e-ISSN 2827-8356
p-ISSN 2827-9999

Analisis Keamanan Web Samsat Menggunakan Metode Owasp (Open Web Application Security Project)

Samsat web security analysis using the OWASP method

Zarifah Aina[1], Shafira Nur Rahmadiyah[2], Setti Manullang[3]
[1,2,3]Ilmu Komputer, Universitas Islam Negeri Sumatera Utara
E-mail: [1]zarifahaina1@gmail.com, [2]shafirarahmadiyah@gmail.com,
[3]settimanullang4o@gmail.com.

Abstrak

Terdapat situs web tempat masyarakat dapat berkomunikasi dan bertukar informasi antara pemerintah dan masyarakat umum. Website adalah suatu jenis media yang berisi informasi yang dapat diakses dari mana saja melalui internet dan dapat diakses dari mana saja di dunia. Pengujian sistem keamanan aplikasi berbasis website merupakan aspek krusial di era pengembangan aplikasi berbasis web yang pesat. Penelitian ini dilakukan pada website Samsat. Pemeliharaan website Samsat menjadi kendala bagi administrator. Permasalahan tersebut selalu terselesaikan, dan dapat diselesaikan ketika terjadi bencana. Tanpa sistem manajemen yang efektif, teknologi informasi apa pun akan mampu mendukung organisasi atau lembaga tertentu dengan sendirinya.. Berdasarkan latar belakang tersebut, maka dibutuhkan analisis dari website Samsat tersebut agar mengetahui apakah web tersebut telah aman. Metode penelitian menggunakan OWASP (*Open Web Application Security Project*).

Kata kunci : Keamanan, Analisis, Website

Abstract

There are websites where people can communicate and exchange information between the government and the general public. A website is a type of media that contains information that can be accessed from anywhere on the internet and is accessible from anywhere in the world. Pengujian sistem keamanan aplikasi berbasis website is a crucial aspect in the era of web-based application development that involves pesat. This study was conducted on the Samsat website. The Samsat website's maintenance is an issue for the administrator. These problems are always resolved, and they can be resolved when a bencana occurs. Without an effective management system, any information technology will be able to support a particular organization or institution on its own. Based on this background, an analysis of the Samsat website is needed to find out whether the website is safe. The research method uses OWASP (Open Web Application Security Project).

Keywords: Security, Analysis, Website

1. PENDAHULUAN

Di zaman digital seperti sekarang, proses pertukaran dan penyebaran informasi berlangsung dengan sangat cepat berkat kehadiran internet. Kemajuan teknologi internet telah menciptakan kemudahan dalam mengakses dan mendistribusikan berbagai informasi. Hal ini menjadikan informasi sebagai sumber daya yang memiliki nilai tinggi, bukan hanya untuk perorangan, melainkan juga untuk instansi pemerintahan dan perusahaan swasta[1]. Di tengah maraknya pengembangan aplikasi berbasis web saat ini, evaluasi aspek keamanan sistem menjadi suatu keharusan yang tidak bisa diabaikan. Perkembangan yang signifikan dalam penggunaan

e-ISSN 2827-8356
p-ISSN 2827-9999

aplikasi web telah memicu munculnya berbagai bentuk serangan siber yang semakin canggih. Sayangnya, perhatian terhadap aspek keamanan seringkali tidak dijadikan prioritas utama dan cenderung dikesampingkan dalam proses pengembangan. Seringkali masalah keamanan berada di urutan kedua, atau bahkan di urutan terakhir dalam daftar hal-hal yang dianggap penting [2]. Menurut Sarwono, website dapat didefinisikan sebagai sebuah platform digital yang memuat berbagai laman informasi, yang dapat diakses secara universal melalui jaringan internet. Pada dasarnya, sebuah website terdiri dari rangkaian kode pemrograman yang berisi instruksi-instruksi, yang kemudian diterjemahkan melalui sebuah browser [3].

Pemanfaatan situs web telah memfasilitasi interaksi dan pertukaran informasi antara instansi pemerintah dan publik melalui jaringan internet. Banyak manfaat dari website yang digunakan di bidang pemerintahan. Penggunaan platform digital ini memberikan beragam keuntungan bagi sektor pelayanan publik. Akan tetapi, implementasi website dalam mendukung aktivitas pelayanan Samsat Medan Selatan kepada masyarakat juga menghadirkan tantangan berupa potensi kejahatan siber yang membutuhkan penanganan secara cermat. Berbagai celah keamanan yang terdapat pada situs web dapat mengancam kerahasiaan data dan informasi penting milik instansi pemerintah, khususnya yang tersimpan di sistem Samsat Medan Selatan[4].

Layanan jaringan dan keamanan aplikasi merupakan elemen penting dalam memastikan integritas, kerahasiaan, dan ketersediaan data dan layanan di jaringan. Ketika ketergantungan pada teknologi informasi meningkat, ancaman terhadap sistem jaringan menjadi lebih beragam dan kompleks. Layanan jaringan dan keamanan aplikasi adalah praktik dan prosedur yang digunakan untuk melindungi jaringan komputer dari ancaman dan serangan yang merusak seperti virus, malware, peretas, dan peretasan data. Tujuan utama layanan jaringan dan keamanan aplikasi adalah untuk melindungi data dan informasi sensitif dari kehilangan dan akses tidak sah. Layanan jaringan dan keamanan aplikasi mencakup berbagai tindakan dan teknologi untuk mencegah, mengidentifikasi, dan merespons ancaman keamanan. Ini termasuk penggunaan firewall, VPN, dan teknologi keamanan jaringan lainnya. Ketika volume data dan kompleksitas aplikasi meningkat, jaringan menjadi target utama penjahat dunia maya. Serangan yang berhasil dapat mengakibatkan hilangnya data penting, kerugian finansial, dan kerusakan reputasi. Oleh karena itu, memastikan keamanan layanan jaringan dan aplikasi menjadi prioritas utama bagi organisasi atau perusahaan [5].

Samsat sendiri adalah singkatan dari Sistem Administrasi Manunggal Satu Atap (*One-stop Administration Services Office*), yaitu sistem administrasi yang dibentuk untuk memperlancar dan mempercepat pelayanan kepentingan masyarakat yang kegiatannya diselenggarakan dalam satu gedung. Di dalam Samsat, terdapat tiga instansi, yaitu Kepolisian Republik Indonesia (POLRI), Badan Pendapatan Daerah (Bapenda), dan PT Jasa Raharja. Dalam konteks evaluasi keamanan sistem berbasis web, Open Web Application Security Project (OWASP) Top 10 hadir sebagai salah satu pendekatan pengujian yang dapat diandalkan. Metodologi ini, yang dikembangkan oleh komunitas OWASP, menyajikan daftar sepuluh celah keamanan paling kritis yang berpotensi mengancam integritas platform website[6].

Dalam penelitian ini penulis ingin berfokus pada evaluasi aspek keamanan Website Samsat dengan mengimplementasikan metodologi Open Web Application Security Project (OWASP). Pemilihan kerangka kerja OWASP didasarkan pada track record keberhasilannya dalam mengidentifikasi berbagai celah atau kerentanan keamanan sistem melalui serangkaian pengujian komprehensif. Meski demikian, untuk mendapatkan hasil analisis yang optimal, diperlukan pertimbangan terhadap beberapa faktor selama proses pengujian sistem keamanan.

2. METODE PENELITIAN

2.1 OWASP (*Open Web Application Security Project*)

Proyek Keamanan Aplikasi Web Terbuka (OWASP) adalah kerangka kerja sumber terbuka yang berfokus pada peningkatan keamanan perangkat lunak aplikasi. Open Web Application Security Project (OWASP) merupakan framework terbuka yang mengutamakan

e-ISSN 2827-8356
p-ISSN 2827-9999

peningkatan aspek keamanan pengembangan perangkat lunak aplikasi. OWASP adalah organisasi yang didirikan untuk menemukan kerentanan keamanan pada aplikasi situs web [7]. OWASP Top 10, atau yang biasa disebut OWASP10, adalah daftar yang diterbitkan oleh komunitas OWASP yang berisi daftar 10 kerentanan keamanan teratas yang dapat membahayakan keamanan sebuah situs web. Daftar ini terus berkembang dan berubah seiring berkembangnya teknologi situs web. OWASP Top 10 pertama kali diterbitkan pada tahun 2003, dengan pembaruan kecil berikutnya pada tahun 2004, 2007, 2010, dan 2017 (OWASP, 2018). Tujuan utama pembentukan OWASP Top 10 adalah untuk mengkategorikan berbagai kerentanan yang sering terjadi atau sering terjadi dalam insiden keamanan sistem, dan untuk membantu pengembang dan pengguna memahami pentingnya keamanan aplikasi [8].

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

2.1.1: Broken Access Control

Dalam daftar kerentanan umum (CWE), kontrol akses yang rusak lebih sering muncul di aplikasi dibandingkan kategori lainnya. Jika otentikasi dan pembatasan akses tidak digunakan dengan benar, peretas dan penyerang dapat memperoleh akses ke sistem Anda. Dengan kata lain, kontrol akses yang rusak dapat memungkinkan akses tidak sah dan menciptakan kerentanan terhadap file dan data sensitif. Cara mengkodekan tindakan khusus, seperti menghapus akun administratif atau akun pengguna dengan autentikasi multifaktor, dapat membantu Anda menghindari kontrol akses yang lemah terkait manajemen kredensial.

2.2.2: Cryptoraphic Failures

Fokusnya di sini adalah pada API atau antarmuka pemrograman aplikasi yang digunakan untuk terhubung ke sumber eksternal dan menyediakan layanan. Dalam hal ini, kebocoran data sensitif dan sistem yang terinfeksi peretas menjadi fokus utama kegagalan kriptografi, karena layanan pihak ketiga seperti GMaps dapat melakukan serangan dengan transmisi data yang tidak aman. Risiko berbagi data pribadi dapat dikurangi dengan mengambil langkah-langkah seperti manajemen sistem yang tepat dan enkripsi data.

2.3.3: Injection

Peretas dapat membuat kode yang tidak dilindungi dan memasukkannya ke dalam program tertentu. Karena program yang diinjeksi seringkali tidak dapat menemukan data yang diinjeksi. Karena sistem memastikan bahwa pengguna dipercaya, pelaku pelanggaran sistem dapat menemukan area aman dan data sensitif. LDAP, CRLF, dan injeksi SQL adalah jenis injeksi. Tes OWASP membantu menemukan masalah injeksi dan memberikan solusi.

2.4.4: Insecure Design

OWASP menyediakan daftar bahaya yang terkait dengan desain yang tidak aman. Studi tahun 2021 memiliki pendekatan baru yang disebut Insecure Design. Kerentanan ini telah terbukti dapat diatasi melalui pengujian penetrasi. Selain menyediakan referensi arsitektural, organisasi harus memperluas penggunaan desain, pola, dan pemodelan ancaman yang aman.

2.5.5: Security Misconfiguration

Kesalahan konfigurasi keamanan sangat penting dalam OWASP Top 10 karena dapat menunjukkan perubahan pada perangkat lunak yang dapat dikonfigurasi. Termasuk tipe tambahan seperti XML (XXE). Kesalahan konfigurasi kontrol akses hampir sama. Selain itu, bagian ini membahas kesalahan konfigurasi yang dapat meningkatkan risiko Anda dengan mengizinkan penyerang mengakses sistem Anda. Pengujian dinamis membantu menemukan kesalahan pemeriksaan konfigurasi di aplikasi Anda, yang digunakan untuk mengatasi masalah ini.

2.6.6: Vulnerable and Outdated Components

Peretas dapat memasukkan dan mengedit kode. Hal ini mungkin disebabkan oleh komponen pihak ketiga atau ketergantungan yang tidak aman.Jenis serangan ini dapat diatasi secara internal dengan menganalisis konfigurasi perangkat lunak. Melalui analisis, pemeriksa dan pemrogram dapat menemukan bagian sistem yang tidak aman sebelum aplikasi dirilis.

2.7.7: Identification and Authentication Failures

e-ISSN 2827-8356
p-ISSN 2827-9999

Kegagalan otentikasi dan implementasi otentikasi adalah salah satu faktor ini, seperti halnya eksekusi sesi yang salah. Sangat berbahaya jika penyerang dapat menyalin peran pengguna yang berwenang. Otentikasi multifaktor adalah cara penting untuk meminimalkan kelemahan dan kesalahan otentikasi. Alat pemindaian DAST dan SCA dapat digunakan untuk menemukan dan memperbaiki masalah seperti kesalahan implementasi yang terjadi saat program sebelumnya mengeksekusi kode.

2.8. 8: Software and Data Integrity Failures

Pada survei tahun 2021, kesalahan integritas data dan perangkat lunak menjadi kategori baru yang berfokus pada opsi pembaruan perangkat lunak. Saluran pipa CI/CD dan data penting. Kategori ini merupakan hasil dari Common Vulnerability and Exposures/Common Vulnerability Scoring System "CVE/CVSS" dan merupakan salah satu dampak dari sistem ini. Pada studi tahun 2017, deserialisasi tidak aman termasuk dalam kategori ini. Deserialisasi adalah pengambilan data, dokumen, atau item yang disimpan di disk yang dapat digunakan untuk menjalankan kode terbuka atau sistem kode untuk menggagalkan serangan. Elemen dapat terstruktur atau biner menggunakan desain tradisional seperti JSON atau XML. Bug terjadi ketika penyerang mengeksploitasi aplikasi tertentu dengan menggunakan informasi yang tidak dapat diandalkan untuk memulai penolakan layanan, menggunakan kode yang tidak dapat diprediksi, atau mengubah perilaku sistem. Baik pengujian penetrasi maupun penggunaan alat keamanan dapat mencegah serangan tersebut, namun deserialisasi merupakan upaya yang signifikan. Dalam kebanyakan kasus, pengguna harus menghindari sumber dan objek serial yang tidak melindungi sistem mereka dari serangan cyber.

2.9.9: Security Logging and Monitoring Failures

Ada risiko kesalahan manusia karena kegagalan login atau praktik pemantauan yang buruk. Penyerang di seluruh dunia mengandalkan kurangnya pemantauan login. Kontrol akses login dan kesalahan validasi data server dapat mendeteksi tindakan mencurigakan di sistem Anda. Selain itu, pengujian penetrasi dapat membantu Anda menemukan tempat di mana login tidak mencukupi.

2.10.10: Server-Side Request Forgery

Ini adalah item baru dalam daftar yang berfokus pada pengujian. Pemalsuan sisi server dikaitkan dengan cakupan pemeriksaan eksploitasi dan dampaknya di atas rata-rata. Level ini adalah situasi tim keamanan. Selain itu, Anda perlu memastikan data itu penting. Untuk memenuhi beragam kebutuhan keamanan aplikasi perusahaan, pembaruan rutin merupakan upaya untuk membangun sistem berkelanjutan yang membantu perusahaan membangun infrastruktur keamanan penting. Pemrogram, auditor, dan bahkan manajer program termasuk dalam kategori ini. Pengujian penetrasi aplikasi meliputi pengujian pengembangan dan pengujian fungsional. Aplikasi ini mudah digunakan dan menggabungkan berbagai pendekatan[9].

Pengujian keamanan sistem informasi khususnya aplikasi website bertujuan untuk menemukan kerentanan dan kesalahan pada sistem serta memberikan solusi dan penanggulangan yang dapat dilakukan untuk memberikan keamanan dan kenyamanan kepada pengguna sistem. Audit keamanan suatu sistem informasi tidak hanya memberikan hasil mengenai kerentanan keamanan saja, namun juga memberikan pemahaman mengenai ruang lingkup sistem yang dibuat. Pengujian dilakukan dengan menggunakan prosedur pengujian penetrasi berdasarkan kerangka kerja tertentu [10].

e-ISSN 2827-8356
p-ISSN 2827-9999

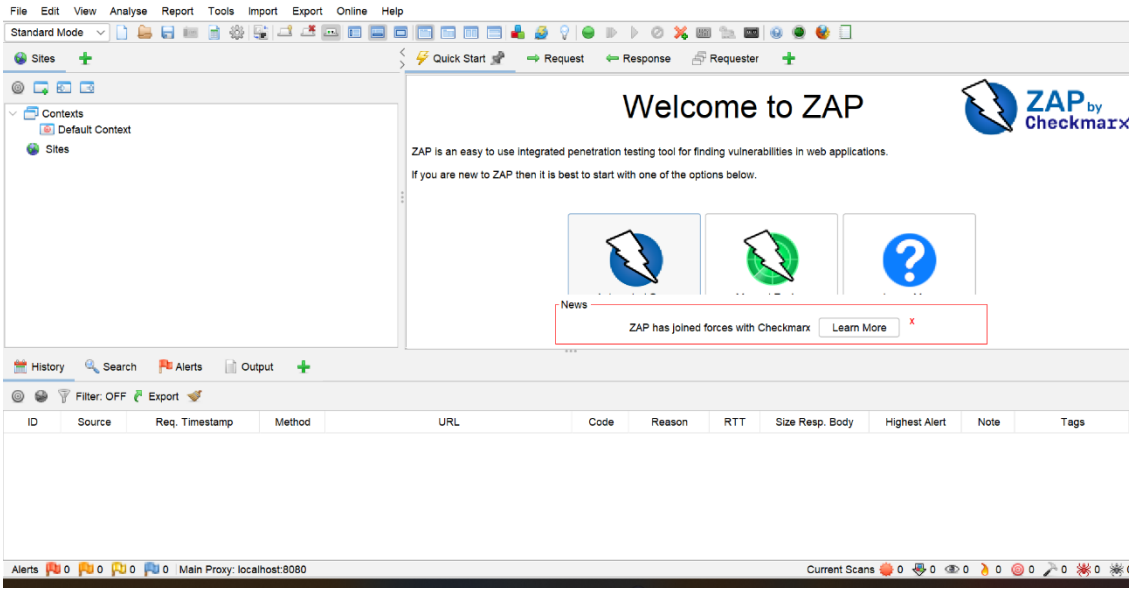
Gambar 2.1 OWASP Top 10

3. HASIL DAN PEMBAHASAN

Saat menggunakan alat OWASP ZAP, bagan peringatan memberikan gambaran ambang batas ancaman risiko yang terdeteksi pada saat itu. Potensi kerentanan keamanan pada situs web target ditunjukkan pada diagram ini, yang dikategorikan berdasarkan sensitivitas redaman yang ditunjukkan oleh kerentanan tersebut di atas. Studi ini mengidentifikasi dua ambang batas yang signifikan dan, jika dieksploitasi, dapat mengakibatkan kesalahan signifikan pada sistem atau data. Selain itu, terdapat juga ancaman tingkat menengah yang meskipun tidak sepenting ancaman tingkat tinggi, namun memerlukan perhatian untuk memastikan eksploitasi yang lebih luas. Pada tingkat rendah, kami menemukan tujuh ancaman yang umumnya berisiko rendah namun harus dimitigasi untuk mencegah potensi risiko kumulatif. Saat ini terdapat lima temuan bermanfaat yang memberikan rincian tambahan tentang potensi kerentanan, namun tidak dianggap sebagai ancaman langsung.

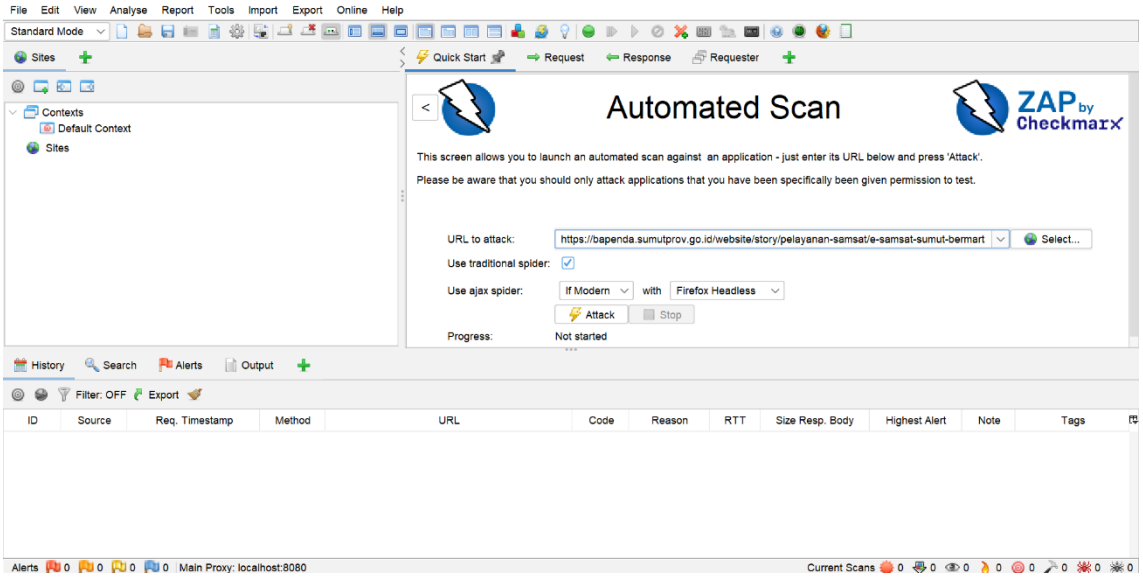
Berikut ini merupakan tampilan user interface dari OWASPZap. Sebelum memasukan url target, langkah pertama yang dilakukan ialah memilih pilihan scanning, automated scan atau manually scan.

e-ISSN 2827-8356
p-ISSN 2827-9999



Gambar 3.1 User Interface Aplikasi OWASPZap

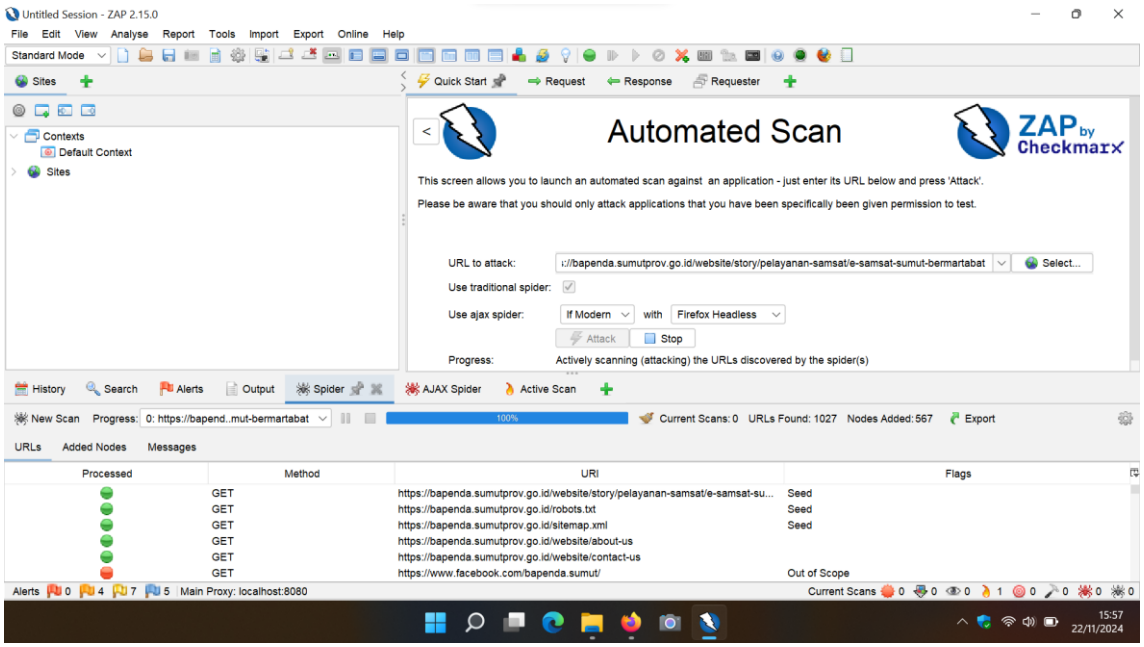
User interface (UI) OWASP Zap merupakan tampilan antarmuka yang memungkinkan pengguna berinteraksi dengan aplikasi ini. UI-nya terdiri dari beberapa elemen utama seperti menu bar, toolbar, panel-panel kerja, dan area hasil scanning.



Gambar 3.2 URL Target

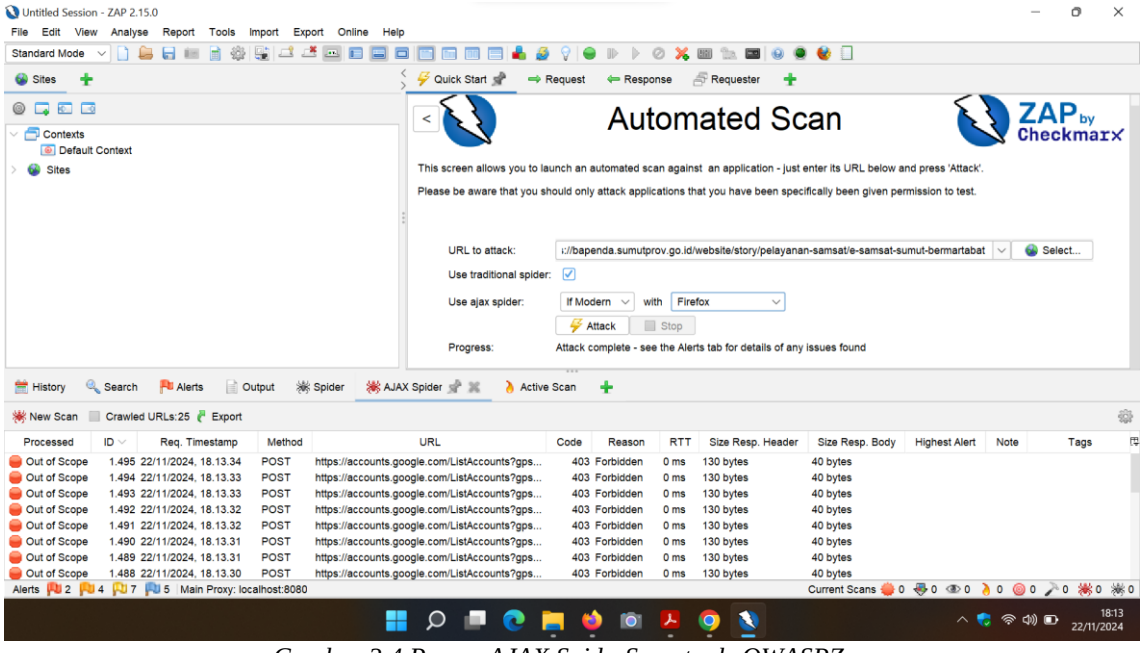
Setelah pengguna memasukkan *URL* tujuan, seperti gambar 3, langkah selanjutnya yang dilakukan yaitu melakukan *attack* atau proses *scanning* pada *website* tersebut. Setelah melakukan *attack*, aplikasi akan melakukan *scanning* pada *website* target (*crawling*) dan memulai proses *scanning* awal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

e-ISSN 2827-8356
p-ISSN 2827-9999



Gambar 3.3 Proses SpiderScan tools OWASPZap

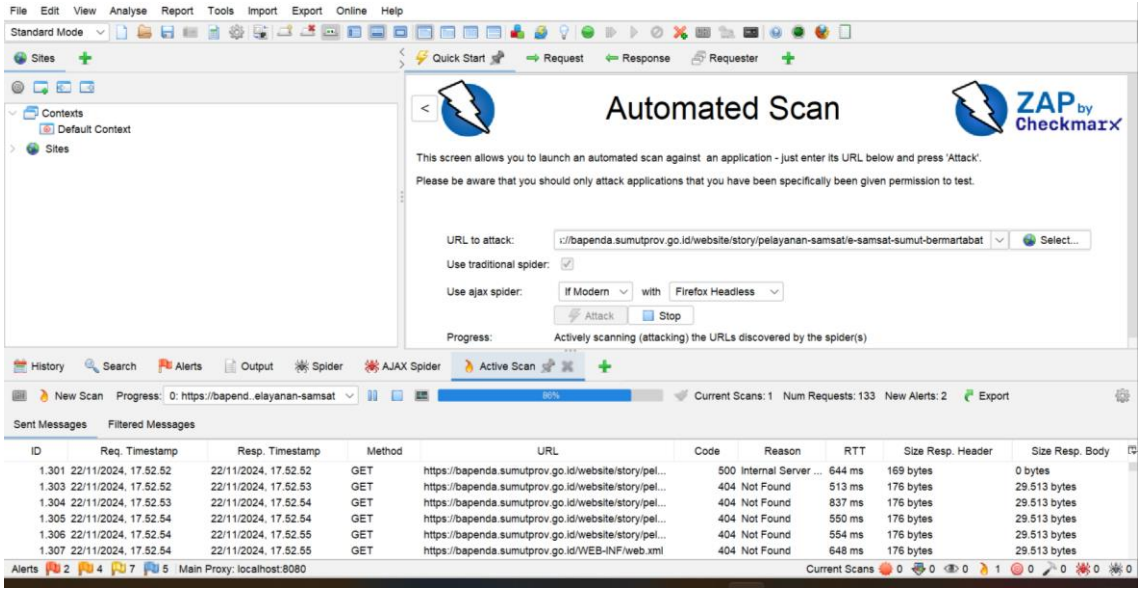
SpiderScan merupakan fitur di OWASP Zap yang bekerja seperti laba-laba menjelajahi website. Prosesnya dimulai dari satu URL, kemudian secara otomatis mengumpulkan informasi tentang struktur dan konten yang ada di dalamnya.



Gambar 3.4 Proses AJAX SpiderScan tools OWASPZap

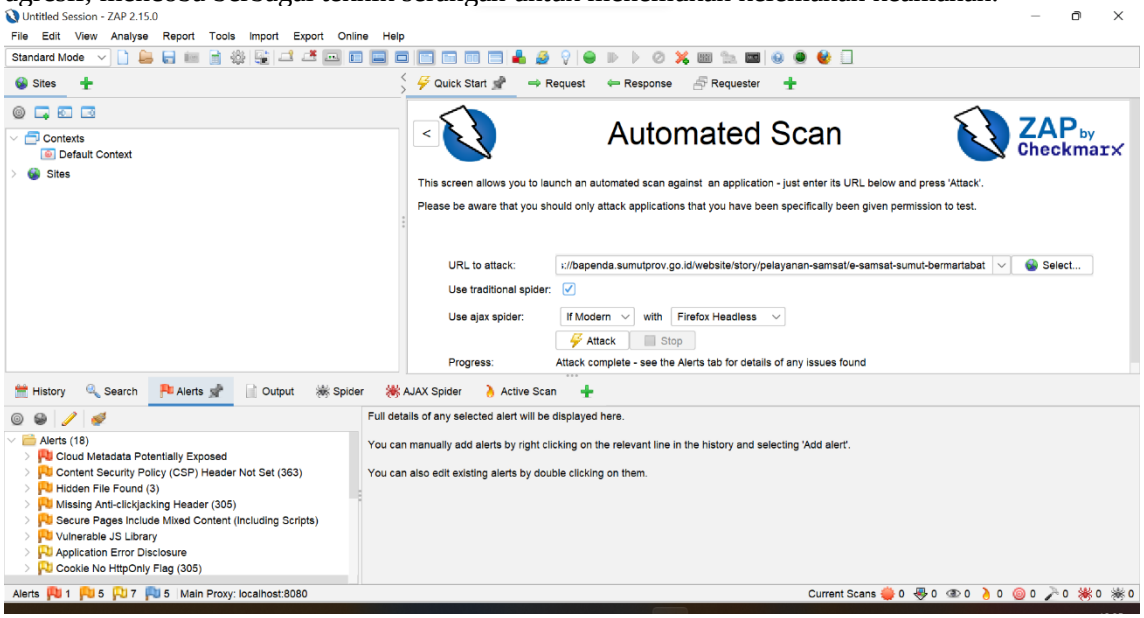
Ajax SpiderScan merupakan fitur khusus di owasp zap yang dirancang untuk menjelajahi konten website dinamis yang menggunakan teknologi AJAX (Asynchronous JavaScript and XML). Ajax dapat mendeteksi dan memproses konten yang dimuat secara dinamis melalui JavaScript, seperti popup, menu dropdown, menu yang muncul setelah diklik atau konten yang diperbarui tanpa reload halaman.

e-ISSN 2827-8356
p-ISSN 2827-9999



Gambar 3.5 Proses Active Scan aplikasi OWASPZap

Active scan merupakan fitur pengujian keamanan otomatis di OWASPZap yang secara aktif mencoba menemukan kerentanan keamanan pada website target dengan permintaan yang dimodifikasi secara aktif ke server target. Active scan bekerja dengan menguji website secara agresif, mencoba berbagai teknik serangan untuk menemukan kelemahan keamanan.



Gambar 3.6 Hasil dari proses scanning

Hasil dari proses *scanning* pada *website* samsat menunjukkan adanya 18 tanda bahaya (*alerts*) dengan tingkat ancaman dari yang terendah (*Medium*) sampai dengan yang tertinggi (*Low*). dari celah keamanan tersebut yaitu , *Hight 2 Medium* 4, *Low* 7, *Informational* 5.

Berdasarkan hasil pengujian yang dilakukan pada website samsat melalui Vulnerability Scanner tools owasp-zap didapatkan 18 kerentanan yaitu:

Tabel 1. Hasil Vulnerability Assessment

No	Celah Keamanan	Tingkat Ancaman
1	SQL Injection	Tinggi

e-ISSN 2827-8356
p-ISSN 2827-9999

No	Celah Keamanan	Tingkat Ancaman
2	Cross-Site Scripting (XSS)	Tinggi
3	Cross-Site Request Forgery (CSRF)	Menengah
4	Broken Authentication	Menengah
5	Sensitive Data Exposure	Menengah
6	Insecure APIs	Menengah
7	Buffer Overflow	Rendah
8	Security Misconfiguration	Rendah
9	Weak Password Policies	Rendah
10	Missing Security Headers	Rendah
11	Malicious File Upload	Rendah
12	Outdated Software Components	Rendah
13	Hardcoded Secrets in Code	Rendah
14	Denial of Service (DoS)	Informatif
15	Improper Access Control	Informatif
16	Default Credentials	Informatif
17	Insecure Deserialization	Informatif
18	Insufficient Logging and Monitoring	Informatif

Hasil pengujian dengan OWASP-ZAP pada website SAMSAT mengidentifikasi 18 kerentanan, termasuk SQL Injection dan XSS (ancaman tinggi), serta CSRF dan Broken Authentication (ancaman menengah). Beberapa kerentanan rendah seperti Weak Password Policies juga terdeteksi. Disarankan untuk segera memperbaiki celah berisiko tinggi, meningkatkan konfigurasi keamanan, dan melakukan pemantauan secara rutin.

4. KESIMPULAN DAN SARAN

Berdasarkan dua kali pengujian yang telah dilakukan dengan menggunakan tools OWASP ZAP dalam website SAMSAT, teridentifikasi total 18 celah keamanan. Celah-celah ini terdiri dari 2 tingkat ancaman tinggi, 4 tingkat ancaman menengah, 7 tingkat ancaman rendah rendah, dan 5 ancaman informatif. Yang disebut Tingkat Ancaman Tinggi menunjukkan adanya potensi kerentanan serius yang dapat mengakibatkan kegagalan sistem atau kehilangan data jika dieksploitasi. Untuk itu, perlu segera dilakukan pemeriksaan tingginya tingkat keamanan. yang ditemukan, implementasi pembaruan berkala dan pemantauan keamanan sistem secara rutin untuk mencegah celah keamanan baru, serta melakukan pengujian keamanan secara berkala menggunakan metode OWASP untuk memastikan keamanan situs web minimum.

DAFTAR PUSTAKA

[1] G. Pramudya and A. Mukti, “Analisis Kerentanan Keamanan Website Dengan Menggunakan Metode Penetration Testing dan Framework OWASP (Studi Kasus : Website Instansi XYZ),” 2020.

[2] B. Ghozali, K. Kusrini, and S. Sudarmawan, “Mendeteksi Kerentanan Keamanan Aplikasi Website Menggunakan Metode Owasp (Open Web Application Security Project) Untuk Penilaian Risk Rating,” *Creat. Inf. Technol. J.*, vol. 4, no. 4, p. 264, 2019, doi: 10.24076/citec.2017v4i4.119.

e-ISSN 2827-8356
p-ISSN 2827-9999

[3] T. Susilawati, F. Yuliansyah, M. Romzi, and R. Aryani, "Membangun Website Toko Online Pempek Nthree Menggunakan Php Dan Mysql," *J. Tek. Inform. Mahakarya*, vol. 3, no. 1, pp. 35–44, 2020.

[4] I. O. Riandhanu, "Analisis Metode Open Web Application Security Project (OWASP) Menggunakan Penetration Testing pada Keamanan Website Absensi," *J. Inf. dan Teknol.*, vol. 4, no. 3, pp. 160–165, 2022, doi: 10.37034/jidt.v4i3.236.

[5] R. Raharja, *Keamanan jaringan*. Jawa Timur: KBM Indonesia, 2024.

[6] Y. Yudiana, A. Elanda, and R. L. Buana, "Analisis Kualitas Keamanan Sistem Informasi E-Office Berbasis Website Pada STMIK Rosma Dengan Menggunakan OWASP Top 10," *CESS (Journal Comput. Eng. Syst. Sci.*, vol. 6, no. 2, p. 185, 2021, doi: 10.24114/cess.v6i2.24777.

[7] 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 Ilm. Penelit. dan Pembelajaran Inform.*, vol. 5, no. 1, p. 45, 2020, doi: 10.29100/jipi.v5i1.1565.

[8] A. Dharmawan, Y. Prihati, and H. Listijo, "Penetration testing menggunakan OWASP top 10 pada domain xyz.ac.id," *Jelc*, vol. 8, no. 1, pp. 1–9, 2022.

[9] E. Nurelasari and D. Gumilang Al Farabi, "Analisis Keamanan Sistem Website Menggunakan Metode Open Web Application Security Project (Owasp) Pada Simantep.Id," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 8, no. 3, pp. 3049–3054, 2024, doi: 10.36040/jati.v8i3.9314.

[10] 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, [Online]. Available: https://www.sciencedirect.com