Jurnal Minfo Polgan (JMP)
Volume 15, Nomor 2, Mei-Aagustus 2026
DOI : https://doi.org/10.33395/jmp.v15i2.16236
e-ISSN : 2797-3298
p-ISSN : 2089-9424

Vulnerability Assessment Website Gratis Kabeh Banyumas Using ISSAF and OWASP ZAP

[1]Nur Wahid, [2*]Mohammad Imron, [3]Darso
[1,2*, 3]Ilmu Komputer, Universitas Amikom Purwokerto, Purwokerto, Indonesia
*Korespondensi: imron@amikompurwokerto.ac.id

Submit : 18 Mei 2026 | **Diterima** : 10 Jun 2026 | **Terbit** : 20 Jun 2026

ABSTRACT

Digital transformation in the public sector is driving government agencies to provide web-based services to improve public access to services. However, this also significantly increases cybersecurity risks, particularly for government websites that handle sensitive user data. This study aims to analyze security vulnerabilities on the Gratis Kabeh Banyumas website using the Information Systems Security Assessment Framework (ISSAF) and the OWASP ZAP tool. The research methodology includes the planning, information gathering, vulnerability scanning, exploitation testing, risk level analysis, and reporting stages. OWASP ZAP was used to perform automated scans to identify potential vulnerabilities, which were then validated through direct attack simulations such as Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), and Clickjacking. The research results indicate the presence of security vulnerabilities, predominantly classified as Low and Medium risk, particularly in security configuration weaknesses such as the absence of a Content Security Policy (CSP) and Anti-CSRF Token. These vulnerabilities have the potential to threaten the confidentiality, integrity, and availability of the system if exploited by malicious actors. Based on these analysis results, it can be concluded that the system still requires fundamental security improvements. The implications of this research are the need for technical improvement recommendations such as input validation, the implementation of secure coding, and periodic security testing. This research is expected to increase awareness of website security in digital public services. For future research, it is recommended to use a combination of other security tools and expand the research scope to obtain more comprehensive and accurate results.

Keywords: *ISSAF, OWASP ZAP, Public Service, Vulnerability Assessment, Website Security.*

ABSTRAK

Transformasi digital di sektor publik mendorong instansi pemerintah untuk menyediakan layanan berbasis website guna meningkatkan aksesibilitas pelayanan masyarakat. Namun, hal ini juga meningkatkan risiko keamanan siber secara signifikan, terutama pada website pemerintah yang mengelola data sensitif pengguna. Penelitian ini bertujuan untuk menganalisis kerentanan keamanan pada website Gratis Kabeh Banyumas menggunakan metode Information Systems Security Assessment Framework (ISSAF) dan tools OWASP ZAP. Metode penelitian meliputi tahap perencanaan, pengumpulan informasi, pemindaian kerentanan, pengujian eksploitasi, analisis tingkat risiko, dan pelaporan. OWASP ZAP digunakan untuk melakukan pemindaian otomatis guna mengidentifikasi potensi kerentanan, yang kemudian divalidasi melalui simulasi serangan langsung seperti Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), dan Clickjacking. Hasil penelitian menunjukkan adanya celah keamanan yang didominasi oleh tingkat risiko Low dan Medium, khususnya pada kelemahan konfigurasi keamanan seperti ketiadaan Content Security Policy (CSP) dan Anti-CSRF Token. Kerentanan ini berpotensi mengancam kerahasiaan, integritas, dan ketersediaan sistem jika dieksploitasi oleh pihak tidak bertanggung jawab. Berdasarkan hasil analisis tersebut, dapat disimpulkan bahwa sistem masih membutuhkan peningkatan keamanan mendasar. Implikasi dari penelitian ini adalah perlunya rekomendasi perbaikan teknis seperti validasi input, penerapan secure coding, serta pengujian keamanan secara berkala. Penelitian ini diharapkan dapat meningkatkan kesadaran keamanan website pada layanan publik digital. Untuk penelitian selanjutnya, disarankan agar menggunakan kombinasi tools keamanan lainnya serta memperluas objek penelitian agar memperoleh hasil yang lebih komprehensif dan akurat.

Kata Kunci: ISSAF, OWASP ZAP, Public Service, Vulnerability Assessment, Website Security.

PENDAHULUAN

Pemanfaatan teknologi digital pada instansi pemerintah terus meningkat untuk mendukung pelayanan publik yang lebih cepat, transparan, dan mudah diakses masyarakat. Transformasi digital ini menjadi prioritas strategis di Indonesia, terutama sejak percepatan digitalisasi pada masa pandemi COVID-19. Namun, di balik kemudahan tersebut, peningkatan penggunaan teknologi digital diiringi dengan meningkatnya ancaman keamanan siber yang menargetkan sistem informasi pemerintah (Aslan et al., 2023). Aplikasi layanan publik berbasis web memiliki risiko keamanan yang tinggi karena mengelola berbagai data penting milik pengguna.

Berbagai serangan seperti SQL Injection, Cross-Site Scripting (XSS), dan Cross-Site Request Forgery (CSRF) kerap terjadi akibat kesalahan konfigurasi keamanan. Kegagalan dalam menjaga aspek Confidentiality, Integrity, dan Availability (CIA) dapat menurunkan tingkat kepercayaan masyarakat dan menyebabkan kerugian besar. Oleh karena itu, pendekatan vulnerability assessment sangat diperlukan untuk mengidentifikasi, mengukur, dan memprioritaskan celah keamanan dalam suatu sistem informasi secara preventif (Alhamed & Rahman, 2023).

Website Gratis Kabeh Banyumas yang dikelola oleh Dinas Kependudukan dan Pencatatan Sipil sejak tahun 2022 memfasilitasi layanan administrasi krusial seperti pembuatan akta dan Kartu Keluarga (KK). Walaupun website ini menangani data sensitif dan pernah mengalami insiden perubahan skrip (script tampering), sistem ini belum pernah menjalani pengujian keamanan siber secara komprehensif, melainkan hanya sebatas uji kelayakan aplikasi.

Untuk mendukung urgensi evaluasi pada sistem tersebut, terdapat beberapa penelitian terdahulu yang menjadi acuan (Ashari et al., 2023) dalam penelitiannya menunjukkan bahwa penggunaan tools OWASP ZAP sangat efektif dalam mendeteksi dan mengotomatisasi pemindaian kerentanan pada aplikasi web. Selanjutnya, (Khan et al., 2023)menjelaskan bahwa Information Systems Security Assessment Framework (ISSAF) mampu memberikan pendekatan metodologi yang sistematis dan komprehensif dalam proses penetration testing. Sementara itu, (Darojat et al., 2022). menyoroti bahwa sebagian besar pengujian kerentanan web masih terbatas pada tahap identifikasi (pemindaian) tanpa disertai pengujian eksploitasi langsung, sehingga belum memberikan gambaran nyata mengenai tingkat risiko spesifik yang dihadapi sistem.

Berdasarkan permasalahan dan celah dari penelitian-penelitian terdahulu tersebut, penelitian ini bertujuan untuk menganalisis kerentanan keamanan pada website Gratis Kabeh Banyumas menggunakan metode ISSAF dan tools OWASP ZAP. Nilai tambah (novelty) dari penelitian ini adalah dilakukannya pengujian eksploitasi secara langsung terhadap kerentanan yang ditemukan melalui simulasi serangan. Penelitian ini diharapkan dapat memberikan analisis dampak yang lebih mendalam, rekomendasi perbaikan yang komprehensif, serta berkontribusi dalam meningkatkan keamanan layanan publik digital.

METODE PENELITIAN

Metode yang diterapkan dalam penelitian ini berfokus pada pengujian kerentanan sistem untuk mengetahui potensi kelemahan keamanan website pada website Gratis Kabeh Banyumas. Pendekatan ini merupakan metode sistematis yang digunakan untuk menemukan serta mengevaluasi kelemahan pada sistem informasi guna meminimalkan potensi eksploitasi oleh pihak tidak bertanggung jawab (Muharrom & Saktiansyah, 2023).

Metode yang digunakan dalam penelitian ini mengacu pada Information Systems Security Assessment Framework (ISSAF), yang menyediakan tahapan pengujian keamanan secara terstruktur mulai dari tahap perencanaan hingga pelaporan hasil. ISSAF dikenal sebagai salah satu framework yang mampu memberikan pendekatan komprehensif dalam melakukan evaluasi keamanan sistem informasi (Sanjaya et al., 2023). Sehingga ISSAF dipilih karena menyediakan tahapan penetration testing yang lebih lengkap mulai dari information gathering hingga reporting dibanding framework lainnya.

Pendekatan pengujian yang digunakan adalah black-box testing, yaitu metode pengujian yang dilakukan tanpa mengetahui struktur internal sistem, sehingga pengujian difokuskan pada interaksi pengguna dengan sistem untuk mengidentifikasi potensi kerentanan

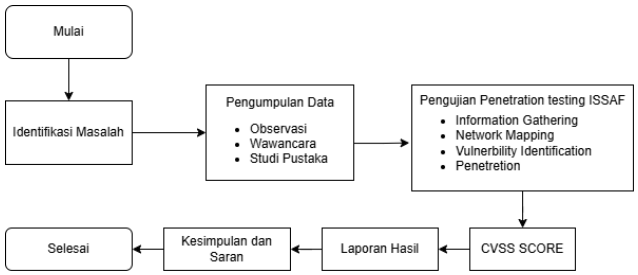
yang dapat dieksploitasi (Cholifah et al., 2022).

Pada tahap awal, dilakukan proses information gathering untuk mengumpulkan informasi terkait sistem target, seperti domain, alamat IP, teknologi yang digunakan, struktur endpoint, serta mekanisme request dan response pada aplikasi. Tahap pengumpulan informasi dasar ini dilakukan menggunakan tools seperti Ping, Whois, dan WhatWeb. Selanjutnya, untuk memetakan jaringan dan mendeteksi layanan yang berjalan, digunakan Nmap dan Nikto. Secara keseluruhan, proses identifikasi hingga pengujian eksploitasi didukung oleh penggunaan OWASP ZAP untuk pemindaian kerentanan secara otomatis dan Burp Suite untuk analisis lanjutan (Malik et al., 2022).

Pengujian dilakukan menggunakan tools OWASP ZAP untuk mendeteksi kerentanan aplikasi web melalui teknik automated scanning, yang mencakup proses spidering, passive scanning, dan active scanning. Selain itu, dilakukan pengujian lanjutan melalui simulasi serangan seperti Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), dan Clickjacking untuk mengetahui dampak nyata dari kerentanan yang ditemukan. (Albahli & M. Melad, 2022).

Tahapan penelitian meliputi identifikasi masalah, pengumpulan data, proses pengujian, analisis hasil, serta penyusunan laporan dan kesimpulan. Alur penelitian dalam studi ini disusun secara sistematis untuk menggambarkan tahapan yang dilakukan selama proses penelitian dari awal hingga akhir (Tobi & Kampen, 2024).

Alur Penelitian



Gambar 1. Alur Penelitian

Berdasarkan Gambar 1, penelitian dimulai dari tahap identifikasi masalah untuk menentukan fokus penelitian serta ruang lingkup pengujian. Selanjutnya dilakukan pengumpulan data melalui observasi, wawancara, dan studi pustaka guna memperoleh informasi terkait sistem yang akan diuji.

Tahap berikutnya adalah pengujian keamanan menggunakan metode Information Systems Security Assessment Framework (ISSAF) yang meliputi 4 proses pengujian meliputi Information Gathering, Network Mapping, Vulnerability Scanning, dan Penetration. Pada tahap information gathering menggunakan tools seperti Ping, Whois, dan WhatWeb untuk mengumpulkan data awal sistem. Selanjutnya, Network Mapping dilakukan menggunakan Nmap dan Nikto guna mengidentifikasi port yang terbuka serta mendeteksi kelemahan pada server web. Setelah itu, proses vulnerability scanning dijalankan menggunakan OWASP ZAP untuk mendeteksi potensi kerentanan pada aplikasi web secara otomatis. Terakhir, penetration dilakukan memanfaatkan Burp Suite untuk analisis dan simulasi serangan. Keseluruhan proses ini merupakan bagian penting dalam penetration testing karena membantu dalam mengidentifikasi titik lemah sistem secara sistematis (Bella et al., 2023).

Selain itu, dilakukan pengujian lanjutan melalui simulasi serangan seperti Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), dan Clickjacking untuk mengevaluasi dampak nyata dari kerentanan yang ditemukan terhadap sistem. Hasil pengujian kemudian dianalisis menggunakan metode Common Vulnerability Scoring System (CVSS) untuk menentukan tingkat risiko setiap kerentanan secara kuantitatif (Saha et al., 2023).

Tahap akhir adalah penyusunan laporan hasil pengujian yang berisi temuan kerentanan, tingkat risiko, serta rekomendasi perbaikan. Proses ini diakhiri dengan penarikan kesimpulan dan pemberian saran sebagai bentuk evaluasi terhadap keamanan sistem yang diuji.

Metode Pengujian Keamanan (ISSAF)

Tahapan pengujian dalam penelitian ini mengacu pada konsep penetration testing yang diadaptasi dari kerangka kerja Information Systems Security Assessment Framework (ISSAF)

(Ferrag et al., 2023). Proses pengujian disusun secara sistematis untuk mengidentifikasi, menguji, serta menganalisis kerentanan keamanan pada aplikasi web. Adapun tahapan yang dilakukan meliputi:

1. Perencanaan

Tahap awal penelitian difokuskan pada penentuan ruang lingkup, target sistem, serta mekanisme pengujian yang akan digunakan mengidentifikasi target sistem, serta menetapkan metode dan perangkat yang digunakan. Pada tahap ini juga dilakukan penentuan batasan pengujian guna memastikan proses pengujian berjalan sesuai dengan etika dan tujuan penelitian. Tools yang digunakan dalam penelitian ini antara lain OWASP ZAP untuk pemindaian kerentanan dan Burp Suite untuk analisis serta pengujian eksploitasi (Albahli & M. Melad, 2022).

2. Pengumpulan Informasi

Tahap pengumpulan informasi dilakukan untuk mengumpulkan data awal mengenai sistem target sebagai dasar dalam proses pengujian keamanan. Informasi yang dikumpulkan meliputi domain, alamat IP, struktur endpoint, teknologi yang digunakan, serta mekanisme komunikasi antara client dan server. Proses ini dilakukan dengan pendekatan pasif dan aktif. Pendekatan pasif dilakukan tanpa interaksi langsung yang signifikan dengan sistem target, sedangkan pendekatan aktif dilakukan dengan melakukan permintaan (request) untuk memperoleh informasi teknis.

Pada tahap information gathering, digunakan beberapa tools seperti Ping untuk mengetahui konektivitas dan respons server, Whois untuk memperoleh informasi registrasi domain, serta WhatWeb untuk mengidentifikasi teknologi yang digunakan pada website. Tahap ini bertujuan untuk memperoleh gambaran awal mengenai sistem target sehingga dapat membantu dalam menentukan strategi pengujian keamanan pada tahap selanjutnya. Informasi yang diperoleh meliputi domain, layanan server, teknologi web, dan konfigurasi dasar sistem yang berpotensi menjadi titik awal identifikasi kerentanan (Deepa et al., 2022).

3. Pemindaian Kerentanan

Tahap pemetaan jaringan dilakukan untuk mengidentifikasi struktur jaringan serta layanan yang berjalan pada sistem target. Proses ini bertujuan untuk mengetahui port yang terbuka, service yang aktif, serta potensi titik masuk (entry point) yang dapat dimanfaatkan dalam proses pengujian keamanan. Pada tahap ini dilakukan proses pemindaian terhadap sistem target untuk memperoleh informasi mengenai konfigurasi jaringan dan layanan yang tersedia. Informasi yang diperoleh meliputi nomor port, jenis layanan (service), serta versi perangkat lunak yang digunakan. Data tersebut sangat penting untuk menentukan kemungkinan celah keamanan yang dapat dieksploitasi pada tahap selanjutnya.

Pada tahap network mapping, digunakan tools Nmap untuk melakukan pemindaian port serta identifikasi layanan (service detection) yang berjalan pada sistem, sedangkan Nikto digunakan untuk mendeteksi kelemahan pada server web seperti konfigurasi yang tidak aman dan komponen yang rentan. Tahap ini bertujuan untuk memperoleh gambaran mengenai attack surface sistem, yaitu seluruh titik yang berpotensi menjadi jalur masuk bagi penyerang. Informasi yang diperoleh kemudian digunakan sebagai dasar dalam proses identifikasi kerentanan dan pengujian eksploitasi pada tahap selanjutnya (Y. He & P. De Roure, 2023).

4. Pengujian Eksploitasi

Tahap ini merupakan proses pengujian lanjutan terhadap kerentanan yang telah teridentifikasi pada tahap sebelumnya. Pengujian dilakukan dengan mensimulasikan serangan nyata menggunakan pendekatan black- box testing, di mana penguji diasumsikan tidak memiliki akses terhadap struktur internal sistem. Pendekatan ini bertujuan untuk mengevaluasi keamanan sistem dari sudut pandang eksternal serta didukung oleh penelitian terkait penetration testing yang menekankan evaluasi keamanan dari perspektif penyerang (Alkhurayyif & Saad Almarshdy, 2024).

Fokus pengujian eksploitasi dalam penelitian ini mencakup beberapa jenis kerentanan utama pada aplikasi web. Pengujian Cross-Site Scripting (XSS) dilakukan melalui penyisipan payload pada input pengguna untuk mengetahui apakah aplikasi telah menerapkan validasi dan sanitasi input dengan baik. Selanjutnya, pengujian Cross-Site Request Forgery (CSRF) dilakukan dengan memanipulasi request tanpa mekanisme validasi token keamanan guna mengidentifikasi kemungkinan penyalahgunaan sesi pengguna. Selain itu, dilakukan pula pengujian Clickjacking dengan mencoba memuat halaman website ke dalam elemen iframe untuk mengetahui apakah sistem telah menerapkan perlindungan keamanan antarmuka pengguna, seperti penggunaan header keamanan X-Frame-Options atau Content Security Policy (CSP).

Pengujian eksploitasi ini bertujuan untuk mengevaluasi sejauh mana kerentanan yang ditemukan dapat dimanfaatkan serta untuk mengukur dampak yang ditimbulkan terhadap aspek keamanan sistem, seperti kerahasiaan, integritas, dan ketersediaan data, yang merupakan prinsip utama dalam keamanan informasi serta sejalan dengan klasifikasi kerentanan pada OWASP Top 10 dan penelitian terkait penetration testing yang menekankan pentingnya evaluasi dampak kerentanan terhadap sistem.

5. Pelaporan

Tahap pelaporan merupakan fase akhir dalam proses pengujian yang bertujuan untuk mendokumentasikan seluruh hasil pengujian yang telah dilakukan. Pada tahap ini, hasil pengujian dianalisis untuk menentukan tingkat risiko setiap kerentanan dengan mengacu pada standar Common Vulnerability Scoring System (CVSS) yang digunakan untuk mengukur tingkat keparahan kerentanan, serta diklasifikasikan berdasarkan kategori kerentanan pada OWASP Top 10 sebagai acuan umum dalam keamanan aplikasi web. Pendekatan ini juga sejalan dengan penelitian sebelumnya yang menyatakan bahwa analisis kerentanan menggunakan CVSS dan klasifikasi OWASP dapat membantu dalam menentukan prioritas penanganan risiko keamanan secara efektif (Lashkari et al., 2022).

Pada tahap ini, hasil pengujian dianalisis menggunakan metode Common Vulnerability Scoring System (CVSS) untuk menentukan tingkat risiko kerentanan. Penilaian CVSS memberikan skor numerik yang dikategorikan menjadi tingkat risiko tertentu, seperti critical, high, medium, dan low (Mohamad Noor & N. Ithnin, 2022). Selain itu, hasil pengujian juga diklasifikasikan berdasarkan standar OWASP Top 10 sebagai acuan dalam mengidentifikasi jenis kerentanan pada aplikasi web (M. Alenezi & K. Almustafa, 2023).

Tabel 1. Klasifikasi tingkat risiko berdasarkan CVSS

CVSS Score	Risk Level
9.0–10.0	Critical
7.0–8.9	High
4.0–6.9	Medium
0.1–3.9	Low
0.0	None

Berdasarkan Tabel 1, klasifikasi tingkat risiko kerentanan ditentukan berdasarkan rentang nilai CVSS yang diperoleh. Semakin tinggi nilai CVSS, maka semakin besar potensi ancaman serius terhadap aspek CIA (Confidentiality, Integrity, Availability). Kategori Critical dan High memerlukan tindakan mitigasi segera, sedangkan kategori Medium dan Low tetap memerlukan perhatian untuk mencegah celah tersebut menjadi jalur eksploitasi lebih lanjut.

Sementara itu, kerentanan dengan kategori medium dan low tetap perlu diperhatikan karena dapat menjadi celah yang dimanfaatkan dalam kondisi tertentu (Holm et al., 2023). Oleh karena itu, klasifikasi ini digunakan sebagai dasar dalam menentukan prioritas penanganan kerentanan agar proses mitigasi dapat dilakukan secara efektif dan tepat sasaran. Selain menggunakan CVSS, kerentanan yang ditemukan juga diklasifikasikan berdasarkan kategori OWASP Top 10 untuk mengidentifikasi jenis dan karakteristik kerentanan pada aplikasi web (Albahli & M. Melad, 2022). Berikut adalah daftar sepuluh kategori kerentanan yang disajikan pada Tabel 2.

Tabel 2. Daftar kategori kerentanan OWASP Top 10 (2021)

Code	Vulnerability Categories
A01:2021	Broken Access Control
A02:2021	Cryptographic Failures
A03:2021	Injection
A04:2021	Insecure Design
A05:2021	Security Misconfiguration
A06:2021	Vulnerable and Outdated Components
A07:2021	Identification and Authentication Failures
A08:2021	Software and Data Integrity Failures

Code	Vulnerability Categories
A09:2021	Security Logging and Monitoring Failures
A10:2021	Server-Side Request Forgery (SSRF)

Selain menggunakan metrik risiko CVSS, penelitian ini juga mengklasifikasikan temuan berdasarkan sepuluh kategori kerentanan pada OWASP Top 10 (2021) seperti yang disajikan pada Tabel 2. Standar ini digunakan sebagai acuan umum untuk mengidentifikasi karakteristik dan jenis kelemahan pada aplikasi web secara global, sehingga mempermudah proses pemetaan kerentanan sesuai dengan standar industri keamanan siber saat ini.

Tabel 3. Analisis Hasil Pengujian Kerentanan

No	Kerentanan	Kategori OWASP	Tingkat Risiko (CVSS)	Dampak	Rekomendasi
1	Cross-Site Scripting (XSS)	A03: Injection	Tinggi	Eksekusi script berbahaya pada sisi klien, pencurian data pengguna, dan manipulasi tampilan	Melakukan validasi dan sanitasi input serta menerapkan output encoding
2	Cross-Site Request Forgery (CSRF)	A05: Security Misconfiguration	Kritis	Perubahan data tanpa sepengetahuan pengguna dan potensi pengambilalihan akun	Mengimplementasikan CSRF token serta melakukan validasi terhadap request
3	Clickjacking	A05: Security Misconfiguration	Medium	Manipulasi interaksi pengguna	Menambahkan X-Frame- Options atau CSP

Berdasarkan Tabel 3, kerentanan Cross-Site Scripting (XSS) termasuk dalam kategori Injection dengan tingkat risiko tinggi, karena memungkinkan penyerang untuk mengeksekusi script berbahaya pada sisi klien (OWASP, 2023). Sementara itu, kerentanan Cross-Site Request Forgery (CSRF) termasuk dalam kategori Security Misconfiguration dengan tingkat risiko kritis, karena memungkinkan penyerang melakukan manipulasi data tanpa sepengetahuan pengguna (OWASP, 2023). Oleh karena itu, kedua kerentanan tersebut memerlukan penanganan yang serius melalui penerapan mekanisme keamanan yang tepat untuk mengurangi potensi risiko terhadap sistem.

HASIL DAN PEMBAHASAN

Bagian ini menyajikan hasil pengujian keamanan yang telah dilakukan serta pembahasan terhadap kerentanan yang ditemukan pada website Gratis Kabeh Banyumas. Hasil yang diperoleh berasal dari proses vulnerability scanning menggunakan OWASP ZAP serta pengujian lanjutan melalui simulasi serangan untuk mengetahui dampak nyata terhadap sistem.

Perencanaan

Tahap perencanaan dilakukan untuk menentukan ruang lingkup pengujian, metode yang digunakan, serta tools yang akan digunakan dalam proses pengujian keamanan. Pendekatan yang digunakan adalah black-box testing, yaitu pengujian yang dilakukan tanpa mengetahui struktur internal sistem. Dalam penelitian ini, penerapan metode black-box testing dilakukan dengan cara melakukan pengujian langsung terhadap fitur yang tersedia pada aplikasi dari sisi pengguna, seperti form input dan endpoint sistem, tanpa mengakses kode sumber maupun konfigurasi internal. Pengujian dilakukan melalui simulasi serangan seperti Cross-Site Scripting (XSS) dan Cross-Site Request Forgery (CSRF) untuk mengidentifikasi kerentanan yang dapat dieksploitasi oleh pihak eksternal.

Selain itu, hasil pengujian yang diperoleh akan dianalisis menggunakan metode Common Vulnerability Scoring System (CVSS) untuk menentukan tingkat risiko dari setiap kerentanan yang ditemukan, sehingga dapat digunakan sebagai dasar dalam menentukan prioritas penanganan keamanan sistem. Tahap perencanaan juga mencakup kesepakatan

mengenai Rules of Engagement (RoE) yang didapatkan melalui proses wawancara dengan pihak pengelola. Pihak Dinas Kependudukan dan Pencatatan Sipil memberikan izin penuh untuk aktivitas pengujian tanpa batasan teknis khusus, dengan syarat utama bahwa pengujian tersebut tidak merusak fungsionalitas sistem. Selain itu, untuk mencegah terjadinya gangguan pada pelayanan administrasi masyarakat, disepakati bahwa aktivitas pemindaian dan eksploitasi direkomendasikan untuk dieksekusi pada malam hari atau saat hari libur operasional.

Pengumpulan Informasi dan Pemindaian Kerentanan

Tahap ini bertujuan untuk mengidentifikasi potensi kerentanan pada sistem melalui proses information gathering dan pemindaian menggunakan tools OWASP ZAP. Proses information gathering dilakukan untuk memahami struktur aplikasi, termasuk endpoint, parameter input, serta mekanisme request dan response yang digunakan. Informasi ini menjadi dasar dalam menentukan titik-titik yang berpotensi menjadi celah keamanan.

Langkah pertama dalam information gathering adalah melakukan pengujian konektivitas menggunakan perintah ping untuk memastikan server target aktif, memberikan respon dengan baik, serta untuk mengetahui alamat IP dari sistem target. Berikut adalah hasil pengujian ping pada Gambar 2.

```
root@LAPTOP-H3NMOLOU:~# ping gratiskabeh.banyumaskab.go.id
PING gratiskabeh.banyumaskab.go.id (103.105.190.138) 56(84) bytes of data.
64 bytes from 103.105.190.138: icmp_seq=1 ttl=51 time=30.3 ms
64 bytes from 103.105.190.138: icmp_seq=2 ttl=51 time=30.0 ms
64 bytes from 103.105.190.138: icmp_seq=3 ttl=51 time=28.9 ms
64 bytes from 103.105.190.138: icmp_seq=4 ttl=51 time=29.4 ms
64 bytes from 103.105.190.138: icmp_seq=5 ttl=51 time=29.8 ms
^C
--- gratiskabeh.banyumaskab.go.id ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 15525ms
rtt min/avg/max/mdev = 28.934/29.672/30.269/0.466 ms
root@LAPTOP-H3NMOLOU:~#
```

Gambar 2. Pengujian konektivitas server menggunakan Ping

Setelah server dipastikan aktif, proses pengumpulan informasi dilanjutkan menggunakan tools Whois dan WhatWeb untuk mendapatkan data registrasi domain dan teknologi server pada website target.

```
nur_wahid@LAPTOP-H3NMOLOU:~$ whois banyumaskab.go.id
Domain Name: BANYUMASKAB.GO.ID
Registry Domain ID: 42141_DOMAIN_ID-ID
Registrar WHOIS Server:
Registrar URL: domain.go.id
Updated Date: 2025-11-15T07:45:33Z
Creation Date: 2007-04-17T13:23:29Z
Registry Expiry Date: 2027-02-01T23:59:59Z
Registrar: Kementerian Komunikasi dan Digital Republik Indonesia
Registrar IANA ID: 1
Registrar Abuse Contact Email: helpdeskdomain@mail.komdigi.go.id
Registrar Abuse Contact Phone:
Domain Status: ok
Name Server: NS1.BANYUMASKAB.GO.ID
Name Server: NS2.BANYUMASKAB.GO.ID
Name Server: NS3.BANYUMASKAB.GO.ID
DNSSEC: unsigned
URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of WHOIS database: 2026-05-07T11:07:39Z <<<
```

Gambar 3. Pengumpulan Informasi Menggunakan Whois

```
nur_wahid@LAPTOP-H3NMOLOU:~$ whatweb banyumaskab.go.id
http://banyumaskab.go.id [301 Moved Permanently] Apache, HTTPServer[
Apache], IP[103.105.190.24], RedirectLocation[https://banyumaskab.go
.id/], Title[301 Moved Permanently]
https://banyumaskab.go.id/ [200 OK] Apache, CodeIgniter-PHP-Framewor
k, Cookies[ci_session], HTML5, HTTPServer[Apache], HttpOnly[ci_sessi
on], IP[103.105.190.24], Script, Title[Pemerintah Kabupaten Banyumas
], UncommonHeaders[content-security-policy]
```

Gambar 4. Identifikasi Teknologi Menggunakan WhatWeb

Setelah proses pengumpulan informasi awal, langkah selanjutnya adalah melakukan network mapping untuk mengetahui port yang terbuka, layanan yang aktif, serta potensi kelemahan awal pada server web. Berikut adalah hasil pemindaian menggunakan tools Nmap dan Nikto.

```
root@LAPTOP-H3NMOLOU:~# nmap -sV gratiskabeh.banyumaskab.go.id
Starting Nmap 7.80 ( https://nmap.org ) at 2025-11-02 17:23 WIB
Nmap scan report for gratiskabeh.banyumaskab.go.id (103.105.190.138)
Host is up (0.059s latency).
Other addresses for gratiskabeh.banyumaskab.go.id (not scanned): 64:ff9b::6769:be8a
Not shown: 986 filtered ports
PORT     STATE  SERVICE     VERSION
21/tcp   open   ftp         vsftpd 3.0.3
22/tcp   open   ssh         OpenSSH 8.0 (protocol 2.0)
25/tcp   open   tcpwrapped
80/tcp   open   http        nginx 1.14.1
110/tcp  open   tcpwrapped
113/tcp  closed ident
143/tcp  open   imap?
443/tcp  open   ssl/http    nginx 1.14.1
445/tcp  open   netbios-ssn Samba smbd 4.6.2
2000/tcp open   cisco-sccp?
3306/tcp open   tcpwrapped
5060/tcp open   sip?
8010/tcp open   ssl/xmpp?
9090/tcp closed zeus-admin
```

Gambar 5. Pemetaan Jaringan Menggunakan Nmap

```
nur_wahid@LAPTOP-H3NMOLOU:~$ nikto banyumaskab.go.id
- Nikto v2.1.5
---------------------------------------------------------------------------
+ ERROR: No host specified

       -config+            Use this config file
       -Display+           Turn on/off display outputs
       -dbcheck            check database and other key files for syntax errors
       -Format+            save file (-o) format
       -Help               Extended help information
       -host+              target host
       -id+                Host authentication to use, format is id:pass or id:pass:realm
       -list-plugins       List all available plugins
       -output+            Write output to this file
       -nossl              Disables using SSL
       -no404              Disables 404 checks
       -Plugins+           List of plugins to run (default: ALL)
       -port+              Port to use (default 80)
       -root+              Prepend root value to all requests, format is /directory
       -ssl                Force ssl mode on port
       -Tuning+            Scan tuning
       -timeout+           Timeout for requests (default 10 seconds)
       -update             Update databases and plugins from CIRT.net
       -Version            Print plugin and database versions
       -vhost+             Virtual host (for Host header)
                + requires a value

        Note: This is the short help output. Use -H for full help text.
```

Gambar 6. Deteksi Kelemahan Server Menggunakan Nikto

Selain menggunakan tools pemindaian, proses *information gathering* juga divalidasi melalui wawancara dengan pengelola sistem. Dari segi infrastruktur pertahanan awal, diketahui bahwa website telah diintegrasikan dengan protokol SSL dan sistem firewall yang difasilitasi oleh instansi terkait (Kominfo). Untuk menjamin ketersediaan data (*Availability*) apabila terjadi insiden siber yang tidak diinginkan, sistem pengarsipan didukung oleh dua server cadangan (*backup*). Informasi ini menjadi konteks penting dalam mengevaluasi apakah kerentanan yang ditemukan nantinya mampu menembus lapis pertahanan yang telah disiapkan tersebut.

Selanjutnya, pemindaian (*vulnerability scanning*) dilakukan menggunakan OWASP ZAP untuk mendeteksi kelemahan pada konfigurasi keamanan dan pengelolaan request secara otomatis. Hasil pemindaian menunjukkan bahwa website Gratis Kabeh Banyumas masih memiliki beberapa kerentanan dengan tingkat risiko *Low* dan *Medium*.

Kerentanan yang teridentifikasi meliputi *Absence of Anti-CSRF Tokens*, *Content Security Policy (CSP) Header Not Set*, *Missing Anti-clickjacking Header*, serta kelemahan pada pengaturan cookie dan server disclosure. Temuan ini menunjukkan bahwa sistem belum sepenuhnya menerapkan mekanisme keamanan berbasis header dan perlindungan terhadap manipulasi *request*.

Kondisi tersebut mengindikasikan adanya potensi eksploitasi oleh pihak tidak bertanggung jawab, khususnya pada serangan yang memanfaatkan kelemahan konfigurasi keamanan dan pengelolaan sesi. Oleh karena itu, diperlukan pengujian lanjutan untuk mengevaluasi sejauh mana kerentanan tersebut dapat dimanfaatkan serta dampaknya terhadap keamanan sistem. Tahap ini bertujuan untuk mengidentifikasi potensi kerentanan pada sistem melalui proses information gathering dan pemindaian menggunakan tools keamanan, yang

merupakan bagian penting dalam metode penetration testing (OWASP, 2023).

Tabel 4. Hasil Vulnerability Assessment Website

No	Kerentanan	Risiko	Confidence
1	Absence of Anti-CSRF Tokens	Medium	Low
2	Content Security Policy (CSP) Header Not Set	Medium	High
3	Cross-Domain Misconfiguration	Medium	Medium
4	HTTP to HTTPS Insecure Transition in Form Post	Medium	Medium
5	Missing Anti-clickjacking Header	Medium	Medium
6	Vulnerable JS Library	Medium	Medium
7	Cookie Without Secure Flag	Low	Medium
8	Cookie Without SameSite Attribute	Low	Medium
9	Cross-Domain JavaScript Source File Inclusion	Low	Medium
10	Server Leaks Information via "X-Powered-By" HTTP Response Header Field	Low	Medium
11	Server Leaks Version Information via "Server" HTTP Response Header Field	Low	High
12	Strict-Transport-Security Header Not Set	Low	High
13	Timestamp Disclosure - Unix	Low	Low
14	X-Content-Type-Options Header Missing	Low	Medium
15	Authentication Request Identified	Informational	Low
16	Information Disclosure - Suspicious Comments	Informational	Low
17	Modern Web Application	Informational	Medium
18	Retrieved from Cache	Informational	Medium
19	Session Management Response Identified	Informational	Medium
20	User Agent Fuzzer	Informational	Medium
21	User Controllable HTML Element Attribute (Potential XSS)	Informational	Low

Berdasarkan Tabel 4, dapat diketahui bahwa hasil pemindaian didominasi oleh kerentanan dengan tingkat risiko Low, diikuti oleh kategori Informational, dan tingkat risiko Medium. Kondisi ini menunjukkan bahwa meskipun tidak ditemukan kerentanan berisiko kritis secara otomatis, sistem masih memiliki banyak kelemahan pada konfigurasi keamanan dasar dan pengelolaan informasi server yang berpotensi dieksploitasi lebih lanjut oleh penyerang.

Tabel 5. Ringkasan Tingkat Risiko Kerentanan

Risk Level	Jumlah
Medium	6
Low	8
Informational	7
Total	**21**

Berdasarkan Tabel 5, terlihat jelas bahwa tidak ditemukan kerentanan dengan tingkat risiko High maupun Critical. Sebaran temuan terdistribusi pada tingkat risiko Low (8 temuan), Informational (7 temuan), dan Medium (6 temuan). Hal ini menunjukkan bahwa profil risiko aplikasi secara mayoritas bersumber dari kelemahan tingkat menengah ke bawah, yang utamanya berkaitan dengan konfigurasi keamanan dasar dan pengaturan header pelindung. Secara lebih rinci, kerentanan dengan tingkat risiko Medium umumnya berkaitan dengan kelemahan konfigurasi keamanan yang krusial, seperti tidak adanya Content Security Policy (CSP), Absence of Anti-CSRF Tokens, serta Missing Anti-clickjacking Header. Celah ini

berpotensi dimanfaatkan oleh penyerang untuk melakukan eksploitasi seperti Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), dan Clickjacking.

Sementara itu, kerentanan dengan tingkat risiko Low cenderung berkaitan dengan kelemahan pada pengaturan header dan bocornya informasi server. Meskipun tidak berdampak langsung secara kritis, celah ini tetap dapat dimanfaatkan sebagai langkah awal dalam proses eksploitasi yang lebih lanjut. Adapun temuan pada kategori Informational umumnya hanya bersifat sebagai catatan teknis terkait identitas sistem yang tidak menimbulkan ancaman langsung. Secara keseluruhan, distribusi tingkat risiko ini menegaskan bahwa sistem masih memiliki kelemahan mendasar pada aspek konfigurasi dan validasi keamanan yang perlu segera diperbaiki untuk meminimalkan potensi serangan siber.

Penetration Testing Analysis

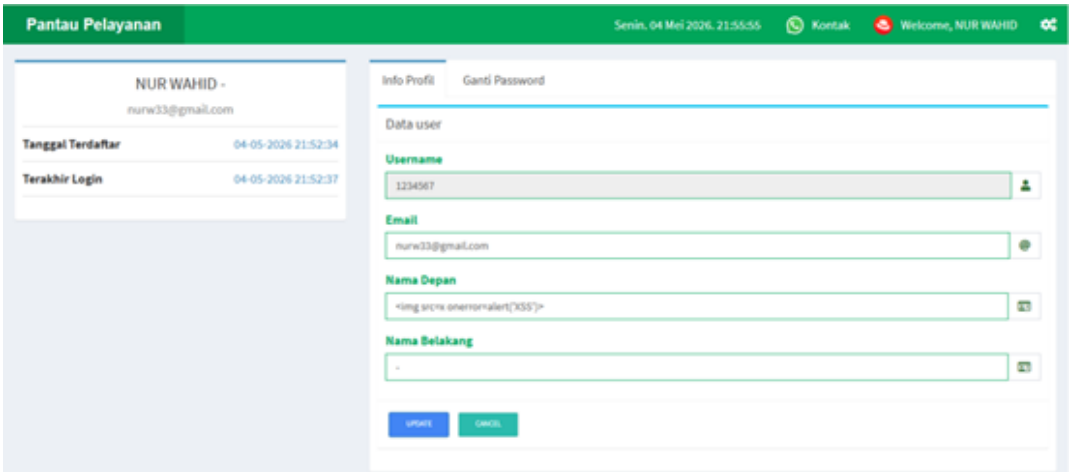
Untuk memastikan validitas temuan dari proses vulnerability scanning, dilakukan pengujian lanjutan terhadap kerentanan yang telah teridentifikasi. Pengujian dilakukan menggunakan pendekatan black-box testing, sehingga proses pengujian dilakukan tanpa mengetahui struktur internal sistem.

Pengujian Cross-Site Scripting (XSS)

Injeksi Payload

Tahap ini merupakan proses utama dalam pengujian penetrasi, di mana payload XSS disisipkan ke berbagai form input yang terdapat pada website. Payload yang digunakan memanfaatkan atribut HTML seperti onerror, onclick, dan onload yang sering menjadi celah keamanan dalam aplikasi web.

Pengujian dilakukan pada beberapa titik input, yaitu pada form login dan form register. Pada halaman profil pengguna, payload diinjeksikan ke dalam kolom input, khususnya pada field “Nama Depan”. Seperti ditunjukkan pada Gambar 7, proses injeksi dilakukan dengan menyisipkan script ke dalam form yang tersedia. Hasil pengujian menunjukkan bahwa payload tidak berhasil dieksekusi oleh sistem, yang ditandai dengan tidak munculnya pop-up alert setelah data diperbarui. Hal ini menunjukkan bahwa sistem telah menerapkan mekanisme validasi atau sanitasi input dengan cukup baik, sehingga mampu mencegah eksekusi script berbahaya.



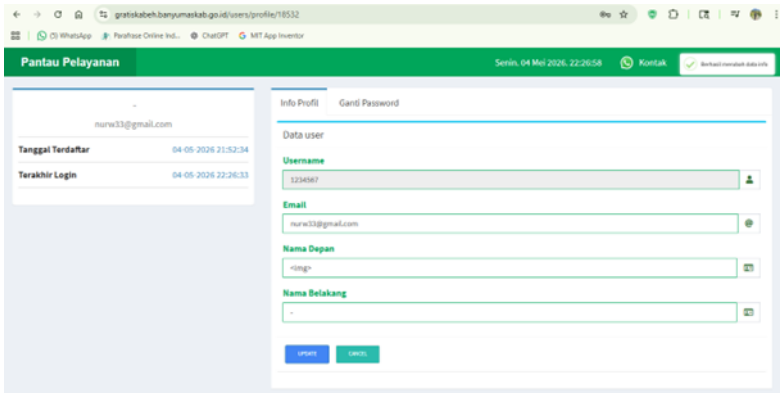
Gambar 7. Injeksi payload XSS pada halaman profil pengguna

Berdasarkan Gambar 7, terlihat bahwa payload telah berhasil disisipkan ke dalam field input pada halaman profil. Namun, sistem tidak mengeksekusi payload tersebut, yang menunjukkan bahwa terdapat mekanisme validasi atau sanitasi input yang mampu mencegah serangan XSS pada fitur tersebut. Pengujian selanjutnya dilakukan pada halaman register. Gambar 8 menunjukkan proses injeksi payload pada kolom “Nama”. Pengujian ini bertujuan untuk mengidentifikasi apakah sistem melakukan validasi dan sanitasi input secara menyeluruh.

Gambar 8. Injeksi payload pada form register

Berdasarkan Gambar 8, payload telah berhasil disisipkan ke dalam field input pada halaman register. Namun, hasil pengujian menunjukkan bahwa payload tidak berhasil dieksekusi oleh sistem setelah data dikirim. Tidak ditemukan adanya pop-up alert atau indikasi eksekusi script pada sisi klien. Hal ini menunjukkan bahwa sistem telah menerapkan mekanisme validasi atau sanitasi input yang cukup baik dalam mencegah serangan Cross-Site Scripting (XSS).

Hasil pengujian menunjukkan bahwa payload berhasil dieksekusi pada sisi klien, yang ditandai dengan munculnya pop-up alert seperti yang ditunjukkan pada Gambar 9. Payload yang digunakan adalah <img src=x onerror=alert('XSS')>, yang memanfaatkan event handler pada elemen HTML.



Gambar 9. Hasil pengujian XSS pada halaman profil pengguna

Berdasarkan Gambar 9, payload yang telah disisipkan tidak berhasil dieksekusi oleh sistem. Hal ini ditunjukkan dengan tidak munculnya pop-up alert serta tidak adanya indikasi eksekusi script pada sisi klien. Input yang mengandung script hanya ditampilkan sebagai teks biasa. Hal ini menunjukkan bahwa sistem telah menerapkan mekanisme validasi atau sanitasi input dengan baik, sehingga mampu mencegah serangan Cross-Site Scripting (XSS).

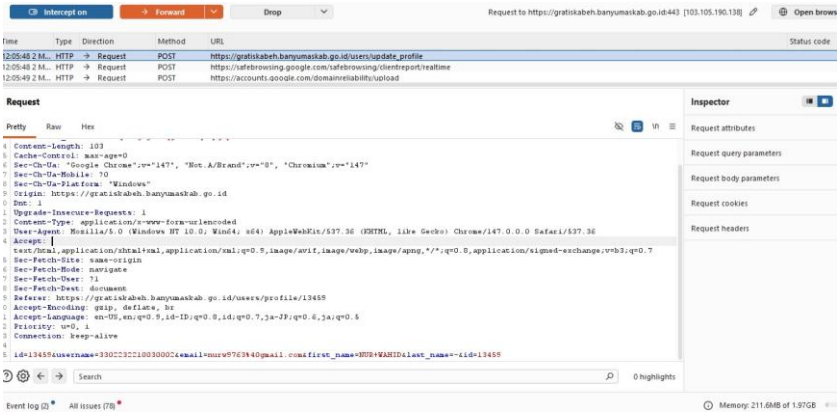
Pengujian Cross-Site Request Forgery (CSRF)

Injeksi Payload

Pengujian CSRF dilakukan dengan mensimulasikan pengiriman request menggunakan halaman HTML tanpa adanya interaksi langsung dari pengguna. Tujuan dari pengujian ini adalah untuk mengidentifikasi apakah sistem memiliki mekanisme validasi request, seperti penggunaan token keamanan (anti-CSRF token).

Hasil pengujian menunjukkan bahwa request dapat dikirim ke sistem hanya dengan

menekan tombol pada halaman simulasi. Setelah tombol ditekan, sistem langsung menerima parameter yang dikirim melalui request tanpa adanya proses validasi tambahan.

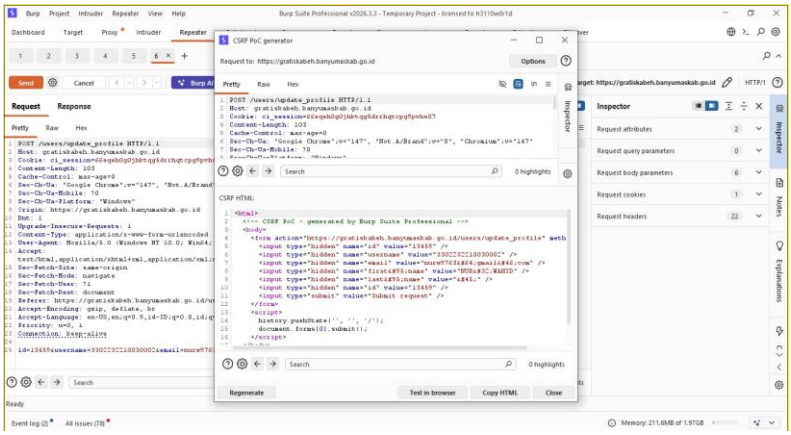


Gambar 10. Request perubahan data pengguna menggunakan Burp Suite

Berdasarkan Gambar 10, terlihat bahwa request perubahan data pengguna dikirim menggunakan metode POST ke endpoint /users/update_profile dengan menyertakan beberapa parameter seperti username, email, dan nama pengguna. Namun, pada request tersebut tidak ditemukan adanya mekanisme keamanan berupa token anti-CSRF (anti-CSRF token).

Hal ini menunjukkan bahwa sistem tidak melakukan validasi terhadap sumber request yang dikirim, sehingga memungkinkan request diproses tanpa memastikan bahwa permintaan tersebut berasal dari pengguna yang sah. Kondisi ini mengindikasikan bahwa sistem rentan terhadap serangan Cross-Site Request Forgery (CSRF), di mana penyerang dapat memanfaatkan sesi pengguna yang sedang aktif untuk mengirimkan request berbahaya tanpa sepengetahuan pengguna. Kerentanan ini berpotensi dimanfaatkan oleh penyerang untuk melakukan perubahan data pengguna, seperti mengubah informasi profil atau melakukan aksi lain tanpa izin pengguna. Oleh karena itu, diperlukan mekanisme pengamanan seperti penggunaan token CSRF untuk memastikan keaslian setiap request yang dikirim ke server.

Selanjutnya, pada Gambar 11 ditunjukkan bahwa setelah tombol ditekan, request dikirim menggunakan metode GET melalui parameter pada URL tanpa adanya mekanisme verifikasi keaslian request. Hal ini menyebabkan halaman diarahkan ke form login dengan membawa parameter yang telah dimodifikasi pada URL. Kondisi tersebut menunjukkan bahwa sistem tidak memiliki mekanisme validasi yang memadai terhadap request yang dikirim, sehingga memungkinkan terjadinya serangan Cross-Site Request Forgery (CSRF). Penyerang dapat memanfaatkan celah ini untuk mengirimkan request berbahaya melalui URL tanpa sepengetahuan pengguna.



Gambar 11. Pembuatan proof of concept (PoC) serangan CSRF menggunakan Burp Suite

Berdasarkan Gambar 11, terlihat bahwa request yang telah ditangkap sebelumnya dapat dikonversi menjadi sebuah proof of concept (PoC) dalam bentuk HTML menggunakan fitur

CSRF PoC generator pada Burp Suite. Halaman HTML tersebut berisi parameter yang sama dengan request asli dan dapat digunakan untuk mengirimkan request secara otomatis ke server tanpa interaksi langsung dari pengguna.

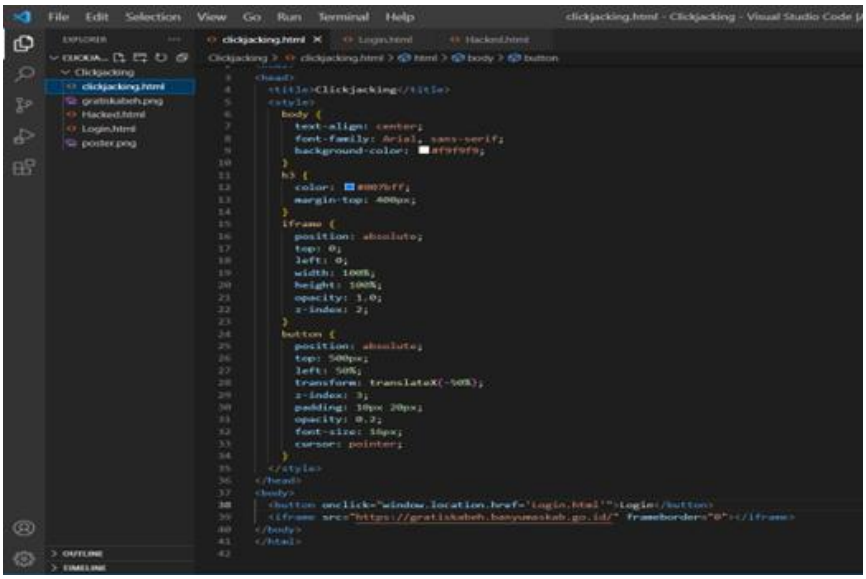
Hal ini menunjukkan bahwa request dapat direplikasi dan dikirim ulang dari sumber eksternal, yang mengindikasikan bahwa sistem tidak memiliki mekanisme perlindungan terhadap serangan CSRF, seperti penggunaan token keamanan (anti-CSRF token). Dengan adanya PoC tersebut, penyerang dapat dengan mudah membuat halaman berbahaya yang secara otomatis mengirimkan request ke sistem ketika korban mengakses halaman tersebut.

Pengujian Clickjacking

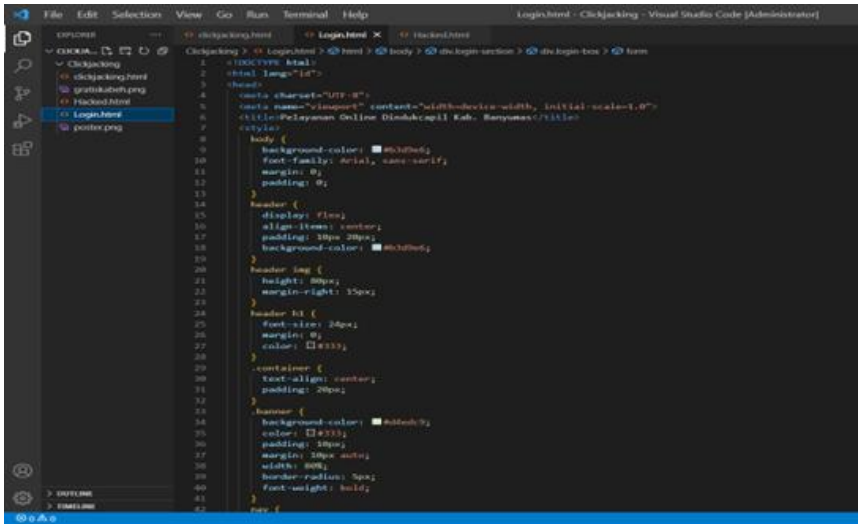
Simulasi Serangan

Tahap ini dilakukan untuk mengidentifikasi apakah website rentan terhadap serangan clickjacking. Pengujian dilakukan dengan mencoba menampilkan halaman website target ke dalam elemen iframe pada halaman lain. Tujuan dari pengujian ini adalah untuk mengetahui apakah sistem memiliki mekanisme perlindungan terhadap penyematan halaman oleh domain lain.

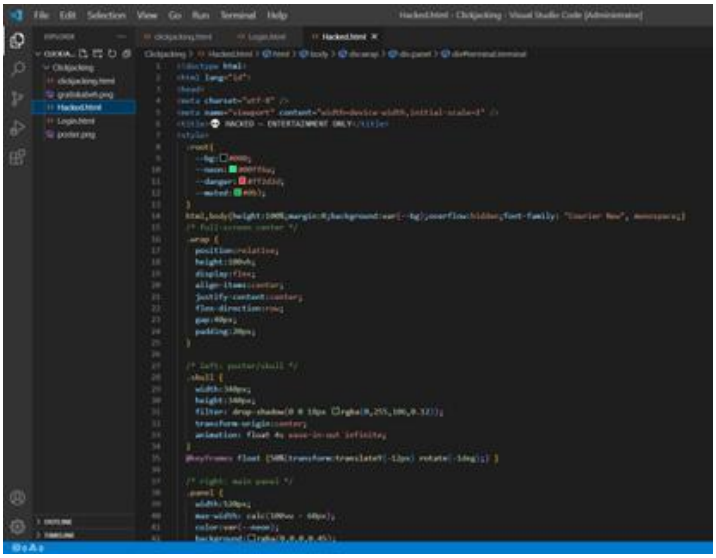
Untuk mensimulasikan serangan ini, dibuat skenario Proof of Concept (PoC) yang terdiri dari beberapa file HTML pendukung. File clickjacking.html bertindak sebagai halaman utama yang memuat iframe website target, sedangkan Login.html dan Hacked.html digunakan untuk memanipulasi antarmuka pengguna. Berikut adalah struktur kode skrip pendukung eksploitasi.



Gambar 12. Skrip untuk memuat elemen iframe



Gambar 13. Skrip untuk manipulasi tampilan



Gambar 14. Skrip untuk halaman peringatan eksploitasi

Berdasarkan kode-kode di atas, halaman clickjacking.html memuat website Gratis Kabeh Banyumas dalam sebuah iframe dan menempatkan elemen manipulasi di atasnya. Visualisasi dari manipulasi antarmuka pengguna tersebut dapat dilihat pada Gambar 15 berikut.

Gambar 15. Tampilan antarmuka manipulasi

Ketika pengguna berinteraksi dengan halaman tersebut, mereka tanpa sadar melakukan klik pada elemen website target yang sebenarnya. Hasil dari eksekusi serangan ini terbukti berhasil dan mengarahkan pengguna ke halaman "Hacked" yang telah disiapkan oleh penyerang, seperti yang ditunjukkan pada Gambar 16 berikut.

Gambar 16. Tampilan halaman setelah eksploitasi clickjacking

Keberhasilan memuat website target ke dalam iframe dan mengeksekusi aksi manipulasi ini membuktikan bahwa sistem rentan terhadap serangan clickjacking. Selain itu, dilakukan juga

analisis terhadap header keamanan pada response server untuk memastikan apakah terdapat konfigurasi keamanan seperti X-Frame-Options atau Content Security Policy (CSP) dengan directive frame-ancestors. Kedua mekanisme tersebut berfungsi untuk mencegah halaman website ditampilkan dalam iframe oleh pihak yang tidak berwenang.

Analisis dan Pembahasan

Hasil pengujian menunjukkan bahwa sebagian besar kerentanan yang ditemukan berada pada kategori Medium dan Low, dengan dominasi pada aspek konfigurasi keamanan dan validasi input. Hal ini mengindikasikan bahwa sistem masih memiliki kelemahan mendasar dalam penerapan mekanisme keamanan aplikasi web. Kerentanan Cross-Site Scripting (XSS) yang berhasil dieksploitasi menunjukkan bahwa sistem belum menerapkan validasi input dan output encoding secara optimal. Kondisi ini memungkinkan penyerang untuk menyisipkan script berbahaya yang dapat dijalankan pada sisi pengguna, sehingga berpotensi menyebabkan pencurian data, session hijacking, serta manipulasi tampilan halaman.

Di sisi lain, kerentanan Cross-Site Request Forgery (CSRF) menunjukkan bahwa sistem tidak memiliki mekanisme validasi request yang memadai. Tanpa adanya token keamanan, sistem tidak dapat membedakan antara request yang sah dan request yang dikirim oleh pihak tidak bertanggung jawab. Hal ini membuka peluang bagi penyerang untuk melakukan aksi tanpa sepengetahuan pengguna. Selain itu, tidak adanya Content Security Policy (CSP) serta security header lainnya memperbesar risiko eksploitasi, terutama pada serangan berbasis injeksi dan manipulasi konten. Hal ini menunjukkan bahwa keamanan sistem masih berfokus pada level dasar dan belum mencakup perlindungan menyeluruh pada sisi aplikasi.

Jika dibandingkan dengan penelitian sebelumnya, hasil penelitian ini menunjukkan kesamaan pada dominasi kerentanan kategori menengah yang disebabkan oleh kurangnya implementasi security header dan validasi input. Namun, penelitian ini memiliki keunggulan karena tidak hanya mengidentifikasi kerentanan, tetapi juga melakukan pengujian eksploitasi secara langsung sehingga memberikan gambaran yang lebih komprehensif terhadap tingkat keamanan sistem.

KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan menggunakan metode Information Systems Security Assessment Framework (ISSAF) dan tools OWASP ZAP, dapat disimpulkan bahwa website Gratis Kabeh Banyumas masih memiliki sejumlah kerentanan keamanan dengan tingkat risiko Low dan Medium. Kerentanan yang ditemukan didominasi oleh kelemahan pada konfigurasi keamanan, seperti tidak adanya Content Security Policy (CSP), Anti-CSRF Token, serta beberapa security header yang belum diterapkan secara optimal. Hasil pengujian lanjutan menunjukkan bahwa kerentanan Cross-Site Scripting (XSS) dan Cross-Site Request Forgery (CSRF) dapat dieksploitasi secara langsung. Pada pengujian XSS, payload berhasil dieksekusi pada sisi klien, yang menunjukkan bahwa sistem belum menerapkan validasi dan sanitasi input secara memadai. Sementara itu, pada pengujian CSRF, perubahan data dapat dilakukan tanpa validasi token keamanan, yang membuktikan bahwa sistem belum memiliki mekanisme perlindungan terhadap manipulasi request. Temuan ini menunjukkan bahwa kerentanan yang terdapat pada sistem tidak hanya bersifat teoritis, tetapi juga memiliki dampak nyata terhadap keamanan, terutama pada aspek kerahasiaan, integritas, dan ketersediaan data. Hasil eksploitasi menunjukkan bahwa kerentanan yang ditemukan bersifat praktis dan dapat dimanfaatkan secara langsung oleh penyerang. Oleh karena itu, diperlukan penerapan mekanisme keamanan yang lebih baik, seperti validasi input, output encoding, implementasi Content Security Policy (CSP), penggunaan Anti-CSRF Token, serta penerapan secure coding practices. Penelitian ini memiliki keterbatasan pada penggunaan satu tools yaitu OWASP ZAP serta ruang lingkup pengujian yang terbatas pada satu objek website. Oleh karena itu, penelitian selanjutnya disarankan untuk menggunakan kombinasi tools keamanan lainnya serta memperluas objek penelitian agar memperoleh hasil yang lebih komprehensif dan akurat.

REFERENSI

Albahli, & M. Melad. (2022). Analysis of Optimal Operation of Charging Stations Based on Dynamic Target Tracking of Electric Vehicles. *Electronics*, *11*(19), 3175. https://doi.org/10.3390/electronics11193175

Alhamed, M., & Rahman, M. M. H. (2023). A Systematic Literature Review on Penetration Testing in Networks: Future Research Directions. *Applied Sciences*, *13*(12), 6986. https://doi.org/10.3390/app13126986

Alkhurayyif, Y., & Saad Almarshdy, Y. (2024). Adopting Automated Penetration Testing Tools. *Journal of Information Security and Cybercrimes Research*, *7*(1), 51–66. https://doi.org/10.26735/RJJT2453

Ashari, I. F. A., Affandi, M., Putra, H. T., & Nur, M. T. (2023). Security Audit for Vulnerability Detection and Mitigation of UPT Integrated Laboratory (ILab) ITERA Website Based on OWASP Zed Attack Proxy (ZAP). *Jurnal JTIK (Jurnal Teknologi Informasi Dan Komunikasi)*, *7*(1), 24–34. https://doi.org/10.35870/jtik.v7i1.657

Aslan, Ö., Aktuğ, S. S., Ozkan-Okay, M., Yilmaz, A. A., & Akin, E. (2023). A Comprehensive Review of Cyber Security Vulnerabilities, Threats, Attacks, and Solutions. *Electronics*, *12*(6), 1333. https://doi.org/10.3390/electronics12061333

Cholifah, W. N., Yulianingsih, Y., & Sagita, S. M. (2022). Pengujian Black Box Testing pada Aplikasi Action & Strategy Berbasis Android dengan Teknologi Phonegap. *STRING (Satuan Tulisan Riset Dan Inovasi Teknologi)*, *3*(2), 206. https://doi.org/10.30998/string.v3i2.3048

Darojat, E. Z., Sediyono, E., & Sembiring, I. (2022). Vulnerability Assessment Website E-Government dengan NIST SP 800-115 dan OWASP Menggunakan Web Vulnerability Scanner. *JURNAL SISTEM INFORMASI BISNIS*, *12*(1), 36–44. https://doi.org/10.21456/vol12iss1pp36-44

Deepa, B., Prabadevi, P., Maddikunta, T., Gadekallu, & C. L. Chowdhary. (2022). Unsupervised virtual view synthesis from monocular video with generative adversarial warping. *Computers & Electrical Engineering*, *96*, 107460. https://doi.org/10.1016/j.compeleceng.2021.107460

Ferrag, L., Maglaras, H., Janicke, J., Jiang, & T. Shu. (2023). A review on orthogonal time–frequency space modulation: State-of-art, hotspots and challenges. *Computer Networks*, *224*, 109597. https://doi.org/10.1016/j.comnet.2023.109597

Bella, P., Biondi, S., Bognanni, & S. Esposito. (2023). A joint strategy for service deployment and task offloading in satellite–terrestrial IoT. *Computer Networks*, *225*, 109656. https://doi.org/10.1016/j.comnet.2023.109656

Holm, M. Ekstedt, & D. Andersson. (2023). Efficient Private Comparison Queries Over Encrypted Databases Using Fully Homomorphic Encryption With Finite Fields. *IEEE Transactions on Dependable and Secure Computing*, *18*(6), 2861–2874. https://doi.org/10.1109/TDSC.2020.2967740

Khan, I., Ahmad, M., & Kim, D. (2023). Tackling Age of Information in Access Policies for Sensing Ecosystems. *Sensors*, *23*(7), 3456. https://doi.org/10.3390/s23073456

Lashkari, G., D. Gil, & M. S. I. Mamun. (2022). Multiple Cues-Based Robust Visual Object Tracking Method. *Electronics*, *11*(3), 345. https://doi.org/10.3390/electronics11030345

M. Alenezi, & K. Almustafa. (2023). Physical Modeling and Intelligent Prediction for Instability of High Backfill Slope Moisturized under the Influence of Rainfall Disasters. *Applied Sciences*, *13*(7), 4218. https://doi.org/10.3390/app13074218

Mohamad Noor, & N. Ithnin. (2022). Multi-method Approach for User Experience of Selfie-taking Mobile Applications. *International Journal of Advanced Computer Science and Applications*, *13*(9). https://doi.org/10.14569/IJACSA.2022.01309101

Muharrom, M., & Saktiansyah, A. (2023). Analysis of Vulnerability Assessment Technique Implementation on Network Using OpenVas. *International Journal of Engineering and Computer Science Applications (IJECSA)*, *2*(2), 51–58. https://doi.org/10.30812/ijecsa.v2i2.3297

Albahli, & M. Melad. (2022). Analysis of Optimal Operation of Charging Stations Based on Dynamic Target Tracking of Electric Vehicles. *Electronics*, *11*(19), 3175. https://doi.org/10.3390/electronics11193175

Malik, N., Ahmad, & R. Khan. (2022). Importance of Memory Management Layer in Big Data Architecture. *International Journal of Advanced Computer Science and Applications*, *13*(5). https://doi.org/10.14569/IJACSA.2022.0130554

Saha, M., A. Rahman, & S. A. Hossain. (2023). Study of Single and Multipass f–rGO Inkjet-Printed Structures with Various Concentrations: Electrical and Thermal Evaluation. *Sensors*, *23*(4), 2058. https://doi.org/10.3390/s23042058

Sanjaya, I. G. A. S., Sasmita, G. M. A., & Arsa, D. M. S. (2023). Evaluasi Keamanan Website Lembaga X Melalui Penetration Testing Menggunakan Framework ISSAF. *Jurnal Ilmiah Merpati (Menara Penelitian Akademika Teknologi Informasi)*, 113. https://doi.org/10.24843/JIM.2020.v08.i02.p05

Tobi, H., & Kampen, J. K. (2024). Research design: the methodology for interdisciplinary research framework. *Quality & Quantity*, *52*(3), 1209–1225. https://doi.org/10.1007/s11135-017-0513-8

Y. He, P. R., & P. De Roure. (2023). Artificial Intelligence–Based Ethical Hacking for Health Information Systems: Simulation Study. *Journal of Medical Internet Research*, *25*, e41748. https://doi.org/10.2196/41748