Jurnal Ilmu Komputer dan Sistem Informasi (JIRSI) **e-ISSN 2830-3954**
Volume: 2, Nomor : 2, Mei 2023 : 141-154 **p-ISSN 2830-6031**
https://jurnal.unity-academy.sch.id/index.php/jirsi/index

Analisis Keretanan Website Dengan Aplikasi Owasp Zap

Website Train Analysis With Owasp Zap App

Abdul Fattah Hasibuan*[1], Tommy[2], Divi Handoko[3]
[1,2,3] Prodi Teknik Informatika Universitas Harapan Medan , Jl. HM. Joni No.70 C, Teladan Bar Kota Medan, Sumatera Utara, Indonesia.

[1] hasibuan428@gmail.com *, [2]tomshirakawa@gmail.com, [3]divihandoko@gmail.com

Abstrak

Keamanan website sering kali diabaikan oleh para developer termasuk website kecil atau website profile perusahaan yang mempunyai data yang mungkin bisa dicuri oleh para pihak yang tidak bertanggung jawab. Diperlukannya selft test guna untuk mengetahui celah keamanan yang kemungkinan bisa diserang. Maka dari itu vulnerability assessement diperlukan dalam mencari celah keamanan. Vulnerability Assessment (VA) adalah proses pemindaian sistem untuk mengetahui keretanan dan celah yang ada pada sebuah sistem website, celah ini memberikan backdoor ke penyerang untuk dapat menyerang sistem korban. Owasp-zap merupakan aplikasi yang dapat mencari sebuah celah keamanan yang terdapat pada sistem website, setelah informasi mengenai celah didapatkan maka developer dapat melakukan berubahan atau menambahkan skript terhadap sebuah sistem agar celah keamanan dapat diatasi guna menjaga integritas data dari serangan pihak yang tidak bertanggung jawab.

***Kata Kunci**: OWASP,OWASP-ZAP,Vulnerabillity, Website, Backdoor.*

Abstract

Website security is often ignored by developers, including small websites or company profile websites that have data that may be stolen by irresponsible parties. Self-test is needed in order to find out security holes that might be attacked. Therefore, a vulnerability assessment is needed to find security holes. Vulnerability Assessment (VA) is a system scanning process to find vulnerabilities and loopholes in a website system, this loophole provides a backdoor for attackers to attack the victim's system. Owasp-zap is an application that can look for a security hole in the website system, after information about the vulnerability is obtained, the developer can make changes or add scripts to a system so that security holes can be overcome in order to maintain data integrity from attacks by irresponsible parties.

***Keywords**: OWASP, OWASP ZAP, Vulnerability, Website, Backdoor.*

1. Pendahuluan

Perkembangan website di Indonesia sekarang ini sangat pesat, hal ini terjadi karena semakin bertambahnya jumlah pengguna layanan internet dari tahun ke tahun.

Beberapa website yang sering diakses oleh pengguna diantaranya search engine, e-commerce, sosial media, portal berita dan lain – lain. Akan tetapi dibalik kemudahan layanan yang disediakan oleh setiap website, ternyata terdapat beberapa masalah pada celah keamanan. Dengan memanfaatkan celah kemanan ini seseorang dapat melakukan serangan pada website tersebut.

Keamanan website merupakan satu hal penting dalam perancangan sebuah website. Namun, masih banyak pengembang website yang kurang berhati-hati dalam meningkatkan keamanan website nya. Pengembang situs web harus menerapkan keamanan situs web yang baik di awal perancangan situs web, karena mungkin suatu saat situs web yang telah dibangun akan menjadi target kerusakan oleh peretas. Selain itu, pengembang situs web harus sering mengikuti tren serangan terbaru agar dapat menjaga dan meningkatkan situs web dari hal-hal yang tidak diinginkan. Vulnerability Assessment merupakan sebuah keretanan, kekurangan atau celah pada sistem, yang dapat dimanfaatkan oleh satu atau lebih penyerang untuk melakukan serangan yang dapat membahayakan kerahasiaan, integritas, atau ketersediaan suatu system. Dalam tataran konsep proses VA merupakan proses yang sangat kompleks karena melibatkan seluruh komponen dalam TI. Dalam tataran teknis pun merupakan proses yang beresiko mengingat adanya peluang untuk merusak atau mengganggu kinerja sistem yang berlangsung. Vulnerability testing banyak digunakan untuk meningkatkan kesadaran tentang pentingnya keamanan informasi. Penilaian kerentanan bisa mendeteksi hampir semua celah kerentanan yang biasanya terjadi pada sebuah system. Beberapa faktor dari dari celah keamanan bisa terjadi. karena kurangnya sistem pengamanan website dan kekeliruan programmer ketika melakukanpengcodingan. Salah satu cara untuk melakukan evaluasi keamanan website menggunakan perangkat lunak yang khusus dirancang untuk mengetahui kerentanan yang ada pada suatu sistem yaitu Acunetix Vulnerability Scanner Pentest-tools.com, vulnerability scanner, OWASP ZAP [1].

Berdasarkan dari latar belakang masalah diatas, maka peneliti ingin menganalisis serta mengimplementasi dalam mencari keretanan website dengan menggunakan OWASP ZAP. Dimana dengan menganalisis keretanan website lab.scarpe dapat membantu menigkatkan keamanan dari website tersebut.

2. Metodologi

2.1. Kerangka Kerja Penelitian

Berikut kerangka kerja penelitian yang dilakukan dalam melakukan analisis kerentanan suatu website menggunakan aplikasi OWASP ZAP. Dalam penelitian ini akan menggunakan website Lab.Scarpe sebagai bahan percobaan dalam penelitian ini untuk menguji bagaimana kerentanan website tersebut tehadap serangan-serangan yang dilakukan melalui aplikasi OWASP ZAP. Berdasarkan gambar 1, tahapan yang pertama adalah mengumpulkan data untuk memperoleh informasi yang dibutuhkan dalam rangka mencapai tujuan penelitian. Kemudian melakukan studi literatur dari beberapa sumber jurnal dan buku dalam rangka mencari informasi mengenai penyerangan dan kerentanan website. Selanjutnya

menganalisa kerentanan website menggunakan aplikasi OWASP ZAP dan mencari solusinya. Langkah terakhir menarik kesimpulan dari hasil penelitian yang dilakukan.

Gambar 1. Metode Penelitian

2.2. Pengumpulan Data

Pengumpulan data yang digunakan peneliti ada tiga cara, yaitu :

a. Studi Pustaka

Peneliti mempelajari, meneliti dan menelaah berbagai literatur dari berbagai sumber seperti buku, jurnal
ilmiah, situs internet dan bacaan lainnya yang berkaitan dengan penelitian yang dilakukan.

b. Observasi

Observasi dilakukan untuk memperoleh data dan informasi yang akurat tentang penelitian yang dilakukan dengan cara melakuan penelitian dan percobaan secara langsung pada website yang akan digunakan.

3. Hasil dan Pembahasan

Dalam perkembangannya, teknologi telah banyak mempengaruhi cara manusia dalam melakukan banyak hal termasuk dalam hal mencari suatu jasa, salah satunya website lab.scarpe (labscarpe.my.id) yang merupakan salah satu situs penyedia jasa yang bergerak dibidang kebersihan sepatu dan yang menarik untuk diteliti adalah bagaimana website lab.scarpe dalam menjaga keamanan website. Keamanan website adalah komponen penting untuk melindungi dan mengamankan situs web dan server. Situs web dipindai untuk mengetahui kerentanan dan malware melalui perangkat lunak keamanan situs website. Kerentanan dalam teknologi informasi, adalah cacat dalam kode atau desain yang menciptakan titik kompromi keamanan potensial untuk titik akhir atau jaringan. Kerentanan menciptakan kemungkinan vektor serangan, dimana penyusup dapat menjalankan kode atau mengakses memori sistem target. Tools Owasp-Zap adalah fitur untuk melakukan penetration testing pada sebuah website. Lab.scarpe adalah salah satu platform yang penulis sediakan sebagai bahan analisis. Dengan tool Owasp-Zap pendeteksian kerentanan pada website lab.scarpe dapat dilakukan serta analisa dari hasil kerentanan

tersebut juga dapat disimpulkan agar dapat menjadi acuan untuk memperbaiki sistem lab.scarpe yang telah ada.

3.1 Internet

Menurut [2] Internet adalah komunikasi global yang menghubungkan seluruh komputer di dunia meskipun berbeda sistem operasi dan mesin. Internet adalah jaringan komputer yang menghubungkan antar jaringan secara global, internet dapat juga dapat disebut jaringan alam suatu jaringan yang luas. Seperti halnya jarigan komputer lokal maupun jaringan komputer area, internet juga menggunakan protokol komunikasi yang sama yaitu TCP/IP (Tranmission Control Protocol / Internet Protocol).

Menurut [3] Internet adalah suatu jaringan komunikasi yang memiliki fungsi untuk menghubungkan antara satu media elektronik dengan media elektronik yang lain dengan cepat dan tepat. Jaringan komunikasi tersebut, akan menyampaikan beberapa informasi yang dikirim melalui transmisi sinyal dengan frekuensi yang telah disesuaikan. Untuk standar global dalam penggunaan jaringan internet sendiri menggunakan TCP / IP (Transmission Control Protocol / Internet Protocol). Istilah TCP / IP merupakan bentuk protokol pertukaran paket yang digunakan oleh berbagai pengguna global / dunia. Kemudian, proses untuk menghubungkan antara rangkaian internet disebut dengan "internetworking". Menurut salah satu ahli dalam bidang IT, Onno W. Menjelaskan bahwa pengertian internet adalah suatu media yang digunakan untuk mengefisienkan proses komunikasi menggunakan aplikasi seperti website, email, atau voip. Di zaman yang sudah serba canggih ini keberadaan internet bagaikan kebutuhan pokok yang sulit untuk diabaikan. Internet sangat berpengaruh dalam kehidupan sehari-hari bagi semua kalangan, baik orang dewasa maupun anak-anak. Banyak sekali manfaat internet dalam kehidupan sehari-hari. Berikut ini merupakan beberapa manfaat yang dapat diperoleh dari penggunaan internet bagi manusia dalam masa pandemic maupun tidak.

3.2 Website

Menurut [4] Website adalah kumpulan semua halaman web yang fungsinya untuk menampilkan berbagai informasi dalam bentuk tulisan, gambar dan suara dari sebuah domain yang terbentuk dalam suatu rangkaian yang saling terkait. Suatu halaman web yang sudah terhubung dengan suatu halaman web lain biasanya disebut dengan hyperlink, sedangkan teks yang terhubung dengan suatu halaman web lain biasanya disebut dengan hyperlink, sedangkan teks yang terhubung oleh teks lain disebut sebagai hypertext. Website merupakan kumpulan berbagai halaman media informasi dalam suatu domain yang dapat diakses oleh siapapun menggunakan jaringan internet.

Website adalah kumpulan dari halaman web yang sudah dipublikasikan di jaringan internet dan memiliki domain/URL (Uniform Resource Locator) yang dapat diakses semua pengguna internet dengan cara mengetikan alamatnya. Hal ini dimungkinkan dengan adanya teknologi World Wide Web (WWW) Halaman website biasanya

berupa dokumen yang ditulis dalam format Hyper Text Markup Language (HTML), yang bisa diakses melalui HTTP, HTTPS adalah suatu protokol yang menyampaikan berbagai informasi dari server website untuk ditampilkan kepada para user atau pemakai melalui web browser [2].

3.3 Keamanan Website

Keamanan suatu website merupakan salah satu prioritas yang sangat utama bagi seorang pengololah atau pengguna situs. Kebanyakan pengguna hanya mengutamakan design tampilan dan konten apa supaya menarik pengunjung sebanyak banyaknya. Jika seorang pengolah atau pengguna mengabaikan keamanan suatu website maka yang dirugikan adalah pengguna itu sendiri karena seseorang dapat mengambil data-data penting pada suatu website dan bahkan pula dapat dapat mengacak-acak tampilan website tersebut. Paling utama keamanan sebuah situs adalah melindungi komputer, aplikasi dan jaringannya dengan tujuan mengamankan informasi yang berada didalamnya [5].
Menurut [6] Keamanan merupakan sebagian aspek utama yang ada pada sistem jaringan internet. Pada dasarnya internet dibangun melalui jaringan komputer yang saling terhubung, dalam hal ini banyaknya pengguna yang dihubungkan dalam suatu jaringan berpengaruh terhadap keamanan data atau informasi, serta keamanan informasi menjadi rentan terhadap serangan dan dapat disalahgunakan.

3.4 OWASP

Open Web Application Security Project (OWASP) adalah sebuah framework yang bersifat open source yang berfokus dalam memperbaiki keamanan software aplikasi. OWASP merupakan organisasi yang dibangun untuk menemukan celah keamanan dari sebuah aplikasi website. Berdasarkan standar yang dikeluarkan oleh OWASP terdapat sebelas langkah yang dapat dilakukan untuk menilai dan menguji keamanan pada sebuah website, berupa: Information Gathering, Configuration Management, Secure Transmission, Authentication, Session Management, Authorization, Cryptography, Data Validation, Denial of Service, Error Handling" [7].
Menurut [5] Open Web Aplication Security project (OWASP) adalah sebuah organisasi internasional yang bersifat non-profit, didirikan oleh Open Web Aplication Security project (OWASP) foundation pada 21 April 2004 diAmerika serikat. Open Web Aplication Security project (OWASP) fokus pada peningkatan keamanan perangkat lunak dan didedikasikan untuk memungkinkan organisasi dalam mengembangkan, memperoleh, mengoprasikan dan memelihara aplikasi terpercaya untuk menjamin keamanan yang dibuat atau dikembangkan. Open Web Aplication Security project (OWASP) memiliki misi untuk mengamankan software sehingga orang-orang dan organisasi dapat membuat keputusan terhadap resiko keamanan yang benar.

3.5 OWASP-ZAP

Zed Attack Proxy (ZAP) adalah alat pengujian penetrasi open-source gratis yang dikelola di bawah payung Open Web Application Security Project (OWASP). ZAP

dirancang khusus untuk menguji aplikasi web dan fleksibel serta dapat diperluas. Pada intinya, ZAP adalah apa yang dikenal sebagai "proxy man-in-the-middle." Itu berdiri di antara browser penguji dan aplikasi web sehingga dapat mencegat dan memeriksa pesan yang dikirim antara browser dan aplikasi web, memodifikasi konten jika diperlukan, dan kemudian meneruskan paket tersebut ke tujuan. Ini dapat digunakan sebagai aplikasi yang berdiri sendiri, dan sebagai proses daemon. OWASP ZAP (Zed Attack Proxy) merupakan sebuah aplikasi yang digunakan untuk penetration testing dalam menemukan vulnerabilities/celah keamanan pada suatu aplikasi website. ZAP menyediakan scanner secara automatis [7]. OWASP ZAP adalah sebuah tools vulnerabilities scanner yang dibuat oleh organisasi OWASP tools ini adalah suatu proyek dari OWASP yang paling aktif karena terus dikembangkan tools ini bersifat open source sehingga siapa saja juga bisa mengembangkan tools ini [8].

3.6 OWASP TOP 10

Menurut [8] OWASP Top 10 atau yang biasa disebut OWASP 10 adalah sebuah metode yang dirilis oleh komunitas OWASP yang berisikan 10 daftar teratas celah keamanan yang dapat mengancam keamanan suatu website daftar ini terus berkembang dan berubah-ubah mengikuti perkembangan teknologi website yang terus berkembang.

Parameter yang digunakan pada OWASP TOP 10 adalah sebagai berikut:

1. A1 – Injection Celah injeksi, seperti SQL, OS dan LDAP injection terjadi ketika data yang berbahaya dikirim ke interpreter sebagai bagian dari perintah atau query. Data berbahaya milik penyerang dapat mengelabui interpreter untuk mengeksekusi perintah yang tidak diinginkan atau mengakses data secara ilegal.
2. A2 Broken Authentication Fungsi aplikasi yang berhubungan dengan otentikasi dan manajemen sesi sering tidak diterapkan dengan benar, sehingga memungkinkan penyerang untuk mengambil password-password, kunci-kunci, token-token sesi, atau untuk mengeksploitasi celah implementasi lainnya untuk mengambil identitas pengguna.
3. A3 Sensitive Data Exposure Banyak aplikasi web yang tidak melindungi data rahasia dengan baik. Seperti kartu kredit, nomor PIN, dan kewenangan-kewenangan otentikasi. Penyerang akan mencuri atau merubah data yang diproteksi dengan lemah, untuk melakukan tindak pencurian identitas, kejahatan lewat kartu kredit, pencurian identitas, atau kejahatan lainnya. Data rahasia pantas terlindungi dengan baik menggunakan sandi enkripsi ketika tinggal atau dalam transit, sebagai pencegahan khusus atas tidak kejahatan ketika ketika tampil di browser.
4. A4 XML External Entities (XXE) Banyak bahasa XML versi lebih tua atau yang tidak dikonfigurasi dengan baik mengevaluasi referensi entitas eksternal dalam dokumen XML. Entitas eksternal dapat digunakan untuk mengungkapkan file internal menggunakan URL file, pembagian file internal, pemindaian port internal, eksekusi kode jarak jauh, dan penolakan serangan layanan.

5. A5 Broken Access Control Pembatasan pada apa yang diizinkan oleh pengguna yang diautentikasi sering kali tidak dilakukan dengan benar. Penyerang dapat mengeksploitasi keretanan ini untuk mengakses fungsionalitas dan / atau data yang tidak sah, seperti mengakses akun pengguna lain, melihat file sensitif, memodifikasi data pengguna lain, mengubah hak akses, dll.
6. A6 Security Misconfiguration Keamanan yang baik memerlukan konfigurasi keamanan yang terperinci dan telah menyeluruh pada framework aplikasi, aplikasi pada server, web server, database dan sistem operasi. Semua pengaturan ini harus didefinisikan, diimplementasikan dan dipelihara, karena terdapat banyak aplikasi yang dirilis tanpa konfigurasi default yang aman. Termasuk menjaga semua software untuk tetap terbarukan (up to date).
7. A7 Cross-Site Scripting (XSS) Celah XSS terjadi ketika sebuah aplikasi mengambil data berbahaya dan mengirimkannya ke web browser dengan tanpa memvalidasi atau melepaskan konten tersebut dengan benar. XSS mengijinkan penyerang mengeksekusi script di browser target sehingga dapat leluasa mengambil alih data pengguna, merubah tampilan website target, atau mengarahkan korban ke laman yang berbahaya.
8. A8 Insecure Deserialization Eksploitasi deserialization adalah agak sulit, seperti di luar rak eksploitasi jarang bekerja tanpa perubahan atau menyesuaikan eksploitasi kode yang mendasarinya.
9. A9 Using Components with Known Vulnerabilities Komponen, seperti libraries, frameworks, dan modul perangkat lunak lainnya, dijalankan dengan hak yang sama seperti aplikasi. Jika komponen rentan dieksploitasi, serangan semacam itu dapat memfasilitasi hilangnya data serius atau pengambilalihan server. Aplikasi dan API yang menggunakan komponen dengan kerentanan yang diketahui dapat merusak pertahanan aplikasi dan memungkinkan berbagai serangan dan berdampak.
10. A10 Insufficient Logging & Monitoring Logging/ Pencatatan dan pemantauan yang tidak memadai, ditambah dengan integrasi yang hilang atau tidak efektif dengan respons insiden, memungkinkan penyerang untuk menyerang system lebih lanjut, me mpertahankan exploitasi, beralih ke lebih banyak sistem, dan mengutakatik, mengekstrak, atau menghancurkan data. Sebagian besar studi pelanggaran menunjukkan waktu untuk mendeteksi pelanggaran lebih dari 200 hari, biasanya terdeteksi oleh pihak eksternal daripada proses internal atau pemantauan.

3.7 Implementasi

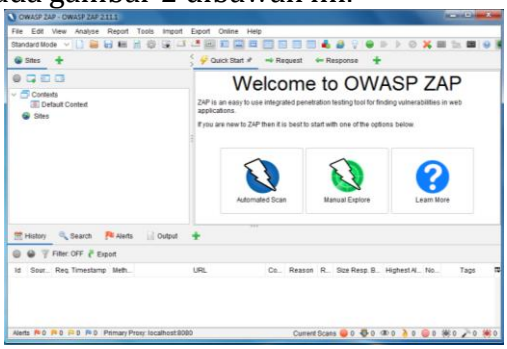
Implementasi sistem dilakukan dengan meyerang website labscarpe.my.id dengan menggunakan aplikasi owasp-zap yang nanti akan diujicoba untuk mengetahui kerentanan atau celah yang terdapat di website labscarpe.my.id. Adapun halaman halaman yang terdapat pada website lab.scarpe. Solusi pengamanan web dari gangguan atau serangan hacker, dapat dilakukan dengan cara self test yaitu pengujian yang dilakukan terhadap web lab.scarpe secara legal dengan aktifitas menyerupai hacker. Self test dapat dilakukan dengan menggunakan aplikasi Owasp-

Zap dengan automated scanner atau manual expore yang bertujuan untuk mendapatkan hasil keretanan terhadap website agar dapat menghindari serangan yang tidak diinginkan terjadi pada website tersebut.

Maka dari itu diperlukan pengujian terhadap suatu website guna untuk menjaga keamanan data dari pihak yang tidak bertanggung jawab. Hal yang dibutuhkan untuk scanning vulnerability adalah sebuah link website yang sudah dipublikasikan ke internet dan sudah mendapaatkan ijin untuk dapat melakukan scanning vulnerability website.

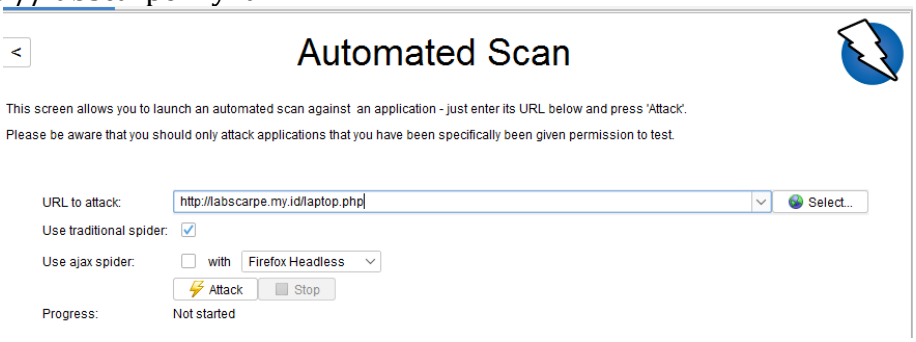
Berikut adalah langkah-langkah dalam mencari celah keamanan dengan menggunakan owasp-zap melalui Automated Scanner dan Manual Expore:

Pertama silahkan buka apliaksi Owap-Zap. Jika sudah terbuka maka akan ada 2 pilihan yang disediakan oleh aplikasi owasp-zap yaitu Automated scan dan Manual expore, dapat dilihat pada gambar 2 dibawah ini.



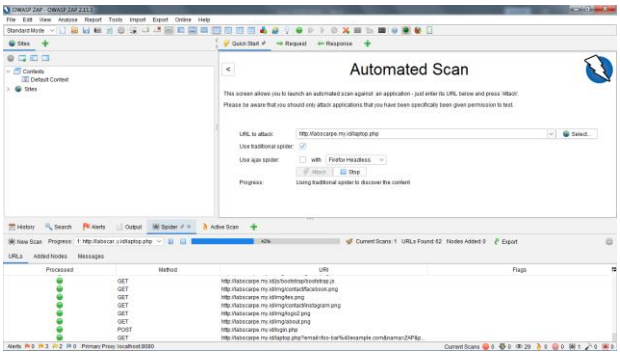
Gambar 2. Tampilan Awal Owasp-Zap.

Jiika ingin menggunakan Automated Scan maka klik Automated Scan, lalu masukan link http://labscarpe.my.id



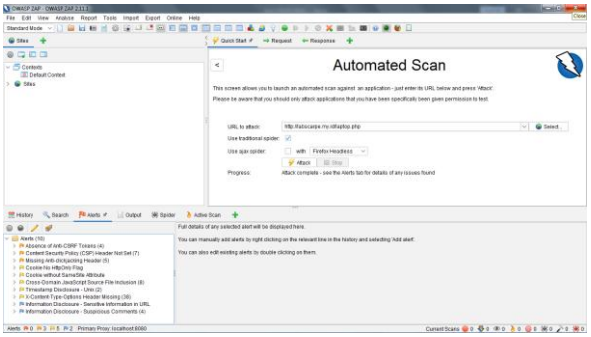
Gambar 3. Tampilan Automated Scan.

Kemudian klik attack maka aplikasi owasp-zap akan langsung memperoses hasil dari link tersebut, bisa dilihat pada gambar 4.



Gambar 4. Tampilan Proses Automated Scan.

Setelah proses scanning sudah selesai maka akan terlihat beberapa kerentanan, menurut Owasp-Zap website lab.scarpe memiliki 8 keretanan yang diantaranya 3 kerentanan dengan level medium, 5 kerentanan dengan level low. Bisa dilihat pada gambar 5.



Gambar 5. Tampilan Hasil Kerentanan.

Berdasarkan hasil pengujian yang dilakukan pada website lab.scarpe melalui Vulnerability Scanner tools owasp-zap didapatkan 8 kerentanan yaitu:

Tabel 1. Hasil Vulnerability Assessment

Nama	Risk	Number Of Instances
Absence of Anti-SCRF Tokens	Medium	2
Contect Security Polici (CSP) Header Not Set	Medium	7
Missing Anti-clickjacking Header	Medium	5
Cookie No HttpOnly Flag	Low	1
Cookie without Samesite Attribute	Low	1
Cross-Domain JavaScript Souce File Inclusion	Low	21
Timestamp Disclosure - Unix	Low	2
X-Content-Type-Option Header Missing	Low	13

Dari 8 kerentanan yang telah didapatkan maka hal yang harus dilakukan adalah memberikan solusi terkait keretanan yang dapat menyebabkan website terserang

pihak yang tidak bertanggung jawab atau kemungkinan terjadinya kebocoran data. Untuk solusi masalah terkait dapat dilihat pada penjelasan di bawah ini.

A. Absence Of Anti-CSRF Tokens

Absence Of Anti-CSRF Tokens yang terdeteksi melalui aplikasi owasp-zap. Serangan cross site request forgery (CSRF) dapat terjadi disebabkan karena tidak adanya mekanisme perlindungan terhadap token keamanan (request token) pada sebuah website, sehingga penyerang dapat mengirim suatu request (sumbit suatu form) secara illegal. Serangan CSRF dapat sukses dilakukan dengan memaksa pengguna untuk melakukan permintaan mengubah data seperti profil pribadi, alamat email, dan bahkan yang lebih berbahaya adalah melakukan transfer dana.
Ada beberapa cara yang dapat dilakukan dalam mengatasi celah keamanan tersebut dengan mengaktifkan CSRF protection menggunakan framework Codeigniter 3.0. Mekanisme CSRF protection sebenarnya sudah ada di Codeigniter sejak versi 2.0 namun di versi Codeigniter 3.0 terdapat beberapa pengaturan tambahan yang disediakan untuk meningkatkan tingkat keamanan. Dengan mengaktifkan CSRF protection maka untuk semua POST request harus menyertakan CSRF token agar request tersebut dinilai valid. Ada dua cara yang dapat dilakukan untuk mengatasi hal ini. Yang pertama bisa menambahkan action atau fungsi controller yang berupa request AJAX ke dalam konfigurasi $config[‘csrf_exclude_uris’] sehingga CSRF protection akan di hilangkan untuk action atau fungsi tersebut. Dan yang kedua bisa menambahkan CSRF token untuk semua AJAX request POST yang ingin dilindungi. Jika AJAX request menggunakan jquery, maka dapat menggunakan fungsi $.ajaxSetup untuk secara otomatis menyertakan CSRF token di semua AJAX request yang telah dibuat.

```
var csfrData = {};
csfrData['<?php echo $this->security->get_csrf_token_name
(); ?>'] = '<?php echo
$this->security->get_csrf_hash(); ?>';
$.ajaxSetup({
data: csfrData
});
```

Gambar 6. Perlindungan CSRF menggunakan JavaScript.

B. Content Security Policy (CSP) Header Not Set
Content Security Policy (CSP) header atau meta tag untuk mengatur resource dari sumber mana saja yang diperbolehkan untuk dieksekusi oleh web browser, yang dimaksud resource seperti javascript, css, font image, video, audio, dan sebagainya. Dengan content security policy dapat meminimalisir potensi-potensi seragan XSS pada website dengan memerintahkan browser untuk tidak me-load dan mengeksekusi resource dari sumber yang tidak diinginkan. Penerapan content security policy dapat dituliskan pada response header, dan berikut ini adalah format penulisan header content security policy sebagai berikut:

```
<directive-a> <source-1> <source-2> <source-n>
```

Gambar 7. Gambar Penulisan Policy Directive.

C. Missing Anti Clickjacking Header

Missing anti clickjacking header adalah sebuah keretanan yang mengakibatkan website dapat terserang serangan clickjacking. Serangan ini bertujuan untuk mengelabui user agar mengklik sesuatu yang sebenarnya tidak diinginkan oleh user. Bahayanya, identitas pribadi seperti email dan password bisa dicuri melalui teknik penyerangan ini. Untuk menghindari hal tersebut diperlukan HTTP header tambahan yaitu X-Frame-Option, X Frame Option berfungsi untuk menjaga keamanan dari serangan sejenis Clickjacking yaitu dengan mengnonaktifkan iframe atau object didalam website. Berikut bisa dilihat pada gambar 9 adalah contoh X Frame Option yang hanya mengijinkan request dari url sendiri yang terletak didalam .htaccess :

```
<httpProtocol>
    <customHeaders>
        <remove name="x-powered-by"/>
        <add name="x-frame-option" value="DENY"/>
    </customHeader>
</httpProtocol>
```

Gambar 8. Penerapan X Frame Option.

D. Cookie No HttpOnly Flag

Keretanan dari Cookie No HttpOnly flag dapat menyebabkan terjadinya serangan Cross Site Scripting (XSS), kebocoran dari Cookie juga dapat menyebabkan penyerang dengan mudah mengakses Cookie dan menggunakan ini sebagai media mencari tahu atau mengambil informasi sensitif yang terkandung dalam Cookie. Keretanan ini dapat di atasi dengan setcookie() dan setrawcookie(), dan bisa mennambahkan sintaks pada .htaccess. Bisa dilihat pada Gambar 3.17 berikut adalah contoh sintaks setcookie() dan setrawcookie():

```
setcookie($name, $value, $expire, $domain, $secure, $httponly)
setrawcookie($name, $value, $expire,$domain, $secure, $httponly)
```

Gambar 9. Perilindungan Dari Serangn XSS.

E. Cookie without SameSite Attribute

Samesite adalah properti yang dapat diatur dalam cookie HTTP untuk mencegah serangan cross site request forgery (CSRF). Saat Samesite diatur ke Lax, cookie dikirim ke dalam permintaan di situs yang sama dan dalam permintan GET dari situs lain, kemudian tidak dikirim ke dalam GET yang merupakan cross domain. Jika Cookie ingin hanya bisa diakses dalam konteks pihak pertama, developer memiliki opsi untuk menerapkan salah satu dari dua configurasi Samesite=Lax atau Samesite=Strict guna mencegah akses eksternal. Berikut adalah gambar sintaks dari penggunaan samesite=strict dan samesite=Lax :

```
app.post("login", (req, res)){
    cost cookie = "user-fattah; samesite=strict; secure";

    res.setHeader("set-cookie", [cookie])
    res.send("ok")
}
```

```
app.post("login", (req, res)){
    cost cookie = "user-fattah; samesite=lax; secure";

    res.setHeader("set-cookie", [cookie])
    res.send("ok")
}
```

Gambar 10. Contoh Dari Penggunaan Samesite=Strict dan Samesite=Lax.

F. Cross-Domain JavaScript Source Attribut

Peringatan keamanan yang dapat mempengaruhi sistem website yang menjalankan satu atau beberapa file Javascript dari domain pihak ketiga. Jika pihak ketiga sengaja atau tidak disengaja menyimpan konten berbahaya konten tersebut dapat ditambahkan dan dijalankan di sistem website. Kemungkinan ini bisa terjadi karena javascript eksternal tidak divalidasi hal ini dapat meyebabkan data pengguna bocor. Data sensitif bisa berupa data autentikasi pengguna seperti token, ID sesi, cookie atau informasi pribadi email, nomor telpon, alamat. Kelemhan ini juga kurang lebih sama dengan Cookie without Samesite Attribute, Cookie No HttpOnly flag.

G. Timestamp Disclosure Unix

Timestamp Disclosure – Unix merupakan kerentanan yang disebabkan oleh tampilnya informasi timestamp unix pada browser. Kerentanan ini dapat dimanfaatkan oleh penyerang sebagai sarana pengumpulan informasi untuk melakukan serangan. Verifikasi dilakukan dengan cara memeriksa timestamp yang tampil mengandung informasi penting atau tidak secara manual. Masalah kerentanan ini bisa diabaikan jika timestamp yang ditampilkan tidak kritis, jika tidak code program pada sistem harus dimodifikasi untuk tidak mengungkapkan informasi timestamp.

H. X Content Type Option

X Content Type Options adalah salah satu komponen HTTP Security Header yang memberikan response terhadap permintaan Multipurpose Internet Mail Extensions atau MIME type dan itu menunjukkan sifat dan format file dokumen. Opsi X Content Type Options digunakan untuk mengamankan data MIME dari kemungkinan sniffing. Sniffing sebuah aktifitas untuk memonitor dan mengcapture semua jenis konten yang melewati jaringan, beberapa contohnya adalah MIME type yang meliputi script, style. Untuk mengatasi masalah ini harus mengaktifkan fungsi X Content di .htaccess yaitu dengan menambahkan:

```
Header set X-Content-Type-Option nosniff
```

Gambar 11. Penggunaan dari X-Content-Type-Option.

4. Kesimpulan

Dari analisis yang dilakukan peneliti analisis kerentanan website dengan aplikasi owasp-zap pada website Lab.Scarpe didapatkan kesimpulan, pencarian kerentanan dengan menggunakan aplikasi Owasp Zap dapat dilakukan, hal tersebut dibuktikan dengan hasil pengujian yang telah didapatkan oleh peneliti. Langkah yang dilakukan untuk mendapatkan hasil pengujian dengan menggunakan Owasp-Zap diantaranya: input URL target yang ingin dilakukan vulnerability assessment, lalu dilakukan pencarian index pada target pengujian dan mendapatkan hasil report keseluruhan. Di dapatkan 8 kerentanan yang diantaranya, 3 kerentanan level medium yaitu Absence of Anti-SCRF Tokens, Contect Security Polici (CSP) Header Not Set, Missing Anti-clickjacking Header, dan 5 kerentanan level medium yaitu Cookie No HttpOnly Flag, Cookie without Samesite Attribute, Cross-Domain JavaScript Souce File Inclusion, Timestamp-Type-Option Header Missing, X-Content-Type-Option Header Missing. Dan beberapa kerentanan ini dapat menyebabkan serangan diantaranya, serangan SCRF (Cross-Site Request Forgery) yang menyebabkan user mengeksekusi suatu tindakan yang sebenarnya tidak dikehendaki oleh user itu sendiri contohnya seperti menghapus akun secara permanen, mentranfer uang ke suatu akun, dan Serangan XSS yang bertujuan mencuri data sensitif melalui inputan website yang sudah di injection coding berbahaya sehingga hal yang di input tidak terkirim ke website sebenarnya melainkan terkirim ke pihak yang tidak bertanggung jawab.

DAFTAR PUSTAKA

[1] A. Zirwan, “Pengujian dan Analisis Kemanan Website Menggunakan Acunetix Vulnerability Scanner,” J. Inf. dan Teknol., vol. 4, no. 1, pp. 70–75, 2022, doi: 10.37034/jidt.v4i1.190.

[2] . N., A. Ibrahim, and A. Ambarita, “Sistem Informasi Pengaduan Pelanggan Air Berbasis Website Pada Pdam Kota Ternate,” IJIS - Indones. J. Inf. Syst., vol. 3, no. 1, p. 10, 2018, doi: 10.36549/ijis.v3i1.37.

[3] D. Maharani, F. Helmiah, and N. Rahmadani, “Manfaat Menggunakan Internet dan Website P2ada Masa Pandemi Covid-19,” J. Pengabdi. Masy. Inform., vol. 1, no. 1, pp. 1–7, 2021, doi: 10.25008/abdiformatika.v1i1.130.

[4] T. A. Kinaswara, N. R. Hidayati, and F. Nugrahanti, “Rancang Bangun Aplikasi Inventaris Berbasis Website Pada Kelurahan Bantengan | Kinaswara | Prosiding Seminar Nasional Teknologi Informasi dan Komunikasi (SENATIK),” Pros. Semin. Nas. Teknol. Inf. dan Komun., vol. 2, no. 1, pp. 71–75, 2019, [Online]. Available: http://prosiding.unipma.ac.id/index.php/SENATIK/article/view/1073.

[5] Y. Mulyanto, E. Haryanti, and J. Jumirah, “Analisis Keamanan Website Sman 1 Sumbawa Menggunakan Metode Vulnerability Asesement,” J. Inform. Teknol. dan Sains, vol. 3, no. 3, pp. 394–400, 2021, doi: 10.51401/jinteks.v3i3.1260.

[6] G. H. Editya and S. Mulyati, “Aplikasi Mobile One Time Password Menggunakan Algoritma MD5 dan SHA1 untuk Meningkatkan Keamanan Website,” Skanika, vol. 1, no. 2, pp. 618–623, 2018.

[7] G. Guntoro, L. Costaner, and M. Musfawati, "Analisis Keamanan Web Server Open Journal System (Ojs) Menggunakan Metode Issaf Dan Owasp (Studi Kasus Ojs Universitas Lancang Kuning)," JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform., vol. 5, no. 1, p. 45, 2020, doi:

[8] Y. Yudiana, A. Elanda, and R. L. Buana, "Analisis Kualitas Keamanan Sistem Informasi E-Office Berbasis Website Pada STMIK Rosma Dengan Menggunakan OWASP Top 10," CESS (Journal Comput. Eng. Syst. Sci., vol. 6, no. 2, p. 185, 2021, doi: 10.24114/cess.v6i2.24777.10.29100/jipi.v5i1.1565.