

PENGUJIAN WEBSITE DINAS SOSIAL SURABAYA MENGGUNAKAN METODE PENETRATION TESTING DAN OWASP TOP 10

Bregas Arya Bagaskara¹, Mohammad Idhom², Henni Endah Wahanani³

¹²³Informatika, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional "Veteran" Jawa Timur

Jln. Rungkut Madya, Gn. Anyar, Kec. Gn. Anyar, Surabaya, Jawa Timur 60294

¹ bregas909@gmail.com, ² idhom@upnjatim.ac.id, ³ henniendah@upnjatim.ac.id

Abstract

Information system security is crucial in the development of information technology. Advancements in technology have opened potential security gaps that could be exploited by malicious actors. This study aims to identify and analyzing vulnerabilities on the Surabaya Social Service website using penetration testing with the OWASP Top 10 approach. As a public service platform, this website is at risk of cyberattacks that could endanger user data. The research follows five stages: information gathering, footprinting & scanning, vulnerability assessment, exploitation, and analyze & report. Using the OWASP Top 10 framework, the study evaluates the ten most critical web application vulnerabilities. The results reveal six primary issues: Browsable Web Directories, web.config File Information Disclosure, Content Security Policy (CSP) Header Not Set, Strict-Transport-Security Header Not Set, Timestamp Disclosure - Unix, and X-Content-Type-Options Header Missing. To mitigate these vulnerabilities, implementing security headers such as Content-Security-Policy (CSP), Strict-Transport-Security (HSTS), and X-Content-Type-Options: nosniff is recommended. Additionally, securing directories and sensitive configuration files is essential to reduce data leakage risks. This research offers valuable insights to enhance the security of government websites and safeguard them against cyber threats.

Keywords : Penetration testing, OWASP Top 10, Website, Cyber, Vulnerability, Security.

Abstrak

Keamanan sistem informasi menjadi faktor penting dalam pengembangan teknologi informasi. Kemajuan teknologi membuat banyaknya potensi celah keamanan yang dapat dimanfaatkan oleh pihak yang tidak bertanggung jawab. Dari hal tersebut dilakukan penelitian yang bertujuan untuk mengidentifikasi dan menganalisis kerentanan pada *website* Dinas Sosial Surabaya menggunakan metode *penetration testing* dengan pendekatan OWASP Top 10. Sebagai salah satu aplikasi pelayanan publik, *website* Dinas Sosial Surabaya memiliki potensi menjadi target serangan siber yang dapat mengancam keamanan data pengguna. Penelitian ini melibatkan lima tahap dalam metode *penetration testing*, yaitu *information gathering*, *footprinting & scanning*, *vulnerability assessment*, *exploitation*, serta *analyze & report*. Pendekatan OWASP Top 10 digunakan untuk mengevaluasi kerentanan, dengan fokus pada sepuluh kerentanan paling kritis yang dapat membahayakan aplikasi *web*. Hasil pengujian mengungkapkan enam kerentanan utama pada *website* ini, yaitu *Browsable Web Directories*, *web.config File Information Disclosure*, *Content Security Policy (CSP) Header Not Set*, *Strict-Transport-Security Header Not Set*, *Timestamp Disclosure - Unix*, dan *X-Content-Type-Options Header Missing*. Untuk mengatasi kerentanan tersebut, disarankan penerapan header keamanan seperti *Content-Security-Policy (CSP)*, *Strict-Transport-Security (HSTS)*, dan *X-Content-Type-Options: nosniff*. Selain itu, pengamanan terhadap direktori dan *file* konfigurasi sensitif perlu dilakukan untuk meminimalkan risiko kebocoran data. Penelitian ini memberikan wawasan penting dalam meningkatkan keamanan *website* pemerintah sehingga lebih terlindungi dari ancaman siber.

Kata kunci : Penetration testing, OWASP Top 10, Website, Siber, Kerentanan, Keamanan.

1. PENDAHULUAN

Keamanan sistem informasi merupakan salah satu isu penting dalam perkembangan teknologi informasi dan komunikasi [1]. Seiring dengan kemajuan teknologi informasi, *website* tidak hanya berfungsi sebagai media untuk menyebarkan informasi tetapi juga menjadi target potensial serangan siber. Oleh karena itu, jaringan komputer harus dilindungi secara menyeluruh dari berbagai bentuk serangan, upaya penyusupan, maupun aktivitas pemindaian oleh pihak-pihak yang tidak bertanggung jawab [2]. Ancaman keamanan yang terus berubah dan berkembang membutuhkan upaya yang konsisten untuk menemukan dan menanggulangi berbagai potensi celah keamanan.

Menurut Global Cybersecurity Index (GCI) tahun 2020 yang diterbitkan oleh International Telecommunication Union (ITU), Saat ini Indonesia menempati peringkat ke-24 dari 194 negara dalam hal kewanman siber yang meningkat signifikan dari posisi ke-41 pada tahun 2018 [3]. Hal ini menunjukkan kemajuan dalam perlindungan keamanan siber di tingkat nasional.

Di tahun 2022, Badan Siber dan Sandi Negara (BSSN) mencatat 370,02 juta serangan siber di Indonesia, mengalami peningkatan sebesar 38,72% dari tahun sebelumnya yang berjumlah 266,74 juta serangan [4]. Berdasarkan lanskap keamanan siber indonesia tahun 2023, sektor pemerintahan adalah yang paling terdampak oleh insiden siber, seperti kebocoran data, serangan *ransomware*, kerusakan tampilan *website* (*defacement*), indikasi serangan DDoS, dan pemantauan proaktif terhadap dugaan insiden siber [5]. Data tersebut menggambarkan bahwa sektor pemerintahan masih menghadapi tantangan besar dalam mengatasi ancaman serangan siber.

Sebagai ilustrasi dari gambaran tersebut, sebuah penelitian terkait kerentanan keamanan *website* pemerintah kabupaten kediri yang menggunakan teknik *penetration testing* berhasil mengidentifikasi beberapa celah kewanman. Salah satu celah keamanan tersebut memungkinkan pengungkapan data sensitif, seperti *username* dan *password*, yang dapat digunakan untuk mengakses halaman cPanel admin [6].

Dalam hal ini, institusi atau lembaga pemerintahan seperti Dinas Sosial Surabaya yang menyediakan layanan sosial melalui *website* resmi, harus menjamin bahwa sistem informasi yang digunakan aman dan terlindungi. *Penetration Testing*, sebagai salah satu contoh teknik uji keamanan yang efektif Untuk mendeteksi

kemungkinan celah keamanan yang bisa dieksploitasi oleh pihak yang tidak berwenang.

Penetration testing atau *pentest* adalah simulasi serangan terhadap sebuah sistem yang dilakukan oleh individu atau tim dengan tujuan membantu mengidentifikasi kerentanan terhadap aplikasi atau jaringan guna memberikan evaluasi serta pengukuran untuk perbaikan berkelanjutan [7]. Salah satu penelitian terkait, yaitu oleh Ikhsan (2024), membahas pengujian tingkat keamanan *website* www.sadikun.com milik PT. Sadikun Niaga Mas Raya terhadap potensi serangan pihak luar. Penelitian ini menggunakan metode *penetration testing* dengan tahapan seperti *footprinting*, *scanning*, *exploitation*, dan *reporting*, yang berhasil mengidentifikasi sejumlah kerentanan pada *website* tersebut [8]. Untuk mendukung hasil penelitian tersebut diperlukan pendekatan yang terstruktur dan menyeluruh dalam mengidentifikasi serta mengelola risiko keamanan. Salah satu pendekatan yang dapat diterapkan untuk mencapai tujuan tersebut adalah OWASP Top 10.

Open Web Application Security Prokect (OWASP) Top 10 merupakan panduan terkemuka yang dirancang untuk mengidentifikasi dan mengatasi kerentanan keamanan pada aplikasi web [9]. Metode ini memberikan pemahaman tentang risiko ancaman keamanan yang paling relevan dan paling berdampak pada aplikasi web di era digital saat ini.

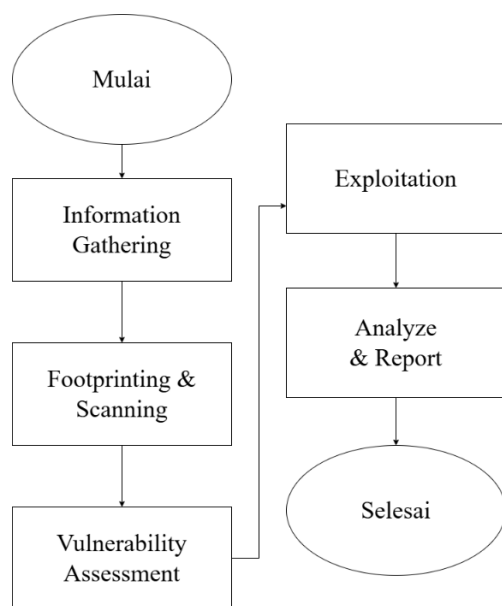
Seperti contoh penggunaan metode OWASP Top 10 penelitian yang dilakukan oleh Armando & Rosalina (2023) menggunakan OWASP Top 10 sebagai kerangka utama untuk mengidentifikasi dan mengklasifikasikan kelemahan keamanan pada *website* resmi pemerintah Kota Tangerang, tangerangkota.go.id. Dengan metode *penetration testing* berbasis pendekatan *black box*, penelitian ini menemukan beberapa celah keamanan kritis, seperti *SQL Injection*, *Stored XSS*, *Reflected XSS*, *Insecure Direct Object Reference* (IDOR), dan *Information Disclosure*. Temuan ini menegaskan bahwa OWASP Top 10 sangat relevan dan efektif sebagai standar keamanan *website* karena mampu mengidentifikasi kerentanan yang sering muncul dalam pengujian keamanan saat ini. Dengan standar yang menyeluruh, OWASP Top 10 menjadi alat yang tepat untuk memastikan perlindungan data dan informasi penting dari ancaman keamanan siber [10].

Studi kasus pada *website* dinassosial.surabaya.go.id dianggap penting karena institusi ini memberikan layanan publik melewati *platform online* dan memberikan akses

informasi yang penting. Penelitian ini bertujuan untuk menganalisis kerentanan keamanan pada *website* tersebut menggunakan teknik *penetration testing* dengan pendekatan OWASP Top 10. Melalui metode dan pendekatan tersebut, diharapkan dapat teridentifikasi berbagai risiko dan ancaman keamanan, serta rekomendasi perbaikan untuk memperkuat keamanan *website* tersebut. Hasil penelitian ini diharapkan dapat berkontribusi dalam meningkatkan keamanan informasi dan layanan yang diberikan oleh Dinas Sosial Surabaya, sekaligus menjadi acuan bagi instansi pemerintah lainnya dalam mengelola keamanan aplikasi web mereka.

2. METODOLOGI PENELITIAN

Penetration testing dirancang secara terstruktur untuk memastikan bahwa setiap tahapan dilakukan untuk mendukung tujuan utama penelitian serta menghasilkan hasil yang valid dan dapat diandalkan. Tahapan tersebut dirangkum dalam alur penelitian yang meliputi *Information Gathering*, *Footprinting & Scanning*, *Vulnerability Assessment*, *Exploitation*, *Analyze & Report*, dan *Selesai*.



Gambar 1. Alur Penelitian

2.1 Information Gathering

Tahap *information gathering* adalah Langkah awal dimana data ataupun informasi yang didapatkan akan digunakan untuk menyusun porses pengujian yang akurat dan menyeluruh [11].

TABEL I. ALAT UNTUK INFORMATION GATHERING DAN RENCANA PENGGUNAANYA

No	Alat	Rencana penggunaan
1	Wappalyzer	Menggunakan ekstensi browser wappalyzer pada halaman website target
2	Nslookup	Menggunakan perintah pada terminal kali linux seperti berikut : nslookup dinassosial.surabaya.go.id
3	Whois	Menggunakan perintah pada terminal kali linux seperti berikut : whois <alamat_ip_target>
4	Nmap	Menggunakan perintah pada terminal kali linux seperti berikut : nmap dinassosial.surabaya.go.id
5	Dirsearch	Menggunakan perintah pada terminal kali linux seperti berikut : dirsearch dinassosial.surabaya.go.id -x 300-600
6	Google Dorking	Menggunakan kata-kata pencarian di google berikut ini: 1. Site:dinassosial.surabaya.go.id filetype:sql 2. inurl:admin site:dinassosial.surabaya.go.id 3. intext:phpMyAdmin inurl:dinassosial.Surabaya.go.id 4. inurl:/.git site:dinassosial.surabaya.go.id 5. intitle:"Index of /database" site:dinassosial.surabaya.go.id

2.2 Footprinting & Scanning

Tahap *footprinting & scanning* merupakan tahapan yang penting dalam proses *penetration testing* pada *website* Dinas Sosial Surabaya. Pada tahap ini, dilakukan pemindaian terhadap target untuk mengidentifikasi kerentanan yang

berpotensi untuk dieksploitasi. Untuk mendeteksi kerentanan tersebut digunakan beberapa alat antara lain OWASP ZAP, Nessus dan OpenVAS [8].

2.3 Vulnerability Assessment

Setelah pemindaian kerentanan dilakukan, tahap *vulnerability assessment* bertujuan untuk mengklasifikasikan kerentanan yang terindikasi berdasarkan tingkat keparahan dan dampaknya. Klasifikasi ini menggunakan standar OWASP Top 10 2021, yang membantu menentukan jenis terhadap setiap kerentanannya [12].

2.4 Exploitation

Tahap *exploitation* merupakan Langkah vital dalam alur penetration testing karena dilakukan pengujian terhadap kerentanan yang telah teridentifikasi pada *website* Dinas Sosial Surabaya[13]. Dalam tahap ini digunakan alat-alat yang ditampilkan pada tabel dibawah ini:

TABEL II. ALAT UNTUK EXPLOITASI

No	Alat	Kegunaan
1	Firefox browser	Sebagai browser untuk melakukan pengujian dengan menggunakan fitur inspeksi.
2	Burp Suite	Sebagai penangkap <i>request</i> yang masuk dalam <i>website</i> yang nantinya akan dimodifikasi agar menghasilkan <i>response</i> yang diinginkan.
3	Bettercap	Sebagai alat untuk melakukan pantauan terhadap aktifitas jaringan pada <i>website</i> target dan menganalisa lalu lintas data [14].
4	Sqlmap	Sebagai alat untuk mendeteksi dan mengeksploitasi kerentanan <i>SQL Injection</i> pada aplikasi web secara otomatis [15].

2.5 Analyze & Report

Tahap *analyze & report* merupakan langkah akhir pada alur penelitian. Semua temuan kerentanan dievaluasi berdasarkan tingkat keparahan dan dampaknya terhadap sistem untuk membuat rekomendasi perbaikan yang

dibutuhkan pada *website* Dinas Sosial Surabaya [16].

3. HASIL DAN PEMBAHASAN

Pada bagian ini akan membahas hasil dan analisis dari pengujian yang menggunakan metode penetration testing dengan pendekatan OWASP Top 10. Bagian ini berfokus pada temuan celah keamanan yang diperoleh pada *website* Dinas Sosial Surabaya.

3.1 Information Gathering

Tahap information gathering pada *website* Dinas Sosial Surabaya secara efektif berhasil mengumpulkan beberapa data dan informasi yang sangat penting untuk memahami struktur dan potensi kerentanan pada *website*. Data dan informasi tersebut disajikan dalam bentuk tabel dibawah ini.

TABEL III. HASIL DARI INFORMATION GATHERING

No	Alat	Kegunaan
1	Wappalyzer	<i>Website</i> ini menggunakan <i>framework</i> antarmuka pengguna (UI) Laravel dan Bootstrap versi 4.3.1, serta dilengkapi dengan reCAPTCHA untuk meningkatkan keamanan. Berbagai <i>library</i> juga digunakan untuk mendukung fitur-fitur pada <i>website</i> , seperti JQuery versi 3.2.1, Moment.js versi 2.18.1, SweetAlert2, DataTables 1.10.22, dan Select2. Sebagai web server, <i>website</i> ini mengandalkan Nginx, yang juga berfungsi sebagai <i>reverse proxy</i> untuk mengelola lalu lintas data secara efisien.
2	Nslookup	<i>Ip Address website</i> adalah 112.xxx.xxx.xx (Ipv4).
3	Whois	Alamat IP 112.xxx.xxx.xx yang terdaftar pada APNIC (Asia Pacific Network Information Centre) berlokasi di Kantor Pemerintah Kota Surabaya, Jalan Jimerto 25-27, Surabaya, dengan email

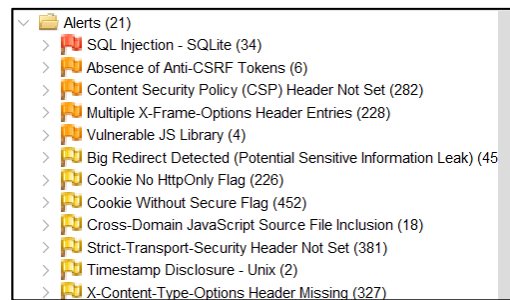
		abuse@surabaya.go.id dan terakhir diperbarui pada 18 Juni 2020.
4	Nmap	Layanan port : 1. Port 80/tcp dengan service http 2. Port 113/tcp dengan service ident 3. Port 443/tcp dengan service https
5	Dirsearch	Direktori yang terbuka: 1. /404.html 2. /admin 3. /favicon.ico 4. /faq 5. /index.php 6. /info 7. /login 8. /manual/index.html 9. /profile 10. /robot.txt 11. /search 12. /template/ 13. /web.config
6	Google Dorking	Tidak ada tautan yang mengandung informasi atau data sensitif.

3.2 Footprinting & Scanning

Tahap *footprinting & scanning* dilakukan untuk mengidentifikasi kerentanan dengan menggunakan beberapa alat khusus seperti OWASP ZAP, Nessus, dan OpenVAS untuk memetakan area rentan dan mengungkapkan celah keamanan yang ada.

3.2.1 OWASP ZAP

Untuk mengidentifikasi kerentanan yang ada pada *website* Dinas Sosial Surabaya digunakan alat OWASP ZAP dengan memanfaatkan fitur *automated scan* dan *manual scan*. Proses pemindaian dilakukan menggunakan *traditional spider*, *active scan* dan *manual active* untuk mengidentifikasi potensi kerentanan [17].

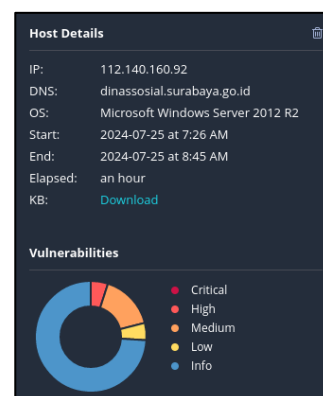


Gambar 2. Hasil Pemindaian Menggunakan OWASP ZAP

Gambar 2 menampilkan hasil pemindaian OWASP ZAP pada *website* *dinassosial.surabaya.go.id* mengungkap 1 kerentanan tinggi, 4 kerentanan sedang, 7 kerentanan rendah, dan 9 kerentanan informasional.

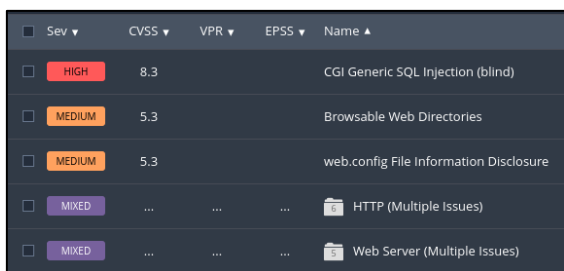
3.2.2 Nessus

Untuk pemindaian lebih lanjut pada *website* Dinas Sosial Surabaya digunakan alat Nessus yang menggunakan fitur *Web Application Tests* untuk mendeteksi berbagai potensi kerentanan. Alat ini dibuat untuk mengidentifikasi kerentanan, kesalahan konfigurasi, serta berbagai risiko keamanan lainnya pada perangkat yang terhubung ke jaringan [18].



Gambar 3. Hasil Pemindaian Menggunakan Nessus

Gambar 3 menampilkan hasil Pemindaian Nessus pada *website* *dinassosial.surabaya.go.id* yang mengidentifikasi 1 kerentanan tinggi, 3 kerentanan sedang, 1 kerentanan rendah, dan 32 kerentanan informasional.

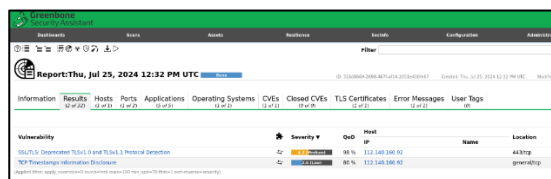


Gambar 4. Beberapa Kerentanan Yang Teridentifikasi Pada Nessus

Gambar 4 menampilkan beberapa contoh kerentanan yang teridentifikasi seperti contoh kerentanan *CGI Generic SQL Injection (blind)* dengan Tingkat skor 8.3 dan keparahan tinggi, serta kerentanan *Browsable Web Directories* dan *web.config File Information Disclosure* dengan score 5.3 dan tingkat keparahan sedang.

3.2.3 OpenVAS

Untuk memperluas identifikasi kerentanan pada website Dinas Sosial Surabaya digunakan alat OpenVAS dengan *menggunakan fitur Task Wizard* untuk pemindaian cepat dan *Advanced Task Wizard* untuk deteksi kerentanan secara kompleks dan akurat.



Gambar 5. Hasil Pemindaian Menggunakan OpenVAS

Berdasarkan gambar 5 Pemindaian OpenVAS pada *website* dinassosial.surabaya.go.id menemukan 2 kerentanan, yaitu 1 kerentanan sedang dan 1 kerentanan rendah.

3.3 Vulnerability Assessment

Setelah tahap pemindaian, tahap *vulnerability assessment* dilakukan dengan memilah potensi kerentanan yang terklasifikasi berdasarkan tingkat kerentanan dan berdasarkan kategori OWASP Top 10.

TABEL IV. VULNERABILITY ASSESSMENT BERDASARKAN KATEGORI OWASP TOP 10

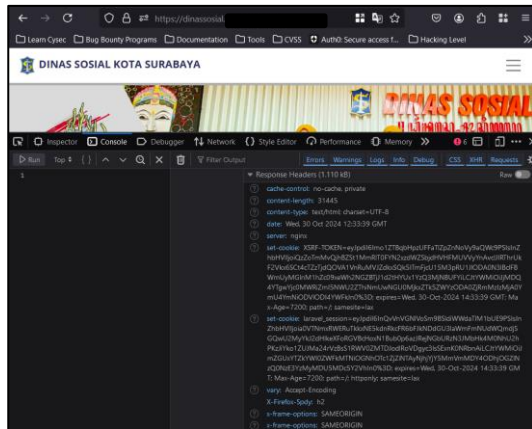
No	Kerentanan	Severity	Kategori OWASP Top 10
1	SQL Injection – SQLite	High	A03 - Injection
2	CGI Generic SQL Injection (blind)	High	A03 - Injection
3	Browsable Web Directories	Medium	A05 - Security Misconfiguration
4	web.config File Information Disclosure	Medium	A06 - Vulnerable and Outdated Components
5	Absence of Anti-CSRF Tokens	Medium	A07 - Identification and Authentication Failures
6	Content Security Policy (CSP) Header Not Set	Medium	A04 - Insecure Design
7	Multiple X-Frame-Options Header Entries	Medium	A05 - Security Misconfiguration
8	Strict-Transport-Security Header Not Set	Low	A05 - Security Misconfiguration
9	Timestamp Disclosure - Unix	Low	A06 - Vulnerable and Outdated Components
10	X-Content-Type-Options Header Missing	Low	A05 - Security Misconfiguration

3.4 Exploitation

Pada tahap exploitation, dilakukan pengujian terhadap sepuluh kerentanan yang telah diklasifikasikan sebelumnya, yaitu 2 kerentanan dengan tingkat keparahan tinggi (*high*), 5 kerentanan sedang (*medium*), dan 3 kerentanan rendah (*low*). Pengujian ini bertujuan untuk mengidentifikasi dan memvalidasi setiap kerentanan yang ditemukan. Proses pengujian dilakukan secara bertahap, dimulai dari kerentanan dengan tingkat keparahan rendah hingga keparahan dengan tingkat tinggi. Pendekatan ini digunakan untuk menganalisis dampak dari masing-masing kerentanan terhadap keamanan website Dinas Sosial Surabaya secara menyeluruh dan bertahap.

3.4.1 X-Content-Type-Options Header Missing

Kerentanan *X-Content-Type-Options Header Missing* terjadi karena server tidak menyetel *header nosniff*, yang penting untuk mencegah risiko penembakan jenis konten.

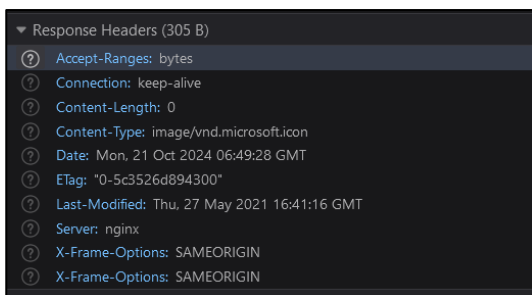


Gambar 6. Response Header Tanpa Header Nosniff

Gambar 6 menampilkan, meskipun website Dinas Sosial Surabaya belum mengimplementasikannya header nosniff, pengawasan secara langsung pada fitur yang terkait membuat dampak kerentanan ini minim terhadap keamanan keseluruhan website.

3.4.2 Timestamp Disclosure - Unix

Kerentanan *Timestamp Disclosure - Unix* terjadi saat server mengungkapkan informasi waktu dalam format *Unix Timestamp*, yang dapat memberi petunjuk kepada penyerang tentang pola aktivitas sistem.



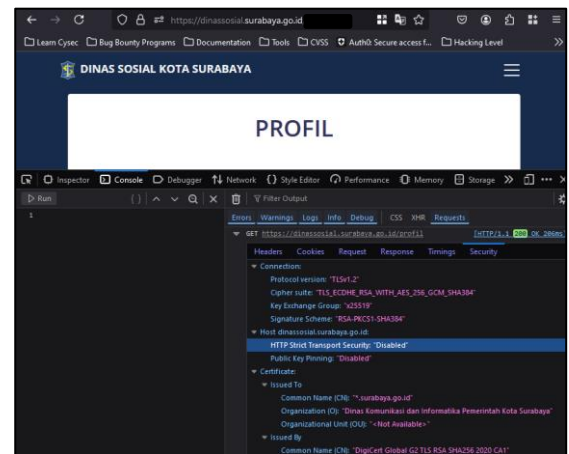
Gambar 7. Response Header Dengan Unix Timestamp

Berdasarkan Gambar 7, pada halaman <https://dinassosial.surabaya.go.id/template/xxx/xxx/xxx/xxxxxx.png>, response header menunjukkan timestamp "Last-Modified: Thu, 27 May 2021 16:41:16 GMT," mengindikasikan waktu terakhir file tersebut dimodifikasi.

3.4.3 Strict-Transport-Security Header Not

Kerentanan *Strict-Transport-Security Header Not Set* terjadi ketika server tidak menyertakan HSTS dalam response header,

memungkinkan serangan *man-in-the-middle* seperti *SSL stripping* dan *HSTS Hijacking*.



Gambar 8. Response Header Dengan HSTS Disable

Berdasarkan Gambar 8, website Dinas Sosial Surabaya belum mengonfigurasi HSTS. Namun, pengujian dengan *HSTS Hijacking* menggunakan Bettercap menunjukkan bahwa data sensitif seperti *username* dan *password* tetap aman saat melakukan *login*.

3.4.4 Multiple X-Frame-Options Header Entries

Kerentanan *Multiple X-Frame-Options Header Entries* terjadi ketika server mengirimkan lebih dari satu entri *X-Frame-Options* dalam response header. Header ini mencegah serangan *clickjacking* dengan mengontrol tampilan situs dalam elemen *iframe*. Meskipun terdapat beberapa entri *X-Frame-Options*, *iframe* gagal memuat website Dinas Sosial Surabaya. Browser modern dan terstandarisasi memilih satu nilai dan mengabaikan lainnya, sehingga perlindungan terhadap *clickjacking* tetap efektif. Dengan demikian, website Dinas Sosial Surabaya ini tidak rentan terhadap kerentanan tersebut.

3.4.5 Content Security Policy (CSP) Header Not Set

Kerentanan *Content Security Policy (CSP) Header Not Set* terjadi ketika server tidak mengirimkan CSP dalam response header. CSP melindungi dari serangan seperti XSS dan injeksi data dengan membatasi sumber konten yang dapat dimuat.

```
14:19:27.937 >> setTimeout("console.log('hello world halaman login')", 500)
14:19:27.956 ← 16515
14:19:28.457 hello world halaman login
```

Gambar 9. Output Console Browser Halaman Login

Berdasarkan gambar 9 Dilakukan pengujian skrip `'setTimeout("console.log('hello world halaman login')", 500)'` yang berhasil dijalankan. Skrip tersebut seharusnya diblokir oleh kebijakan CSP. Hal ini menunjukkan bahwa *website* tersebut rentan karena tidak mengatur CSP dengan baik.

3.4.6 Absence of Anti-CSRF Tokens

Absence of Anti-CSRF Tokens terjadi ketika aplikasi web tidak menggunakan token untuk melindungi dari serangan *Cross-Site Request Forgery* (CSRF), yang dapat mengeksploitasi tindakan tidak sah tanpa sepengetahuan pengguna. Setelah dianalisis, halaman yang terindikasi merupakan *website template* UI "argon" yang merupakan *framework frontend*. Meskipun *template* ini digunakan, kerentanan *Absence of Anti-CSRF Tokens* tidak berpengaruh pada *website* utama Dinas Sosial Surabaya, sehingga dapat disimpulkan bahwa *website* tersebut tidak rentan terhadap kerentanan ini.

3.4.7 web.config File Information Disclosure

Web.config File Information Disclosure adalah kerentanan yang terjadi ketika *file web.config*, yang umumnya digunakan dalam aplikasi berbasis .NET pada server web, terbuka untuk publik atau dapat diakses oleh pihak yang tidak berwenang.



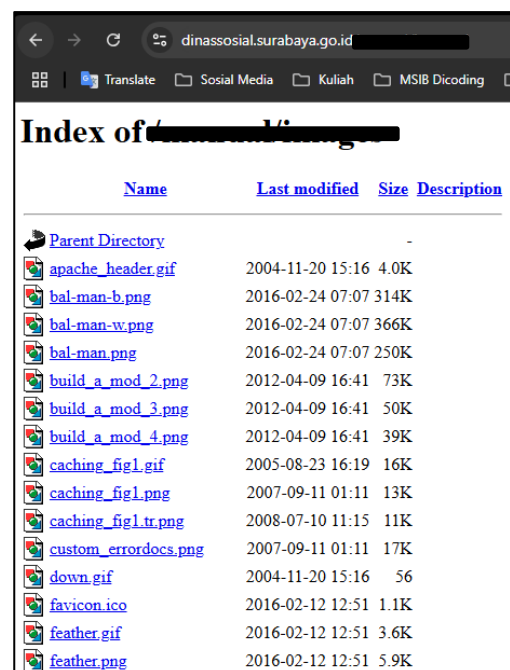
```
1 HTTP/1.1 200 OK
2 Server: nginx
3 Date: Fri, 01 Nov 2024 02:57:34 GMT
4 Content-Length: 1194
5 Connection: keep-alive
6 Last-Modified: Thu, 27 May 2021 16:41:16 GMT
7 ETag: "4aa-5c352ed04100"
8 Accept-Ranges: bytes
9 X-Frame-Options: SAMEORIGIN
10 X-Frame-Options: SAMEORIGIN
11
12 <!--
13 RewriteEngine requires Microsoft URL Rewrite Module for IIS
14 Download: https://www.microsoft.com/en-us/download/details.aspx?id=47327
15 Debug Help:
16 https://docs.microsoft.com/en-us/iis/extensions/url-rewrite-module/using-failed-req-
17 uest-tracing-to-trace-rewrite-rules
18 -->
19 <configuration>
20 <system.webServer>
21 <rewrite>
22 <rules>
23 <rule name="Imported Rule 1" stopProcessing="true">
24 <conditions>
25 <ignoreCase="false" negate="true" />
26 </conditions>
27 <action type="Redirect" redirectType="Permanent" url="/(R:1)" />
28 </rule>
29
30 <rule name="Imported Rule 2" stopProcessing="true">
31 <conditions>
32 <add input="{REQUEST_FILENAME}" matchType="IsDirectory">
33 <ignoreCase="false" negate="true" />
34 <add input="{REQUEST_FILENAME}" matchType="IsFile" ignoreCase="false"
35 <negate="true" />
36 </conditions>
```

Gambar 10. Response Website Menggunakan Burp Suite

Gambar 10 menunjukkan bahwa *file* ini dapat diakses oleh pengguna, memberikan informasi mengenai aturan penulisan ulang pada server IIS. Seharusnya, akses terhadap *file* ini harus dibatasi dengan *response* "404 Not Found" atau "403 Forbidden". Oleh karena itu, dapat disimpulkan bahwa *website* Dinas Sosial Surabaya rentan terhadap kerentanan ini karena *file web.config* dapat diakses tanpa perlindungan yang memadai.

3.4.8 Browseable Web Directories

Browseable Web Directories adalah kerentanan yang terjadi ketika direktori server web dapat diakses publik tanpa otentikasi, memungkinkan pengguna atau penyerang melihat daftar *file* dan folder dalam direktori tertentu.



Name	Last modified	Size	Description
Parent Directory	-	-	-
apache_header.gif	2004-11-20 15:16	4.0K	
bal-man-b.png	2016-02-24 07:07	314K	
bal-man-w.png	2016-02-24 07:07	366K	
bal-man.png	2016-02-24 07:07	250K	
build_a_mod_2.png	2012-04-09 16:41	73K	
build_a_mod_3.png	2012-04-09 16:41	50K	
build_a_mod_4.png	2012-04-09 16:41	39K	
caching_fig1.gif	2005-08-23 16:19	16K	
caching_fig1.png	2007-09-11 01:11	13K	
caching_fig1.tr.png	2008-07-10 11:15	11K	
custom_errordocs.png	2007-09-11 01:11	17K	
down.gif	2004-11-20 15:16	56	
favicon.ico	2016-02-12 12:51	1.1K	
feather.gif	2016-02-12 12:51	3.6K	
feather.png	2016-02-12 12:51	5.9K	

Gambar 11. Tampilan Endpoint Yang Terindikasi

Gambar 11 menunjukkan data-data yang terdapat pada url yang terindikasi kerentanan yaitu <https://dinassosial.surabaya.go.id/mxxxx/ixxx>, dimana *file-file* tersebut berisi tentang informasi terkait server yang digunakan oleh *website* Dinas Sosial Surabaya. Hal ini mengindikasikan bahwa *website* tersebut rentan terhadap kerentanan *Browseable Web Directories*.

3.4.9 SQL Injection – SQLite

SQL Injection – SQLite merupakan kerentanan di mana penyerang dapat menyisipkan kode SQL berbahaya untuk memanipulasi *database SQLite*. Halaman *login*

website Dinas Sosial Surabaya terindikasi menunjukkan adanya perbedaan waktu pemrosesan *query* saat parameter *request* ditambahkan skrip SQL, yang awalnya mengindikasikan potensi *Time-Based SQL Injection*. Namun, perbedaan waktu tersebut disimpulkan sebagai hasil dari variasi acak dan beban pada *request*, bukan manipulasi skrip SQL. Pengujian lebih lanjut menggunakan tools sqlmap untuk mendeteksi *SQL Injection* secara otomatis juga tidak berhasil mengakses nama *database*, sehingga *website* ini dinyatakan tidak rentan terhadap *SQL Injection*.

3.4.10 CGI Generic SQL Injection (Blind)

CGI Generic SQL Injection (Blind) adalah teknik eksploitasi pada aplikasi web yang menggunakan *Common Gateway Interface (CGI)* untuk memproses input pengguna, di mana keberhasilan injeksi ditentukan melalui perubahan tidak langsung seperti waktu *response* atau kondisi tertentu. Fitur *search* pada *website* Dinas Sosial Surabaya teridikasi memiliki kerentanan tersebut karena terdapat elemen tambahan pada beberapa parameter. Namun setelah dianalisa, Elemen HTML tambahan yang muncul pada fitur pencarian *website* adalah *response* normal terhadap input pengguna, bukan hasil kesalahan *query*. Oleh karena itu, *website* Dinas Sosial Surabaya tidak rentan terhadap *CGI Generic SQL Injection (Blind)*.

3.5 Analyze & Report

Pada tahap *analyze & report* disajikan temuan dari hasil pengujian yang dilakukan pada *website* Dinas Sosial Surabaya yang kemudian dievaluasi untuk menghasilkan rekomendasi perbaikan.

3.5.1 Penetration Testing Report

Berikut ini adalah laporan pengujian yang dilakukan pada *website* Dinas Sosial Surabaya. Laporan tersebut disajikan dalam bentuk tabel yang berisi nama kerentanan serta status kerentanan tersebut.

TABEL V. LAPORAN HASIL PENETRATION TESTING

No	Kerentanan	Status
1	SQL Injection – SQLite	Tidak ditemukan
2	CGI Generic SQL Injection (blind)	Tidak ditemukan

3	Browsable Web Directories	Ditemukan
4	web.config File Information Disclosure	Ditemukan
5	Absence of Anti-CSRF Tokens	Tidak ditemukan
6	Content Security Policy (CSP) Header Not Set	Ditemukan
7	Multiple XFrame-Options Header Entries	Tidak ditemukan
8	Strict-Transport-Security Header Not Set	Ditemukan
9	Timestamp Disclosure - Unix	Ditemukan
10	X-Content-Type-Options Header Missing	Ditemukan

Tabel 5 merangkum hasil pengujian pada *website* Dinas Sosial Surabaya dimana dari 10 kerentanan yang terindikasi 6 diantaranya terbukti dan berhasil ditemukan, kerentanan tersebut yaitu *Browsable Web Directories*, *web.config File Information Disclosure*, *Content Security Policy (CSP) Header Not Set*, *Strict-Transport-Security Header Not Set*, *Timestamp Disclosure – Unix* dan *X-Content-Type-Options Header Missing*.

3.5.2 Rekomendasi Perbaikan

Berdasarkan standar keamanan OWASP Top 10, pengujian tersebut menemukan celah keamanan yang perlu segera diperbaiki dan berikut ini adalah saran cara untuk memperbaikinya.

1. Untuk mengatasi kerentanan *Browsable Web Directories*, administrator dapat menonaktifkan *directory listing* melalui konfigurasi server dan memastikan direktori tanpa indeks diarahkan ke halaman *error 403*.
2. Untuk mencegah kebocoran informasi melalui *web.config*, administrator harus membatasi akses hanya untuk pengguna berhak dan memperkuat konfigurasi server IIS dengan aturan pembatasan akses. Menambahkan *header HTTP X-Content-Type-Options: nosniff* serta melakukan pengujian rutin akan membantu melindungi *file* dan mengurangi risiko kebocoran data.

3. Untuk meningkatkan keamanan *website* *dinassosial.surabaya.go.id*, disarankan menerapkan *header Content-Security-Policy (CSP)* untuk mengontrol sumber daya yang dimuat. CSP dapat diterapkan melalui *header HTTP* atau elemen `<meta>`, seperti:

```
<meta http-equiv="Content-Security-Policy" content="default-src 'self';  
img-src https://*; child-src  
'none';" /> atau aturan yang lebih spesifik  
untuk subdomain terpercaya.
```

4. Untuk mengatasi kerentanan *Strict-Transport-Security Header Not Set*, administrator server harus menambahkan header HSTS dengan konfigurasi *Strict-Transport-Security: max-age=31536000; includeSubDomains; preload*. Ini memastikan semua koneksi menggunakan HTTPS, menghindari serangan *man-in-the-middle (MITM)*, dan memaksa pengalihan otomatis dari HTTP ke HTTPS.
5. Untuk mengatasi kerentanan *Timestamp Disclosure - Unix*, server web harus dikonfigurasi untuk menyembunyikan informasi *timestamp Unix* dan data waktu sistem dari pengguna. Hal ini dapat dilakukan dengan memodifikasi konfigurasi aplikasi dan server untuk menghindari penyertaan detail waktu dalam *respons HTTP* dan *log error*. Kebijakan *logging* yang aman juga harus diterapkan untuk mencegah pencatatan informasi yang tidak relevan.
6. Untuk mengatasi kerentanan *X-Content-Type-Options Header Missing*, server web harus mengonfigurasi header *X-Content-Type-Options* dengan nilai `"nosniff"` untuk mencegah browser mendeteksi jenis konten secara otomatis, mengurangi risiko serangan seperti *MIME sniffing* dan *Cross-Site Scripting (XSS)*

4. Kesimpulan dan Saran

4.1 Kesimpulan

Hasil penelitian ini, pengujian keamanan *website* Dinas Sosial Surabaya dilakukan dengan metode *penetration testing*, yang mencakup lima tahap yaitu *information gathering*, *footprinting & scanning*, *vulnerability assessment*, *exploitation*, dan *analyze & report* dengan pendekatan yang digunakan untuk evaluasi adalah OWASP Top 10 2021, yang juga menjadi dasar untuk

rekomendasi perbaikan menghasilkan 10 indikasi kerentanan pada *website*, yang kemudian diuji lebih lanjut dan ditemukan 6 kerentanan utama. Kerentanan tersebut meliputi *Browsable Web Directories*, *web.config File Information Disclosure*, *Content Security Policy (CSP) Header Not Set*, *Strict-Transport-Security Header Not Set*, *Timestamp Disclosure - Unix*, dan *X-Content-Type-Options Header Missing*.

4.2 Saran

Untuk penelitian selanjutnya, disarankan agar mengembangkan atau memadukan metode pengujian keamanan lain, seperti NIST Cybersecurity Framework atau ISO/IEC 27001, untuk memperoleh perspektif yang lebih menyeluruh terkait keamanan aplikasi web, terutama di sektor pemerintahan.

5. UCAPAN TERIMA KASIH

Penulis menyampaikan rasa terima kasih yang sebesar-besarnya kepada semua pihak yang telah memberikan dukungan selama proses penelitian ini, baik dalam bentuk dukungan moral maupun material. Tanpa bantuan, bimbingan, dan kontribusi dari berbagai pihak, penelitian ini tidak akan dapat diselesaikan dengan baik dan lancar. Penulis juga menghargai setiap masukan, saran, serta dukungan yang telah diberikan selama pelaksanaan penelitian ini.

Daftar Pustaka:

- [1] Y. W, R. Anto, D. Teguh Yuwono, and Y. Yuliadi, "Deteksi Serangan Vulnerability Pada Open Jurnal System Menggunakan Metode Black-Box," *J. Inform. dan Rekayasa Elektron.*, vol. 4, no. 1, pp. 68–77, 2021, doi: 10.36595/jire.v4i1.365.
- [2] A. M. I. W. Hidayat, "Analisis Perbandingan Sistem Autentikasi Port Knocking dan Single Packet Authorization pada Server Raspbian," vol. 2, no. 1, pp. 28–37, 2019.
- [3] ITU, *Global Cybersecurity Index 2020*. International Telecommunication Union, 2020. [Online]. Available: https://www.itu.int/dms_pub/itu-d/opb/str/D-STR-GCI.01-2021-PDF-E.pdf
- [4] F. Pratiwi, "BSSN Catat 370,02 Juta Serangan Siber ke Indonesia pada 2022 - DataIndonesia.id," 2023. <https://dataindonesia.id/internet/detail/bssn-catat-37002-juta-serangan-siber-ke-indonesia-pada-2022> (accessed Jun.

- 14, 2024).
- [5] BSSN, "Lanskap Keamanan Siber Indonesia," no. 70, 2023, [Online]. Available: <https://www.bssn.go.id/wp-content/uploads/2024/03/Lanskap-Keamamanan-Siber-Indonesia-2023.pdf>
- [6] Firda, S. Putri, Y. B. Utomo, and H. Kurniadi, "Analisa Celah Keamanan Pada Website Pemerintah Kabupaten Kediri Menggunakan Metode Penetration Testing Melalui Kali Linux," *Pros. SEMNAS INOTEK (Seminar Nas. Inov. Teknol.*, vol. 7, no. 1, pp. 52–59, 2023, [Online]. Available: <https://proceeding.unpkediri.ac.id/index.php/inotek/article/view/3411>
- [7] U. S. D. of the Interior, "Penetration Testing | U.S. Department of the Interior," 2023. <https://www.doi.gov/ocio/customers/penetration-testing> (accessed Mar. 07, 2024).
- [8] M. F. F. Ikhsan, E. I. Alwi, and T. Hasanuddin, "Website vulnerability analysis PT . Sadikun Niaga Mas Raya Uses the Owasp Penetration Testing Method," *Int. J. Multidiscip. Res. Growth Eval.*, vol. 05, no. 01, pp. 418–425, 2024.
- [9] D. F. Priambodo, A. D. Rifansyah, and M. Hasbi, "Penetration Testing Web XYZ Berdasarkan OWASP Risk Rating," *Teknika*, vol. 12, no. 1, pp. 33–46, 2023, doi: 10.34148/teknika.v12i1.571.
- [10] Y. Armando and R. Rosalina, "Penetration Testing Tangerang City Web Application With Implementing OWASP Top 10 Web Security Risks Framework," *JISA(Jurnal Inform. dan Sains)*, vol. 6, no. 2, pp. 105–109, 2023, doi: 10.31326/jisa.v6i2.1656.
- [11] I. Odun-Ayo *et al.*, "Evaluating Common Reconnaissance Tools and Techniques for Information Gathering," *J. Comput. Sci.*, vol. 18, no. 2, pp. 103–115, 2022, doi: 10.3844/jcssp.2022.103.115.
- [12] M. F. Safitra, M. Lubis, and A. Widjajarto, "Security Vulnerability Analysis using Penetration Testing Execution Standard (PTES): Case Study of Government's Website," *ACM Int. Conf. Proceeding Ser.*, pp. 139–145, 2023, doi: 10.1145/3592307.3592329.
- [13] Y. Khera, D. Kumar, S. Sujay, and N. Garg, "Analysis and Impact of Vulnerability Assessment and Penetration Testing," *Proc. Int. Conf. Mach. Learn. Big Data, Cloud Parallel Comput. Trends, Perspectives Prospect. Com.* 2019, no. May, pp. 525–530, 2019, doi: 10.1109/COMITCon.2019.8862224.
- [14] M. Mada, "Install Bettercap di Kali Linux 2020.x | by Muhammad Mada | MADATECH | Medium," Jan. 21, 2021. <https://medium.com/madatech/install-bettercap-di-kali-linux-20-x-fa2600ff381f> (accessed Nov. 06, 2024).
- [15] "sqlmap: automatic SQL injection and database takeover tool." <https://sqlmap.org/> (accessed Jun. 14, 2024).
- [16] F. Heiding, E. Süren, J. Olegård, and R. Lagerström, "Penetration testing of connected households," *Comput. Secur.*, vol. 126, 2023, doi: 10.1016/j.cose.2022.103067.
- [17] G. Kusuma, "Implementasi Owasp Zap Untuk Pengujian Keamanan Sistem Informasi Akademik," *J. Teknol. Inf. J. Keilmuan dan Apl. Bid. Tek. Inform.*, vol. 16, no. 2, pp. 178–186, 2022, doi: 10.47111/jti.v16i2.3995.
- [18] "Nessus Vulnerability Scanner." <https://www.tenable.com/products/nessus> (accessed Jun. 14, 2024).