakukan *penetration testing* menggunakan *framework ISSAF* pada sebuah LMS (*Leraning Management System*).

Penelitian terkait penggunakan Framework ISSAF juga telah dilakukan (Nugroho and Christanto 2023). Penelitian ini melakukan penetration testing dengan menggunakan Framework ISSAF dan OWASP yang menunjukkan hasil pengujian bahwa target *website* tidak dapat ditembus karena telah dilengkapi fitur keamanan yang mumpuni.

METODE

Penelitian ini dilakukan melalui pendekatan *Penetration Testing* dengan menggunakan framework ISSAF dan OWASP. Penetration testing sendiri adalah metode aktif non-standar untuk menilai pertahanan jaringan dengan mengikuti prosedur berurutan dan interaktif yang terdiri dari beberapa fase, dimulai dari pengumpulan informasi hingga pelaporan hasil yang diperoleh (Ghanem and Chen 2019). Selain itu *penetration testing* juga dikenal sebagai metode pengujian keamanan yang meniru serangan nyata dengan cara mengakses sistem seperti pengguna biasa, bertujuan untuk menemukan dan mengeksploitasi kerentanan tanpa mengetahui detail internal sistem atau *black box testing* (WSTG - Latest | OWASP Foundation n.d.). Berikut diagram alur penelitian secara umum dapat dilihat pada Gambar 1.



Gambar 1. Alur Penelitian

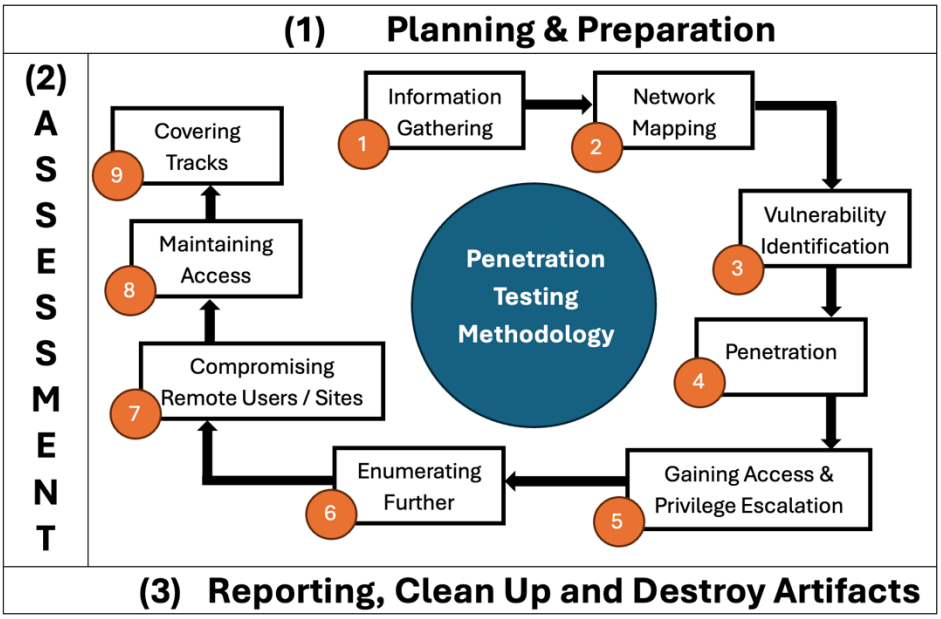
Penelitian ini menggunakan dua pendekatan yaitu menggunakan Framework ISSAF dan menggunakan OWASP WSTG.

Framework ISSAF

Metodologi ISSAF dalam *penetration testing* dirancang untuk memastikan evaluasi menyeluruh terhadap keamanan jaringan, sistem, dan aplikasi. ISSAF menggunakan tiga fase utama yaitu *planning and preparation*, *assessment*, serta *reporting, clean-up and destroy artefacts* (Anggraeni, Zen, and Pranata 2022; Rathore et al. 2005; Teknlogi Informatika dan Komputer Thamrin et al. 2023; Umar et al. 2023).

- **Fase 1: Planning and Preparation**
  Fase *Planning and Preparation* mencakup penandatanganan perjanjian formal (*assessment agreement*) sebagai dasar pelaksanaan serta perlindungan hukum bagi kedua belah pihak. Pada tahap ini juga ditetapkan tim pelaksana, jadwal kegiatan, serta jalur eskalasi yang akan digunakan. Selanjutnya, dilakukan identifikasi pihak yang menjadi kontak utama, penyelenggaraan rapat pembukaan untuk mengonfirmasi ruang lingkup, pendekatan, dan metodologi, serta penetapan kasus uji yang akan dilaksanakan. Seluruh aktivitas pada fase ini bertujuan untuk memastikan proses pengujian berjalan secara terstruktur dan sesuai dengan kesepakatan awal.

- **Fase 2: Assessment**
  Fase *assessment* merupakan inti dari pelaksanaan *penetration testing* dengan menggunakan pendekatan berlapis untuk memperoleh pemahaman komprehensif mengenai kerentanan sistem. Pendekatan ini meliputi *information gathering, network mapping, vulnerability identification, penetration, gaining access & privilege escalation, enumeration further, compromising remote users/sites, maintining access, covering tracks*. Selain itu, dilakukan juga enumerasi lanjutan, kompromi terhadap pengguna atau situs jarak jauh, pemeliharaan akses, serta penyamaran jejak. Tahapan ini dirancang untuk mensimulasikan serangan nyata sehingga dapat mengukur tingkat keamanan sistem secara lebih akurat. Berikut tahapan *penetration testing* dapat dilihat pada Gambar 2 (Rathore et al. 2005).



Gambar 2. *Assessment*

- **Fase 3: Reporting, Clean-up and Destroy Artefacts**
  Fase *Reporting* dalam metodologi ISSAF mencakup penyampaian hasil pengujian secara lisan dan tertulis. Pelaporan lisan dilakukan segera jika ditemukan kerentanan kritis agar organisasi dapat segera melakukan mitigasi. Sementara itu, pelaporan akhir berbentuk laporan tertulis yang sistematis, mencakup ringkasan manajemen, ruang lingkup, metodologi, hasil pengujian, daftar kerentanan, serta rekomendasi perbaikan beserta prioritas tindak lanjut. Selanjutnya, fase *Clean-up and Destroy Artefacts* dilakukan untuk memastikan seluruh data, file, atau informasi yang ditinggalkan selama proses pengujian dihapus dari sistem. Apabila tidak memungkinkan untuk dihapus secara langsung, artefak yang tersisa didokumentasikan dalam laporan teknis agar dapat ditangani oleh tim teknis.

OWASP WSTG

OWASP *Web Security Testing Guide* (WSTG) merupakan panduan komprehensif yang dikembangkan secara kolaboratif oleh para profesional keamanan siber dan relawan, dengan tujuan memberikan kerangka praktik terbaik dalam pengujian keamanan aplikasi web dan layanan web. Sebagai salah satu referensi utama, WSTG banyak digunakan oleh penetration testers maupun organisasi di seluruh dunia untuk memastikan keandalan sistem keamanan digital (Rafeli, Seta, and Widi 2022; WSTG - v4.2 | OWASP Foundation n.d.).

OWASP ZAP

OWASP ZAP *(Zed Attack Proxy)* adalah sebuah *tools open source* yang digunakan untuk pengujian terhadap suatu sistem apakah memiliki kerentanan keamanan (Alazmi and De Leon 2022; Aydos et al. 2022). OWASP dirancang sebagai *tools* untuk *penetration testing*, OWASP ZAP menjadi *tools* yang cocok untuk melakukan pencarian terhadap kerentanan sebuah sistem (Celah Keamanan Website Menggunakan Tools OWASP ZAP Di Kali Linux et al. 2025)(Abdullah n.d.; Samgir et al. 2024).

HASIL

Penetration testing dilakukan dengan beberapa instrumen menggunakan instrumen yang dimiliki oleh WSTG. Proses pengujian dilakukan melalui enam tahapan utama: *Gathering Publicly Available Information, Network Scanning, System Profiling, Service Profiling, Application Testing*, serta *Vulnerability Identification/Assessment*. Tahapan dan kategori pengujian tersebut dapat dilihat pada Tabel 1 berikut:

Tabel 1
Tahapan dan Kategori Pengujian

No	Tahapan	Kategori
1	*Conduct Search Engine Discovery Reconnaissance for Information Leakage*	*Gathering Publicly Available Information*
2	*whois information gathering*	*Gathering Publicly Available Information*
3	*Virus scanner*	*Gathering Publicly Available Information*
4	*Server history*	*Gathering Publicly Available Information*
5	*Review Webserver Metafiles for Information Leakage*	*Gathering Publicly Available Information*
6	*Review Webpage Content for Information Leakage*	*Gathering Publicly Available Information*
7	*Fingerprint Web Application Framework*	*Gathering Publicly Available Information*
8	*Detection's firewall*	*Gathering Publicly Available Information*
9	*Ping network*	*Network Scanning*
10	*Nmap host network scanner*	*Network Scanning*
11	*Nmap system profiling*	*System Profiling*
12	*Nmap service profiling*	*Service Profiling*
13	*Test File Extensions Handling for Sensitive Information*	*Application Testing*
14	*Enumerate Infrastructure and Application Admin Interfaces*	*Application Testing*
15	*Test HTTP Methods*	*Application Testing*
16	*Test File Permission*	*Application Testing*
17	*Test Account Provisioning Process*	*Application Testing*
18	*Testing for Account Enumeration and Guessable User Account*	*Application Testing*
19	*Testing for Credentials Transported over an Encrypted Channel*	*Application Testing*
20	*Testing for Default Credentials*	*Application Testing*
21	*Testing for Weak Lock Out Mechanism*	*Application Testing*
22	*Testing for Browser Cache Weaknesses*	*Application Testing*
23	*Testing for Weak Password Change or Reset Functionalities*	*Application Testing*
24	*Testing Directory Traversal File Include*	*Application Testing*
25	*Testing for Session Management Schema*	*Application Testing*
26	*Testing for Cookies Attributes*	*Application Testing*
27	*Testing for Session Fixation*	*Application Testing*
28	*Testing Session Timeout*	*Application Testing*
29	*Testing for Reflected Cross Site Scripting*	*Application Testing*
30	*Testing for Stored Cross Site Scripting*	*Application Testing*
31	*Testing for SQL Injection*	*Application Testing*
32	*Testing for Weak Transport Layer Security*	*Application Testing*

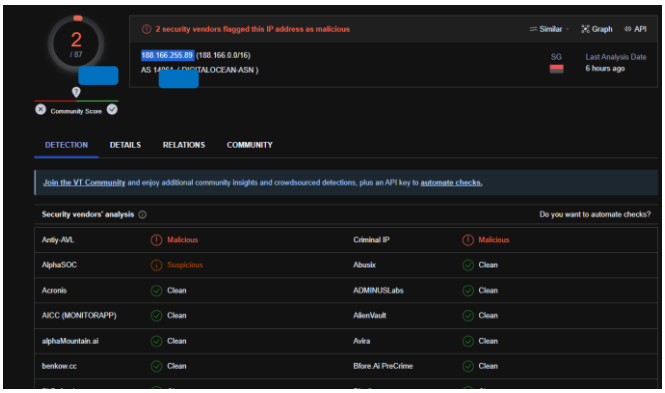
Kemudian berdasarkan hasil uji tersebut didapatkan hasil seperti pada Tabel 2 berikut:

Tabel 2
Hasil Uji *Peneteration Testing*

No	Tahapan	Tools	Keterangan
1	*Conduct Search Engine Discovery Reconnaissance for Information Leakage*	Google search engine	*pass*
2	*whois information gathering*	Whois	*pass*
3	*Virus scanner*	Virustotal	*not pass*
4	*Server history*	Netcraft	*pass*
5	*Review Webserver Metafiles for Information Leakage*	Google search engine, Dirb, Burp Suite	*pass*
6	*Review Webpage Content for Information Leakage*	Google search engine	*pass*
7	*Fingerprint Web Application Framework*	Whatweb, Wappalyzer	*pass*
8	*Detection's firewall*	Whatwaf	*pass*
9	*Ping network*	Ping	*pass*
10	*Nmap host network scanner*	Nmap	*pass*
11	*nmap system profiling*	Nmap	*pass*
12	*nmap service profiling*	Nmap	*not Pass*
13	*Test File Extensions Handling for Sensitive Information*	Google search engine, Burp Suite	
14	*Enumerate Infrastructure and Application Admin Interfaces*	Burp Suite	*pass*
15	*Test HTTP Methods*	Burp Suite	*pass*
16	*Test File Permission*	Burp Suite	*pass*
17	*Test Account Provisioning Process*	Burp Suite	*pass*
18	*Testing for Account Enumeration and Guessable User Account*	Burp Suite	*pass*
19	*Testing for Credentials Transported over an Encrypted Channel*	Burp Suite	*pass*
20	*Testing for Default Credentials*	Burp Suite	*not pass*
21	*Testing for Weak Lock Out Mechanism*	Burp Suite	*pass*
22	*Testing for Browser Cache Weaknesses*	Burp Suite	*pass*
23	*Testing for Weak Password Change or Reset Functionalities*	Burp Suite	*pass*
24	*Testing Directory Traversal File Include*	Burp Suite, Nmap	*pass*
25	*Testing for Session Management Schema*	Burp Suite	*pass*
26	*Testing for Cookies Attributes*	Burp Suite	*pass*
27	*Testing for Session Fixation*	Burp Suite	*pass*
28	*Testing Session Timeout*	Burp Suite	*pass*
29	*Testing for Reflected Cross Site Scripting*	Burp Suite	*pass*
30	*Testing for Stored Cross Site Scripting*	Burp Suite	*pass*
31	*Testing for SQL Injection*	Burp Suite	*pass*
32	*Testing for Weak Transport Layer Security*	Sslscan, Firefox search engine	*pass*

Berdasarkan 32 jenis instrumen yang digunakan, terdapat 29 instrumen berhasil lolos uji dan 3 lainnya tidak lolos uji. 3 instumen tersebut adalah:

a. Virus Scanner (Identified Viruses)
Pada instumen ini untuk melakukan scanning menggunakan virus total seperti pada Gambar 3.

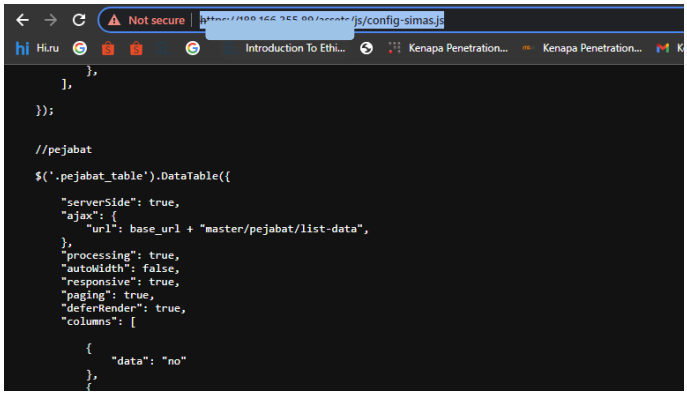


Gambar 3. Virus Total

Setelah melakukan scanning terhadap IP Address yang dimiliki oleh *website* Instansi XYZ, ditemukan IP Address tersebut masuk ke dalam kategori *malicious.*

b. ***Test File Extensions Handling for Sensitive Information***

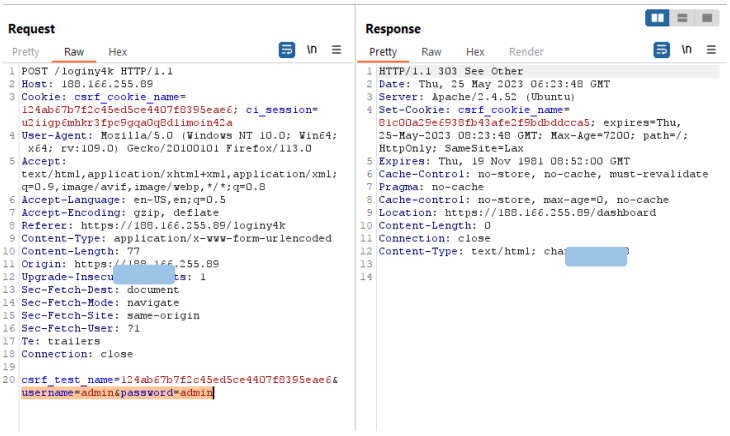
Pada instrumen ini ditemukan akses ke file config dengan format javascript. Pada file javascript ini terdapat beberapa konfigruasi dan path yang seharusnya hanya boleh diakses oleh user yang memiliki credential valid. Hasil uji dapat dilihat pada Gambar 4.



Gambar 4. *File Extension Handling For Sensitive Information*

c. ***Testing for Default Credentials***

Pada instrument ini ditemukan kerentanan terkait penggunaan *Default Credential.* Hasil uji menunjukkan penggunaan *default credential* seperti username: admin dan password: admin. Berdasarkan *default credential* tersebut *attacker* berhasil masuk ke halaman dashboard yang hanya boleh diakses oleh level user administrator. Hasil uji dapat dilihat pada Gambar 5.



Gambar 5. *Testing for default credential*

Kemudian pengujian lanjutan dilakukan dengan menggunakan *tools* OWASP ZAP untuk melakukan *Vulnerability Identification/Assessment*. Berdasarkan hasil laporan dari OWASP ZAP ditemukan hasil seperti Tabel 3 berikut:

Tabel 3
Summary of Alert

Risk Level	Number of Alert
High	2
Medium	4
Low	5
Informational	3

Berdasarkan pada Tabel 3, didapatkan *risk level high* sebanyak 2, *medium* sebanyak 4, *low* sebanyak 5 dan *informational* sebanyak 3.
Kemudian untuk detail dari *alert* tersebut dapat dilihat pada Tabel 4 berikut:

Tabel 4
Detail Alert

Name	Risk Level	Number of Instance
Cloud Metadata Potentiality Exposed	High	1
Path Traversal	High	1
Absence of Anti-CSRF Tokens	Medium	3
Content Security Policy (CSP) Header Not Set	Medium	4
Hidden File Found	Medium	1
Missing Anti-clickjacking Header	Medium	3
Cookie Without Secure Flag	Low	5
Server Leaks Version Information via "Server" HTTP Response Header Field	Low	25
Strict-Transport-Security Header Not Set	Low	23
Timestamp Disclosure – Unix	Low	1
X-Content-Type-Options Header Missing	Low	22
Information Disclosure – Suspicious Comments	Informational	8
Re-examine Cache-control Directives	Informational	1
User Agent Fuzzer	Informational	144

PEMBAHASAN

Hasil *penetration testing* menunjukkan adanya tiga instrumen yang tidak lolos uji dengan tingkat kerentanan yang cukup signifikan. Berdasarkan pengujian menggunakan OWASP ZAP, diperoleh klasifikasi risiko dengan 2 kerentanan tingkat *high*, 4 tingkat *medium*, dan 5 tingkat *low*. Temuan ini mengindikasikan bahwa *website* institusi XYZ masih memiliki celah keamanan yang dapat dieksploitasi oleh pihak tidak berwenang jika tidak segera dilakukan tindakan mitigasi.

Berdasarkan hasil pengujian tersebut, disusun 12 rekomendasi perbaikan keamanan sebagaimana ditunjukkan pada Tabel 5. Rekomendasi ini mencakup aspek teknis dan kebijakan yang berfungsi sebagai langkah preventif maupun kuratif.

Tabel 5
Rekomendasi

No.	Issue	Rekomendasi
1	*Identified Viruses*	Segera lakukan backup file pada *website* dan hapus file yang teridentifikasi secara manual.
2	*Website Not Secure*	Pastikan sertifikat SSL sudah terpasang dengan benar. Jika muncul error seperti ERR_SSL_PROTOCOL_ERROR, lakukan pengecekan ulang instalasi sertifikat. Tambahkan https:// ke URL. Pastikan ikon gembok muncul pada browser.
3	*File Extensions That Have Sensitive Information*	Lakukan pengecekan rutin terhadap file ekstensi yang terekspos. Tidak boleh ada informasi sensitif yang dapat diakses publik.

4	*Default Credential*	Terapkan kebijakan kredensial yang kuat, tidak menggunakan password default, dan gunakan kombinasi yang sulit ditebak.
5	*Missing Header (HSTS)*	Tambahkan HTTP Strict Transport Security (HSTS) pada header: Strict-Transport-Security: max-age=31536000; includeSubDomains
6	*Missing Header (X-Frame-Options)*	Tambahkan X-Frame-Options untuk mencegah clickjacking: X-Frame-Options: SAMEORIGIN
7	*Missing Header (SameSite Cookies)*	Perbarui atribut session cookies agar lebih aman dengan menambahkan: SameSite=Strict
8	*Open Port*	Hindari penggunaan port default 22 (SSH) yang rawan brute-force. Gunakan port yang lebih tinggi (misal 31000, 41762) atau batasi akses hanya dari IP tertentu.
9	*Website Application Firewall (WAF)*	Gunakan WAF seperti Cloudflare untuk menambah lapisan keamanan, menyembunyikan informasi server, serta melindungi dari serangan umum.
10	*Port Expose*	Sembunyikan port sensitif (22, 25) dari pemindaian jaringan. Biarkan port umum (80, 443) terbuka sesuai kebutuhan layanan.
11	*User Policy*	Buat kebijakan user role yang ketat. Misalnya, perubahan kredensial oleh user harus menunggu persetujuan admin agar ada validasi.
12	*Session Quality*	Tingkatkan keamanan session. Jangan gunakan session default framework (misalnya CodeIgniter). Disarankan beralih ke JWT (JSON Web Token) untuk manajemen session.

KESIMPULAN

Berdasarkan hasil penetration testing yang telah dilakukan, dapat disimpulkan bahwa pendekatan Framework ISSAF dan OWASP terbukti sangat efektif digunakan sebagai pedoman dalam melakukan pengujian keamanan sistem, terutama ketika dikombinasikan dengan instrumen yang dimiliki oleh OWASP WSTG. Proses pengujian yang mencakup enam tahapan utama berhasil mengidentifikasi kerentanan kritis pada *website* institusi XYZ, antara lain *Identified Viruses, File Extensions That Have Sensitive Information,* dan *Default Credentials*. Selain itu, melalui pengujian menggunakan OWASP ZAP diperoleh klasifikasi risiko dengan 2 kerentanan tingkat high, 4 tingkat medium, 5 tingkat low, serta 3 kategori informational. Temuan ini menunjukkan bahwa sistem yang digunakan masih memiliki kelemahan signifikan yang dapat dimanfaatkan oleh pihak tidak berwenang apabila tidak segera dilakukan mitigasi. Untuk meningkatkan ketahanan sistem, penelitian ini menyusun 12 rekomendasi keamanan, yaitu: (1) segera melakukan *backup* dan penghapusan file yang teridentifikasi sebagai virus, (2) memastikan pemasangan sertifikat SSL secara benar, (3) melakukan pemeriksaan rutin terhadap file ekstensi yang berisi informasi sensitif, (4) menerapkan kebijakan kredensial yang kuat dan menghindari penggunaan password *default*, (5) menambahkan *header* HSTS pada konfigurasi server, (6) menambahkan *header X-Frame-Options* untuk mencegah *clickjacking*, (7) memperbarui atribut *SameSite* pada *cookies*, (8) mengamankan *port* layanan dengan menghindari penggunaan *port default* atau membatasi akses, (9) menerapkan *Website Application Firewall* (WAF), (10) menyembunyikan *port* sensitif dari pemindaian publik, (11) membuat kebijakan *user role* yang ketat dengan mekanisme validasi, dan (12) meningkatkan kualitas manajemen *session* dengan menggunakan JWT. Dengan implementasi rekomendasi tersebut, diharapkan institusi XYZ dapat memperkuat pertahanan *website* terhadap potensi serangan siber, sekaligus menjamin aspek *confidentiality*, *integrity*, dan *availability (CIA Triad)* dalam mendukung keberlangsungan layanan kesehatan di era digital.

REFERENSI

Abdullah, Hilmi S. n.d. “Evaluation of Open Source Web Application Vulnerability Scanners.” *Academic Journal of Nawroz University*. doi:10.25007/ajnu.v9n1a532.

Alazmi, Suliman, and Daniel Conte De Leon. 2022. “A Systematic Literature Review on the Characteristics and Effectiveness of Web Application Vulnerability Scanners.” *IEEE Access* 10:33200–219. doi:10.1109/ACCESS.2022.3161522.

Anggraeni, Ditya Putri, Bita Parga Zen, and Mega Pranata. 2022. “Security Analysis On Websites Using The Information System Assessment Framework (Issaf) And Open Web Application Security Version 4 (OWASPv4) Using The Penetration Testing Method.” *Jurnal Pertahanan: Media Informasi Tentang Kajian*

Dan Strategi Pertahanan Yang Mengedepankan Identity, Nasionalism Dan Integrity 8(3):497–506. doi:10.33172/JP.V8I3.1777.

Aydos, Murat, Çiğdem Aldan, Evren Coşkun, and Alperen Soydan. 2022. "Security Testing of Web Applications: A Systematic Mapping of the Literature." *Journal of King Saud University - Computer and Information Sciences* 34(9):6775–92. doi:10.1016/J.JKSUCI.2021.09.018.

Celah Keamanan Website Menggunakan Tools OWASP ZAP Di Kali Linux, Analisis, Calvin Bernandra Putra Pura, Try Yudha Maulana, Aldi Februri, Tamsir Ariyadi, Universitas Bina Darma, JlnJendral Ahmadyani No, Kecamatan Seberang Ulu, and Kota Palembang Sumatra Selatan. 2025. "Analisis Celah Keamanan Website Menggunakan Tools OWASP ZAP Di Kali Linux." *JUSTER : Jurnal Sains Dan Terapan* 4(1):46–51. doi:10.57218/JUSTER.V4I1.1341.

Ghanem, Mohamed C., and Thomas M. Chen. 2019. "Reinforcement Learning for Efficient Network Penetration Testing." *Information 2020, Vol. 11, Page 6* 11(1):6. doi:10.3390/INFO11010006.

Mario, Fransiskus, Hartono Tjiptabudi, and Ricky Imanuel Ndaumanu. 2024. "Evaluasi Celah Keamanan Website Dana Pensiun X Melalui Penetration Testing Berdasarkan ISSAF Framework." *Jurnal Algoritma* 21(2):9–17. doi:10.33364/ALGORITMA/V.21-2.1644.

Nugroho, Verseveranda Setyo, and Febrian Wahyu Christanto. 2023. "Analisis Keamanan Website Dengan Information System Security Assessment Framework (Issaf) Dan Open Web Application Security Project (OWASP)." *NERO (Networking Engineering Research Operation)* 8(2):145–56. doi:10.21107/NERO.V8I2.19712.

Palomo-Duarte, Manuel, Juan Antonio Caballero-Hernandez, Esra Abdullatif Altulaihan, Abrar Alismail, and Mounir Frikha. 2023. "A Survey on Web Application Penetration Testing." *Electronics 2023, Vol. 12, Page 1229* 12(5):1229. doi:10.3390/ELECTRONICS12051229.

Rafeli, Albestty Islamyati, Henki Bayu Seta, and I. Wayan Widi. 2022. "Pengujian Celah Keamanan Menggunakan Metode OWASP Web Security Testing Guide (WSTG) Pada Website XYZ." *Informatik : Jurnal Ilmu Komputer* 18(2):97–103. doi:10.52958/IFTK.V18I2.4632.

Rathore, Balwant, Mark Brunner, Miguel Dilaj, Omar Herrera, Piero Brunati, Rama Subramaniam, Subash Raman, and Umesh Chavan. 2005. "Information Systems Security Assessment Framework (ISSAF) Draft 0.2."

Samgir, Amit Bharat, Vitthal Gutte, Kishor Kolhe, and Dhanashri R. Patil. 2024. "Automated Penetration Testing Architecture Using Metasploit and OWASP ZAP for Web Applications." *2nd International Conference on Sustainable Computing and Smart Systems, ICSCSS 2024 - Proceedings* 649–57. doi:10.1109/ICSCSS60660.2024.10625033.

Saragih, Elpi Aprianti, Rezki Kurniati, and Nurmi Hidayasari. 2025. "A Analisis Keamanan Website SMP Negeri 2 Bagan Sinembah Dengan Framework ISSAF." *Jurnal Teknologi Sistem Informasi Dan Sistem Komputer TGD* 8(2):173–83. doi:10.53513/JSK.V8I2.11744.

Teknlogi Informatika dan Komputer Thamrin, Jurnal MH, Imam Riadi, Yudi Kurniawan, and Irhash Ainur Rafiq. 2023. "Analisis Keamanan Website Menggunakan Information System Security Asessment Framework(ISSAF)." *Jurnal Teknologi Informatika Dan Komputer* 9(1):126–36. doi:10.37012/JTIK.V9I1.1439.

Umar, Rusydi, Imam Riadi, Muhammad Ihya, and Aulia Elfatiha. 2023. "Analisis Keamanan Sistem Informasi Akademik Berbasis Web Menggunakan Framework ISSAF." *Jutisi : Jurnal Ilmiah Teknik Informatika Dan Sistem Informasi* 12(1). https://ojs.stmik-banjarbaru.ac.id/index.php/jutisi/article/view/1191.

Umar, Rusydi, Imam Riadi, and Sonny Abriantoro Wicaksono. 2024. "Security Analysis of Learning Management System Using Penetration Testing with ISSAF Framework: LMS Security Analysis Using the Pentest Method with the ISSAF Framework." *PIKSEL : Penelitian Ilmu Komputer Sistem Embedded and Logic* 12(1):59–68. doi:10.33558/PIKSEL.V12I1.8331.

WSTG - Latest | OWASP Foundation. n.d. Retrieved August 20, 2025. https://owasp.org/www-project-web-security-testing-guide/latest/2-Introduction/README#penetration-testing.

WSTG - v4.2 | OWASP Foundation. n.d. Retrieved August 21, 2025. https://owasp.org/www-project-web-security-testing-guide/v42/.