CyberSecurity dan Forensik Digital e-ISSN: 2615-8442
Vol. 8, No. 1, Mei 2025, hlm. 25-33

Pengujian dan Mitigasi Kerentanan *Website* Sistem Informasi Akademik Universitas Ma'arif Nahdlatul Ulama Kebumen dengan OWASP ZAP

Eko Setiawan[1], Fahmi Fachri [2]

[1,2] Teknik Informatika, Fakultas Teknik, Universitas Ma'arif Nahdlatul Ulama
Email: [1]ekosetiawancc7@gmail.com, [2] fahmifachriumnu@gmail.com

Abstrak

Penggunaan sistem informasi akademik berbasis web di lingkungan pendidikan tinggi semakin krusial untuk mendukung proses manajemen data akademik. Namun, tingginya ketergantungan pada aplikasi web juga meningkatkan risiko terhadap serangan siber. *Website* Sistem Informasi Akademik Universitas Ma'arif Nahdlatul Ulama Kebumen sempat mengalami insiden peretasan yang menyebabkan tampilan berubah menjadi iklan judi online, meskipun saat ini telah dipulihkan. Berdasarkan insiden tersebut, tujuan penelitian ini dilakukan untuk mengidentifikasi potensi kerentanan lainnya dan memberikan rekomendasi mitigasi. Penelitian menggunakan metode pengujian keamanan berbasis OWASP Web Security Testing Guide (WSTG) dan alat bantu OWASP Zed Attack Proxy (ZAP). Hasil pengujian menunjukkan adanya tiga kerentanan utama, yaitu Content Security Policy (CSP) Header Not Set, HTTP to HTTPS Insecure Transition in Form Post, dan Missing Anti-clickjacking Header. Kendati tidak ditemukan celah XSS aktif dan semua transmisi data telah dienkripsi melalui HTTPS, sistem tetap belum memiliki perlindungan terhadap Clickjacking. Mitigasi yang direkomendasikan mencakup penerapan header CSP, konfigurasi HSTS, serta penambahan X-Frame-Options atau frame-ancestors. Implementasi mitigasi ini diharapkan dapat meningkatkan keamanan sistem informasi akademik dari potensi serangan siber di masa mendatang.

Kata kunci: Keamanan *Website*, OWASP ZAP, Wireshark, XSS, Clickjacking, OWASP Top 10

Testing and Mitigation of Website Vulnerabilities in the Academic Information System of Universitas Ma'arif Nahdlatul Ulama Kebumen using OWASP ZAP

Abstract

The use of web-based academic information systems in higher education has become increasingly vital for managing academic data. However, this reliance on web applications also increases the risk of cyberattacks. The Academic Information System website of Universitas Ma'arif Nahdlatul Ulama Kebumen previously experienced a hacking incident in which the display was altered to show online gambling advertisements, although it has since been restored. This research aims to identify other potential vulnerabilities and provide mitigation recommendations. The study employs security testing based on the OWASP Web Security Testing Guide (WSTG) and utilizes the OWASP Zed Attack Proxy (ZAP) tool. The results reveal three main vulnerabilities: Content Security Policy (CSP) Header Not Set, HTTP to HTTPS Insecure Transition in Form Post, and Missing Anti-clickjacking Header. Although no active XSS exploit was found and all data transmissions were encrypted via HTTPS, the system lacks protection against clickjacking attacks. Recommended mitigation includes implementing CSP headers, enabling HTTP Strict Transport Security (HSTS), and adding X-Frame-Options or frame-ancestors directives. These measures are expected to enhance the security of the academic information system and protect user data from future cyber threats.

***Keywords**: Website Security, OWASP ZAP, Wireshark, XSS, Clickjacking, OWASP Top 10*

1. PENDAHULUAN

Dokumen Penggunaan aplikasi web sebagai media informasi dalam dunia pendidikan semakin berkembang, terutama di perguruan tinggi. Aplikasi web berperan dalam menyajikan informasi mengenai universitas kepada masyarakat umum, mencakup sejarah, visi dan misi, kompetensi, profil institusi, serta informasi akademik terkini (R.A. Dwi Ayu Puspitasari 2021). Keberadaan sistem informasi akademik memungkinkan mahasiswa untuk mengakses informasi dengan lebih mudah dan cepat,

sehingga meningkatkan efisiensi dalam proses akademik.

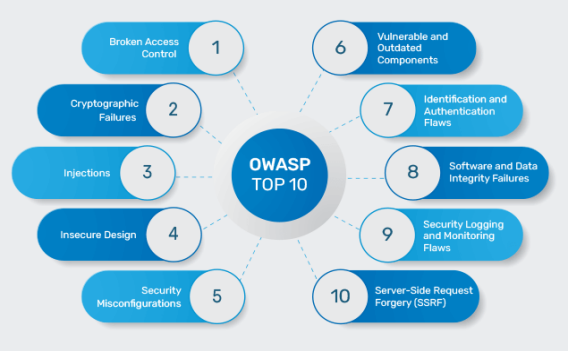
Namun, meskipun aplikasi web memberikan kemudahan dalam akses informasi, keberadaannya juga rentan terhadap ancaman keamanan siber. Menurut laporan Badan Siber dan Sandi Negara (BSSN) yang bekerja sama dengan Indonesia Honeynet Project (IHP), terdapat 12.895.554 serangan siber yang menargetkan aplikasi berbasis web pada tahun 2024 (Muhammad Fauzi et al. 2024). Data ini menunjukkan bahwa serangan siber terhadap aplikasi web merupakan isu yang serius dan tidak dapat diabaikan. Dengan meningkatnya penggunaan sistem informasi akademik dalam dunia pendidikan, keamanan aplikasi web menjadi aspek yang krusial untuk diperhatikan.

Salah satu kasus serangan siber yang pernah terjadi adalah peretasan terhadap Sistem Informasi Universitas Ma'arif Nahdlatul Ulama (UMNU) Kebumen, di mana tampilan *website* sempat berubah menjadi iklan judi online. Meskipun sistem informasi tersebut telah diperbaiki dan dikembalikan ke kondisi normal, insiden ini menunjukkan bahwa masih ada kemungkinan kerentanan yang belum terdeteksi sepenuhnya. Oleh karena itu, penelitian ini dilakukan untuk mengevaluasi apakah masih terdapat kelemahan keamanan pada sistem informasi akademik UMNU Kebumen. Selain itu, penelitian ini juga akan memberikan langkah-langkah mitigasi untuk mengurangi risiko yang dapat dimanfaatkan oleh pihak yang tidak bertanggung jawab.

Berbagai jenis serangan siber, seperti SQL Injection, Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), serta kelemahan dalam autentikasi, dapat menyebabkan pencurian data akademik, perubahan informasi secara tidak sah, atau bahkan penghentian layanan secara tiba-tiba (Alwi Putra Supendi 2024). Oleh karena itu, diperlukan pengujian keamanan secara berkala untuk memastikan bahwa sistem informasi akademik aman dari berbagai ancaman siber.

Dalam penelitian ini, pengujian keamanan akan dilakukan dengan menggunakan *OWASP Zed Attack Proxy* (ZAP), sebuah alat open-source yang dikembangkan oleh *Open Web Application Security Project* (OWASP) dan digunakan untuk mendeteksi berbagai jenis kerentanan keamanan dalam aplikasi web (Ramadhan, 2024). Pengujian ini akan berfokus pada OWASP Top 10, yaitu sepuluh jenis kerentanan keamanan aplikasi web yang paling umum dan berisiko tinggi menurut OWASP.

Selain menggunakan OWASP ZAP, penelitian ini juga menerapkan *metode Web Security Testing Guide* (WSTG) sebagai panduan dalam melakukan pengujian keamanan. WSTG dipilih karena merupakan standar yang telah banyak digunakan dalam pengujian keamanan aplikasi web dan mencakup berbagai aspek penting, seperti autentikasi, otorisasi, enkripsi, serta validasi input. Dengan mengikuti metodologi WSTG, pengujian akan lebih sistematis dan menyeluruh, sehingga hasil analisis dapat memberikan rekomendasi mitigasi yang lebih akurat dan relevan.



Gambar 1 OWASP Top 10

Proses pengujian akan mencakup pemindaian terhadap berbagai aspek keamanan aplikasi, seperti pengelolaan autentikasi, enkripsi data, serta validasi input pengguna. Dengan menggunakan OWASP ZAP dan metodologi WSTG, penelitian ini bertujuan untuk mengidentifikasi kerentanan yang terdapat pada *website* sistem informasi akademik Universitas Ma'arif Nahdlatul Ulama Kebumen. Selain itu, penelitian ini juga akan menyajikan langkah-langkah mitigasi untuk mengurangi risiko yang ditemukan. Dengan demikian, hasil penelitian ini diharapkan dapat berkontribusi dalam meningkatkan keamanan sistem informasi akademik serta melindungi data institusi dari ancaman siber dan potensi kerusakan akibat serangan. Kontribusi utama dari penelitian ini adalah melakukan pengujian keamanan secara menyeluruh dengan pendekatan kombinatif antara OWASP ZAP dan WSTG terhadap sistem informasi akademik yang sebelumnya telah mengalami insiden peretasan, serta menyajikan bukti konkret berupa payload dan response yang digunakan dalam pengujian untuk memperkuat rekomendasi mitigasi yang diberikan.

2. KAJIAN PUSTAKA

2.1. Keamanan Aplikasi Web

Menurut (Alamsyah 2021) Keamanan pada sebuah aplikasi web merupakan aspek penting yang harus dimiliki. Memasang firewall, anti virus, atau software sejenis pada komputer atau router yang terhubung langsung atau berada dalam jaringan dengan server aplikasi web dapat membantu mengamankan aplikasi web.

2.2. OWASP TOP 10

OWASP Top 10 merupakan daftar yang disusun oleh Open Web Application Security Project (OWASP) untuk mengidentifikasi sepuluh jenis kerentanan keamanan paling signifikan yang umum terjadi pada aplikasi web (Ela Nurelasari and Difa Gumilang Al Farabi 2024). Daftar ini diperbaharui secara berkala dengan mempertimbangkan perkembangan teknologi, tren serangan terbaru, serta

hasil analisis dari berbagai insiden keamanan siber yang terjadi di seluruh dunia.

2.3. Zed Attack Proxy

ZAP merupakan alat open-source yang dikembangkan oleh OWASP untuk membantu *Penetration Testing* dan pengembang dalam menemukan serta mengatasi celah keamanan yang ada dalam sebuah aplikasi web (Handaya and Islamadina 2025).

2.4. Web Security Testing Guide

WSTG merupakan panduan pengujian keamanan aplikasi web yang dikembangkan oleh *Open Web Application Security Project* (OWASP)(Sya'bani and Rahma 2022). Panduan ini dirancang untuk membantu penguji keamanan, pengembang, serta administrator sistem dalam mengidentifikasi dan mengevaluasi celah keamanan pada aplikasi berbasis web. WSTG mencakup berbagai metode dan teknik yang digunakan dalam pengujian keamanan guna memastikan bahwa aplikasi web telah dikonfigurasi dengan aman dan terlindungi dari potensi eksploitasi oleh pihak yang tidak bertanggung jawab.

2.5. *Penetration Testing*

Penetration testing merupakan kegiatan untuk menilai keamanan sistem yang telah dibangun dengan cara melakukan simulasi serangan yang umum digunakan oleh peretas. Dalam penelitian yang dilakukan oleh (Fachri et al. 2021) pentest berhasil menemukan kelemahan pada sebuah *website* melalui eksploitasi terhadap celah keamanan pada port 22 yang terbuka, sehingga menunjukkan pentingnya pengujian sistem secara menyeluruh untuk mengidentifikasi potensi risiko.

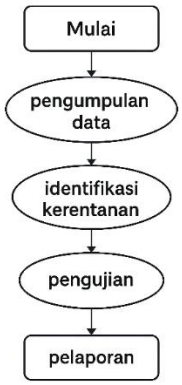
2.6. Penelitian serupa

Beberapa penelitian di Indonesia telah menerapkan pendekatan yang sejenis dalam pengujian keamanan sistem berbasis web. misalnya dalam penelitian (Akhson and Fachri 2025) menggunakan OWASP ZAP mampu menemukan kerentanan sql injection dengan tingkat risiko tinggi

Penelitian ini mengadopsi pendekatan serupa, namun dengan kontribusi yang lebih spesifik, yaitu penerapan metodologi OWASP ZAP dan panduan Web Security Testing Guide (WSTG) untuk mengevaluasi sistem informasi akademik Universitas Ma'arif Nahdlatul Ulama Kebumen yang sebelumnya pernah mengalami insiden peretasan. Penelitian ini berkontribusi dalam penyusunan langkah mitigasi berbasis temuan aktual, serta menyajikan perbandingan kondisi keamanan sebelum dan sesudah dilakukan pengujian, sesuatu yang belum banyak dibahas secara mendalam dalam penelitian terdahulu di Indonesia. Gap analysis menunjukkan bahwa sebagian besar penelitian terdahulu cenderung terbatas pada identifikasi kerentanan secara umum tanpa pengujian berdasarkan kategori WSTG yang terstruktur, tanpa dokumentasi konfigurasi alat yang rinci, serta belum menyertakan validasi hasil dan pertimbangan etis secara menyeluruh sebagaimana yang dilakukan dalam penelitian ini

3. METODOLOGI

Penelitian ini menggunakan kerangka kerja OWASP *Web Security Testing Guide* (WSTG) karena panduan ini lebih teknis dan spesifik dalam menguji aplikasi web dibandingkan metode lain seperti NIST SP 800-115 yang lebih bersifat kebijakan, atau PTES yang lebih fokus pada aspek manajerial. WSTG dinilai lebih cocok karena memiliki struktur modular yang selaras dengan kemampuan OWASP ZAP dalam mendeteksi kerentanan berbasis kategori seperti Authentication, Authorization, dan Session Management. Pengujian dilakukan menggunakan OWASP ZAP versi 15.**1** dengan *metode Active Scan* dan *Passive Scan* untuk memastikan identifikasi kerentanan dilakukan secara menyeluruh tanpa melibatkan eksploitasi yang berbahaya. Dalam pelaksanaan pengujian ini, peneliti tetap memperhatikan pertimbangan etika *(ethical considerations)*, termasuk tidak merusak sistem target, tidak mengambil atau menyebarkan data sensitif, serta hanya melakukan pengujian setelah memperoleh izin resmi dari pihak Universitas Ma'arif Nahdlatul Ulama Kebumen.berlangsung



Gambar 2 Alur Penelitian

Adapun penjelasan tahap-tahap sebagai berikut :

3.1. Pengumpulan Data

Tahap awal penelitian ini dilakukan dengan menentukan *website* yang akan diuji, yaitu umnu.ac.id. Setelah itu, dilakukan uji konektivitas dasar menggunakan perintah *ping* untuk memperoleh alamat IP dari *website* target serta memastikan bahwa server dapat diakses dan mengukur waktu respons dari server (Irfan MurtiRaazi 2023).

3.2. Identifikasi Kerentanan

tahap selanjutnya adalah mengidentifikasi potensi kerentanan pada *website* sistem informasi akademik UMNU kebumen menggunakan OWASP

ZAP. Proses ini dilakukan dengan melakukan pemindaian keamanan (*scanning*) terhadap aplikasi web untuk mendeteksi kelemahan yang dapat dieksploitasi. scanning tidak hanya membantu menemukan kerentanan yang ada, tetapi juga memberikan gambaran yang jelas tentang seberapa efektif mekanisme keamanan yang diterapkan (Yuzar and Rahmatulloh 2025).

3.3. Pengujian Kerentanan

Setelah melakukan identifikasi kerentanan menggunakan OWASP ZAP, tahap selanjutnya adalah menguji kelemahan yang memiliki tingkat risiko Medium ke atas. Pengujian ini bertujuan untuk memahami dampak dari kerentanan yang ditemukan serta memastikan apakah kelemahan tersebut dapat dieksploitasi lebih lanjut.

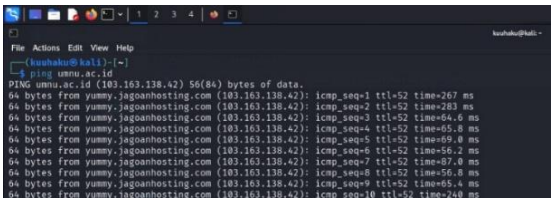
3.4. Pelaporan

Tahap akhir dari penelitian ini difokuskan pada penyusunan langkah-langkah mitigasi terhadap kerentanan yang ditemukan pada *website* Sistem Informasi Akademik UMNU Kebumen. Mitigasi dilakukan berdasarkan hasil analisis tingkat risiko dari temuan kerentanan yang telah diidentifikasi sebelumnya, dengan tujuan untuk meningkatkan ketahanan sistem terhadap potensi serangan siber

4. PEMBAHASAN

4.1 Pengumpulan Data

Menentukan alamat IP *website* sebagai langkah awal dalam proses identifikasi potensi kerentanan pada Sistem Informasi Akademik UMNU Kebumen. Data yang diperoleh dari proses ini akan digunakan untuk menganalisis keamanan *website*.Untuk mendapatkan alamat IP, dilakukan uji konektivitas dasar menggunakan perintah ping, yang memungkinkan pengecekan apakah server dapat dijangkau serta mengukur waktu respons dari server.



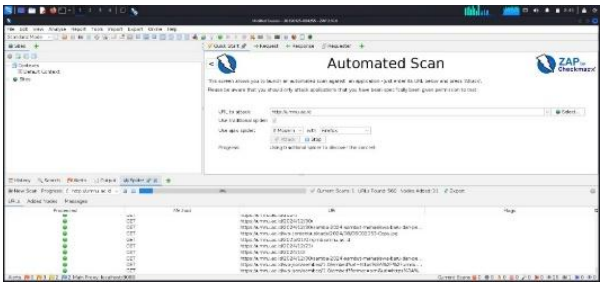
Gambar 3 Pengambilan Ping Web

4.2 Identifikasi Kerentanan

Setelah melakukan pengambilan data dan menentukan alamat IP *website* Sistem Informasi Akademik UMNU Kebumen, langkah selanjutnya adalah mengidentifikasi potensi kerentanan menggunakan *OWASP ZAP*. Proses ini bertujuan untuk menemukan celah keamanan yang dapat dieksploitasi oleh pihak yang tidak bertanggung jawab.

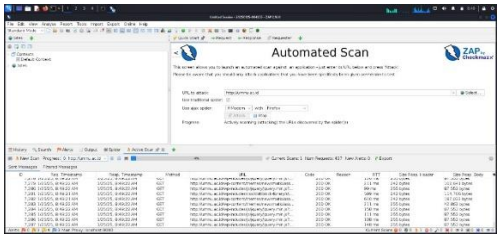
4.1.1. Proses Pemindaian Menggunakan OWASP ZAP

Identifikasi kerentanan dilakukan dengan menjalankan pemindaian menggunakan *OWASP ZAP*, yang berfungsi sebagai proxy untuk menganalisis lalu lintas HTTP antara klien dan server. Pemindaian ini mencakup *Spidering* yaitu Merayapi halaman-halaman *website* secara otomatis untuk mengidentifikasi semua endpoint yang tersedia.



Gambar 4 *Spidering*

Active Scan yaitu Melakukan uji penetrasi dengan mengirimkan berbagai payload untuk mendeteksi kelemahan berdasarkan OWASP Top 10.



Gambar 5 *Active Scan*

4.2.2. Hasil Identifikasi Kerentanan

Berdasarkan hasil pemindaian *OWASP ZAP*, ditemukan beberapa kerentanan dengan tingkat risiko Medium ke atas, antara lain

Tabel 1 Hasil Pemindaian

Alert type	Risk	Count
Content Security Policy (CSP) Header Not Set	Medium	934 (8,490.9%)
HTTP to HTTPS Insecure Transition in Form Post	Medium	2 (18.2%)
Missing Anti-clickjacking Header	Medium	545 (4,954.5%)
Cross-Domain JavaScript Source File Inclusion	Low	8 (72.7%)
Strict-Transport-Security Header Not Set	Low	2731 (24,827.3%)
Timestamp Disclosure - Unix	Low	28 (254.5%)

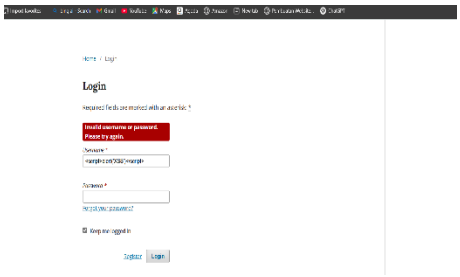
Alert type	Risk	Count
X-Content-Type-Options Header Missing	Low	1855 (16,863.6%)
Charset Mismatch	Informational	178 (1,618.2%)
Information Disclosure - Suspicious Comments	Informational	933 (8,481.8%)
Re-examine Cache-control Directives	Informational	1247 (11,336.4%)
User Agent Fuzzer	Informational	323 (2,936.4%)

4.2. Pengujian

Setelah melakukan identifikasi kerentanan menggunakan OWASP ZAP, tahap selanjutnya adalah menguji lebih dalam dua kerentanan utama yang ditemukan, yaitu (CSP) Header Not Set, HTTP to HTTPS Insecure Transition in Form Post, Missing Anti-clickjacking Header Pengujian dilakukan untuk mengetahui sejauh mana dampak dari masing-masing kerentanan serta bagaimana potensi eksploitasi terhadap keamanan *website* Sistem Informasi Akademik UMNU Kebumen.

4.2.1. Pengujian Kerentanan Content Security Policy (CSP) Header Not Set

Kerentanan *(CSP) Header Not Set* ditemukan pada sistem, yang berarti tidak ada kebijakan eksplisit yang mengatur sumber skrip yang diperbolehkan untuk dijalankan. CSP berfungsi sebagai mekanisme keamanan yang dapat mencegah berbagai serangan, termasuk *Cross-Site Scripting (XSS*), dengan membatasi eksekusi skrip hanya dari sumber yang dipercaya (Angga Putrawansyah PB and Tata Sutabri 2024). Ketiadaan CSP dapat meningkatkan risiko serangan berbasis skrip, terutama jika terdapat celah sanitasi input yang memungkinkan injeksi skrip berbahaya (Supriadi et al. 2024). Selain itu, ketidakhadiran pengaturan CSP pada sistem ini juga mencerminkan (OWASP– Security Misconfiguration 2021) dalam OWASP Top 10, di mana konfigurasi keamanan yang tidak tepat atau tidak lengkap dapat membuka celah bagi potensi eksploitasi. Dalam hal ini, tidak adanya kebijakan CSP yang jelas merupakan contoh dari kesalahan konfigurasi yang dapat dimanfaatkan oleh penyerang untuk menjalankan skrip berbahaya atau memanfaatkan celah keamanan lainnya dalam aplikasiUntuk menguji apakah absennya CSP dapat dieksploitasi dalam serangan XSS, dilakukan injeksi skrip pada beberapa area yang memungkinkan. Untuk menguji apakah absennya Content Security Policy (CSP) dapat dieksploitasi dalam serangan Cross-Site Scripting (XSS), dilakukan injeksi skrip pada beberapa area yang memungkinkan. Pengujian difokuskan pada kolom *Input* dengan cara menginputkan skrip berbahaya untuk melihat apakah hasil pencarian mencetak kembali input tanpa proses sanitasi. Selain itu, pengujian juga dilakukan melalui parameter URL dengan menyisipkan skrip di dalamnya untuk mengamati apakah skrip tersebut dieksekusi saat halaman dimuat. Beberapa payload yang digunakan dalam pengujian ini antara lain <script>alert('XSS')</script>, <img src="x" onerror="alert('XSS')">, "><script>alert('XSS')</script>, dan "><h1>TEST</h1>.



Gambar 6 payload dan respon

Payload tersebut mewakili teknik injeksi XSS yang umum digunakan untuk menguji tingkat kerentanan aplikasi terhadap serangan skrip berbahaya

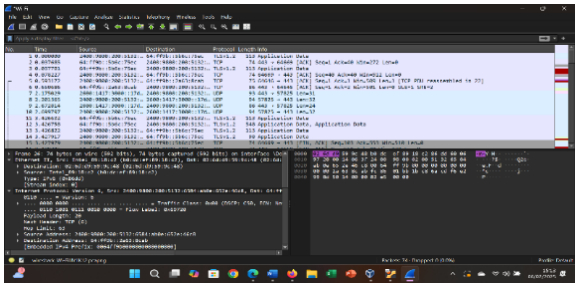
Analisis lebih mendalam menunjukkan bahwa sistem kemungkinan telah melakukan penyaringan terhadap karakter-karakter berbahaya, seperti tanda kurung sudut (<, >), serta menghindari pencetakan langsung input pengguna ke halaman tanpa proses encoding atau escaping. Ini merupakan indikasi bahwa validasi dan sanitasi input dilakukan secara server-side atau pada framework web yang digunakan.

Namun demikian, ketiadaan CSP tetap dianggap sebagai celah potensial, karena jika suatu saat mekanisme sanitasi gagal atau terjadi kesalahan konfigurasi pada halaman tertentu, serangan XSS tetap dapat terjadi. Oleh karena itu, penambahan kebijakan CSP sangat direkomendasikan sebagai *defense-in-depth*, untuk memastikan bahwa hanya skrip dari sumber yang terpercaya yang dapat dijalankan, menambah satu lapisan perlindungan lagi terhadap XSS dan serangan serupa.

4.2.2. Pengujian Kerentanan HTTP Ke HTTPS *Insecure Transition in Form Post*

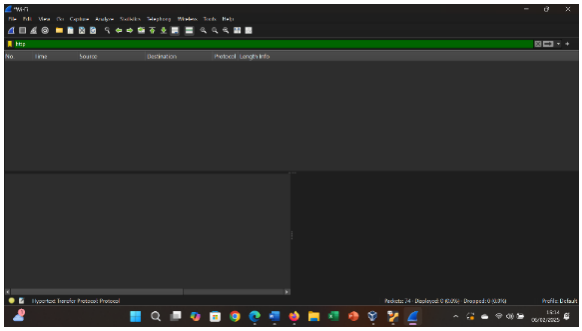
Kerentanan ini terjadi ketika suatu situs web memuat halaman melalui HTTP (tanpa enkripsi) dan kemudian mengirimkan data form (seperti login atau input pengguna) ke server melalui HTTPS**.** Hal ini berisiko karena pada tahap awal, koneksi HTTP dapat disadap menggunakan teknik *sniffing***,** memungkinkan penyerang melakukan *Man-in-the-Middle (MITM)* Attack untuk mencuri atau

memodifikasi data sebelum dikirimkan melalui HTTPS (Jurnal and Rahman 2024). Kerentanan ini dapat dikategorikan dalam (OWASP– Cryptographic Failures 2021) dalam OWASP Top 10, di mana pengiriman data sensitif melalui HTTP yang tidak terenkripsi membuka peluang bagi penyerang untuk melakukan serangan *Man-in-the-Middle* (MITM). Proses pengalihan yang tidak aman dari HTTP ke HTTPS dapat mengekspos data pengguna sebelum koneksi aman terbentuk, meningkatkan risiko penyadapan dan modifikasi data yang sedang dalam perjalananDalam pengujian ini, teknik *sniffing* digunakan untuk mengidentifikasi apakah ada transmisi HTTP sebelum terjadi perpindahan ke HTTPS (Hae and Sulistyo 2021). Pengujian dilakukan menggunakan *Wireshark* untuk menangkap dan menganalisis paket data yang dikirimkan oleh pengguna ketika mengakses halaman login atau form lainnya (Luthfansa and Rosiani 2021).



Gambar 7 Lalu Lintas Jaringan

Gambar di atas menunjukkan lalu lintas jaringan saat pengguna mengakses halaman login di *website* Sistem Informasi Akademik UMNU Kebumen. Karena banyaknya lalu lintas yang tercatat dalam Wireshark, kita dapat menggunakan filter HTTP untuk menyaring paket-paket yang terkait guna menentukan apakah kerentanan HTTP ke HTTPS Insecure Transition in Form Post terjadi.Namun setelah dilakukan pem- *filter*-an menggunakan HTTP tidak ditemukan adanya paket HTTP yang mengandung data form login.



Gambar 8 Lalu Lintas jaringan setelah difilter

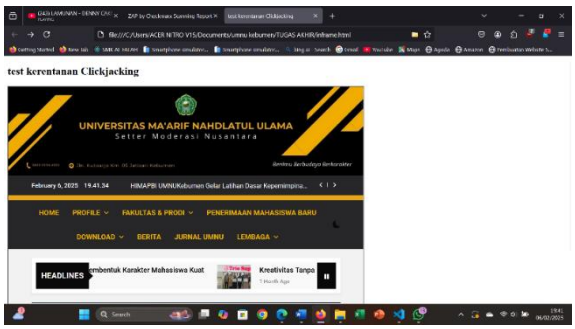
Semua permintaan yang terkait dengan halaman login langsung dikirim melalui HTTPS (port 443) tanpa melalui HTTP terlebih dahulu. Hasil ini menunjukkan bahwa *website* UMNU Kebumen telah menerapkan HTTPS secara penuh, sehingga serangan berbasis MITM melalui HTTP tidak dapat dilakukan. Dengan konfigurasi ini, potensi serangan MITM berbasis sniffing tidak dapat dilakukan, karena seluruh komunikasi dienkripsi sejak awal. Analisis mendalam menunjukkan bahwa konsistensi penggunaan protokol HTTPS pada seluruh proses, termasuk saat mengakses halaman login, mencerminkan penerapan praktik keamanan web yang baik. Meskipun hasil pengujian tidak menemukan transmisi data melalui HTTP, penerapan HSTS (HTTP Strict Transport Security) tetap direkomendasikan untuk menutup sepenuhnya celah keamanan terkait transisi protokol, seperti serangan SSL stripping. Dengan HSTS, browser akan dipaksa menggunakan HTTPS pada kunjungan berikutnya, sehingga perlindungan terhadap data pengguna menjadi lebih kuat dan berkelanjutan, terutama pada area sensitif seperti autentikasi.

4.3.3. Pengujian kerentanan *Missing Anti-Clickjacking Header*

Clickjacking adalah teknik di mana penyerang memasukkan halaman web dalam bingkai transparan di situs web palsu (Anugrah Utama and Supardi 2024). Dengan teknik ini, pengguna dapat secara tidak sadar melakukan tindakan berbahaya seperti mengklik tombol penting, mengubah pengaturan akun, atau bahkan memberikan izin akses tanpa menyadarinya. Kerentanan *Clickjacking* ini juga berhubungan dengan *Security Misconfiguration* dalam OWASP Top 10, di mana penggunaan elemen-elemen web atau komponen yang rentan terhadap penyusupan dapat dimanfaatkan oleh penyerang untuk menyembunyikan elemen-elemen berbahaya pada halaman yang tampak aman. Teknik ini memanfaatkan kelemahan dalam pengaturan antarmuka pengguna dan dapat dieksploitasi untuk menipu pengguna dalam melakukan tindakan yang tidak diinginkan, seperti memberikan izin akses atau memanipulasi pengaturan tanpa sepengetahuan mereka. Pengujian dilakukan dengan mencoba memuat halaman *website* dalam *iframe* menggunakan kode berikut:

```
<!DOCTYPE html>
<html>
<head>
      <title>test kerentanan Clickjacking
</title>
</head>
<body>
    <h2>test kerentanan Clickjacking</h2>
        <iframe src="https://umnu.ac.id/"
width="800" height="600"></iframe>
</body>
</html>
```

Setelah dijalankan, halaman berhasil dimuat dalam iframe, yang menunjukkan bahwa *website* tidak memiliki perlindungan terhadap Clickjacking.



Gambar 9 Pengujian Clickjacking

Selain itu, analisis terhadap HTTP Response Headers menunjukkan bahwa tidak terdapat header keamanan seperti X-Frame-Options atau Content-Security-Policy (frame-ancestors), yang seharusnya digunakan untuk mencegah serangan ini

4.3. Pelaporan

Bagian ini memaparkan langkah mitigasi terhadap kerentanan yang ditemukan berdasarkan hasil pengujian, guna meningkatkan keamanan sistem informasi akademik UMNU Kebumen. Terkait temuan *Content Security Policy* (CSP) *Header Not Set*, meskipun tidak ditemukan eksploitasi XSS, sistem tetap berisiko terhadap serangan jika sanitasi input gagal. Pada penlitian yang dilakukan (Akhmad Dzihan Kamaly 2022) penambahan *header* CSP dapat mencegah XSS. Oleh karena itu, disarankan untuk menambahkan *header* CSP seperti *Content-Security-Policy*: default-src 'self'; script-src 'self'; object-src 'none'; frame-ancestors 'none'; base-uri 'self';, serta menggunakan report-uri untuk pelaporan pelanggaran, dan melakukan audit input secara berkala.

Pada temuan HTTP ke HTTPS Insecure Transition in Form Post, seluruh form login telah menggunakan HTTPS, namun tetap direkomendasikan untuk mengaktifkan "*STS (Strict-Transport-Security: max-age=31536000; includeSubDomains; preload)"* seperti yang dilakukan pada penelitian yang dilakukan oleh (Basyirah et al. 2023) berhasil menurunkan tingkat resiko kerentanan pada HSTS yang tidak ada. Terakhir, pada temuan *Missing Anti-Clickjacking Header*, *website* dapat dimuat dalam iframe eksternal, sehingga perlu ditambahkan header *X-Frame-Options: DENY* atau *SAMEORIGIN*, dalam penelitian yang dilakukan (Setya Putra and Santoso 2025) direkomendasikan penambahan header tersebut sebagai salah satu langkah mitigasi yang dapat diterapkan untuk meningkatkan keamanan sistem. serta penggunaan CSP dengan directive frame-ancestors 'none'; untuk mencegah serangan *clickjacking*, disertai audit penggunaan iframe.

Tabel 2 Hasil Penelitian

Jenis Kerentanan	Teknik/Tool Pengujian	Rekomendasi mitigasi
CSP Not Set	Menyisipkan payload seperti: `<script>alert('XSS')</script>` `<img src="x" onerror="alert('XSS')">` `"><h1>TEST</h1>` pada input form dan URL.	Terapkan header HTTP CSP untuk membatasi sumber daya yang dapat dijalankan browser.
Insecure Transition (HTTP sebelum HTTPS)	Menggunakan Wireshark untuk memantau lalu lintas jaringan;	Terapkan HTTP *Strict Transport Security* (HSTS) dengan *header Strict-Transport-Security*.
clickjacking	Membuat halaman HTML eksternal dengan iframe yang memuat halaman target: <iframe src="https://umnu.ac.id">	Tambahkan header *X-Frame-Options: DENY* atau *SAMEORIGIN*, atau gunakan CSP dengan *frame-ancestors 'none'* untuk mencegah pembajakan frame.

4.4. Research Limitation (Keterbatasan)

Penelitian ini memiliki keterbatasan terkait dengan akses terhadap kode sumber web yang diuji. Karena tidak memiliki hak akses untuk melakukan perubahan langsung pada kode atau konfigurasi sistem, penelitian ini hanya dapat memberikan rekomendasi mitigasi berdasarkan temuan kerentanannya. Hal ini menyebabkan keterbatasan dalam mengimplementasikan langkah mitigasi secara langsung dan menguji dampak perbandingan antara kondisi sebelum dan setelah mitigasi. Oleh karena itu, analisis yang dilakukan terbatas pada rekomendasi teoritis tanpa pengujian implementasi mitigasi pada sistem yang diuji.

5. KESIMPULAN DAN SARAN

Penelitian ini dilakukan berdasarkan insiden peretasan yang pernah terjadi pada sistem informasi akademik UMNU Kebumen, di mana tampilan situs sempat berubah menjadi iklan judi online. Meskipun tampilan telah diperbaiki, penelitian ini bertujuan untuk memastikan apakah masih terdapat kerentanan lain yang berpotensi disalahgunakan serta memberikan rekomendasi mitigasi. Berdasarkan pengujian menggunakan OWASP ZAP dan pendekatan OWASP WSTG, ditemukan beberapa temuan kerentanan yaitu absennya header Content Security Policy (CSP), potensi risiko dari transisi HTTP ke HTTPS tanpa HSTS, serta tidak adanya proteksi terhadap serangan clickjacking. Meskipun tidak ditemukan eksploitasi aktif pada celah-celah

tersebut, kerentanan tersebut tetap memerlukan mitigasi sebagai langkah *preventif* dalam meningkatkan keamanan aplikasi web. Rekomendasi seperti penambahan header keamanan (CSP, HSTS, dan X-Frame-Options) dan penguatan sanitasi input dapat membantu mencegah serangan siber di masa mendatang serta memperkuat keandalan sistem informasi akademik. disarankan agar penelitian dilanjutkan dengan akses terhadap kode sumber aplikasi, sehingga dapat dilakukan mitigasi langsung pada level kode serta evaluasi efektivitas perbandingan sistem sebelum dan sesudah mitigasi diterapkan.

DAFTAR PUSTAKA

Akhmad Dzihan Kamaly, Umar Yunan Kurnia Septo Hediyanto ,Adityas Widjajarto. 2022. "Analisis Security Mitigation Terhadap *Website* Akademik Penunjan Administrasi Di Institusi XYZ Mengunakan Metode Penetration Testing Execution Standard (PTES)."

Akhson, Sidik Maulana, and Fahmi Fachri. 2025. "Analisis Keamanan *Website* SMK Wongsorejo Gombong Terhadap Serangan SQL Injection Dengan PTES."

Alamsyah, Hendri. 2021. "Penerapan Sistem Keamanan WEB Menggunakan Metode WEB Aplication Firewall." *Jurnal Amplifier Mei* 11.

Alwi Putra Supendi. 2024. *ANALISA KERENTANAN APLIKASI WEB MENGGUNAKAN FRAMEWORK MITRE ATT&CK DENGAN METODE SIMULASI RED TEAM: STUDI KASUS DI PT. NURUL FIKRI CIPTA INOVASI*.

Angga Putrawansyah PB, and Tata Sutabri. 2024. "Analisis Keamanan Aplikasi Rekam Medis Elektronik Mengunakan Metode Penetration Testing Pada UPTD RSD Besemah." *Router : Jurnal Teknik Informatika dan Terapan* 2(4): 01–12. doi:10.62951/router.v2i4.268.

Anugrah Utama, Delta, and Reno Supardi. 2024. "Analisis Keamanan *Website* Menggunakan PTES (Penetration Testing Execution And Standart)." *Jurnal Media Infotama* 20(0736): 341139. http://info.cern.ch.

Basyirah, Aulia, Umar Yunan, Kurnia Septo Hediyanto, and Muhammad Fathinuddin. 2023. 7 Jurnal Sains Komputer & Informatika (J-SAKTI *Optimisasi Strategi Security Mitigation Dengan Vapt Pada Website Absensi Praktikan Dan Asisten Laboratorium Praktek*.

Ela Nurelasari, and Difa Gumilang Al Farabi. 2024. 8 Jurnal Mahasiswa Teknik Informatika *ANALISIS KEAMANAN SISTEM WEBSITE MENGGUNAKAN METODE OPEN WEB APPLICATION SECURITY PROJECT (OWASP) PADA SIMANTEP.ID*.

Fachri, Fahmi, Abdul Fadlil, Imam Riadi, Ahmad Dahlan, Yogyakarta Jln Soepomo, and Informasi Artikel. 2021. "Analisis Keamanan Webserver Menggunakan Penetration Test." *JURNAL INFORMATIKA* 8(2). http://ejournal.bsi.ac.id/ejurnal/index.php/ji.

Hae, Yacob, and Wiwin Sulistyo. 2021. 8 *Analisis Keamanan Jaringan Pada Web Dari Serangan Sniffing Dengan Metode Eksperimen*. http://jurnal.mdp.ac.id.

Handaya, Syahril, and Raihan Islamadina. 2025. *IMPLEMENTASI PENETRATION TESTING PADA APLIKASI WEB SISTEM EVALUASI DATA BIDANG TIK POLDA ACEH MENGGUNAKAN METODE OWASP DAN NIST SP 800-115 1*.

Irfan MurtiRaazi. 2023. *ANALISIS PENILAIAN KEAMANAN SERVER TERHADAP SISTEM INFORMASI MANAJEMEN KEPEGAWAIAN DENGAN METODE NIST SP 800-115 PADA UNIVERSITAS ISLAM NEGERI AR-RANIRY TUGAS AKHIR*.

Jurnal, Halaman, and Rakhmadi Rahman. 2024. "JURNAL RISET TEKNIK KOMPUTER IMPLEMENTASI SSL (SECURE SOCKET LAYER) UNTUK MELINDUNGI TRANSAKSI ONLINE." *JURTIKOM* 1(3). doi:10.69714/563m6d13.

Luthfansa, Zaky Maula, and Ulla Delfana Rosiani. 2021. *Pemanfaatan Wireshark Untuk Sniffing Komunikasi Data Berprotokol HTTP Pada Jaringan Internet*.

Muhammad Fauzi, Rifki, Rudi Hermawan, Dewanto Rosian Adhy, and Siti Maesaroh. 2024. "Analisis Kerentanan Keamanan Web Menggunakan Metode OWASP Dan PTES Di Web Pemerintahan Desa XYZ." *Jurnal Orang Elektro* 13(2). https://XYZ.g-desa.id/,.

OWASP– Cryptographic Failures. 2021. "Cryptographic Failures ."

OWASP– Security Misconfiguration. 2021. "A05:2021 – Security Misconfiguration ."

R.A. Dwi Ayu Puspitasari. 2021. "ANALISA SISTEM INFORMASI AKADEMIK (SISFO) DAN JARINGAN DI UNIVERSITAS BINA DARMA."

Setya Putra, Bagus, and Dwi Budi Santoso. 2025. *Analisis Keamanan Website Berbasis WordPress Melalui Penetration Testing Untuk Meningkatkan Keamanan Digital*.

Supriadi, Dedi, Emi Suryadi, Rudi Muslim, Lalu Delsi Samsumar, and Universitas Teknologi Mataram. 2024. 1 Journal of Data Analytics, Information, and Computer Science (JDAICS) *IMPLEMENTASI VULNERABILITY ASSESSMENT OWASP (OPEN WEB APPLICATION SECURITY PROJECT) PADA WEBSITE UNIVERSITAS TEKNOLOGI MATARAM*.

Sya'bani, Fadila Ahmad, and Fayruz Rahma. 2022. *Hardening Sistem Informasi XYZ Menggunakan Framework OWASP*.

Yuzar, Arnefia, and Alam Rahmatulloh. 2025. 9 Jurnal Mahasiswa Teknik Informatika) *PERBANDINGAN EFEKTIVITAS OWASP ZAP, ACUNETIX, NIKTO MENGGUNAKAN VULNERABILITY SCANNING UNTUK DETEKSI KERENTANAN APLIKASI WEB*. https://opendata.tasikmalayakab.go.id,.