Journal of Informatics and Computer Science Vol. 11 No. 2 Oktober 2025
Universitas Ubudiyah Indonesia
e-ISSN : 2615-5346

ANALISIS PENGUJIAN KEAMANAN APLIKASI WEB UNIVERSITAS UBUDIYAH INDONESIA MENGGUNAKAN *OPEN WORLD WIDE APPLICATION SECURITY PROJECT ZED ATTACK PROXY* (OWASP ZAP)

ANALYSIS OF WEB APPLICATION SECURITY TESTING AT UNIVERSITAS UBUDIYAH INDONESIA USING OPEN WORLD WIDE APPLICATION SECURITY PROJECT ZED ATTACK PROXY (OWASP ZAP)

Shavinatun Husna Dewi[1], Rizka Albar[2], Desita Ria Yusian TB[3], M Bayu Wibawa[4]
[1,2,]Program Studi Sistem Informasi, [3,4,]Program Studi Informatika Fakultas Sains dan Teknologi, Universitas Ubudiyah Indonesia
Jalan Alue Naga. Desa Tibang Kec. Syiah Kuala. Banda Aceh-Indonesia
E-mail: [1]shavinatunh@gmail.com, [2]albar@uui.ac.id, [3]desita@uui.ac.id, mbayuw@uui.ac.id

***Abstrak*—** Pada penelitian ini dilakukan pengujian keamanan aplikasi web Universitas Ubudiyah Indonesia untuk menguji tingkat keamanan pada web dengan menggunakan *Open World Wide Application Security Project Zed Attack Proxy* (OWASP ZAP), OWASP ZAP merupakan aplikasi *open-source* yang berfungsi untuk melakukan identifikasi serta eksploitasi kerentanan terhadap suatu *website*, memiliki desain *user-friendly* dapat memudahkan pengguna dalam melakukan pengujian yang dapat mengamankan *website* mereka. Metode pengujian yang digunakan yaitu *Penetration Testing* dengan model *White Box Testing* yang meliputi pemindaian pasif untuk memetakan potensi celah keamanan tanpa mempengaruhi kinerja sistem, serta pemindaian aktif untuk mengidentifikasi dan menguji kerentanan seperti *SQL Injection, Cross-Site Scripting (XSS),* serta *Broken Authentication*. Hasil pengujian menunjukkan kerentanan dan risiko yang bervariasi, hasil analisis pada penelitian ini diharapkan dapat menjadi langkah awal untuk meningkatkan keamanan aplikasi web yang lebih kuat, konfigurasi keamanan yang lebih baik dan pembaruan perangkat lunak secara rutin terhadap sistem.

Kata kunci: ***Analisis Uji Keamanan, Website, Universitas Ubudiyah Indonesia, OWASP ZAP, Metode Penetrasi Testing.***

Abstract*—This research involves testing the security of the Universitas Ubudiyah Indonesia's web application to assess its security level using Open Web Application Security Project Zed Attack Proxy (OWASP ZAP). OWASP ZAP is an open-source application designed to identify and exploit vulnerabilities in websites. With a user-friendly design, it facilitates users in conducting tests to secure their websites.The testing method employed is Penetration Testing using the White Box Testing model, which includes passive scanning to map potential security gaps without affecting system performance, as well as active scanning to identify and test vulnerabilities such as SQL Injection, Cross-Site Scripting (XSS), and Broken Authentication. The results of the testing indicate varying vulnerabilities and risks. The analysis from this research is expected to serve as an initial step toward enhancing the web application's security, improving security configurations, and ensuring regular software updates for the system.*

Keywords: ***Security Testing Analysis, Website, Universitas Ubudiyah Indonesia, OWASP ZAP, Penetration Testing Method.***

I. PENDAHULUAN

Perkembangan teknologi informasi yang semakin pesat baik di sektor perekonomian, kesehatan, industri maupun di sektor pendidikan mendorong perlunya sistem aplikasi yang aman untuk melindungi data sensitif pengguna. Seiring dengan berjalannya waktu, ancaman keamanan siber menjadi salah satu tantangan utama yang harus dihadapi. Universitas Ubudiyah Indonesia merupakan salah satu perguruan tinggi yang mengandalkan aplikasi web untuk mendukung operasional dan layanan akademiknya. Dalam hal ini memerlukan kepastian bahwa sistem *website* aman dari berbagai potensi serangan. Keamanan aplikasi web menjadi hal yang sangat penting untuk melindungi data senstif, menjaga integritas sistem dan mencegah penyalahgunaan yang dapat merugikan institusi dan penggunanya.

Universitas Ubudiyah Indonesia memerlukan upaya yang dapat mencegah hal yang merugikan tersebut. Salah satu cara yang dapat dilakukan adalah dengan memanfaatkan aplikasi *Open Web Application Security Project Zed Attack Proxy* (OWASP ZAP) yaitu aplikasi *open-source* yang sering digunakan untuk menguji keamanan pada suatu aplikasi web dan merupakan aplikasi yang sangat baik dalam menyelesaikan masalah keamanan *website*. Aplikasi ini dirancang untuk mendeteksi kerentanan keamanan pada aplikasi web melalui berbagai pengujian, mulai dari pengujian deteksi celah keamanan

umum, pengujian serangan injeksi, analisis konfigurasi keamanan dan berbagai pengujian lainnya.

Dalam rangka meningkatkan kemanan data pada *website* Universitas Ubudiyah Indonesia menghadapi beberapa tantangan, diantaranya:

1. Bagaimana aplikasi *Open Web Application Security Project Zed Attack Proxy* (OWASP ZAP) dapat meningkatkan keamanan *website* Universitas Ubudiyah Indonesia?
2. Bagaimana pengaruh pengujian keamanan pada aplikasi *Open Web Application Security Project Zed Attack Proxy* (OWASP ZAP) terhadap *website* Universitas Ubudiyah Indonesia?
3. Apa strategi paling tepat dalam melakukan pembaharuan yang dapat meningkatkan keamanan pada *website* Universitas Ubudiyah Indonesia?

II. STUDI PUSTAKA

A. Universitas Ubudiyah Indonesia

Universitas Ubudiyah Indonesia merupakaan Lembaga pendidikan tinggi swasta yang terletak di kota Banda Aceh, Provinsi Aceh. Universitas Ubudiyah Indonesia (UUI) memperoleh status Universitas pada tanggal 29 April 2014 berdasarkan Ketetapan Menteri Pendidikan No.45/E/O/2014. Universitas Ubudiyah Indonesia saat ini mengelola 3 fakultas, terdiri dari Fakultas Sains dan Teknologi, Ilmu kesehatan dan Fakultas Sosial Sains dan Ilmu Pendidikan. Memiliki visi menjadi *World Class University* akan membuat mahasiswa difasilitasi dengan berbagai kemudahan berbasis teknologi informasi dan komunikasi. Universitas Ubudiyah Indonesia juga salah satu Universitas Swasta terbaik di Aceh saat ini.

B. Uji Keamanan

Pengujian merupakan suatu proses sistematis untuk mengevaluasi kualitas suatu perangkat lunak dengan mengidentifikasi suatu kesalahan atau kekurangan dalam fungsionalitasnya (Helmi et al., 2023).

Menurut Fachri et al. (2021) Keamanan adalah faktor yang sangat penting yang harus diperhatikan dalam membangun sebuah *web-server*. Pengujian keamanan sebuah *web-server* didefinisikan sebagai upaya legal dan resmi untuk mengeksploitasi sistem komputer dengan tujuan meningkatkan keamanan pada sistem.

C. Website

Menurut Putri dan Santoso (2021) *Website* merupakan sekumpulan halaman yang saling terhubung dan diakses melalui internet yang berfungsi sebagai media penyampaian informasi, layanan atau komunikasi. *Website* memiliki berbagai bentuk seperti *website* pribadi, institusi, komersial, non komersial maupun organisasi.

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

Open Web Application Security Project Zed Attack Proxy (OWASP ZAP) adalah organisasi non profit yang berfokus pada peningkatan keamanan perangkat lunak, menyediakan *framework* dan *tools open-source* seperti *Web Security Testing Guide* (WSTG) untuk pengujian keamanan sistem berbasis web (Kuncoro dan Fayruz, 2022).

Application Security Project Zed Attack Proxy (OWASP ZAP) adalah alat sumber terbuka yang digunakan untuk menguji keamanan aplikasi web. Dalam pengujian ini, terdapat beberapa indikator yang sering digunakan untuk menilai kerentanan dan risiko keamanan:

1. Identifikasi Kerentanan *(vulnerability Identification)*

Pada proses ini melibatkan pemindaian aplikasi web untuk mendeteksi potensi celah keamanan, seperti *SQL Injection*, *Cross-Site Scripting (XSS)* dan konfigurasi yang tidak aman. OWASP ZAP dapat menghasilkan laporan yang merinci kerentanan yang ditemukan dan juga dengan deskripsi dan tingkat kerentanannya.

2. Penilaian Resiko *(Risk Assessment)*

Setelah identifikasi kerentanan, langkah selanjutnya adalah menilai risiko berdasarkan dua faktor utama :

- *Likelihood* (Kemungkinan) : Seberapa besar kemungkinan yang terjadi pada kerentanan tersebut akan dieksploitasi oleh pihak yang tidak bertanggung jawab.
- *Impact* (Dampak) : Seberapa besar dampak yang akan ditimbulkan pada *website* jika kerentanan tersebut dieksploitasi.

Penilaian ini selanjutnya akan membantu pengguna menentukan prioritas penanganan kerentanan yang ditemukan.

3. Klasifikasi Kerentanan

Kerentanan yang ditemukan sering diklasifikasikan menggunakan standar seperti :

- *Common Weakness Enumeration (CWE):* yaitu daftar yang berisi berbagai jenis kelemahan dalam desain, kode atau arsitektur aplikasi.
- *Web Application Security Consortium (WASC)* merupakan modul yang berisi informasi teknis, pedoman keamanan dan dokumentasi strategi penyerangan berdasarkan kategori celah keamanan.

E. Penetration Testing

Penetration Testing atau pengujian penetrasi merupakan serangan siber yang disimulasikan secara sah pada sistem

e-ISSN : 2615-5346

komputer untuk mengidentifikasi kerentanan. Proses pengujian mencakup 4 fase utama yaitu : Rekonsiliasi, Pemindaian *(Scanning)*, mendapatkan akses dan mempertahankan akses (Shebli and Behesti, 2018).

F. White-Box Model

Pengujian *White-Box* merupakan pengujian yang difokuskan pada internal sistem yaitu *Source Code Program*, jadi diperlukan akses agar dapat dilakukan pengujian menggunakan model *White-Box*.(Helmi et al., 2023).

White-Box Model merupakan metode pengujian perangkat lunak yang menganalisis struktur internal dan kode perangkat lunak. Dalam pengujian dengan *White-Box* Model penguji harus memastikan perangkat lunak sudah memenuhi target fungsi.

III. METODE

A. Jenis Penelitian

Penelitian ini menggunakan Metode Penetrasi Testing yang mencakup :

1. Rekonsiliasi *(Reconnaissance)*:Pengumpulan informasi tentang *website* target untuk melakukan pengujian.
2. Pemindaian *(Scanning)* yaitu Penggunaan alat teknis untuk menganalisis sistem dan mencari potensi titik masuk. Pada penelitian ini menggunakan aplikasi *Open Web Application Security Project Zed Attack Proxy* (OWASP ZAP) dalam melakukan serangan terhadap target.
3. Mendapatkan Akses *(Gaining Access):* Mengeksploitasi kerentanan yang akan teridentifikasi untuk menyusup ke dalam sistem.
4. Mempertahankan Akses (*Maintaining Access)* untuk membangun keberadaan yang berkelanjutan dan melanjutkan ekstraksi data atau eksploitasi lebih lanjut.

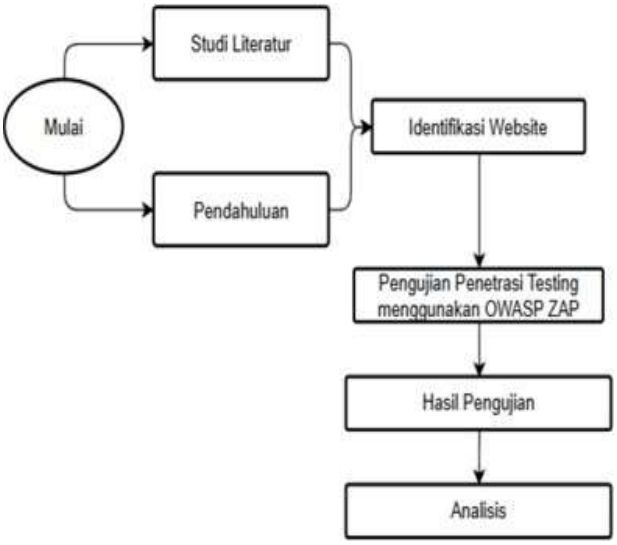
B. Identifikasi *Website*

Website yang digunakan adalah *Website* Universitas Ubudiyah Indonesia : https://uui.ac.id/

Gambar 3.1 *Website* Pengujian Penetrasi Testing.

Penelitian dilakukan dengan melakukan pengujian terhadap *web-server* sebagai tempat berjalannya aplikasi.

C. Rancangan Penelitian



Gambar 3.2 Tahapan Penelitian.

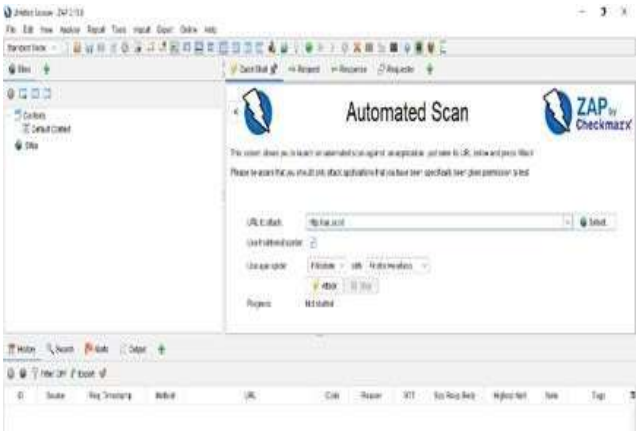
Tahapan pada penelitian ini terbagi menjadi beberapa tahap, yaitu:

1) **Mulai** : Tahap ini merupakan langkah pertama dimulai dari titik awal proses pengujian.
2) **Studi Literatur** : Tahap ini dilakukan untuk melakukan pengumpulan informasi dari berbagai sumber seperti buku, jurnal dan juga artikel terkait pemahaman konsep dan metode pengujian penetrasi (Penetrasi *Testing*).
3) **Pendahuluan** : Tahap ini bertujuan untuk menjelaskan ruang lingkup penelitian. Hal ini mencakup informasi tentang apa yang akan diuji, mengapa pengujian dilakukan dan hasil yang diharapkan.
4) **Identifikasi Website** : Pada langkah ini, *Website* akan diuji, informasi seperti URL dan potensi risiko keamanan akan dianalisis.
5) **Pengujian Penetrasi Testing menggunakan OWASP ZAP** : Pada tahapan ini akan dilakukan pengujian menggunakan aplikasi OWASP ZAP. OWASP ZAP yang digunakan merupakan versi 2.15.0.

IV. HASIL DAN PEMBAHASAN

A. Pengujian Penetrasi Testing menggunakan OWASP ZAP

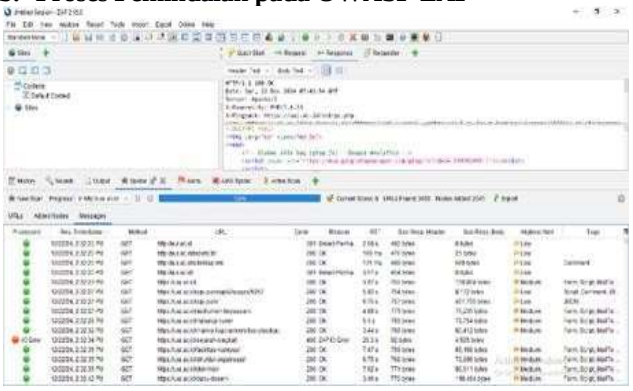
a. Tampilan awal OWASP ZAP



Gambar 4.1 Tampilan Awal pada OWASP ZAP.

Pada Gambar 4.1 merupakan tampilan awal dari aplikasi OWASP ZAP. Tahapan selanjutnya yang dilakukan yaitu pengguna dapat menginput link web pada bagian *"URL to Attack"* untuk melakukan pemindaian *(Scanning*) yang berfungsi untuk mengetahui tingkat keamanan pada *website*.

b. Proses Pemindaian pada OWASP ZAP



Gambar 4.2 Hasil Pemindaian *Website* UUI pada OWASP ZAP

Pada tahapan ini, aplikasi OWASP ZAP akan mencari kerentanan yang ada pada *website* Universitas Ubudiyah Indonesia. Pada Gambar 4.2 dapat dilihat pada *website* Universitas Ubudiyah Indonesia (UUI) memiliki tingkat kerentanan pada level "*Low* (Rendah), *High* (Tinggi), *Medium* (Sedang) dan *Informational* (Informasional)". Kerentanan yang ditemukan OWASP ZAP ini yang kemudian dicurigai dapat mengancam keamanan pada *website* UUI.

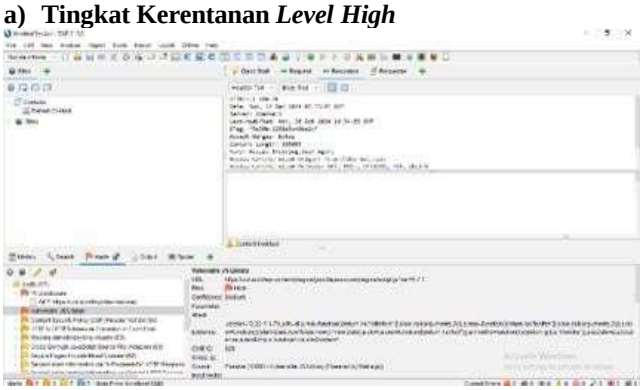
B. Penilaian Resiko (*Risk Assessment*) pada *Website* Universitas Ubudiyah Indonesia

		Confidence				
		User Confirmed	High	Medium	Low	Total
Risk	High	0 (0.0%)	0 (0.0%)	2 (11.1%)	0 (0.0%)	2 (11.1%)
	Medium	0 (0.0%)	1 (5.6%)	2 (11.1%)	0 (0.0%)	3 (16.7%)
	Low	0 (0.0%)	2 (11.1%)	4 (22.2%)	1 (5.6%)	7 (38.9%)
	Informational	0 (0.0%)	0 (0.0%)	3 (16.7%)	3 (16.7%)	6 (33.3%)
	Total	0 (0.0%)	3 (16.7%)	11 (61.1%)	4 (22.2%)	18 (100%)

Gambar 4.3 *Risk Assessment* pada *Website* UUI

Berdasarkan Gambar 4.3 diperoleh data Penilaian Resiko (*Risk Assessment*) pada Tingkat *High* (2) adalah (11.1%), *Medium* (3) adalah (16.7%), *Low* (7) adalah (38.9%) dan *informational* (6) dengan(33.3%).

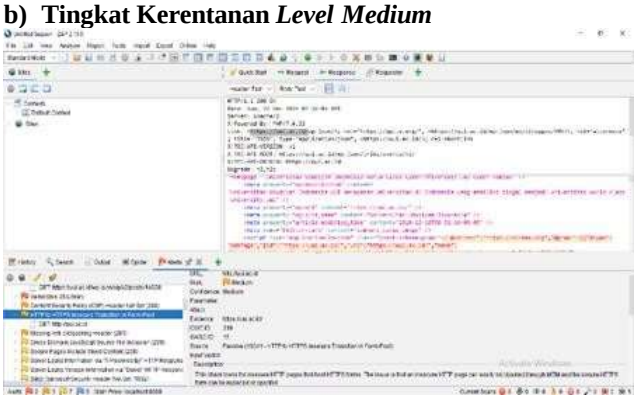
a) Tingkat Kerentanan *Level High*



Gambar 4.4 Tingkat Kerentanan *Level High* pada OWASP ZAP

Pada gambar 4.4 dapat disimpulkan bahwa OWASP ZAP mendeteksi bahwa pustaka *JavaScript* yang digunakan (versi 6.7.1 dari *goodlayers-core*) memiliki kerentanan yang diketahui berdasarkan *Database Retire.js*. Kerentanan ini dapat dieksploitasi oleh penyerang untuk melakukan serangan tertentu seperti *Cross-Site Scripting* (XSS) penyerang dapat menyuntikkan skrip berbahaya ke dalam aplikasi.

b) Tingkat Kerentanan *Level Medium*

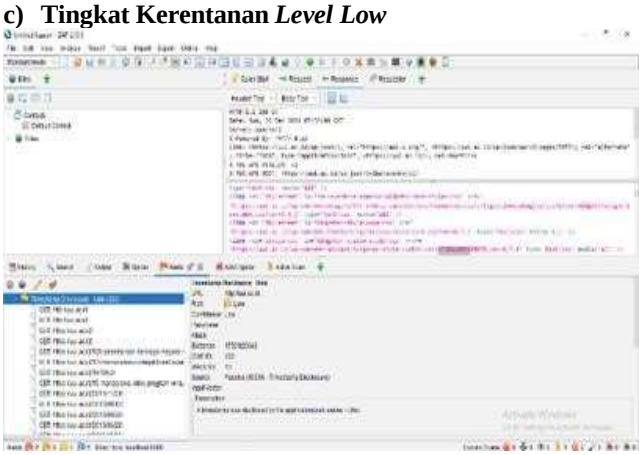


Gambar 4.5 Tingkat Kerentanan *Level Medium* pada OWASP ZAP

e-ISSN : 2615-5346

Pada Gambar 4.5 dapat disimpulkan bahwa OWASP ZAP mendeteksi bahwa terdapat formulir pada halaman dengan protokol HTTP (tidak aman) yang mengarahkan data ke halaman HTTPS (aman). Namun, formulir tersebut masih memulai transmisi melalui HTTP terlebih dahulu. Permasalahan ini membuka peluang bagi penyerang untuk menyadap atau memodifikasi data (seperti informasi *login*). Temuan ini berada pada tingkat *Medium*.

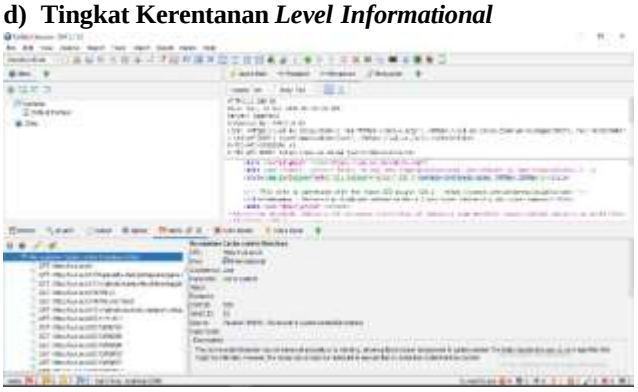
c) Tingkat Kerentanan *Level Low*



Gambar 4.6 Tingkat Kerentanan *Level Low* pada OWASP ZAP

Pada Gambar 4.6 dapat disimpulkan bahwa OWASP ZAP mendeteksi bahwa *Timestamp (unix timestamp)* ditemukan pada elemen situs web, seperti dalam *URL File stylesheet* atau *resource* lain. Hal ini dapat menyebabkan penyerang dapat menggunakan informasi untuk menyusun strategi serangan berdasarkan aset yang baru diunggah, dimodifikasi atau dijadwalkan. Temuan ini termasuk kategori rendah tapi tetap penting untuk diperhatikan, terutama dalam konteks pengamanan informasi, dengan menghilangkan *Timestamp* potensi eksploitasi dapat diminimalkan.

d) Tingkat Kerentanan *Level Informational*

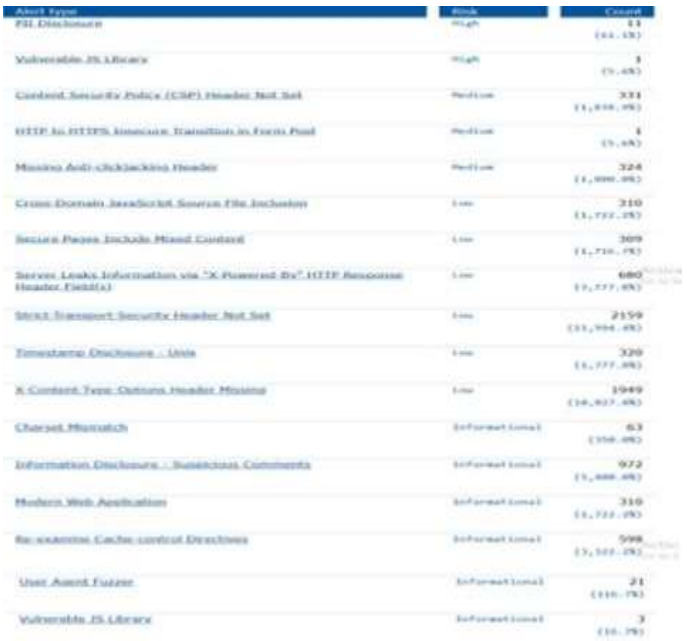


Gambar 4.7 Tingkat Kerentanan *Level Low* pada OWASP ZAP

Gambar 4.7 ini menunjukkan kerentanan konfigurasi pada *server* HTTP situs **uui.ac.id**, yang terkait dengan pengaturan *Header Cache-Control*. Meskipun risikonya rendah, ini tetap penting untuk diperbaiki, terutama jika ada kemungkinan data sensitif diungkapkan.

C. Klasifikasi Kerentanan pada *Website* Universitas Ubudiyah Indonesia

Berdasarkan pengujian pada *Open World Wide Application Security Project Zed Attack Proxy* (OWASP ZAP) terdapat (18) kerentanan secara keseluruhannya sebagai berikut :



Gambar 4.8 Kerentanan secara Keseluruhan pada *Website* UUI

Pada Gambar 4.8 terdapat (18) kerentanan pada tingkat yang bervariasi. Berdasarkan hasil tersebut dapat diperoleh (12) *Common Weakness Enumeration (CWE)* dan *Web Application Security Consortium (WASC)* sebagai berikut:

Tabel 4.1 *CWE dan WASC* pada *Website* UUI

Kerentanan	***Risk Level***	**CWE**	**WASC**
PII Disclosure	*Critical*	CWE-200	WASC-13
Vulnerable JS Library	*High*	CWE-829	WASC-10
Content Security Policy (CSP) Header Not Set	*Medium*	CWE-693	WASC-15
HTTP to HTTPS Insecure Transition in Form Post	*Medium*	CWE-319	WASC-15
Missing Anti-Clickjacking Header	*Medium*	CWE-1021	WASC-15

Cross-Domain JavaScript Source File Inclusion	*Medium*	CWE-829	WASC-15
Secure Pages Include Mixed Content	*Medium*	CWE-614	WASC-15
Server Leaks Information via HTTP Response Headers	*Low*	CWE-200	WASC-13
Strict Transport-Security Header Not Set	*Medium*	CWE-693	WASC-4
Timestamp Disclosure - Unix	*Low*	CWE-200	WASC-13
X-Content-Type-Options Header Missing	*Low*	CWE-693	WASC-15
Re-examine Cache-Control Directives	*Informati o-nal*	CWE-525	WASC-13

1. **CWE-200: *Information Exposure***
   WASC-13: *Information Leakage*

Deskripsi: Data sensitif atau informasi tentang sistem diungkapkan secara tidak sengaja, baik melalui tanggapan HTTP, pesan kesalahan, atau perilaku sistem.

2. **CWE-829: *Inclusion of Functionality from Untrusted Control Sphere***
   WASC-10*: Insufficient Security* Configurability

Deskripsi: Aplikasi menggunakan pustaka atau komponen pihak ketiga yang mengandung kerentanan keamanan.

3. **CWE-693: *Protection Mechanism Failure***
   WASC-15: *Application Misconfiguration*

Deskripsi: Aplikasi gagal mengimplementasikan mekanisme keamanan dengan benar, seperti header keamanan HTTP.

4. **CWE-319: *Cleartext Transmission of Sensitive Information***
   WASC-15: *Application Misconfiguration*

Deskripsi: Data sensitif dikirim melalui jaringan tanpa enkripsi, memungkinkan penyadapan (*man-in-the-middle attack*).

5. **CWE-1021: *Improper Restriction of Rendered UI Layers or Frames***
   WASC-15: *Application Misconfiguration*

Deskripsi: Aplikasi gagal melindungi antarmuka pengguna dengan benar, memungkinkan serangan seperti *clickjacking*.

6. **CWE-829: *Inclusion of Functionality from Untrusted Control Sphere***
   WASC-15: *Application Misconfiguration*

Deskripsi: Aplikasi menyertakan kode dari sumber yang tidak terpercaya, memungkinkan serangan manipulasi.

7. **CWE-614: *Sensitive Data in Insecure Container***
   WASC-15: *Application Misconfiguration*

Deskripsi: Data sensitif (seperti *cookies* atau sumber daya lain) disertakan pada halaman atau diakses melalui koneksi yang tidak aman.

8. **CWE-200: *Information Exposure***
   WASC-13*: Information Leakage*

Deskripsi: Kesalahan konfigurasi dalam pengaturan keamanan aplikasi, seperti header HTTP yang hilang.

9. **CWE-693*: Protection Mechanism Failure***
   WASC-4:*Insufficient Transport Layer Protection*

Deskripsi: Header *Cache-Control* atau *Pragma* tidak disertakan, memungkinkan data sensitif disimpan dalam *cache klien* atau *proxy*.

10. **CWE-200: *Information Exposure***
    WASC-13: *Information Leakage*

Deskripsi: Komentar dalam kode atau tanggapan HTTP mengandung informasi sensitif yang dapat diakses oleh pengguna.

11. **CWE-693: *Protection Mechanism Failure***
    WASC-15: *Application Misconfiguration*

Deskripsi: Sistem mengungkapkan stempel waktu atau informasi waktu lainnya, yang dapat digunakan untuk serangan seperti pengintaian atau penghitungan.

12. **CWE-525: *Missing Cache-Control Directive***
    WASC-13: *Information Leakage*

Deskripsi: Aplikasi tidak menyertakan *header Cache-Control* dengan benar, menyebabkan data disimpan secara tidak aman.

e-ISSN : 2615-5346

V. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Kerentanan risiko pada *website* UUI secara keseluruhan (2) temuan dengan risiko tinggi (11.1%)., (3) temuan dengan risiko sedang (16.7%), (7) temuan dengan risiko rendah (38.9%), (6) temuan yang hanya bersifat informasional (33.3%) yang berarti *Website* UUI memiliki beberapa kelemahan signifikan yang perlu segera ditangani, terutama temuan dengan risiko tinggi dan sedang.

Risiko pada level tinggi yaitu *PII Disclosure* dan *Vulnerable JS Library* dan risiko pada level sedang yaitu *Content Security Policy (CSP) Header Not Set, HTTP to HTTPS Insecure Transition in Form Post* dan *Missing Anti-Clickjacking Header*. Dengan distribusi temuan yang didominasi oleh risiko rendah dan informasional, *Website* UUI dapat dikatakan sudah berada pada tingkat keamanan yang cukup baik. Keamanan yang terus diperbarui adalah kunci untuk menjaga sistem tetap aman dari ancaman baru.

5.2 Saran

Dari hasil penelitian yang dilakukan maka diperoleh beberapa saran yang dapat dikemukakan yaitu :

1. Proses pengujian pengamanan pada *website* harus terus ditingkatkan, terutama temuan dengan risiko pada level tinggi dan sedang.
2. Menutup celah risiko tinggi dan sedang yang ditemukan.
3. Periksa dan hapus komentar di kode sumber yang mungkin berisi informasi sensitif atau detail teknis yang dapat dimanfaatkan oleh penyerang.
4. Pengelola dapat terus melakukan audit keamanan pada sistem secara berkala untuk memastikan keamanan sistem tetap terjaga.

REFERENSI

[1] Fachri, F., adlil, A., Riadi, I., & Dahlan, A.(2021). Analisis Keamanan Webserver menggunakan Penetration Test. *J. Inform*, *8*(2), 183-190.

[2] Germanotta, J. (2021). Metodologi Penetration Testing 2021. Dipetik Desember 23, 2024, dari https://biztech.proxsisgroup.com/metodologi-penetration-testing/.

[3] Hardiani, T., Wijayanto, D., Latifah, N., & Informasi, T. (2022). Data Security Analysis with OWASP framework on website XYZ. *vol*, *6*, 10-20.

[4] Kuncoro, A. W & Fayruz Rahma, S. T., ENG, M.(2022). Analisis Metode Open Web Application Security Project (OWASP) pada Pengujian Keamanan Website:Literature Review. Automata, *3*(1).

[5] M. Z. A. Shebli and B. D. Beheshti (2018) A study on penetration testing process and tools, *2018 IEEE Long Island Systems, Applications and Technology Conference (LISAT)*, Farmingdale, NY, USA, 2018, pp. 1-7, doi: 10.1109/LISAT.2018.8378035.

[6] M. Helmi S.F., Firza P.A., &M. Muharrom A.H.(2023). Pengujian Sistem Jaringan Dokumentasi Dan Informasi Menggunakan Black Box Testing Dan White Box Testing. *Jurnal Publikasi Sistem Informasi Dan Manajemen Bisnis*, *3*(1),213–221. https://doi.org/10.55606/jupsim.v3i1.2447.

[7] Putri, D., & Santoso, A. (2021). Pengembangan Website Interaktif untuk Penyebaran Informasi Sekolah. *Jurnal Teknologi Informasi dan Komunikasi*, Vol. 8, No. 2.

[8] Universitas Ubudiyah Indonesia. (2023). Sejarah Singkat Universitas Ubudiyah Indonesia 2023. Dipetik Desember 24, 2024, dari https://id.wikipedia.org/wiki/Universitas_Ubudiyah_Indonesia/.

[9] Zahra, N. A., Zidane, F. H., & Kuslaila, N. R. (2023). Analisis Keamanan Sistem Informasi Pada Website Pt Sentra Vidya Utama