Jurnal Informatika dan Rekayasa Perangkat Lunak ISSN 2656-2855
Vol. 5, No. 2, September 2023, Hal. 262-272 e-ISSN 2685-5518

Analisis Keamanan *Website* Menggunakan Standar Keamanan *Open Web Application Security Project (OWASP)* Studi Kasus *Website* Penerimaan Mahasiswa Baru Universitas Wahid Hasyim Semarang

Ahmad Zaini[1], Rony Wijanarko[2]
[1,2]Program Studi Teknik Informatika, Fakultas Teknik, Universitas Wahid Hasyim
Email: zainiplonto203@gmail.com

Abstrak

Universitas Wahid Hasyim memiliki website yang memuat informasi dan dokumen yang dipublikasi dan dapat diakses oleh penggunanya. Salah satu website yang paling krusial adalah website Penerimaan Mahasiswa Baru (PMB). Maraknya kebocoran data penduduk di Indonesia membuka mata kita bahwa dibalik kemajuan sebuah digital teknologi terdapat tingkat ancaman yang cukup tinggi. Berdasarkan masalah yang ada dibutuhkan analisis tingkat keamanan website dengan menggunakan standar keamanan Open Web Application Security Project (OWASP), yang bisa meringankan beban pengelola dan pengembang sistem dengan tujuan mencegah dan mengatasi efek resiko yang ditemukan pada website Penerimaan Mahasasiwa Baru Universitas Wahid Hasyim Semarang. Pengujian sistem keamanan yang digunakan oleh peneliti di penelitian ini adalah menggunakan standar keamanan (OWASP) Open Web Application Security Project yang merupakan top 10 dari standar keamanan yang dirilis oleh organisasi (OWASP) yang berisi 10 daftar tertinggi celah keamanan yang mengancam keamanan suatu website, dan menggunakan (OWASP-ZAP) Zed Attack Proxy merupakan aplikasi yang dipergunakan dalam pengujian penetrasi untuk menemukan kerentanan/lubang keamanan pada sebuah aplikasi website. Metode pengujian dengan (OWASP) dapat memberikan bantuan pemilihan tindakan yang yang perlu diambil guna memperkecil kerentanan kebocoran data. Berdasarkan hasil analisis mengunakan (OWASP-ZAP) ditemukan beberapa celah dan kerentan pada website. Berdasarkan hasil pengujian penetration test kualitas keamanan website Penerimaan Mahasiswa Baru ada dalam tingkatan sedang sehingga diperlukan tindakan perbaikan lebih lanjut dari pihak pengembang website untuk meningkatkan keamanan website.

***Kata Kunci**: OWASP Top 10, Keamanan Website, OWASP-ZAP, SQL Injection, Local File Inclusion.*

Abstract

Wahid Hasyim University has a website that contains information and documents that are published and can be accessed by its users. One of the most crucial websites is the New Student Admissions (PMB) website. The increasing number of population data leaks in Indonesia has opened our eyes to the fact that behind the progress of digital technology there is a fairly high level of threat. Based on existing problems, it is necessary to analyze the level of website security using the Open Web Application Security Project (OWASP) security standards, which can ease the burden on system managers and developers with the aim of preventing and overcoming the effects of risks found on the Wahid Hasyim University Semarang New Student Admissions website. The security system testing used by researchers in this study is using the Open Web Application Security Project (OWASP) security standard which is the top 10 of the security standards released by the organization (OWASP) which contains the 10 highest lists of security gaps that threaten the security of a website, and using (OWASP-ZAP) Zed Attack Proxy is an application used in penetration testing to find security vulnerabilities/holes in a website application. The testing method with (OWASP) can help in selecting the actions that need to be taken to reduce the vulnerability of data leaks. Based on the results of the analysis using (OWASP-ZAP), several gaps and vulnerabilities were found on the website. Based on the results of the penetration test, the security quality of the New Student Admissions website is at a moderate level so further corrective action is needed from the website developer to improve website security.

***Kata Kunci**: OWASP Top 10, Web Security, OWASP-ZAP, SQL Injection, Local File Inclusion.*

PENDAHULUAN

Di era *society 5.0* diperlukan konsep pembaruan tatanan dalam masyarakat. Melalui konsep *(society) 5.0* diharapkan kehidupan masyarakat lebih nyaman dan terbuka. Begitu juga dalam dunia pendidikan dibutuhkan perubahan metode pendidikan 5.0 dengan tujuan mendukung revolusi industri kelima, lembaga pendidikan diharapkan mempersiapkan generasi yang berkualitas dan siap untuk menghadapi kemajuan teknologi.

Keamanan pada *website* berarti keamanan informasi dari pengguna dapat menghindari kebocoran data. Resiko rendahnya tingkat keamanan pada sistem menjadi potensi *hacker* untuk masuk ke dalam sistem yang berdampak pada ketidaksesuaian sistem yang telah dibuat.

Maraknya kebocoran data penduduk yang ramai dibahas akhir-akhir ini di Indonesia membuka mata bahwa dibalik kemajuan sebuah sistem teknologi terdapat tingkat ancaman yang cukup tinggi, Pemanfaatan teknologi informasi penggunaan *website* di Universitas Wahid Hasyim memiliki peran penting sebagai media akses informasi dalam waktu yang singkat, *website* ini digunakan oleh para mahasiswa, dosen, staf universitas maupun calon mahasiswa dan alumni.

Universitas Wahid Hasyim memiliki *website* yang berisi informasi dan dokumen yang dipublikasi dan dapat diakses oleh penggunanya. Salah satu sistem yang paling krusial adalah *website* Penerimaan Mahasiswa Baru (PMB). *Website* ini memuat data-data dan informasi pribadi setiap mahasiswa yang berisi data penting dan krusial.

Berdasarkan latar belakang yang ada, diperlukan analisis pengujian keamanan *website*, pada *website* Penerimaan Mahasiswa Baru yang berguna untuk menganalisis *website*, sudah sesuai atau belum dengan standar keamanan untuk melindungi dari tindak kejahatan *cyber* yang bisa merugikan universitas, penulis melakukan penelitian dengan judul "Analisis Keamanan *Website* Menggunakan Standar Keamanan *Open Web Application Security Project* (*OWASP*) Studi Kasus *Website* Penerimaan Mahasiswa Baru Universitas Wahid Hasyim Semarang". Setelah dilakukan analisis tersebut, diharapkan dapat membantu dalam meminimalisir celah pada keamanan *website* Penerimaan Mahasasiswa Baru Universitas Wahid Hasyim.

TINJAUAN PUSTAKA

Di dalam penyusunan penelitian yang dilakukan ini, peneliti menambahkan beberapa referensi dari penelitian - penelitian terdahulu yang dilakukan oleh para peneliti terdahulu sebagai bahan acuan untuk melakukan analisis keamanan *website*. Berikut beberapa penelitian terdahulu yang berhubungan dengan penyusunan penelitian ini:

1. Analisis Keamanan *Web Server Open Journal System (Ojs)* Menggunakan Metode *Issaf Dan Owasp* (Studi Kasus OJS Universitas Lancang Kuning) oleh (Guntoro et al., 2020), latar belakang dari kasus penelitian ini adalah bagaimana menganalisis keamanan sistem *Open Journal System (OJS)* menggunakan metode *(ISSAF) dan (OWASP)* pada Universitas Lancang Kuning.
2. Selanjutnya penelitian oleh (Riandhanu, 2022), dengan judul Analisis Metode *Open Web Application Security Project (OWASP)* Menggunakan *Penetration Testing* pada Keamanan *Website* Absensi. Analisis kerentanan aplikasi berbasis *web* dengan metode *(OWASP)* dengan menggunakan *tools security* untuk mengetahui tingkat keamanan suatu aplikasi.
3. Pada penelitian dengan judul analisis keamanan jaringan *web server* menggunakan *suricata* pada Sekolah Menengah Pertama Negeri 1 Palopo oleh (Layuk, 2021), penelitian ini bertujuan menganalisa keamanan pada *Web Server* dan membantu *administrator* untuk dapat mengetahui mengetahui adanya kemungkinan celah keamanan pada *Web Server* sekolah, sehingga dapat segera dilakukan perbaikan dengan tepat berdasarkan temuan kerentanan atau celah keamanan yang terdapat pada *website* tersebut.

2.1 Keamanan Jaringan

Menurut (Layuk, 2021). Keamanan jaringan merupakan penempatan alat keamanan, kebijakan dan prosedur dengan tujuan untuk mencegah akses tidak sah ke sumber daya jaringan atau kerusakan sumber daya atau data. Keamanan jaringan merupakan upaya pencegahan terhadap serangan pengguna komputer atau pengakses jaringan yang tidak bertanggung jawab.

Sedangkan menurut (Fatma, 2018). Keamanan jaringan *(network security)* dalam jaringan komputer sangat diperlukan untuk memantau akses jaringan dan meminimalkan penyalahgunaan sumber daya jaringan yang tidak sah. Keamanan jaringan memiliki tugas dan dikendalikan oleh administrator jaringan. Secara luas Keamanan sendiri menyangkut 3 elemen dasar yaitu: Keamanan jaringan *(network security)*, Keamanan aplikasi *(application security)*, Keamanan komputer *(computer security).*

2.2 Website

Website merupakan rangkaian halaman data berbasis web yang bisa diakses orang lain di seluruh dunia tanpa ada batasan antara lokasi dan waktu. *Website* ini terdiri dari teks, gambar, audio, video, animasi dan merupakan sarana media untuk membaca atau mengunjungi informasi (Guntoro et al., 2020).

2.3 OWASP TOP 10

Menurut (Guntoro et al., 2020). *Open Web Application Security Project (OWASP)* merupakan *framework open source* yang berfokus pada peningkatan keamanan perangkat lunak aplikasi. *(OWASP)* adalah sebuah komunitas yang dibangun untuk menemukan celah keamanan pada sebuah aplikasi *website.*

Sedangkan menurut (Safitri et al., 2020). Open Web Application Security Project (OWASP) adalah organisasi non *profit* yang fokus di peningkatan keamanan *software* perangkat lunak. *(OWASP)* adalah kerangka kerja yang dipergunakan oleh pengembang dan ahli teknologi untuk pengamanan *website*.

2.4 *OWASP ZAP*

OWASP-ZAP (Zed Attack Proxy) adalah aplikasi yang digunakan untuk sebagai pengujian penetrasi dalam menemukan kerentanan/lubang keamanan yang ada dalam aplikasi *website*. *OWASP-ZAP* menyediakan pemindaian otomatis oleh (Guntoro et al., 2020).

2.5 *Pentesting*

(Penetration Testing) merupakan pendekatan proaktif yang jelas dengan tujuan evaluasi keamanan aset digital dengan secara aktif mengidentifikasi dan juga mengeksploitasi kerentanan yang ada menurut (Filiol, 2021). Strategi pengujian tingkat kerentanan aplikasi web menggunakan metode *absent* (*OWASP)* dengan menggunakan bentuk analisis yang dinamis (*Dynamic Analysis*) yang dilakukan dalam *domain* tempat aplikasi web target beroperasi.

2.6 SQL *Injection*

Menurut (Irawan et al., 2018). *(SQL injection)* adalah suatu teknik yang berguna untuk mengakses dan mengekspoitasi aplikasi web menggunakan data yang disediakan atau disematkan dalam *query (SQL).*

(SQL Injection) merupakan tindakan *hacking* pada aplikasi *client* dengan melakukan modifikasi perintah *(SQL)* yang berada di memori aplikasi *client, (SQL injection)* adalah teknik eksploitasi berbasis aplikasi yang menggunakan *database* guna untuk menyimpan data menurut (Bastian et al., 2020).

2.7 *Local File Inclusion*

Local File Inclusion merupakan kerentanan keamanan yang memungkinkan *hacker* membaca dan melihat file di *server,* termasuk file sensitif (Koprawi, 2020).

Local File Inclusion sering muncul disebabkan kesalahan dalam pengkodean, salah satunya disebabkan dari fungsi seperti fungsi *include* yang tidak tervalidasi dan terfilter dengan baik dan benar. Fungsi *include* yaitu sebuah fungsi dari bahasa pemrograman *PHP* yang memiliki fungsi untuk menyisipkan atau mengasosiasikan sesuatu seperti file ke dalam sebuah halaman pada sebuah *website*. (Begum et al., 2017).

METODE PENELITIAN

3.1 Metode Pengumpulan Data

Pada penelitian ini, metode pengumpulan data adalah faktor terpenting yang harus dijalani untuk dapat melaksanakan analisis dan mengoalah data. Pengumpulan data memiliki tujuan untuk memperoleh informasi yang

dibutuhkan dalam penelitian ini. Dalam pengumpulan data dan informasi adapun beberapa metode yang digunakan oleh peneliti pada penelitian ini yaitu:

1. Observasi
2. Wawancara (*Interview*)
3. Literatur
4. Analisis dan pembuatan laporan

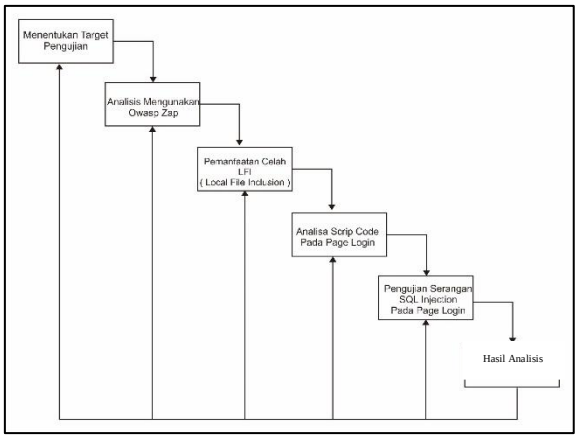
3.2 Jenis Data

1. Data Primer, meruapakan sebuah informasi yang diperoleh langsung oleh peneliti melalui wawancara atau observasi.

2. Data sekunder adalah informasi yang diperoleh peneliti dari media cetak atau online dan bersifat informatif berupa kutipan, kepustakaan, istilah dan jurnal penelitian yang berkaitan dengan penelitian yang dilakukan.

3.3 Tahapan Analisis Website

Berikut adalah tahapan analisis website yang ditunjukkan pada Gambar 1.



Gambar 1. Tahapan Analisis Website

1. Menentukan target pengujian terlebih dahulu
2. Melakukan analisis *website* mengunakan *OWAP-ZAP* untuk mengetahui celah pada *website*
3. Pemanfaatan celah dengan menggunakan *Local File Inclusion* untuk mengambil file *website.*
4. Melakukan analisis pada *webite* yang sudah ditentukan melalui *script code* pada *website* target.
5. Pengunaan *(SQL Injection)* untuk memanfaatkan celah pada *website* target tujuan
6. Memberikan kesimpulan dari hasil pengujian *website*.

HASIL DAN PEMBAHASAN

4.1 Hasil Wawancara Dan Observasi Penulis

Website PMB, menjadi *website* penting yang berisi seluruh data-data penting mahasiswa di Universitas Wahid Hasyim Semarang, peran *website* PMB sangatlah penting untuk memudahkan baik dari calon mahasiswa maupun *administrator*, disini *website* sudah diperbarui dan dalam kondisi baik untuk dalam merespon perintah perintah permintaan maupun dalam pengunaan *web.* terkait keamanan jaringan sudah diterapkan keamanan standar sesuai keamanan *website* tetapi masih ada beberapa celah pada *(SQL injection)* pada *website* PMB Universitas Wahid Hasyim Semarang.

4.2 Implementasi Hadware

Pada penelitian ini perangkat keras *(hardware)* yang digunakan untuk menganalisis keamanan *website* target, adalah *Server* UPT TIK Target. Sedangkan di sisi *client*, menggunakan laptop untuk mengakses *website* target untuk menjalankan aplikasi analisis *OWASP-ZAP* dan berbagai perangkat lainnya. Pertama, pastikan semua perangkat yang ada sudah terhubung dengan *internet.* Jika perangkat sudah dipastikan terkoneksi dengan *internet*, langkah selanjutnya adalah melakukan analisis pada *website* target.

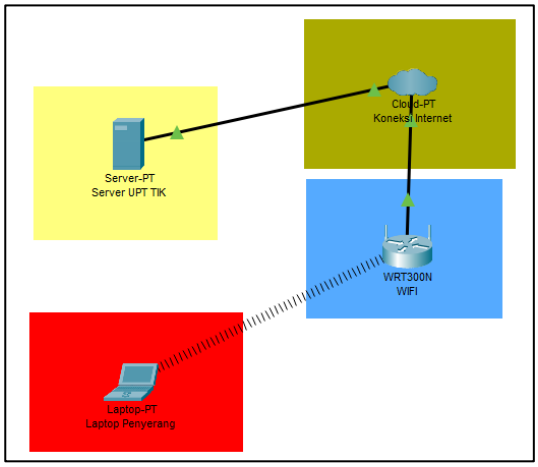
4.3 Implementasi Sofware

Pada tahap ini penulis menyiapkan sebuah aplikasi yang akan digunakan untuk menganalisis keamanan jaringan yang ada. Aplikasi yang digunakan adalah *OWASP Zed Attack Proxy* yang di install pada *OS Kali Linux* melalui *VMware* dan melakukan penilaian atau analisis kerentanan pada *website* PMB Universitas Wahid Hasyim. Dalam hal ini, *sistem operasi Windows 10* juga digunakan oleh sisi *client*, yang digunakan untuk menganalisis *script code* menggunakan *sublime text* dan melakukan *injeksi SQL* pada halaman *website* Penerimaan mahasiswa baru Universitas Wahid Hasyim Semarang.

4.4 Topologi Serangan

Pada Gambar 2, topologi jaringan yang digunakan dalam aliran serangan ini menggambarkan situasi di mana *server* dan *komputer* penyerang berada di *gateway Internet*

yang berbeda. Karena *server* menjalankan *OS Centos* di lokasi yang berbeda. Sedangkan *komputer* penyerang terhubung ke jaringan *nirkabel atau WLAN (Wireless Local Area Network)* sebagai berikut :Topologi Serangan



Gambar 2. Topologi Serangan

4.5 Target Pengujian

Penentuan target pengujian adalah hal awal yang diperlukan untuk menentukan target, serta dapat fokus dalam mengerjakan suatu objek studi kasus dari penentuan target dapat di tentukan target informasi seperti yang ditunjukkan pada Tabel 1.

Tabel 1. Target Informasi

Sistem Operasi	*CentOS*
Hostname	target.ac.id
Web Server	*Apache/2.4.6*
Jenis Aplikasi	*Web*
Jenis Akses	*Internet* dan intranet

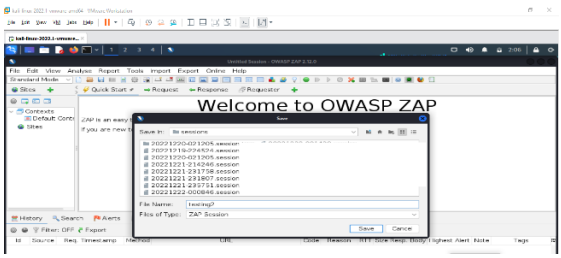
4.6 Implementasi Dan Pengaturan *OWASP ZAP*

Langkah-langkah implementasi dan pengaturan OWASP, sebagai berikut:

1. Menjalankan Software *OWASAP ZAP*

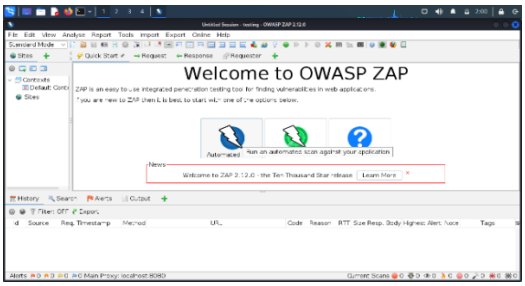
Gambar 2. Membuka Software *OWASAP ZAP*

2. Menyimpan *Sesion*



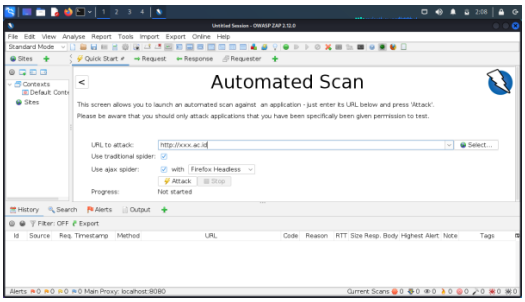
Gambar 2. Menyimpan *Sesion*

3. Scan *Website* Target nenggunakan *Automated Scan*



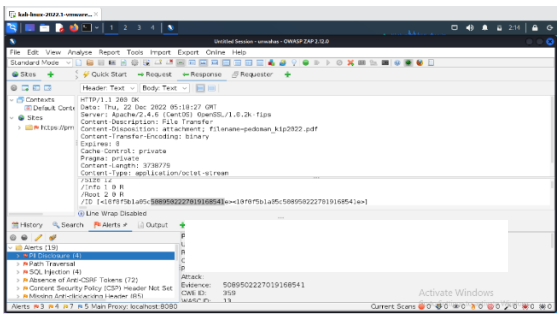
Gambar 3. *Automated Scan*

4. Menentukan Pengaturan Untuk Analisis *Website* Pada *OWASAP ZAP* untuk scan



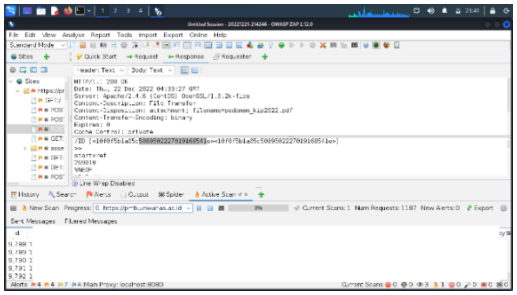
Gambar 4. Menentukan Pengaturan untuk Analisis

5. Menemukan Celah Pada *Website* Target Tujuan

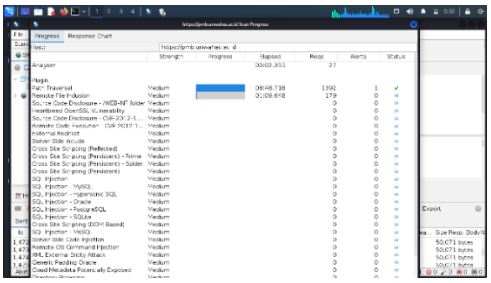


Gambar 5. Celah pada *website*

6. Melanjutkan Proses Scan menggunakan *Active Scan*

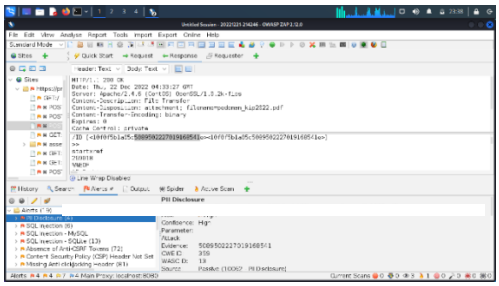


Gambar 6. *Scan* menggunakan *active scan*



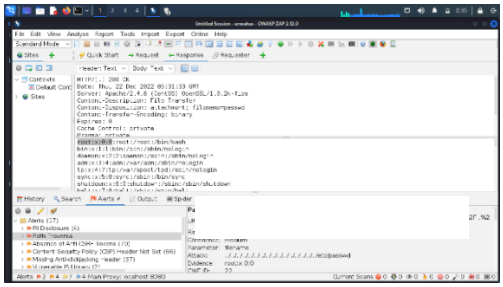
Gambar 7. Daftar *scan website*

7. Pemanfaatan Celah Pada *Website*



Gambar 8. Pemanfaatan celah *website*

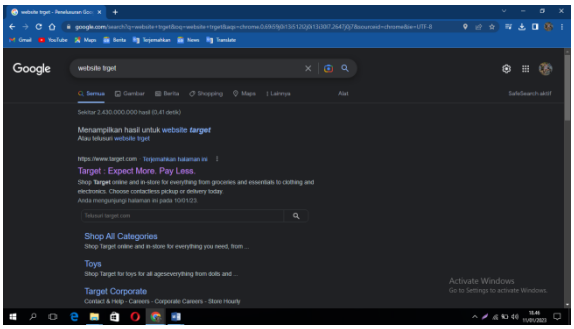
8. Pemanfaatan Celah *Path Traversal*



Gambar 9. Pemanfaatan celah *path tarversal*

4.7 *Pentesting* Pengujian *Website*

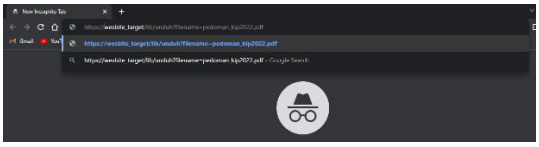
1. Membuka *Website* Target



Gambar 10. Membuka *Website* Target

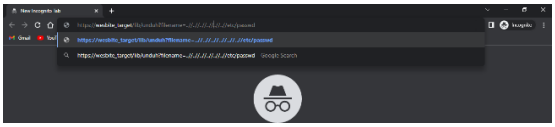
2. Memanfaatkan Celah Path pada Website Menggunakan *LFI* Mengcopy *Link URL*

Gambar 11. Menyalin link Panduan KIP

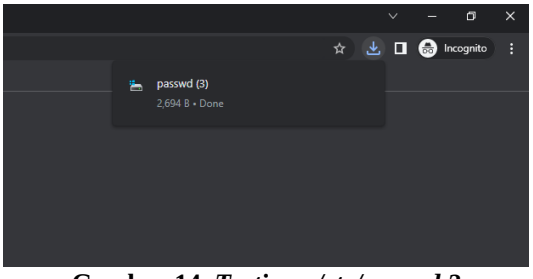


Gambar 12. Menyalin link Panduan KIP

3. Mencoba Melakukan Testing Menggunakan *../etc/passwd*

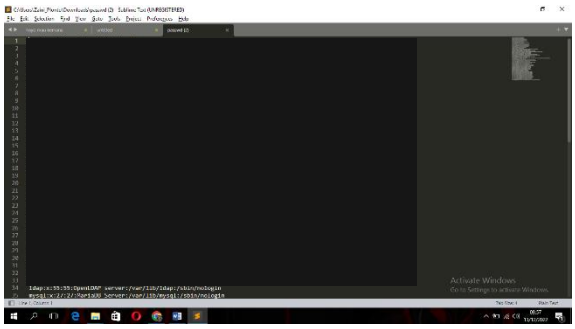


Gambar 13. *Testing ../etc/passwd 1*



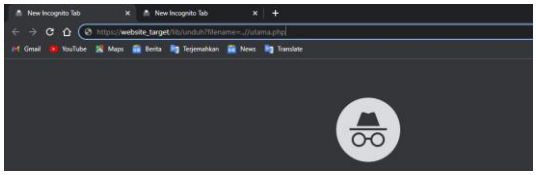
Gambar 14. *Testing ../etc/passwd 2*

4. Mengunduh File *../etc/passwd*

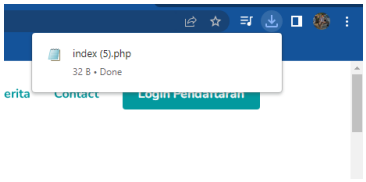


Gambar 15. *Testing ../etc/passwd 3*

5. Memanfaatkan Celah untuk mengunduh akses file utama



Gambar 16. Akses file utama.php 1

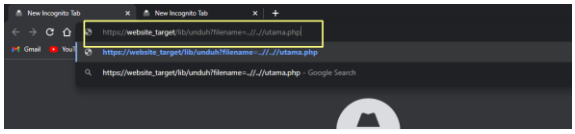


Gambar 17. Akses file utama.php 2

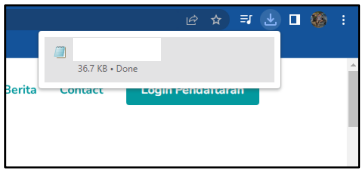
Gambar 18. Akses file utama.php 3

Penulis coba untuk mendownload file utama.php pada website tetapi untuk file utama tidak disimpan pada direktori tersebut

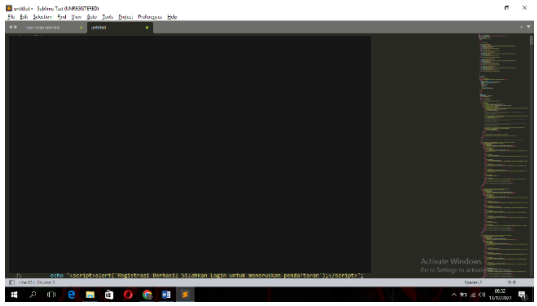
6. Akses File Utama Menggunakan *..//..//utama.php*



Gambar 19. Akses file utama.php 4



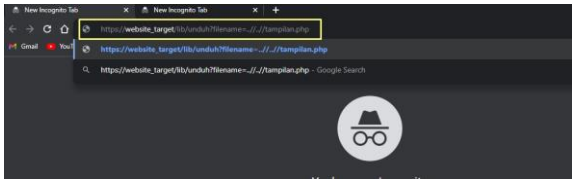
Gambar 20. Akses file utama.php 5



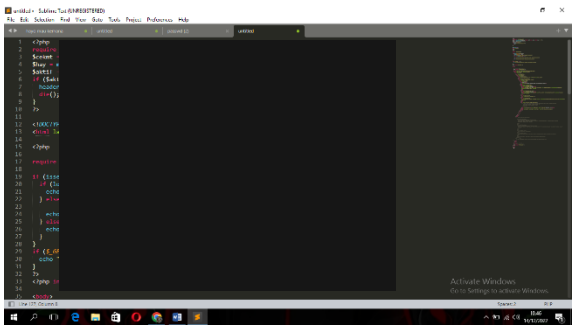
Gambar 21. Akses file utama.php 6

7. Akes File Tampilan *..//..//tampilan.php*

Untuk menganalisis Script pada Tampilan.php



Gambar 22. Akses File Tampilan.Php 7

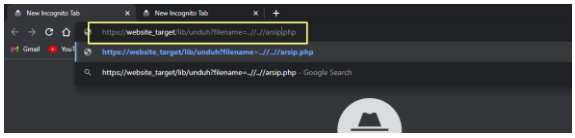


Gambar 23. Membuka File Tampilan.Php

8. Menampilkan File Tampilan.php ternyata untuk analisis masuk ke *website* disimpan di file arsip.php

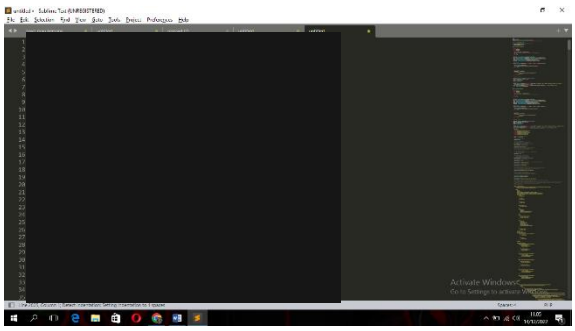
Gambar 24. Membuka file arsip.php

9. Mengunduh File Arsip.php
   ..//..//arsip.php

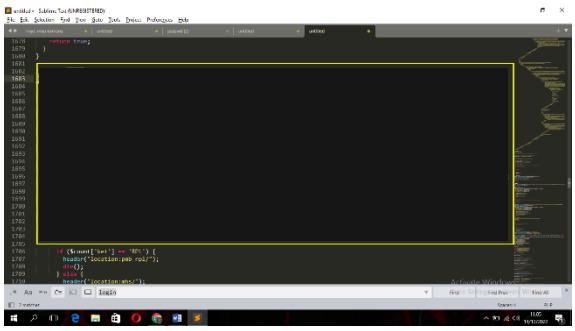


Gambar 21. Unduh File Arsip.Php

10. Menampilkan File arsip.php dan membukanya untuk analisis script



Gambar 22. Membuka file arsip.php



Gambar 23. Analisis scrip code *website*

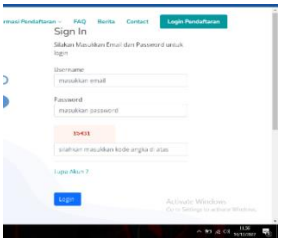
11. Menemukan Celah pada website melalui File Arisp.php

Gambar 24. file *username dan password*

12. Password untuk akses file utama pada website

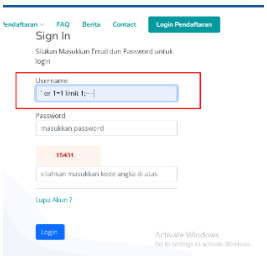
Gambar 25. *Password* file utama

13. Percobaan *SQL Injection* Pada Tampilan website



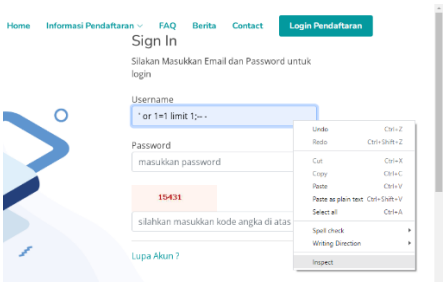
Gambar 26. Tampilan utama

14. Memasukan scrip file *SQL Injection*



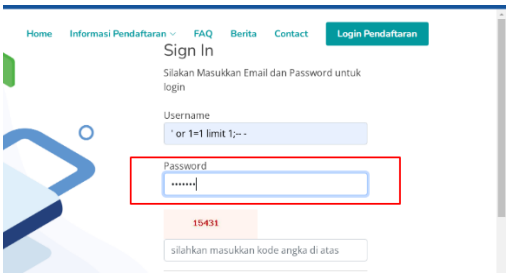
Gambar 27. Web perintah *SQL Injection*

15. *Inspect elemen* merubah karakter



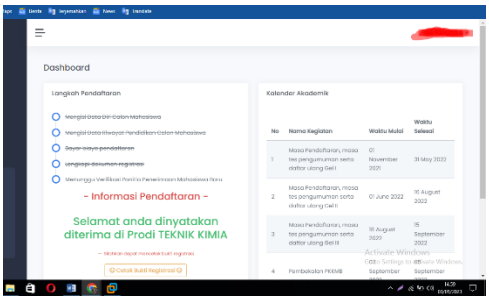
Gambar 28. *Inspect Elemen*

16. Memasukan *Password* yang didapat pada file arsip.php

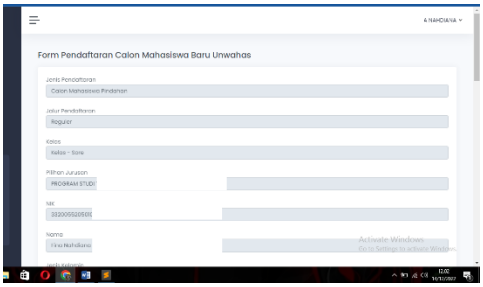


Gambar 29. Memasukan *password*

17. Percobaan Masuk Pada *Website* target



Gambar 30. Akses masuk web 1



Gambar 31. akses masuk web 2

4.8 Analisis Hasil Penetrasi *OWASP ZAP*

Dari pengujian yang telah dilakukan dengan beberapa metode seperti analisis menggunakan *OWASP ZAP* untuk mencari kerentanan website, percobaan injeksi kode, dan pengujian menggunakan *tools*, pada *website* https://xxx.ac.id memiliki 19 kerentanan yaitu :

PII Disclosure (4), Path Traversal, SQL Injection (4), Absence of Anti-CSRF Token (72), Content Security Policy (CSP) Header Not Set (100), Missing Anti-clickjacking Header (85), Vulnerable JS Library (2), Cookie No Http Only Flag (2), Cookie Without Secure Flag (2), Cookie without SameSite Attribute (2), Cross-Domain JavaScript Source File Inclusion (60), Server Leaks Version Information via "Server" HTTP Response Header Field (216), Strict-Transport-Security Header Not Set (188), X-Content-Type-Options Header Missing (173), Content-Type Header Missing, Information Disclosure - Suspicious Comments (40), Modern Web Application (67), Re-examine Cache-control Directives (63), User Controllable HTML Element Attribute (Potential XSS) (7).

4.9 Penetrasi *OWASP ZAP* Berdasarkan *OWASP TOP 10*

Berdasarkan hasil pengujian *OWASP-ZAP*, *website* Penerimaan Mahasiswa Baru Universitas Wahid Hasyim memiliki 19 kerentanan. Dari 19 kerentanan, 7 termasuk dalam daftar OWASP TOP 10. Berikut adalah hasil pengujian OWASP ZAP berdasarkan daftar OWASP TOP 10 yang disajikan pada tabel metode pengujian dan hasil pada tabel di bawah ini.

Tabel 4.2 Tabel metode pengujian dan hasil.

No	Owasp Top 10	Tools	Celah rentan	Tingkat rentan
1	*Broken Access Control*	*Access Control Testing*	Ditemukan	Tinggi
2	*Cryptographic Failures*	*Active Scan Rules*	Ditemukan	Tinggi
3	*Injection*	*Access Control Testing*	Ditemukan	Tinggi
4	*Insecure Design*	*Active Scan Rules*	Ditemukan	Sedang
5	*Security Misconfiguration*	*Access Control Testing*	Ditemukan	Sedang
6	*Vulnerable and Outdated*	*Active Scan Rules*	Ditemukan	Sedang
7	*Identification and Authentication Failures*	*Active Scan Rules*	Ditemukan	Rendah
8	*Software and Data Integrity Failures*	*Active Scan Rules*	Tidak Ditemukan	Tidak Ditemukan
9	*Security Logging and Monitoring Failures*	*Access Control Testing*	Tidak Ditemukan	Tidak Ditemukan
10	*Server-Side Request Forgery*	*Active Scan Rules*	Tidak Ditemukan	Tidak Ditemukan

4.10 Rekomendasi Perbaikan

Untuk meminimalisir kerentanan yang dihasilkan dalam pengujian analisis *website* PMB Universitas Wahid Hasyim maka untuk pengembangan *website* selanjutnya dapat mengikuti saran berikut ini :

1. Memperbaiki dalam akses link *https://xxx.ac.id/assets/file/pedoman_kip2022.pdf* dengan dirubah sebagai berikut untuk meminimalisir celah pada *path*
2. Untuk nama file, gunakan daftar izin ketat yang membatasi kumpulan karakter yang akan digunakan. Gunakan daftar ekstensi file yang diizinkan.
3. *Escape* semua data yang diterima dari klien. Untuk *(SQL injection)* pada login pada username bisa diperkuat lagi menggunakan *mysqli_real_escape_string* yang digunakan untuk lolos dari karakter khusus dalam string untuk digunakan dalam kueri *SQL,*
4. Gunakan pustaka atau kerangka kerja terverifikasi
5. Memastikan *web browser* serta konfigurasi penyeimbang untuk menyetel header Kebijakan-Keamanan-Konten,
6. *Browser Kebijakan Keamanan Konten HTTP Header Modern dan Opsi X-Frame.*
7. *H*arap tingkatkan ke versi terbaru jquery.
8. Pastikan bendera *HttpOnly* disetel untuk semua *cookie.*
9. *cookie* harus selalu diteruskan menggunakan saluran terenkripsi.
10. Kesalahan unik untuk *klien (browser)* saat mencatat data di sisi *server* dan tidak dikomunikasikan ke pengguna
11. Pastikan file sumber *JavaScript* hanya diunduh dari sumber tepercaya.
12. Pastikan *server web, server aplikasi,* penyeimbang muatan, dll.
13. Menerapkan keselamatan lalu lintas yang ketat.
14. Pastikan bahwa *aplikasi/server web* menyetel header *Content-Type dengan benar dan header X-Content-Type-Options ke "nosniff"* untuk semua halaman web.
15. Pastikan setiap halaman menetapkan nilai tipe konten yang spesifik dan sesuai untuk konten yang disajikan. Hapus semua komentar yang mengembalikan informasi yang dapat membantu penyerang dan memperbaiki masalah mendasar yang terkait dengannya.
16. Peringatan informasi, jadi tidak perlu ada perubahan.
17. Pastikan tajuk *HTTP Kontrol Cache diatur ke "no-cache, no-store, must-revalidate".*
18. Validasi semua input dan bersihkan output sebelum menulis ke atribut *HTML.*

SIMPULAN

Tujuan dari pengujian keamanan data yang dilakukan dengan metode *(OWASP)* adalah untuk menguji tingkat keamanan website seleksi mahasiswa baru Universitas Wahid Hasyim. Dari semua fungsi untuk meningkatkan keamanan website, dapat disimpulkan sebagai berikut. Berdasarkan hasil analisis mengunakn *OWASP-ZAP* ditemukan beberapa celah dan kerentan pada *website*, penulis sudah menyampaikan kepada pihak terkait untuk ditindaklnjuti agar website diperbaiki maupun ditingkatkan untuk memperbaiki website dalam meminimalisir celah yang bisa diexploitasi oleh *hacker*. Berdasarkan hasil analisis, kerentanan yang ditemukan hanya dari konfigurasi yang kurang tepat. Dan tidak ditemukan kerentanan lain karena sistem telah berhasil menerapkan beberapa fitur keamanan.

Berdasarkan penelitian yang telah dilakukan, dapat diajukan beberapa saran untuk pengembangan website PMB Universitas Wahid Hasyim Semarang yang dapat bermanfaat dan lebih berkembang lagi. Untuk pengujian terhadap *website* pmb mungkin bisa dimaksimalkan lebih baik lagi kedepannya untuk meminimalisir celah pada *website* PMB Universitas Wahid Hasyim Semarang. Perlu dilakukan pengujian dengan menggunakan metode ISSAF (*Information System Security Assessment Framework*) untuk mengidentifikasi kerentanan pada sisi web server, atau menggunakan pengujian WSTG v4.2 untuk

menentukan pengujian sistem terutama pada sisi pengguna, karena item yang diuji mencakup semua fungsi sistem. Untuk pengujian menggunakan OWASAP TOP 10 dan OWASP ZAP bisa digunakan untuk menganalisis keamanan pada *website* universitas untuk meningkatkan keamanan *website* pada lingkungan universitas.

DAFTAR PUSTAKA

Begum, A., Hassan, M. M., Bhuiyan, T., & Sharif, M. H. (2017). RFI and SQLi based local file inclusion vulnerabilities in web applications of Bangladesh. *IWCI 2016 - 2016 International Workshop on Computational Intelligence*, *June 2018*, 21–25. https://doi.org/10.1109/IWCI.2016.7860332

Fatma, W. D. (2018). *Analisa Keamanan Server Pada Login Page Webserver Dengan Enkripsi Sha 1 Dari Serangan Sql Injection Menggunakansystemoperasi Kali Linux Di Lkp Multi Logika Binjai*.

Guntoro, G., Costaner, L., & Musfawati, M. (2020). Analisis Keamanan Web Server Open Journal System (Ojs) Menggunakan Metode Issaf Dan Owasp (Studi Kasus Ojs Universitas Lancang Kuning). *JIPI (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika)*, *5*(1), 45. https://doi.org/10.29100/jipi.v5i1.1565

Irawan, A. S., Pramukantoro, E. S., & Kusyanti, A. (2018). Pengembangan Intrusion Detection System Terhadap SQL Injection *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya*, *2*(6), 2295–2301.

Koprawi, M. (2020). Dampak dan Pencegahan Serangan File Inclusion: Perspektif Developer. *InfoTekJar : Jurnal Nasional Informatika Dan Teknologi Jaringan*, *5*(1), 40–43. https://doi.org/10.30743/infotekjar.v5i1.1997

LAYUK, K. Y. (2021). *Analisis Keamanan Jaringan Web Server Menggunakan Suricata Pada Sekolah Menengah Pertama Negeri 1 Palopo*. http://repository.uncp.ac.id/412/

Riandhanu, I. O. (2022). Analisis Metode Open Web Application Security Project (OWASP) Menggunakan Penetration Testing pada Keamanan Website Absensi. *Jurnal Informasi Dan Teknologi*, *4*(3), 160–165. https://doi.org/10.37034/jidt.v4i3.236

Safitri, E. M., Ameilindra, Z., & Yulianti, R. (2020). Analisis Teknik Social Engineering Sebagai Ancaman Dalam Keamanan Sistem Informasi: Studi Literatur. *Jurnal Ilmiah Teknologi Informasi Dan Robotika*, *2*(2), 21–26. https://doi.org/10.33005/jifti.v2i2.26

Yudiana, Y., Elanda, A., & Buana, R. L. (2021). Analisis Kualitas Keamanan Sistem Informasi E-Office Berbasis Website Pada STMIK Rosma *(Journal of Computer Engineering, System and Science)*, *6*(2), 185. https://doi.org/10.24114/cess.v6i2.24777