ANALISIS KEAMANAN SISTEM WEBSITE MENGGUNAKAN METODE *OPEN WEB APPLICATION SECURITY PROJECT* (OWASP) PADA SIMANTEP.ID

Ela Nurelasari, Difa Gumilang Al Farabi
Teknologi Informasi, Fakultas Teknik dan Informatika Universitas Bina Sarana Informatika
Jalan Kramat Raya No. 98, Senen, Jakarta Pusat
ela.eur@bsi.ac.id

ABSTRAK

Informasi sangat penting bagi organisasi, bisnis, maupun individu. Program keamanan berfokus pada prinsip *Confidentiality* (kerahasiaan), *Integrity* (integritas), dan *Avaliablity* (ketersediaan) (CIA). Dalam konsep keamanan informasi, ketiga komponen tersebut berfungsi sebagai mata rantai yang saling berhubungan. OWASP *(Open Web Application Security Project) Top 10 Method* adalah salah satu metode pengujian sistem berbasis web yang dirilis oleh komunitas OWASP. Metode ini mencakup sepuluh celah keamanan utama yang dapat mengancam suatu website. Simantep adalah sebuah Perusahaan yang didirikan pada akhir tahun 2019. Kemudian secara resmi menjadi sebuah Perusahaan pada tahun 2022. Merupakan salah satu Perusahaan yang bergerak di bidang penjualan online dan penyewaan Gudang. Untuk mengatasi atau mencegah peretasan terhadap *website* Simantep maka digunakanlah metode *Open Web Application Security Project*. aplikasi OWASPZap yang seberapa mungkin ada celah keamanan pada *website* target berdasarkan tingkat ancaman disini terbagi menjadi beberapa kategori berdasarkan dampak yang ditimbulkan dari celah keamanan tersebut yaitu , *Medium* 3, *Low* 3, *Informational* 4. Hasil dari proses *scanning* pada *website* Simantep.id menunjukkan adanya 10 tanda bahaya (*alerts*) dengan tingkat ancaman dari yang terendah (*Medium*) sampai dengan yang tertinggi (*Low*). Penelitian ini juga dapat digunakan sebagai acuan bagi pengembang *website* untuk meningkatkan kualitas keamanannya. Selain itu penelitian ini juga dapat memberikan kontribusi dalam pengembangan teori dan pengetahuan tentang keamanan *website* terutama dengan OWASP.

Kata kunci : *Open Web Application Security Project, OWASP, Keamanan, OWASPZap.*

1. PENDAHULUAN

Teknologi Informasi adalah suatu teknologi yang digunakan untuk mengolah data termasuk memproses, mendapatkan, menyusun, menyimpan, memanipulasi data dalam berbagai cara untuk menghasilkan informasi yang berkualitas, yaitu informasi yang relevan, akurat dan tepat waktu, yang digunakan untuk keperluan pribadi,bisnis, dan pemerintahan dan merupakan informasi yang strategis pengambilan keputusan.

Perkembangan teknologi informasi merupakan faktor penting bagi kemajuan jaman. Ada beberapa bidang yang menjadi kunci kemajuan teknologi mempengaruhi tingkat kemajuan dalam negara tersebut diantaranya bidang Pendidikan, bidang Ekonomi, bidang Kesehatan, bidang Pemerintahan, dan bidang Sosial Budaya. Pada dasarnya teknologi diciptakan untuk memudahkan pekerjaan manusia. Saat ini teknologi sudah menjadi kebutuhan primer manusia. Bahkan teknologi sudah digunakan di semua segi kehidupan manusia [1].

Hasil terbaru dari survey Asosiasi. Penyelenggara Jasa Internet Indonesia (APJII) mencatat penetrasi internet di Indonesia telah mencapai 78,19% dari total populasi yang sebesar 275.773.901 jiwa.

Penetrasi internet di tanah air meningkat dibandingkan tahun-tahun sebelumnya. Misalnya, pada 2018, penetrasi internet mencapai 64,80%, yang kemudian meningkat menjadi 73,70% dari 2019 hingga 2022%.

Semakin banyak pengguna internet, semakin banyak data yang dapat diakses. Di era internet saat ini, informasi menjadi sangat penting bagi organisasi, perusahaan, dan individu. Semakin banyak orang yang memberi tahu tentang diri mereka di internet, semakin menurunnya privasi. Seiring dengan itu, semakin banyak orang yang menjadi lebih sadar dengan bagaimana informasi mereka dimanfaatkan, dan semakin banyak organisasi yang memperhatikan risiko keamanan informasi yang dapat mengakibatkan kerugian dan dampak negatif terhadap data mereka.

Program keamanan berfokus pada prinsip. *Confidentiality* (kerahasiaan), *Integrity* (integritas), dan *Avaliablity* (ketersediaan) (CIA). Dalam konsep keamanan informasi, ketiga komponen tersebut berfungsi sebagai mata rantai yang saling berhubungan [2].

SIMANTEP adalah sebuah Perusahaan yang didirikan pada akhir tahun 2019. Kemudian secara resmi menjadi sebuah Perusahaan pada tahun 2022. Merupakan salah satu Perusahaan yang bergerak di bidang penjualan online dan penyewaan Gudang. Permasalahan yang akan dibahas adalah tentang sistem keamanan yang ada di *website* Simantep.id menggunakan metode OWASP.

OWASP *(Open Web Application Security Project) Top 10 Method* adalah salah satu metode pengujian sistem berbasis web yang dirilis oleh komunitas OWASP. Metode ini mencakup sepuluh celah keamanan utama yang dapat mengancam suatu *website* [3].

Menurut analisis metode OWASP untuk

pengujian keamanan sistem, alat dan teknik sangat mempengaruhi proses dan hasil. Alat, tujuan, dan metode pengujian semuanya berbeda. Ini dimulai dengan langkah- langkah ini dan diakhiri dengan analisis dan hasil dari rekomendasi tersebut. [4].

Pada penelitian sebelumnya [5] menganalisis dan menguji tingkat kerentanan sistem OJS dengan pengujian menggunakan metode ISSAF dan OWASP, sistem OJS di Universitas Lampung dianggap aman. Penelitian Riandhanu [3] menganalisis dan mengetahui kerentanan pada keamanan web absensi dengan menggunakan Penetration Testing, hasil uji menunjukan kerentanan pada tingkat ancaman kritis(*Critical)*, tiga temuan tinggi (*high)* , empat temuan menengah (*medium)* dan tujuh temuan rendah (*low).*

Adapun penelitian [4] menganalisis dan menguji keamanan sistem dari website tapanuli tengah menggunakan metode OWASP ZAP hasil pengujian menunjukan kerentanan pada tingkat ancaman tinggi (*high),* tanpa temuan, empat temuan menunjukan tingkat ancaman menengah *(medium)* dan tujuh temuan menunjukan tingkat ancaman rendah *(low).*Untuk mengatasi atau mencegah peretasan terhadap *website* Simantep.id maka digunakanlah metode *Open Web Application Security Project*.

2. TINJAUAN PUSTAKA

2.1. Keamanan Web

Keamanan suatu *website* atau web *security system* merupakan salah satu hal terpenting bagi seorang pengelola/ pengguna web. Kebanyakan pengguna hanya fokus pada desain tampilan dan konten untuk menarik pengguna. Jika salah satu pengelola atau jika pengguna mengabaikan keamanan *website*, pengguna sendiri yang dirugikan karena Seseorang dapat mengambil data penting dari situs web dan bahkan mengacak tampilannya (*deface*). Melindungi komputer, aplikasi, dan jaringan sebuah situs web adalah komponen utama keamanan situs web, karena ini memastikan bahwa data yang disimpan di dalamnya aman [5].

2.2. Open Web Aplplication Securty (OWASP)

OWASP, yang merupakan singkatan dari *Open Web Application Security Project*, adalah sebuah komunitas internasional yang berfokus pada keamanan aplikasi web. Tujuan utama dari OWASP adalah meningkatkan keamanan perangkat lunak dengan menyediakan sumber daya, pedoman, dan *tools* yang dapat digunakan oleh para pengembang, profesional keamanan, dan organisasi untuk mengidentifikasi kerentanan keamanan dalam aplikasi web. OWASP didirikan pada tahun 2001 dan telah tumbuh menjadi sebuah organisasi global yang terkenal. Mereka menyediakan berbagai macam proyek terbuka yang berkaitan dengan keamanan aplikasi web, termasuk daftar kerentanan yang paling umum ditemui (OWASP Top 10), pedoman pengembangan yang aman (*OWASP Secure Coding Practices*), serta *tools* dan metodologi pengujian keamanan.

Salah satu kontribusi terpenting dari OWASP adalah OWASP Top 10, yang merupakan daftar kerentanan keamanan aplikasi web yang paling sering ditemui dan berpotensi berbahaya. Daftar ini diperbarui secara berkala dan memberikan panduan tentang kerentanan keamanan aplikasi web. OWASP juga mengadakan konferensi dan acara lokal di berbagai negara untuk mempromosikan kesadaran keamanan aplikasi web dan berbagi pengetahuan tentang praktik terbaik dalam mengatasi kerentanan keamanan. Dengan menyediakan sumber daya dan *tools* yang diperlukan, OWASP berusaha untuk meningkatkan kesadaran dan keamanan aplikasi web di seluruh industri, membantu para pengembang dan organisasi dalam membangun perangkat lunak yang lebih aman [6].

OWASP Top 10 adalah sebuah daftar yang dibuat oleh komunitas OWASP yang berisi 10 daftar teratas kerentanan yang dapat mengancam keamanan suatu website. Bertujuan untuk membuat keamanan perangkat lunak terlihat oleh individu dan organisasi untuk membantu mereka mengambil keputusan yang tepat tentang risiko keamanan perangkat lunak mereka [7].

2.3. OWASP TOP 10

OWASP Top 10, juga dikenal sebagai OWASP 10, adalah daftar yang dibuat oleh komunitas OWASP yang mencakup sepuluh masalah keamanan utama yang dapat mengancam keamanan web. Daftar berikut diperbarui setiap beberapa tahun berdasarkan kombinasi data pengujian keamanan dan survei profesional dalam industri. Versi terbaru dari daftar OWASP TOP 10 dirilis pada tahun 2021. Sumber daya ini memberikan informasi tentang kerentanan atau celah keamanan yang paling umum, contoh setiap jenis, cara terbaik untuk mencegahnya, dan deskripsi tentang bagaimana kerentanan dapat dieksploitasi. OWASP Top 10 sendiri dibuat untuk meningkatkan pengetahuan masyarakat tentang keamanan aplikasi [6]. atau suatu *website* dengan menentukan bahaya celah keamanan yang sering terjadi atau terjadi dalam banyak kasus, seperti yang ditunjukkan dalam gambar di bawah ini.

Gambar 1. OWASP Top 10 2021 [6]

Berikut ini adalah penjelasan lebih lanjut dari masing-masing list baha ya keamanan website OWASP Top 10 2021 :

2.3.1. A1: *Broken Access Control*

Dalam daftar kelemahan umum (CWEs), *Broken Access Control* lebih sering terjadi pada aplikasi daripada kategori lainnya. Ketika autentikasi dan pembatasan akses tidak digunakan dengan benar, peretas dan penyerang dapat mengakses sistem. Dengan kata lain, pengendalian akses yang rusak memungkinkan akses yang tidak legal, yang mampu menyebabkan kerentanan terhadap *file* dan data sensitif. Metode *coding* tindakan khusus seperti menghapus akun manajemen dan akun pengguna *multi-factor authenthication* dapat membantu menghindari kontrol akses yang lemah terkait manajemen kredensial.

2.3.2. A2: *Cryptoraphic Failures*

Di sini, fokusnya adalah API atau *Application Programming Interface*, berfungsi untuk menghubungkan dan menyediakan layanan dari sumber eksternal. Dalam hal ini, Kebocoran data sensitif dan sistem yang telah terinfeksi oleh peretas adalah fokus utama kegagalan kriptografi, karena layanan dari pihak ketiga seperti *GMaps* dapat melakukan serangan dengan menggunakan data transmisi yang tidak aman. Tindakan seperti manajemen sistem yang baik dan enkripsi data dapat mengurangi risiko penyebaran data pribadi.

2.3.3. A3: *Injection*

Peretas dapat membuat kode yang tidak dilindungi dan lalu memasukkan kode yang mereka buat ke dalam program tertentu. Karena program yang terinjeksi seringkali tidak dapat menemukan data terinjeksi. Karena sistem akan memastikan bahwa mereka adalah pengguna yang dapat dipercaya, Penyalahgunaan system dapat menemukan area yang aman dan data rahasia. LDAP, CRLF, dan injeksi SQL adalah beberapa jenis injeksi. Pengujian OWASP membantu menemukan masalah injeksi dan menawarkan solusi.

2.3.4. A4: *Insecure Design*

OWASP menyediakan daftar bahaya yang terkait dengan desain yang tidak aman. Dalam survei 2021, ada pendekatan baru yang disebut *Insecure Design*. Terbukti bahwa kelemahan ini dapat diperbaiki dengan uji penetrasi. Selain menyediakan referensi arsitektur, Perusahaan harus memperluas penggunaan desain yang aman, pola, dan pemodelan ancaman.

2.3.5. A5: *Security Misconfiguration*

Dalam OWASP Top 10, *Security Misconfiguration* keamanan sangat penting karena dapat menunjukkan perubahan perangkat lunak yang dapat dikonfigurasi. Jenis tambahan seperti XML (XXE) termasuk di dalamnya. Kesalahan dalam konfigurasi kontrol akses hampir sama. Selain itu, bagian ini menangani kesalahan konfigurasi yang memiliki kemampuan untuk meningkatkan mengambil risiko dengan menyediakan akeses untuk penyerang memasuki sistem. Pengujian dinamis dapat membantu dalam menemukan kesalahan audit konfgurasi pada aplikasi yang digunakan untuk menyelesaikan permasalahan tersebut.

2.3.6. A6: *Vulnerable and Outdated Components*

Peretas memiliki kemampuan untuk memasuki dan mengedit kode. Ini mungkin disebabkan oleh komponen pihak ketiga dan ketergantungan yang tidak aman. Serangan jenis ini dapat diatasi dari dalam sistem melalui analisis komposisi perangka lunak. Analisis memungkinkan audit atau pemrogram untuk menemukan bagian yang tidak aman sebelum sistem merilis aplikasi.

2.3.7. A7: *Identification and Authentication Failures*

Kesalahan autentikasi dan penerapan autentikasi adalah salah satu dari faktor-faktor ini, juga menjalankan sesi dengan cara yang salah. Sangat berbahaya jika penyerang dapat menyalin peran sebagai pengguna resmi. *Multi-factor authentication* adalah metode penting untuk meminimalkan kelemahan dan kegagalan autentikasi. Ketika alat pemindai DAST dan SCA digunakan, mereka dapat menemukan dan menyelesaikan masalah seperti kesalahan implementasi yang terjadi pada saat program menjalankan kodenya sebelumnya.

2.3.8. A8: *Software and Data Integrity Failures*

Pada survei yang dilakukan pada tahun 2021, kategori baru yang berfokus pada pilihan pembaruan perangkat lunak adalah kegagalan integritas data dan software. *Pipeline CI/ CD* bersama dengan data penting. Kategori ini merupakan konsekuensi dari *Common Vulnerability and Exposures/Common Vulnerability Scoring System (CVE/CVSS)*, yang merupakan salah satu dampak dari sistem tersebut. Sejak survei tahun 2017, deserialisasi yang tidak aman termasuk dalam kategori ini.

Deserialisasi adalah pengambilan data, dokumen, atau item yang tersimpan pada *disk* yang dapat digunakan untuk menjalankan kode terbuka atau sistem kode untuk menghentikan serangan. Melalui penggunaan desain konvensional seperti JSON dan XML, item dapat distrukturkan atau dibinerkan. Kegagalan sering terjadi ketika pelaku menggunakan informasi yang tidak dapat diandalkan untuk memanfaatkan aplikasi terentu, mulai melakukan penolakan layanan untuk menggunakan kode yang tidak dapat diprediksi, atau kode yang tidak dapat diprediksi, untuk mengubah perilaku sistem. Baik pengujian penetrasi maupun penggunaan alat keamanan dapat mencegah serangan itu, meskipun deserialisasi adalah pekerjaan yang cukup besar untuk dilakukan. Kebanyakan kali, pengguna harus menghindari sumber dan objek serial yang gagal melindungi sistem dari serangan siber.

2.3.9. A9: *Security Logging and Monitoring Failures*

Kesalahan manusia dapat terancam oleh kegagalan login dan praktik pemantauan yang tidak memadai. Pelaku ancaman di seluruh dunia bergantung pada kurangnya pemantauan login; kegagalan dalam kontrol akses *login* dan validasi data server memungkinkan mereka untuk menemukan tindakan mencurigakan di dalam sistem. Selain itu, uji penetrasi dapat menemukan lokasi yang tidak memiliki login yang cukup.

2.3.10. A10: *Server-Side Request Forgery*

Ini adalah item baru di list yang fokus pada pengujian. *Forgery* dari pihak server berkatian dengan cakupan pemeriksaan di atas rata-rata untuk eksploitasi dan dampaknya. Pada tingkat ini, situasi di mana tim pengaman. Selain itu, perlu memastikan bahwa data tersebut adalah data penting. Untuk memenuhi kebutuhan yang beragam untuk keamanan aplikasi perusahaan, pembaruan rutin adalah upaya untuk membangun sistem yang berkelanjutan yang dapat membantu perusahaan membangun infrastruktur keamanan penting. Programmer, auditor, dan bahkan manajer program termasuk dalam kategori ini. Uji peentrasi aplikasi termasuk uji pengembangan dan fungsi. Aplikasi ini mudah digunakan dan menggabungkan berbagai pendekatan [6].

2.4. OWASPZap

OWASPZap (*Zed Attaack Proxy*) merupakan sebuah alat *open- socure* yang digunakan untuk melakukan pengujian keamanan apliaksi web. OWASPZap dikembangan oleh organisasi keamanan aplikasi web (OWASP) dan menyediakan berbagai fitur yang berguna dalam mengidentifikasi dan memperbaiki kerentanan keamanan dalam aplikasi web. OWASPZap digunakan untuk melakukan pemindaian keamanan otomatis pada aplikasi web, mengidentifikasi kerentanan yang mungkin ada seperti injeksi SQL, *cross-site scripting* (XSS), *cross-siste request forgery* (CSRF), dan banyak lagi.

Alat ini juga memungkinkan pengguna untuk melakukan serangan pengecoh (*fuzzing*) dan penelusuran sistem web secara manual. OWASPZap memiliki antarmuka pengguna grafis (GUI) yang intuitif, sehingga mudah digunakan oleh pengembang dan peneliti kemanan. Selain itu, OWASPZap juga mendukung otomatisasi melalui antarmuka baris perintah, yang memungkinkan intergrasi dengan proses pengujian keamanan yang ada. OWASPZap sangat populer di kalangan komunitas keamanan aplikasi web karena ketersediaan sumber terbuka, fleksibilitasnya, dan fokusnya pada keamanan aplikasi web, alat ini sering digunakan oleh pengembang, peneliti keamanan, dan profesional keamanan untuk membantu mengidentifikasi dan memperbaiki kerentanan pada sebuah *wesbite* [8].

3. METODE PENELITIAN

Berdasarkan Penelitian pada web Simantep.Id dengan menganalisis keamanan sistem website menggunakan metode *Open Web Application Security Project* (Owasp) terdapat beberapa tahapan, Metode untuk menganalisis data OWASP (Open Web Application Security Project) merupakan serangkaian pendekatan, dan proses yang digunakan untuk mengidentifikasi, mengkategorikan, dan menganalisis kerentanan keamanan aplikasi web. Dengan menerapkan metode ini dapat mengubah data yang dikumpulkan selama analisis menjadi informasi baru yang dapat membantu dalam pemahaman dan pengelolaan risiko keamanan aplikasi web.

Untuk mengolah data OWASP agar menjadi informasi baru, berikut ini merupakan beberapa langkah yang dapat dilakukan :

a. Pemahaman data
   Mengumpulkan data terkait aplikasi web yang akan dianalisis. Ini dapat mencakup hasil pengujian keamanan, catatan log, laporan keamanan, dan informasi lain yang relevan.
b. Identifikasi kerentanan
   Analisis data OWASP melibatkan identifikasi kerentanan keamanan yang terkait dengan aplikasi web.
c. Kategorisasi dan Prioritisasi
   Setelah mengidentifikasi kerentanan, langkah selanjutnya adalah mengkategorikan dan memprioritaskan kerentanan berdasarkan tingkat keparahannya.
d. Analisis Penyebab
   Menganalisis penyebab akar dari kerentanan yang ditemukan. Ini melibatkan mempelajari komponen atau kode yang terlibat dan mencari tahu mengapa kerentanan muncul.
e. Evaluasi Risiko
   Berdasarkan hasil analisis, dapat mengevaluasi risiko yang terkait dengan kerentanan dan menentukan dampak yang mungkin terjadi jika kerentanan tersebut dieksploitasi.
f. Tindakan Perbaikan
   Langkah terakhir dalam metode analisis data OWASP adalah mengambil tindakan perbaikan yang diperlukan untuk mengurangi risiko keamanan aplikasi web. Tindakan ini meliputi *patching*, pembaruan konfigurasi, perbaikan kode, atau peningkatan kebijakan keamanan.

4. HASIL DAN PEMBAHASAN

Alert Chart menggambarkan bahaya dari ancaman yang ditemukan melalui *scanning* menggunakan *tools* OWASPZap, seperti yang ditunjukkan pada tabel dibawah.

Berdasarkan tabel 1 diatas, menunjukkan grafik hasil *scanning* menggunakan aplikasi OWASPZap yang seberapa mungkin ada celah keamanan pada *website* target berdasarkan tingkat ancaman disini terbagi menjadi beberapa kategori berdasarkan

dampak yang ditimbulkan dari celah keamanan tersebut yaitu , *Medium* 3, *Low* 3, *Informational* 4.

Tabel 1. Alert Chart

Alert type	Risk	Count
Content Security Policy (CSP) Header Not Set	Medium	14 (140.0%)
Directory Browsing	Medium	10 (100.0%)
Missing Anti-clickjacking Header	Medium	2 (20.0%)
Cookie No HttpOnly Flag	Low	2 (20.0%)
Cross-Domain JavaScript Source File Inclusion	Low	4 (40.0%)
X-Content-Type-Options Header Missing	Low	15 (150.0%)
Information Disclosure - Suspicious Comments	Informational	1 (10.0%)
Modern Web Application	Informational	2 (20.0%)
Session Management Response Identified	Informational	7 (70.0%)
User Agent Fuzzer	Informational	170 (1,700.0%)
Total		10

Rangkuman kerentanan berdasarkan kategori OWASP Top 10, Hasil pemindaian kerentanan OWASP Top 10 yang berhasil ditemukan diuraikan dalam tabel 2 sebagai berikut :

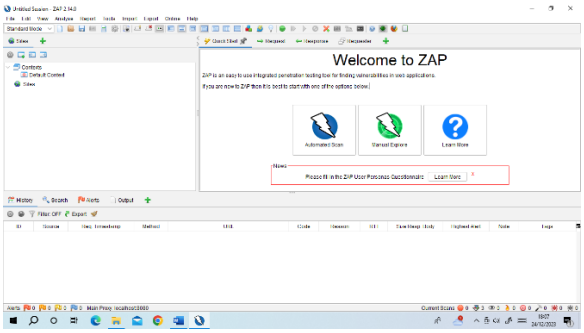
Tabel 2. OWASP Top 10

	OWASP Top 10	Kerentanan
A01	Broken Acces Control	Ditemukan
A02	Cryptographic Failures	Tidak Ditemukan
A03	Injection	Tidak Ditemukan
A04	Insecure Design	Ditemukan
A05	Security Misconfiguration	Ditemukan
A06	Vulnerable and Outdated Components	Tidak Ditemukan
A07	Identification and Authentication Failures	Tidak Ditemukan
A08	Software and Data Integrity Failures	Ditemukan
A09	Security Logging and Monitroing Failures	Tidak Ditemukan
A10	Server-Side Request Forgery (SSRF)	Tidak Ditemukan

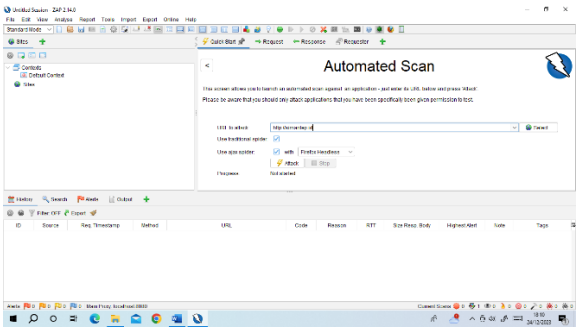
Berdasarkan tabel 2 terdapat kerentanan di A01 *Broken Access Control* dengan menggunakan *tools Active Scan Rules* memiliki tingkat kerentanan *Medium*, A04 *Insecure Design* dengan menggunakan *tools Active Scan Rules*, dengan tingkat kerentanan *Low*, A05 *Security Misconfiguration* dengan menggunakan *tools Access Control Testing* dengan tingkat kerentanan *Low*, A08 *Software and Data Integrity Failures* dengan menggunakan t*ools Active Scan Rules* dengan tingkat kerentanan *Low*.

Berikut ini merupakan tampilan *user interface* dari OWASPZap.

Sebelum memasukan url target, langkah pertama yang dilakukan ialah memilih pilihan *scanning, automated scan* atau *manually scan*.

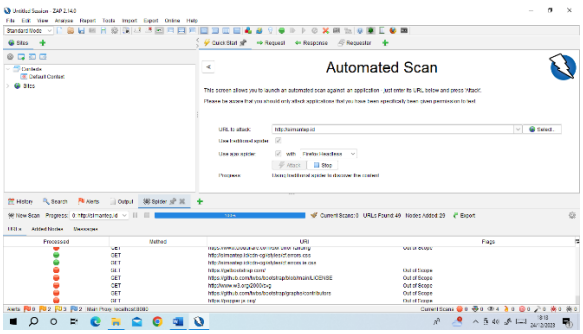


Gambar 2. *User Interface* Aplikasi OWASPZap

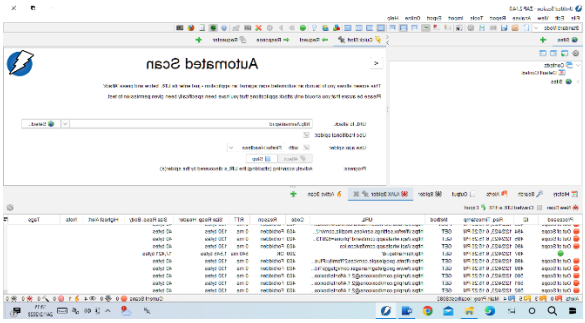


Gambar 3. URL Target

Setelah pengguna memasukkan *URL* tujuan, seperti gambar 3, langkah selanjutnya yang dilakukan yaitu melakukan *attack* atau proses *scanning* pada *website* tersebut. Setelah melakukan *attack*, aplikasi akan melakukan *scanning* pada *website* target (*crawling*) dan memulai proses *scanning* awal untuk menemukan semua indeks di *website* target, seperti yang ditunjukan dalam gambar dibawah ini :

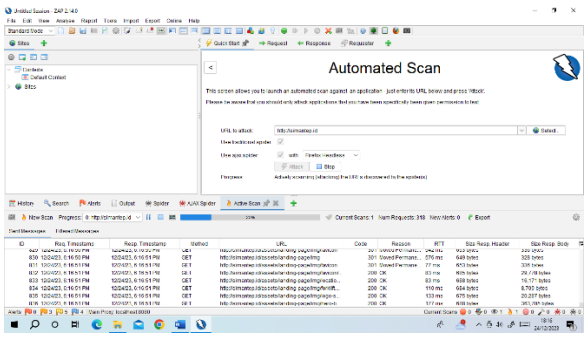


Gambar 4. Proses *SpiderScan tools* OWASPZap



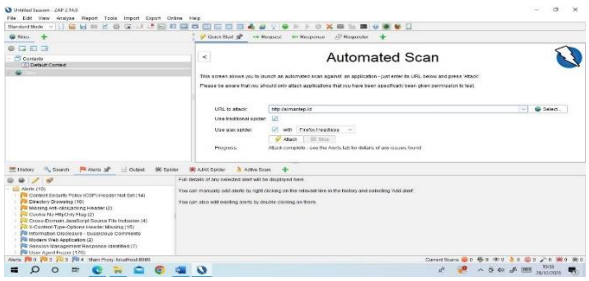
Gambar 5. Proses *AJAX SpiderScan tools* OWASPZap

Tahapan diatas merupakan sebuah proses dari *Spider proggres* atau pengidentifikasian semua index. Setelah menyelesaikan dan memperoleh indeks dari situs web tujuan, secara otomatis akan menuju tahap berikutnya yaitu *Active Scan* yang ditunjukkan pada gambar 6 tujuan dari langkah ini adalah untuk menemukan celah keamanan pada situs web target.



Gambar 6. Proses *Active Scan* aplikasi OWASPZap

Setelah tahap *Active Scan* selesai, aplikasi akan mengirimkan hasil scanning dari website tujuan setelah tahapan Active Scan selesai.



Gambar 7. Hasil dari proses *scanning*

Hasil dari proses *scanning* pada *website* Simantep.id menunjukkan adanya 10 tanda bahaya (*alerts*) dengan tingkat ancaman dari yang terendah (*Medium*) sampai dengan yang tertinggi (*Low*). dari celah keamanan tersebut yaitu , *Medium* 3, *Low* 3, *Informational* 4.

5. KESIMPULAN DAN SARAN

Dalam melakukan vulnerability assessment atau penilaian kerentanan celah keamanan pada website simantep.id menggunakan metode OWASP Top 10 tahun 2021 yang bertujuan untuk menguji tingkat keamanan. menunjukkan grafik hasil *scanning* menggunakan aplikasi OWASPZap yang seberapa mungkin ada celah keamanan pada *website* target berdasarkan tingkat ancaman disini terbagi menjadi beberapa kategori berdasarkan dampak yang ditimbulkan dari celah keamanan tersebut yaitu , *Medium* 3, *Low* 3, *Informational* 4. Metode OWASP Top 10 tahun 2021 masih ideal untuk digunakan sebagai dasar untuk melakukan pengujian keamanan website. Prinsip keamanan CIA (*Confidentiality, Integrity, Availability*) sudah diterapkan pada keamanan sistem pada situs web target dengan domain simantep.id. Hal ini dapat dilihat dari sejumlah keberhasilan yang dicapai dalam mengeksploitasi celah keamanan yang ada.

DAFTAR PUSTAKA

[1] C. A. Cholik, "Perkembangan Teknologi Informasi Komunikasi / Ict Dalam Berbagai Bidang," *Industry and Higher Education*, vol. 2, no. 1. pp. 1689–1699, 2021.

[2] A. S. Sinaga, "KEAMANAN KOMPUTER - Anita Sindar RM Sinaga - Google Buku." 2020.

[3] I. O. Riandhanu, "Analisis Metode Open Web Application Security Project (OWASP) Menggunakan Penetration Testing pada Keamanan Website Absensi," *Jurnal Informasi dan Teknologi*. 2022, doi: 10.37034/jidt.v4i3.236.

[4] K. Nisa, M. A. Putra, R. A. Siregar, and M. D. Irawan, "Analisis Website Tapanuli Tengah Menggunakan Metode Open Web Application Security Project Zap (Owasp Zap)," *Bulletin Of Information Technology (BIT)*, vol. 3, no. 4. pp. 308–316, 2022.

[5] A. W. Kuncoro and F. Rahma, "Analisis Metode Open Web Application Security Project (OWASP) pada Pengujian Keamanan Website: Literature Review," *Automata*, vol. 3, no. 1. pp. 1–5, 2021.

[6] "Beranda - OWASP Top 10 2021." .

[7] S. Hidayatulloh and D. Saptadiaji, "Penetration Testing pada Website Universitas ARS Menggunakan Open Web Application Security Project (OWASP)," *Jurnal Algoritma*, vol. 18, no. 1. pp. 77–86, 2021, doi: 10.33364/algoritma/v.18-1.827.

[8] The Open Web Application Security Project, "Zed Attack Proxy (ZAP)," *Www*.

[9] G. Guntoro, L. Costaner, and M. Musfawati, "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)*, vol. 5, no. 1. p. 45, 2020, doi: 10.29100/jipi.v5i1.1565.

[10] A. Rochman, R. Rohian Salam, and S. Agus Maulana, "Analisis Keamanan Website dengan Information System Security Assessment Framework (Issaf) dan Open Web Application Security Project (Owasp) di Rumah Sakit Xyz," *Jurnal Indonesia Sosial Teknologi*, vol. 2, no. 4. pp. 506–519, 2021, doi: 10.36418/jist.v2i4.124.