ANALISIS KEAMANAN SITUS WEB PEMERINTAH BANYUMAS PADA DOMAIN SUDAGARAN.DESA.ID DENGAN METODE PENETRATION TESTING MENGACU PADA OWASP TOP 10

1st Muhammad Yafi Akmal
Mahasiswa S1 Teknik Informatika
Universitas Telkom Purwokerto
Purwokerto, Indonesia
yafiakmal45@gmail.com

2nd Alon Jala Tirta Segara
Dosen S1 Rekayasa Perangkat Lunak
Universitas Telkom Purwokerto
Purwokerto, Indonesia
alon@ittelkom-pwt.ac.id

3rd Wahyu Adi Prabowo
Dosen S1 Teknik Informatika
Universitas Telkom Purwokerto
Purwokerto, Indonesia
wahyup@telkomuniversity.ac.id

***Abstrak* — Meningkatnya penggunaan internet dari tahun ke tahun seiring dengan perkembangan pesat teknologi informasi dan komunikasi, termasuk website, menjadikan website sebagai salah satu media penting yang digunakan pemerintah untuk menyebarkan dan memperoleh informasi. Namun, perhatian khusus perlu diberikan terhadap keamanan dalam pengembangan website pemerintah, mengingat banyaknya kasus kebocoran data masyarakat yang bersumber dari server milik pemerintah. Hal ini dapat menurunkan kepercayaan masyarakat kepada pemerintah.**

Website Desa Sudagaran yang dikelola oleh Kemendikbud menjadi objek penelitian ini untuk mengidentifikasi dan mengevaluasi tingkat keamanannya. Penelitian dilakukan menggunakan metode penetration testing dan mengacu pada OWASP Top 10, dengan langkah-langkah identifikasi kerentanan dan simulasi penyerangan pada kerentanan yang ditemukan.

Hasil implementasi menunjukkan adanya beberapa kerentanan, seperti SQL Injection, Cross-Site Scripting (XSS), dan kelemahan konfigurasi keamanan. Rekomendasi mitigasi mencakup penerapan header keamanan, validasi input, penggunaan HTTPS, dan pembaruan komponen yang usang. Implementasi langkah mitigasi ini diharapkan dapat meningkatkan keamanan website dan memulihkan kepercayaan masyarakat terhadap layanan berbasis digital.

***Kata kunci*— owasp, penetration testing, website, pemerintah, Banyumas**

I. PENDAHULUAN

Penggunaan internet mengambil tempat yang penting dalam kehidupan sehari hari. Di era digitalisasi dan informasi saat ini, aplikasi web menjadi sesuatu yang penting [1]. Pada tahun 2018, Asosiasi Penyelenggara Jasa Internet Indonesia menyebutkan jumlah pengguna internet di Indonesia tahun 2018 adalah 64,8% dari jumlah penduduk Indonesia. Jumlah tersebut mengalami peningkatan dibandingkan tahun 2017 yakni 54,68%[2]. Sebuah laporan digital dari We Are Social menyoroti bahwa pengguna internet Indonesia mencapai 212,0 juta pada tahun 2023, yaitu setara dengan 77% total populasi di Indonesia[3].

Situs web menjadi alternatif utama bagi perusahaan untuk berkomunikasi, promosi dan berinteraksi dengan khalayak [3]. Tidak hanya perusahaan seperti media berita, instansi pendidikan, sosial media. Pemerintah juga telah menerapkan website sebagai media untuk memudahkan penyebaran informasi kepada masyarakat luas. Dengan adanya website maka kualitas layanan dapat meningkat serta menyediakan peluang yang bagus agar masyarakat dapat berpartisipasi dalam proses demokrasi [4]

Mempertimbangkan data masyarakat yang dikumpulkan dan dikelola oleh pemerintah. Oleh karena itu keamanan data masyarakat perlu dijadikan perhatian lebih oleh pemerintah. Memastikan keamanan penggunaan internet dan sumber dayanya menjadi sangat penting melihat meningkatnya pengguna teknologi ini. Kebocoran data dapat mempengaruhi seseorang dalam hal reputasi, uang dan peluang kehilangan data [5]. Terkait dengan perlindungan data pribadi, Indonesia belum memiliki aturan khusus mengenai perlindungan data pribadi pada level undang-undang. Meskipun demikian, setidaknya terdapat 30 ketentuan perundang-undangan yang mengatur mengenai kewajiban untuk memberikan perlindungan data pribadi diindonesia[2].

Sayangnya masih banyak dugaan kasus kebocoran data yang bersumber dari server pemerintahan. Bocornya ratusan juta data pribadi warga Indonesia yang dipegang oleh BPJS kesehatan[6], 337 juta data masyarakat yang diduga dikelola Direktorat Jenderal Kependudukan dan Pencatatan Sipil Kementerian Dalam Negeri bocor[7], 6 juta data pasien diduga bocor dan dijual di forum online[8].

Selain itu Badan Siber dan Sandi Negara atau dapat di singkat BSSN merangkum trafik anomali serangan siber di Indonesia pada 2023 dengan total 403.990.813 trafik. Hal ini membuktikan besarnya kemungkinan untuk terjadinya penurunan performa perangkat dan jaringan, pencurian data sensitif, hingga perusakan reputasi dan penurunan kepercayaan terhadap suatu organisasi[9].

Melihat banyaknya kasus kebocoran data dari server pemerintah dimana data masyarakan menjadi korban, maka perlu dilakukan analisis keamanan pada website

agar dapat menjadi langkah awal pemerintah sebelum terjadinya eksploitasi celah keamanan website dari orang-orang yang tidak bertanggung jawab.

Sudagaran merupakan salah satu desa yang berlokasi di Kabupaten Banyumas, Provinsi Jawa Tengah, Indonesia. Desa Sudagaran menjadi salah satu desa yang menerapkan teknologi website. Website desa Sudagaran berlokasi di internet dengan domain sudagaran.desa.id. Website desa Sudagaran dikelola langsung oleh Dinas Komunikasi dan Informatika kabupaten Banyumas. Website desa Sudagaran lah yang akan menjadi objek penelitian ini.

Penelitian ini bertujuan untuk menguji serta menganalisis apakah website pemerintah yang dikelola Dinas Komunikasi dan Informatika kabupaten Banyumas yang secara spesifik berdomain sudagaran.desa.id, memiliki keamanan yang sesuai standar owasp top 10.

Penelitian dilakukan dengan metode penetration testing dan berusaha mengidentifikasi apa saja dari daftar OWASP top 10 2021 yang dimiliki oleh website sudagaran.desa.id, domain ini dipilih karena telah mendapat izin secara lisan dari dinas komunikasi dan informatika kabupaten banyumas. Pendekatan penetration testing yang digunakan adalah pendekatan black box.10 Daftar celah keamanan yang dirilis oleh OWASP akan dijadikan acuan sejauh mana penetration testing yang akan dilakukan.

II. KAJIAN TEORI

A. Penetration Testing

Penetration testing atau biasa disingkat pentest adalah pengujian keamanan dengan cara melakukan simulasi serangan siber dengan tujuan menemukan celah keamanan pada sistem komputer atau aplikasi sebelum serangan benar-benar terjadi[10]. Hasil dari pentest pada penelitian ini akan memberikan laporan celah-celah keamanan yang dimiliki sistem kemudian memberikan rekomendasi sebagai bahan evaluasi untuk pencegahan lebih dini.

B. Black Box Testing

Dalam melakukan penetration testing ada 2 metode berdasarkan perspective. Pertama adalah ketika seorang tester memiliki pengetahuan mendalam dari arsitektur, konfigurasi, dan source code sistem target secara menyeluruh atau biasa disebut white box testing. Kemudian ada black box testing yang berfokus pada external perspective dari sistem target[11]. Artinya black box testing tidak memiliki informasi lebih banyak dari white box testing karena tidak memiliki akses ke source code dan sistem[12].

C. Penetration Tools

Penetration tools adalah sebuah alat berbentuk perangkat lunak yang dikembangkan dengan tujuan membantu *pentester* dalam melakukan *penetration testing* agar lebih cepat dan efektif. Beberapa alat yang umum digunakan adalah[12]:

1. Sistem operasi: Kali Linux
2. Credential-cracking tools, untuk mematahkan enkripsi dan melakukan brute-force attack dengan otomatis: medusa, hydra, .john the ripper.
3. Port scanner: nmap, masscan, ZMapp.
4. Vulnerability scanner: Nessus, Core Impact, Netsparker. Khusus web: burp suite, owaspzap.
5. Packet analyzer: wireshark, tcpdump.

Open Web Application Security Project(OWASP)

Open Web Application Security Project atau disingkat sebagai OWASP merupakan komunitas terbuka yang bekerja untuk meningkatkan keamanan perangkat lunak. Berdedikasi untuk membuat sebuah organisasi yang bertujuan untuk menyusun, mengembangkan, memperoleh, dan memelihara aplikasi yang dapat dipercaya. Semua proyek yang dikembangkan oleh OWASP tidak dipungut biaya. OWASP mulai di jalankan pada Desember 2001, dan menjadi badan amal nirlaba Amerika Serikat pada 21 April 2004[13].

OWASP merilis 10 kerentanan yang paling umum terdapat pada *website* atau biasa disebut *OWASP Top 10.* Daftar ini menjadi salah satu dokumen standar serta acuan utama *developers* dan *web application security* dalam melakukan langkah preventif untuk mengamankan *website.*

D. OWASP Top 10 2021

OWASP Top 10 adalah daftar 10 *vulnerability* yang sering dijumpai pada sebuah aplikasi web. Daftar ini berkembang seiring perkembangan teknologi *website.* OWASP Top 10 dibuat dengan tujuan untuk meningkatkan kesadaran tentang keamanan pada aplikasi web dengan mengidentifikasi beberapa risiko celah keamanan yang umum dijumpai dan memiliki risiko tinggi[14].

Daftar OWASP Top 10 terdiri dari:

1. A01:2021-Broken Access Control
2. A02:2021-Cryptographic Failures
3. A03:2021-Injection
4. A04:2021-Insecure Design
5. A05:2021-Security Misconfiguration
6. A06:2021-Vulnerable and Outdated Components
7. A07:2021-Identification and Authentication Failures

8. A08:2021-Software and Data Integrity Failures

9. A09:2021-Security Logging and Monitoring Failures

10. A10:2021-Server-Side Request Forgery

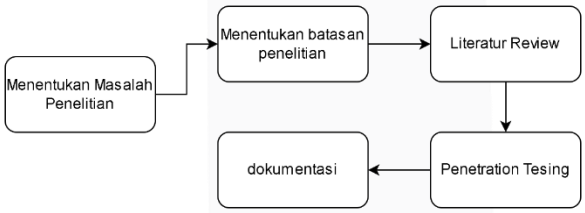
E. CWE

CWE adalah daftar kerentanan perangkat lunak dan perangkat keras yang bisa berakibat pada keamanan sistem, yang dikembangkan oleh komunitas. Daftar CWE dan taksonomi serta skema klasifikasi terkait berfungsi sebagai bahasa yang dapat digunakan untuk mengidentifikasi dan mendeskripsikan kelemahan-kelemahan ini dalam istilah "CWE.". CWE bebas digunakan atau bisa dikatakan open source. CWE akan digunakan untuk mengidentifikasi kerentanan dan standar untuk melakukan mitigasi[15].

III. METODE

Dalam penelitian ini penulis menggunakan website desa sudagaran sebagai objek penelitian serta dinas komunikasi dan informatika sebagai subjek penelitian

Penulis menggunakan pendekatan kualitatif. Alur penelitian ini dapat dilihat pada Gambar. 1



Gambar 1. Alur Penelitian

Penelitian ini akan dimulai dari menentukan masalah penelitian, sehingga penulis dapat menentukan tujuan dari penelitian ini. Dilanjutkan dengan menentukan batasan penelitian, sehingga penelitian ini akan terfokus. Dilanjutkan literatur review untuk menambah wawasan penulis terkait dengan penelitian yang akan dikerjakan penulis. Kemudian memasuki tahapan penetration testing yang kemudian menghasilkan sebuah keluaran yaitu hasil temuan kerentanan pada website. Penelitian ini bisa dikatakan berhasil ketika ditemukannya kerentanan yang dapat dikategorikan kedalam kategori OWASP top 10 2021.

Pendekatan *penetration testing* yang penulis gunakan adalah pendekatan *black box,* di mana *pentester* hanya memiliki sedikit informasi terhadap sistem web yang akan diuji. Penguji tidak melakukan *penetration* secara internal namun menyimulasikan serangan *cyber.*

Diagram alir *penetration testing* dapat dilihat pada **Gambar 2**. yang menggambarkan alur proses penulis dalam melakukan *penetration testing.* Tahapan yang digunakan penulis diambil dari tahapan yang diidentifikasi oleh IBM. Tahapan tahapan tersebut yaitu Tahapan ini digunakan karena lebih mudah dipahami oleh penulis. Penulis sedikit mengubah tahapan untuk menyesuaikan pendekatan yang digunakan penulis.

Gambar 2. Diagram Alir Penetration Testing

1. Reconnaissance

Reconnaissance adalah kegiatan mengumpulkan segala jenis informasi dari situs web yang akan di uji. Dalam hal ini contoh informasi yang akan dikumpulkan adalah seperti: *ip address, email address, subdomain, port* yang terbuka, sistem operasi, *web server* yang digunakan dan masih banyak lagi.

Dalam penelitian ini *Reconnaisance* terbagi menjadi dua kategori yaitu kategori manual dan automasi menggunakan program yang sudah ada. *Recconaisance* manual dilakukan dengan menganalisis kode dengan menggunakan fitur *inspect* dan fitur *view page source* dari browser. Kemudian dilakukan automasi menggunakan software yang sudah tersedia secara open source seperti nmap, whois, recon-ng dan theHarvester.

2. Target Discovery

Dari informasi yang didapatkan melalui tahap *reconnaisance* penguji mengidentifikasi potensi kerentanan berdasarkan pengetahuan penguji. Contohnya adalah ketika terdapat *input field,* maka penguji dapat melakukan *exploitasi* seperti injeksi script atau sql.

Selain indetifikasi manual, penguji akan menggunakan perangkat lunak otomatis untuk mengidentifikasi kerentanan. Dengan menggunakan perangkat lunak otomatis, maka akan lebih akurat dan mengurangi kesalahan manusia.

3. Exploitation

Exploitation merupakan kegiatan melakukan sebuah serangan ke titik-titik yang sudah teridentifikasi memiliki potensi kerentanan saat dilakukan pada tahap *Target Discovery.* Untuk melakukan *exploitation* bisa

dengan cara manual ataupun menggunakan program perangkat lunak untuk melakukan otomatisasi.

4. Documentation

Documentation yaitu melakukan analisis serta melaporkan hasil dari step-step sebelumnya. Sehingga penelitian ini bisa menjelaskan hasil dari analisis keamanan situs web pada *domain* sudagaran.desa.id

Dalam pelaporannya penulis akan menggunakan tabel yang akan menjelaskan jenis kerentanan yang ditemukan pada situs web yang diteliti. Serta memberikan rekomendasi tindakan yang perlu dilakukan pada setiap jenis kerentanan

IV. HASIL DAN PEMBAHASAN

A. Hasil

Dari hasil pengumpulan data secara manual ataupun otomatis data akan diklasifikasi menjadi beberapa jenis seperti jaringan, komponen software, input field, directory dan hasil scanning vulnerability.

Langkah awal dari penelitian ini adalah penggalian informasi seperti lingkup jaringan dan komponen atau software yang digunakan website.

Tabel 1. Identifikasi Jaringan

Parameter	Nilai	Tools
Ip Address	103.163.38.168	Nmap
Port terbuka	80:http 443:tcpwrapped 2000:cisco-sccp 5060:sip? 8010:https	Nmap
Sistem Operasi	Linux	Nmap
Vendor Antivirus	downloads2.kaspersky-labs.com = Kaspersky; update.nai.com = Trellix	Recon-ng module cache_snoop
Email Desa	pemdessudagaran@gmail.com	Website
Name server	ns3.banyumaskab.go.id	Whois
Nomor telepon desa	028 164 432 60	Website
Registrar Organization	Kementerian Komunikasi dan Informatika	Whois

Tabel 2. Identifikasi Software

Parameter	Nilai	Tools
Web Server	Nginx (1.18.0)	Wappalyzer
Reverse proxy	Nginx (1.18.0)	Wappalyzer
Web Framework	CodeIgniter	Wappalyzer
Javascript Libraries	lit-html(3.2.1), lit-element(4.1.1), Lightbox, Axios, TurfJS, Polyfill(3), OWL Carousel, MomentJs(2.18.1), Modernizr(2.8.3), jQuery(1.11.1), DataTables	Wappalyzer
UI Framework	Bootstrap(2)	Wappalyzer

Input field dapat menjadi celah utama masuknya ekploitasi injeksi, sehingga perlu untuk diidentifikasi input field apa saja yang ada pada website desa sudagaran.

Tabel 3. Identifikasi path Input Field

Methode	URL	Parameter
POST	/vote	Ide_pil, alasan, ide_poll
GET	/	cek_nik
GET	/	cek_bantuan
POST	/layanan_mandiri/login	nik, pin
POST	/layanan_mandiri/masuk/registrasi_pin	no._kk, nik, telepon
POST	/layanan_mandiri/masuk/cek_nik	nik
POST	/layanan_mandiri/masuk/cek_kk	no._kk
POST	/layanan_mandiri/masuk/reset	nik
POST	/siteman/auth	username, password
POST	/berita/tambah_komentar/1/{nomor_post}	owner, no_hp, email, komentar, captcha_code
GET	/themes/sederhana/dokhit.php?id=' + id	id

Beberapa directory website dapat menjadi celah keamanan ketika menampilkan halaman atau error tak terduga, yang mungkin tidak seharusnya ditampilkan atau tidak seharusnya di akses oleh publik.

Scanning directory menggunakan alat yaitu dirbuster dengan metode *bruteforce* dan menggunakan directory list yang tersedia di internet secara *open source*. Terdapat beberapa variabel dari hasil scanning dengan menggunakan dirbuster yaitu URL, kode status http, *contect length,* tipe conten, dan URL *redirect* jika kode http 300.

List directory berjumlah sekitar 1,2juta list. Oleh karena itu penulis hanya mengambil beberapa directory list saja berdasarkan kode status http nya. Kode yang akan diambil adalah 200 yang artinya server menerima, memahami dan menyutujui perminataan, kemudian kode 500 yang artinya internal server error. Kode 300 dan 400 akan lebih banyak diabaikan jika tidak ada redirect atau hal-hal yang menarik perhatian.

Kode Status 200

Path berwarna merah memiliki informasi dan celah yang cukup vital

Tabel 4. path dengan kode status 200 dan memiliki content lenght

path		
/LICENSE	/agenda	/albums
/line	/siteman	/composer.lock
/home	/page	/ssl
/database	/system	/download
/composer.json	/themes	/CHANGELOG

Tabel 5. path dengan kode status 200 dan tidak memiliki content lenght

path	
/.vscode/sftp.json	/vendor/autoload.php
/vendor/composer/autoload_psr4.php	/vendor/composer/autoload_real.php
/vendor/composer/autoload_files.php	/vendor/composer/autoload_namespace.php
/vendor/composer/ClassLoader.php	/vendor/composer/LICENSE
/vendor/composer/autoload_classmap.php	/vendor/autoload_statistic.php

Kode Status 500

Kode 500 menampilkan pesan error yang berasal dari server yang terkadang tidak dikelola dengan baik sehingga menampilkan informasi sensitif dari pesan error tersebut.

Tabel 6. path dengan kode status 500

path
/video
/vote
/s/

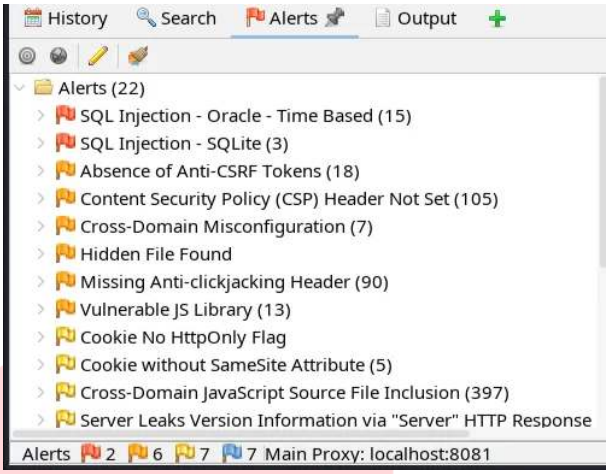
Kode Status 300

Terdapat banyak path dengan status 300, dengan beberapa variasi kode yaitu 301 yang melakukan *redirect* dengan menambahkan slash "/" diujung URL, dan 307 yang melakukan *redirect* dengan mengganti path yang dicari ke path yang lain.

Kode Status 400

Kode 400 juga bervariasi, ada kode 400 jika url memiliki karakter yang tidak diizinkan dan sepertinya respon berasal dari server. Kemudian kode 403 yang merupakan respon dari firewall. Dan kode 404 yang menandakan halaman tidak ditemukan.

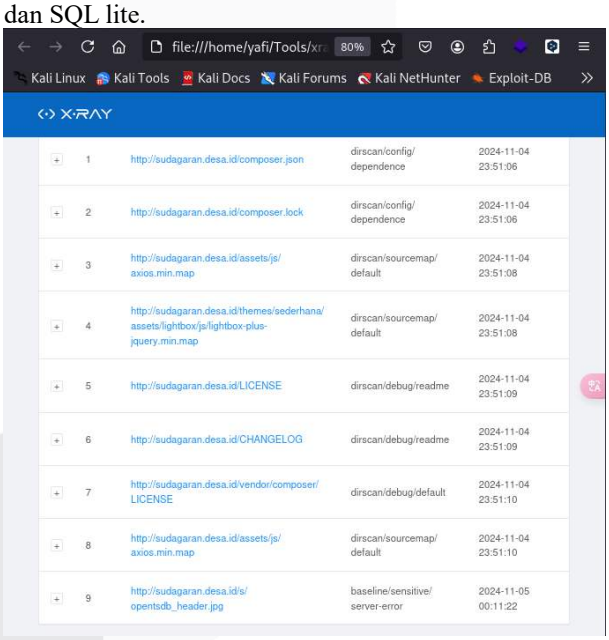
Selanjutnya Vulnerability Scanning, Pada tahap ini *website* akan dilakukan *scanning* dengan beberapa alat *scanning*.



Gambar 3. Hasil Scanning ZAP

Hasil scanning menggunakan ZAP pada kali linux menghasilkan beberapa kerentanan yang teridentifikasi menjadi beberapa kategori prioritas. Pertama adalah high alert dengan icon bendera merah, kemudian medium alert dengan icon bendera jingga, low alert dengan icon bendera kuning dan information alert dengan icon bendera biru.

Berdasarkan scanning tersebut, aplikasi ZAP menilai terdapat beberapa koategori potensi kerentanan dengan high alert yaitu SQL injection dengan beberapa jenis SQL inejction yang teridentifikasi yaitu time based dan SQL lite.



Gambar 4. Hasil Scanning XRAY

Tidak seperti ZAP yang mengkategorikan kerentanan berdasarkan prioritas. Xray memiliki kategori sendiri berdasarkan jenis potensi kerentanan yang ditemukan.

Namun scanning otomatis tidak selalu mengidentifikasi kerentanan dengan benar, karena terkadang aplikasi scanning melaporakan kerentanan yang sebenarnya tidak ada (false positive) atau kerentanan yang ada namun tidak terdeteksi (false

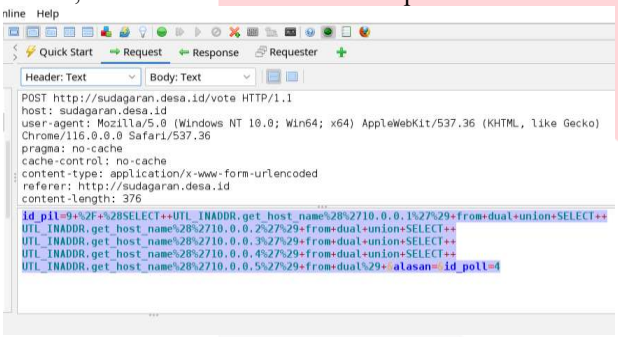
negative). Oleh karena itu masih perlu dilakukan testing manual.

Dari beberapa informasi yang didapatkan dalam tahap reconnaisance dan vulnerability scanning, ada beberapa potensi kerentanan dan pintu masuk penyerang untuk melakukan serangan cyber. Jika tidak terdapat penanganan yang baik dari pengembang maka pintu masuk tersebut dapat dimasuki oleh penyerang dengan berbagai metode eksploitasi.

Selanjutnya adalah Exploitasi, Percobaan eksploitasi akan dimulai dari hasil scanning zap dan xray

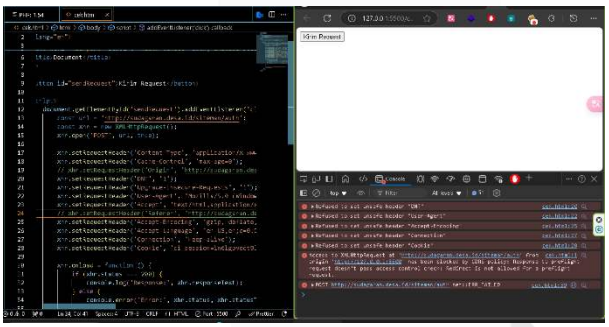
1. SQL Injection

Sql injection yang terdeteksi oleh zap menghasilkan false positive, karena ketika payload dicoba, website tidak memberikan response dan timeout.



Gambar 5. Payload Request yang digunakan ZAP

2. CSRF

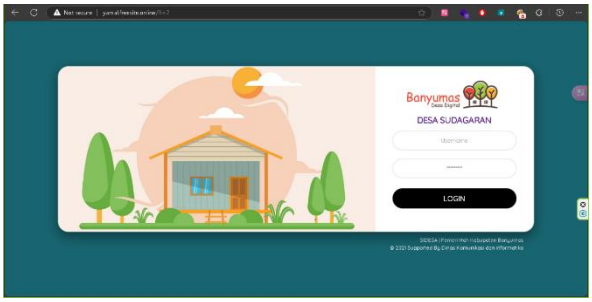


Gambar 6. Gagal melakukan request karena CORS

Pengujian CSRF dilakukan dengan membuat halaman website yang mengirimkan permintaan request menuju domain sudagaran.desa.id.

Dari hasil percobaan tersebut CORS mencegah pengiriman request dari domain yang berbeda. Error Access-Control-Allow-Origin menunjukkan bahwa aplikasi tidak mengizinkan domain eksternal untuk mengakses resource ini.

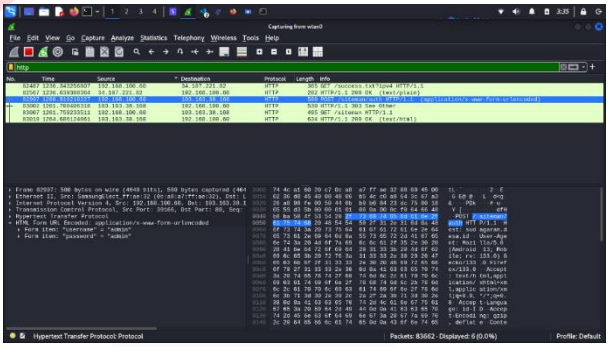
3. Missing anti clickjacking



Gambar 7. Halaman Login Siteman melalui iframe

Pengujian ini dilakukan dengan membuat halaman website yang melakukan *iframing* terhadap website sudagaran. Dan dari hasil percobaan menunjukkan bahwa halaman dapat di-*iframe*, yang menunjukkan tidak adanya perlindungan terhadap clickjacking.

4. Man In The Middle Attack

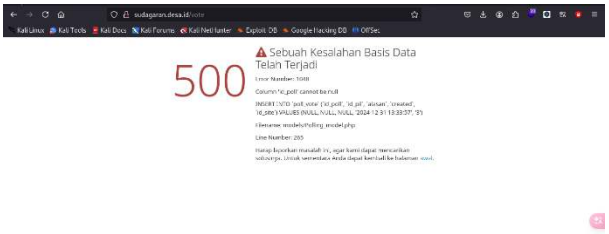


Gambar 8. Hasil Monitoring MITM dengan wireshark

Percobaan ini berusaha menjelaskan kerentanan yang disebabkan oleh penggunaan protokol http yang tidak dienkripsi, menggunakan perangkat Laptop(Linux) dan Android. Percobaan ini berasumsi ip milik pengguna android(192.168.100.60) adalah korban, dan perangkat laptop adalah penyerang yang melakukan arp poisoning pada jaringan lokal yang mana perangkat korban dan penyerang berada di jaringan yang sama dan penyerang dapat memonitoring aktivitas internet korban dengan wireshark. Terlihat terlihat jelas bahwa ketika korban mengakses dan mengirimkan data menuju endpoint /siteman, data yang dikirimkan terlihat jelas.

5. Error Leak

Beberapa endpoint dan form membocorkan error yang seharusnya tidak dibocorkan ke public. Artinya tidak ada penanganan khusus terkait error yang ditimbulkan oleh server. contoh pada form komentar.



Gambar 9. Error dari server yang tidak digeneralisasi

6. Halaman Administrator

Terdapat beberapa *endpoint* yang menuju halaman yang seharusnya atau idealnya disembunyikan atau di batasi aksesnya. Hal ini dikarenakan terekposnya halaman tersebut dapat memberikan jalan masuk pada penyerang untuk melakukan berbagai bentuk penyerangan. Endpoint tersebut antara lain :/ssl, /database, / line

7. Firewall Bypass

Dalam beberapa percobaan firewall berusaha membatasi beberapa payload yang dikirim oleh penulis. Sebagaimana contoh pada gambar, parameter nik terdapat payload yang dilarang oleh firewall.

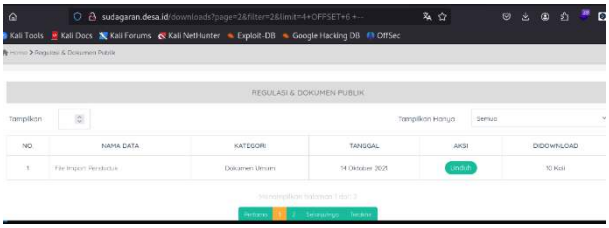
Gambar 10. Request dan Respose dari firewall

Namun dengan membungkus payload tersebut dengan tanda +, firewall dapat di lewati.

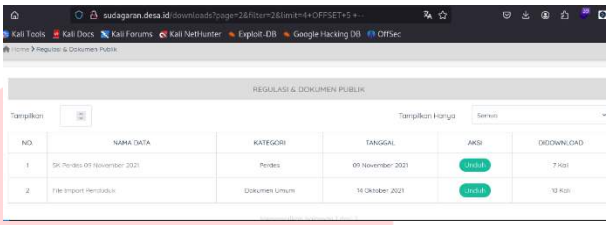
Gambar 11. Request dan Response yang melewati firewall

8. SQL Injection

Percobaan sql injection dilakukan pada halaman downloads yang menampilkan beberapa dokumen publik yang dapat di unduh oleh masyarakat. Percobaan dilakukan dengan menambahkan payload OFFSET pada parameter limit untuk membatasi dokumen yang ditampilkan.

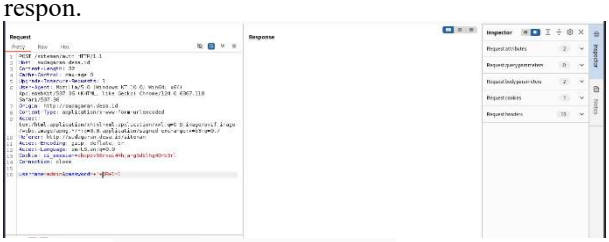


Gambar 12. Hasil Dengan payload OFFSET 6



Gambar 13. Hasil Dengan payload OFFSET 5

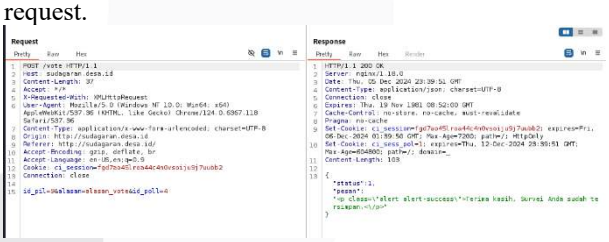
Hal ini menandakan Injeksi berhasil dan mengubah jumlah dokumen yang di tampilkan berdasarkan offset.Percobaan lainnya dilakukan melalui form login siteman, namun server tidak memberikan respon.



Gambar 14. Server tidak memberikan Response

9. Tidak ada batasan dengan fitur vote

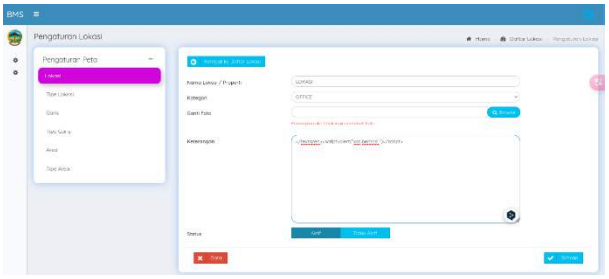
Tidak membatasi jumlah vote pada 1 user. hal ini dapat memicu orang tidak bertanggung jawab untuk memanipulasi nilai vote dengan mengirimkan sejumlah request.



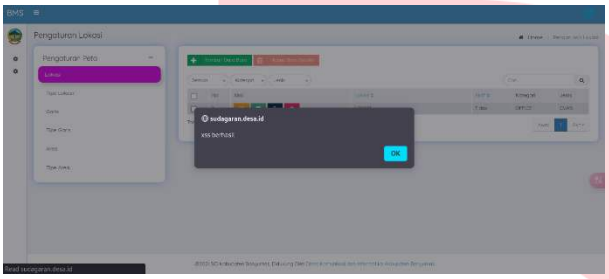
Gambar 15. Request dan Response Fitur Vote

10. XSS pada fitur Administrator

Percobaan xss dilakukan pada fitur admin yang kemungkinan sedang dalam masa pengembangan. Dengan menambahkan payload berikut “</textarea><script>alert(“xss berhasil”)</script>” di dalam kolom input keterangan xss berhasil disimpan dan dijalankan ketika seseorang membuka deskripsi nya.



Gambar 16. payload XSS pada fitur admin



Gambar 17. hasil STORED XSS

Dari hasil percobaan eksploitasi terhadap temuan pada tahap *reconnaisance*, dapat penulis kategorikan beberapa jenis serangan kedalam kategori owasp top 10 untuk kemudahan dalam menganalisis.

B. Pembahasan

1. A01 Broken Akses Control

Melihat eksploitasi terhadap CSRF, SQL inejction, temuan Halaman Administrator, maka dapat disimpulkan website memiliki kerentanan Broken Access Control dengan beberapa detail mapping cwe berikut.

CWE-352: Cross-Site Request Forgery (CSRF). Walaupun CORS mencegah request dari domain berbeda, pengembang perlu memastikan bahwa token CSRF digunakan untuk melindungi dari CSRF secara efektif.

CWE-306: Missing Authentication for Critical Function. Fitur admin tidak seharusnya dapat diakses secara publik apalagi berkaitan dengan fitur untuk menghapus database.

Percobaan bypass halaman login tidak dikatakan berhasil melalui sql injection dikarenakan server tidak merespon payload sql yang diinputkan dan berujung timeout.

2. A02 Crypto Fail

CWE-319: Cleartext Transmission of Sensitive Information. Website tidak menggunakan TLS pada semua page, terkhusus pada halaman login. Percobaan menggunakan teknik MITM berhasil dilakukan. Menandakan website rentan terhadap serangan MITM.

3. A03 Injection

Pada percobaan sebelumnya zap mengidentifikasi beberapa kerentanan dengan prioritas high seperti sql injection tetapi terdapat false positive dalam pengidentifikasiannya. Namun dalam beberapa page yang tidak teridentifikasi oleh zap dapat dipastikan website memilik kerentanan sql injection.

CWE-89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection').dalam percobaan sql injection sebelumnya firewall berhasil dilewati, dan payload sql berhasil mempengaruhi output yang ditampilkan ke website.

CWE-79 Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting'). Dalam percobaan sebelumnya, ditemukan bahwa aplikasi rentan terhadap serangan xss. Aplikasi tidak menetralkan atau salah menetralkan input yang dapat dikontrol pengguna sebelum ditempatkan dalam output yang digunakan sebagai web halaman yang disajikan kepada pengguna lain.

4. A04 Insecure Design

Kerentanan yang termasuk dalam A04 seringkali merupakan masalah pada tingkat desain arsitektur aplikasi. Ini berarti kerentanan tersebut tidak selalu terlihat jelas pada permukaan aplikasi dan membutuhkan pemahaman mendalam tentang bagaimana aplikasi dibangun dan berinteraksi dengan sistem lain. Oleh karena itu kerentanan ini sulit diidentifikasi melalui pendekatan black box.

5. A05 Security Misconfiguration

CWE-1021 Improper Restriction of Rendered UI Layers or Frames. Percobaan iframing menunjukkan bahwa website tidak membatasi atau salah membatasi iframe atau lapisan UI milik aplikasi atau domain lain, yang dapat menyebabkan kebingungan pengguna tentang antarmuka mana yang berinteraksi dengan pengguna.

CWE-209 Generation of Error Message Containing Sensitive Information. Konfigurasi server aplikasi memungkinkan pesan kesalahan yang mendetail, misalnya, pengungkapan query sql yang digunakan, dikembalikan kepada pengguna. Hal ini berpotensi mengekspos informasi sensitif atau kelemahan yang mendasarinya.

Kegagalan firewall dalam mengidentifikasi karakter berbahaya hal ini berarti firewall tidak memvalidasi atau memfilter input dengan benar, yang memungkinkan karakter-karakter berbahaya lolos dan dieksekusi di dalam aplikasi atau sistem. Kemungkinan terdapat konfigurasi firewall yang belum disesuaikan.

6. A06 Vulnerable and Outdated Component

Library jquery-validation berdasarkan CVE-2021-21252. library jquery validation, sebelum versi 1.19.3 berisi satu atau lebih ekspresi reguler yang rentan terhadap ReDoS (Regular Expression Denial of Service)

Library chart berdasarkan CVE-2020-7746. package chart.js sebelum versi 2.9.4, rentan terhadap Polusi prototipe dimana terjadi ketika properti atau nilai tidak aman dimasukkan ke dalam prototipe objek JavaScript.

Library moment berdasarkan CVE-2017-18214. library moment sebelum 2.19.3 memiliki kerentanan regular expression denial of service melalui string tanggal yang dibuat. Objek prototipe adalah dasar dari semua objek dalam JavaScript, dan dengan memanipulasinya, penyerang dapat mengubah perilaku objek yang seharusnya tidak dapat diakses atau diubah.

Library jquery-datatable berdasarkan CVE-2020-28458. Semua versi datatables rentan terhadap polusi objek prototipe javascript karena perbaikan yang tidak lengkap.

Library select2 berdasarkan CVE-2016-10744. versi dibawah 4.0.5 rentan terhadap XSS.

Library bootstrap berdasarkan CVE-2018-20676. XSS dimungkinkan dalam atribut tooltip data-viewport.

Library axios berdasarkan CVE-2020-28168. rentan terhadap SSRF di mana penyerang dapat melewati proksi dengan memberikan URL yang merespons dengan pengalihan ke host atau alamat IP yang dibatasi.

7. A07 Identification and Auth Failure

CWE-308: Use of Single-factor Authentication. Jika reset akun hanya perlu satu data pribadi (NIK) untuk reset akun tanpa langkah otentikasi yang lebih kuat, bisa menyebabkan kegagalan dalam memverifikasi identitas pengguna dengan benar. Ini membuka peluang bagi penyerang untuk mengeksploitasi kekurangan tersebut dan mengakses akun orang lain.

CWE-307: Improper Restriction of Excessive Authentication Attempts. yang merujuk pada kegagalan aplikasi untuk membatasi atau mencegah upaya login berlebihan. Aplikasi yang gagal mendeteksi percobaan login yang berlebihan atau tidak memiliki perlindungan seperti penundaan antara upaya login sangat rentan terhadap serangan brute-force.

8. A08 Software and Data Integrity Failures

Karena pendekatan black-box hanya melihat perilaku eksternal sistem, sulit untuk mendeteksi integritas komponen internal atau manipulasi pipeline CI/CD tanpa melihat konfigurasi, kode sumber, atau pengaturan sistem.

CWE-862: Missing Authorization. Namun dalam formulir vote, dengan beberapa kali percobaan yang dilakukan pada form tersebut, tidak terdapat batasan. Yang berarti siapa saja bisa mengirim beberapa kali vote. Hal ini mengakibatkan data yang ditampilkan dari hasil vote tidak bisa dipercaya integritasnya.

9. A09 Security Logging and Monitoring Failures

Kerentanan ini berfokus pada kurangnya mekanisme logging atau monitoring yang memadai, sehingga ancaman tidak dapat dideteksi atau direspon secara efektif dengan pendekatan blackbox.

CWE-116: Improper Encoding or Escaping of Output. Namun dalam percobaan untuk melewati firewall, terdapat kemungkinan firewall tidak melakukan monitoring yang baik terhadap request dari client.

10. A10 Server Side Request Forgery

SSRF ini sering terjadi ketika aplikasi menerima input dari pengguna yang kemudian diproses untuk membuat permintaan ke server lain, tetapi input tersebut tidak diproses dengan benar. Dalam hal ini penulis tidak menemukan input untuk menguji hal tersebut.

Dari hasil analisis dan pengujian sebelumnya, penulis merangkum hasil nya kedalam table berikut :

Tabel 7. Rangkuman Kerentanan yang ditemukan

Kategori	Kerentanan	Detail
A01 Broken Access Control	Halaman admin ter-ekspos, CSRF	Endpoint sensitif terekspos seperti /database dan /line; token CSRF tidak diterapkan pada halaman login.
A02 Crypto Failures	Cleartext Transmission	Data sensitif dikirim tanpa enkripsi (HTTP). Rentan terhadap MITM
A03 Injection	SQL Injection, dan XSS	Payload SQL Injection berhasil memanipulasi query pada endpoint /downloads, XSS di dalam fitur admin.
A05 Security Misconfiguration	Error Leak	error memaparkan informasi sensitif pada endpoint /s, dan /vote
A06 Outdated Components	Outdate Library	Jquery, Moment.js, Bootstrap rentan terhadap XSS dan lainnya
A07 Auth Failures	Brute Force Vulnerability,	Tidak ada pembatasan upaya login pada halaman login dan registrasi

Kategori	Kerentanan	Detail
A08 Software and data Integrity Filures	Voting Manipulation	Tidak ada batasan jumlah voting pada fitur vote
A09 Security Logging and Monitoring Failures	Bypass Firewall	Payload yang seharusnya diblokir oleh firewall, dapat di lewati dengan menambahkan tanda +.

dapat disimpulkan bahwa dari 10 kategori OWASP Top 10 2021, website desa sudagaran yaitu sudagaran.desa.id teridentifikasi memiliki 8 kerentanan dalam kategori owasp top 10 2021 yaitu selain kategori A4 dan A10.

Selain itu penulis merangkum langkah-langkah mitigasi yang bisa dilakukan oleh pengembang berdasarkan CWE

Tabel 8

Kerentanan	Langkah Mitigasi	Standar Mitigasi
Halaman admin ter-ekspos, CSRF	- Terapkan mekanisme otentikasi kuat pada endpoint sensitif, seperti (/database, /line). - Audit dan batasi akses berdasarkan prinsip least privilege atau berikan akses minimum secara default. - pastikan aplikasi bebas dari XSS, karena kebanyakan perlindungan csrf dapat dilewati dengan XSS. Yaitu dengan melakukan sanitasi dan validasi input. - gunakan csrf token dan terapkan CORS terutama pada halaman login.	CWE -352, 306
Cleartext Transmission	- Konfigurasikan server untuk menggunakan saluran terenkripsi untuk komunikasi, yang mungkin termasuk SSL atau protokol aman lainnya.	CWE -319
SQL Injection, dan XSS	- Gunakan parameterized queries atau prepared statements pada kode backend yang terhubung ke sql. - Lakukan validasi dan sanitasi input. Terapkan output encoding pada semua data dinamis.	CWE -89, CWE -79
Restrict Access, Error Leakage	- Gunakan X-Frame-Options - Nonaktifkan pesan error yang memaparkan informasi sistem. - Jika ada opsi nya, matikan konfigurasi untuk menampilkan error saat runtime.	CWE -209, CWE -1021
Outdate Library	- Monitoring dan Perbarui library ke versi terbaru secara rutin. - Gunakan alat otomatis seperti Dependabot atau Snyk untuk deteksi kerentanan.	CWE -1395
Brute Force Vulnerability	- Batasi upaya login berulang. - Terapkan CAPTCHA untuk mencegah serangan otomatis. - Terapkan autentikasi multifaktor (MFA).	CWE -353, CWE -307, CWE -308
Voting Manipulation	- Kategorikan fitur menjadi anomymous, normal, privilaged, dan administrative area saat pengembangan. Terapkan mekanisme login terhadap vitur vote.	CWE -862
Bypass Firewall	- Configurasi firewall dengan dengan baik agar menerapkan sanitasi dan validasi yang baik.	CWE -116

Langkah Mitigasi yang diterapkan mengikuti standar CWE serta ngacu pada langkah-langkah mitigasi dari OWASP Top 10 2021. Dengan total 17 langkah mitigasi yang dapat diterapkan untuk mengurangi kerentanan yang ada di website desa sudagaran.

V. KESIMPULAN

Berdasarkan hasil pengujian, dapat disimpulkan website rentan terhadap beberapa serangan seperti MITM, XSS, SQL Injection, Akses Tidak Sah, dan Brute Force. Serta beberapa hal yang perlu diperbaiki seperti fitur vote yang tidak dibatasi, belum diterapkan CSRF token, software yang tidak update dan kode yang tidak efektif. Maka penulis dapat menyimpulkan bahwa aplikasi web pemerintah desa sudagaran memiliki kerentanan yang belum memenuhi standar owasp top 10 2021.

Terdapat beberapa langkah mitigasi yang dapat diidentifikasi dengan standar owasp top 10 2021dan CWE dengan total 17 langkah mitigasi yang dapat diterapkan oleh pengembang website desa sudagaran.

REFERENSI

[1] E. A. Altulaihan, A. Alismail, and M. Frikha, "A Survey on Web Application Penetration Testing," Mar. 01, 2023, *MDPI*. doi: 10.3390/electronics12051229.

[2] S. Yuniarti, "PERLINDUNGAN HUKUM DATA PRIBADI DI INDONESIA," *Business Economic, Communication, and Social Sciences)*, vol. 1, no. 1, pp. 147–154, 2019.

[3] A. Zainal Abidin, H. Saputra, and M. Taufiq Sumadi, "Penetration testing on mail server website using the OWASP method," 2023. [Online]. Available: www.ejournal.isha.or.id/index.php/Mandiri

[4] Z. Fang, "E-Government in Digital Era: Concept, Practice, and Development," 2002.

[5] U. Ravindran and R. V. Potukuchi, "A Review on Web Application Vulnerability Assessment and Penetration Testing," *Review of Computer Engineering Studies*, vol. 9, no. 1, pp. 1–22, Mar. 2022, doi: 10.18280/rces.090101.

[6] "BPJS Kesehatan: Data ratusan juta peserta diduga bocor - 'Otomatis yang dirugikan masyarakat', kata pakar," BBC News. Accessed: Apr. 26, 2024. [Online]. Available: https://www.bbc.com/indonesia/indonesia-57196905

[7] "Ratusan juta data Dukcapil Kemendagri diduga bocor, pakar siber: 'Ini peretasan paling parah,'" BBC News. Accessed: Apr. 26, 2024. [Online]. Available: https://www.bbc.com/indonesia/articles/c51v25916zlo

[8] "Sederet Kasus Kebocoran Data Penduduk di Server Pemerintah," Kompas.com. Accessed: Apr. 26, 2024. [Online]. Available: https://www.kompas.com/tren/read/2022/01/08/163000065/sederet-kasus-kebocoran-data-penduduk-di-server-pemerintah

[9] H. SIBURIAN, D. PAKEL, and A. YUSUF, "LANSKAP KEAMANAN SIBER INDONESIA," 2023. Accessed: Nov. 23, 2024. [Online]. Available: https://www.bssn.go.id/wp-content/uploads/2024/03/Lanskap-Keamanan-Siber-Indonesia-2023.pdf

[10] Cisco, "What Is Penetration Testing," Cisco.

[11] A. Ziro, S. Gnatyuk, and S. Toibayeva, "Improved Method for Penetration Testing of Web Applications."

[12] IBM, "What is penetration testing," INternational Businnes Machine.

[13] OWASP Foundation Inc, "About the OWASP Foundation," OWASP Foundation Inc.

[14] OWASP Foundation Inc, "OWASP Top 10," OWASP Foundation Inc.

[15] CWE, "About CWE," Common Weakness Enumeration.