IMPLEMENTASI OWASP TOP 10 DALAM PENGUJIAN PENETRASI WEBSITE : MENGIDENTIFIKASI CELAH KEAMANAN DALAM SISTEM PENGELOLAAN VOTING INDONESIA

(*Implementation Of OWASP Top 10 In Website Penetration Testing: Identifying Security Gaps in Indonesia's Voting Management System*)

Zora Zairina*[1],Raphael Bianco Huwae[1], Andy Hidayat Jatmika[1]
[1]Dept Informatics Engineering, Mataram University
Jl. Majapahit 62, Mataram, Lombok NTB, INDONESIA
Email: ra.zairina8@gmail.com, [Raphael.bianco.huwae, andy]@unram.ac.id

Abstract

The rapid advancement of information technology has had a major influence in every aspect of life, including in government operations. the availability of platforms such as websites can be a medium for disseminating information transparently, especially in Indonesia's democratic process, namely elections. However, this digital convenience also presents cybersecurity challenges. therefore this research focuses on identifying security gaps in the voting management system in Indonesia by conducting penetration testing based on the OWASP Top 10 2021. in conducting penetration testing, it focuses on 10 subdomains on the targeted voting management website. This research process starts from the process of scanning and testing security using various tools such as subfinder, dirsearch, nuclei, ex-param and JSRecon. thus getting the analysis results that of the 10 targets 9 of them have vulnerabilities related to the OWASP Top 10 categories, such as A01-Broken Access Control, A03-Injection, A05-Security Misconfiguration, and A06-Vulnerable and Outdated Components. Broken Access Control is the most commonly found gap, identified in 6 subdomains. this research aims to improve the security of voting management systems in Indonesia while contributing to the sustainability of a more secure digital democracy in Indonesia.

Keywords*: cybersecurity, penetration testing, OWASP Top 10 2021, voting management system, security vulnerabilities*

**Correspondence Author*

1. PENDAHULUAN

Era globalisasi ditandai dengan adanya perkembangan teknologi informasi yang terus berkembang secara pesat. Hal ini memberikan dampak pada kehidupan manusia sehari-hari dan juga dalam keberlangsungan pemerintah dalam menjalankan kepemerintahannya untuk menjaga Negara Kesatuan Republik Indonesia (NKRI)[1]. Salah satu kemudahan yang diberikan oleh teknologi yaitu dengan adanya *website*. *Website* merupakan salah satu wadah berbentuk digital yang memuat informasi dan dapat diakses secara mudah serta bisa dijangkau secara publik melalui internet[2]. *Website* juga kerap kali dijadikan sebagai platform *online* untuk memberikan ketransparansi informasi oleh pemerintah kepada masyarakat.

Indonesia merupakan negara demokrasi yang ditandai dengan kedaulatan berada di tangan rakyat. Salah satu bentuk dari negara demokrasi ini ialah dengan diselenggarakannya pemilu di Indonesia, pemilu di Indonesia diselenggarakan oleh lembaga independen yaitu Komisi Pemilihan Umum (KPU)[3]. Seiring berkembangnya zaman, ilmu pengetahuan mengenai teknologi informasi semakin meningkat dan industrialisasi era 4.0 menciptakan suatu pusat perubahan baru yaitu digitalisasi, ditambah lagi dengan adanya virus covid 19 yang pernah menimpa Indonesia menghasilkan kegiatan dan keputusan diambil via *online*. Hal tersebut mendorong Komisi Pemilihan Umum (KPU) dalam terus memperbarui fitur-fitur dan kegunaan dari *website* yang telah dibentuk oleh KPU itu sendiri. Hal ini diharapkan dapat mempermudah penyebaran informasi dan untuk meningkatkan integritas pemilihan umum di Indonesia ini. *Website* ini dapat diakses secara online dan bisa dijangkau oleh segala macam kalangan masyarakat sehingga dapat menjaga transparansi dalam proses pemilu[4].

Dengan adanya *website* ini rakyat Indonesia dapat ikut andil dalam menjaga dan melaksakan Pemilu di

Indonesia. Kemudahan layanan yang disediakan oleh *website* tersebut diiringi oleh tantangan atau masalah yang akan didapatkan yaitu dengan munculnya ancaman keamanan *cyber*[5]. Keamanan pada *website* sangatlah krusial yang dimana sistem pengelolaan voting ini digunakan dalam mendukung pelaksanaan pemilu dalam memastikan integritas, kerahasiaan dan ketersediaan data. Kurangnya perhatian terhadap masalah tersebut akan menimbulkan risiko yang signifikan bagi instansi kerentanan keamanan dapat dimanfaatkan *attacker* untuk mengambil keuntungan, seperti peretasan, manipulasi data hingga pencurian informasi sensitif. Hal tersebut dapat merugikan instansi baik secara teknis maupun kepercayaan masyarakat terhadap *website*[6].

BSSN menyatakan bahwa terjadi lebih dari 423 juta serangan siber pada periode Januari-November 2020. Pada pertengahan tahun 2020 Indonesia dihebohkan oleh berita terjadinya kebocoran data kependudukan yaitu data pribadi dan sensitif sebanyak 2,3 juta data yang bersumber dari Komisi Pemilihan Umum yang yang tersebar secara *online*. Diduga tidak hanya terjadi kebocoran saja tetapi data-data tersebut dijual oleh *hacker* di forum *dark website*. Kebocoran data tersebut diduga karena lemahnya keamanan infrastruktur dan kurangnya pengawasan dalam penelolaan data. Kebocoran ini dapat dikategorikan sebagai data *breach*, hal ini memberikan dampak sangat luas mulai dari penyalahgunaan data pribadi hingga dapat menurunkan kredibilitas instansi[7]. Terjadinya banyak kasus kebocoran data membuat pemerintah semakin cemas. Dengan itu dibentuklah keamanan pada sistem informasi komputer untuk memastikan bahwa sistem informasi tersebut tidak memiliki celah untuk diakses data secara ilegal atau kepada pihak yang tidak berwenang. Keamanan sistem informasi adalah bagaimana cara agar dapat mencegah penipuan atau *cheating*, kebocoran informasi dan kegagalan fungsi sistem[8].

Penelitian ini bertujuan untuk mengidentifikasi dan menganalisis celah keamanan pada *website* pengelolaan voting di Indonesia dalam meningkatkan keamanan *website*. Dalam melakukan pengujian penetrasi pada *website* ini diperlukan *framework* sebagai kerangka analisis maka dari itu pada penelitian ini memanfaatkan pendekatan OWASP Top 10 2021. Fokus penelitian ini pada pengujian penetrasi yang dilakukan terhadap 10 subdomain *website* terkait sistem pengelolaan voting, untuk menemukan ancaman keamanan yang dapat mengancam integritas, kerahasiaan dan ketersediaan data dalam proses pemilu.

Keunikan dalam penelitian ini terletak pada fokus spesifikasinya pada sistem atau *website* pengelolaan voting di Indonesia, dalam konteks demokrasi banyak yang masih belum di eksplore pada penelitian-penelitian sebelumnya terkhusus dalam ranah keamanan *website*-nya. Selain itu penelitian ini berfokus menggunakan kerangka analisis OWASP Top 10 versi terbaru yaitu tahun 2021 yang memberikan hasil yang lebih relevan dan akurat terhadap ancaman. Dengan didukung oleh alat penetrasi seperti *dirsearch*, nuclei, ex-param dan JSRecon yang mengidentifikasi kerentanan secara komprehensif. Penelitian ini juga memberikan mitigasi dan rekomendasi yang dapat diterapkan oleh pengelola *website*.

Dalam melakukan penelitian ini diperlukan berbagai tahapan dimulai dari studi literatur, kemudian dilanjutkan dengan melakukan pengujian penetrasi. *Penetration testing* adalah suatu proses atau kegiatan untuk menilai keamanan sistem informasi atau *website* dengan mensimulasikan serangan dari berbagai sumber yang tidak diketahui dan berbahaya[9]. Dalam pengujian penetrasi menggunakan alat seperti *dirsearch*, nuclei, ex-param dan JSRecon yang memudahkan proses pengujian. Selain itu dalam melakukan *penetration testing* ini juga diperlukan pedoman analisis kerentanan umum yang tepat pada sebuah sistem informasi. Pada penelitian ini melakukan penetrasi *testing* dengan menggunakan pedoman yang mengacu pada kerentangan umum yang terdapat di dalam OWASP Top 10 . Panduan umum untuk meningkatkan keamanan aplikasi atau *website* ini memiliki 10 analisis risiko yang dapat terjadi pada aplikasi *mobile* maupun *website*. Hasil dari metode ini dapat dijadikan rekomendasi sebagai dasar merumuskan mitigasi yang bertujuan untuk meningkatkan keamanan pada *website* pengelolaan voting di Indonesia ini[10].

2. TINJAUAN PUSTAKA

Penelitian [11] melakukan pengujian pada *website* SIM xxx dengan menggunakan metode OWASP Top 10, dengan menggunakan berbagai macam *tools* di kali linux seperti *tools hydra* dan *dirb* serta melakukan percobaan *configuration port* SSL dan *clickjacking* testing. Penelitian ini menemukan 4 celah keamanan yang perlu dilakukan perbaikan, celah keamanan tersebut adalah *broken authentication*, *sensitive data exposure*, dan *security misconfiguration*. Adapun celah lain yang ditemukan namun tidak termasuk dalam Top 10 keamanan OWASP yaitu *clickjacking*.

Penelitian [12] menguji penetrasi pada *website* XYZ kabupaten XYZ dengan menggunakan pedoman

OWASP Top 10, serta menggunakan metode *black box* untuk mendapatkan hasil pengukuran Tingkat kerentanan pada aplikasi dan menggunakan 2 *tools* yaiti vega dan OWASP ZAP. Nilai dari *vulnerability scanning* yang didapatkan yaitu ada 9 jenis kerentanan 2 *high*, 1 *medium*, dan 6 *low*.

Penelitian [13] melakukan audit dan analisis dengan *website* pemerintah sebagai target menggunakan penetrasi SQL *injection* dan *Cross Site Scripting* (CSS). Penelitian ini juga mengacu pada kerangka analisis OWASP Top 10 2021, dengan hasil penetrasi mendapat 7 celah keamanan pada 10 subdomain target yaitu 4 *injection*, 2 *cryptographic*, dan 1 *security misconfiguration*.

Berdasarkan penjelasan di atas, penulis melakukan penelitian *penetration testing* pada *website* sistem pengelolaan voting di Indonesia menggunakan alat *dirsearch*, nuclei, ex-param dan JSRecon serta melakukan analisis berdasarkan OWASP Top 10 sebagai kerangka analisis. Penelitian ini juga memberikan informasi mengenai nilai *severity* keamanannya dan kategori berdasarkan CWE. Selain itu penelitian ini tidak hanya memberikan analisisnya keamanannya saja tetapi juga memberikan rekomendasi terkait celah keamanan yang didapatkan agar dapat dilakukan untuk meningkatkan keamanan *website* sistem pengelolaan voting di Indonesia.

2.1 Sistem Pengelolaan Voting Indonesia

Sistem pengelolaan voting Indonesia dikelola oleh Komisi Pemilihan Umum (KPU). KPU sebagai pelaksana kegiatan demokrasi ini harus dapat memanfaatkan perkembangan teknologi informasi untuk menjalankan pemilu yang demokratis, efektif dan efisien. Pada saat ini sistem informasi diimplementasikan dalam bentuk *website*. *Website* ini dapat digunakan untuk memudahkan pemilu serentak yang dapat mengurangi permasalahan yang ada. Hal ini juga dapat berfungsi sebagai upaya untuk menjaga integritas *website* yang berfungsi untuk memenuhi kebutuhan pemilu di Indonesia. *Website* ini juga mampu untuk membantu mengakomodir kebutuhan pemilu di Indonesia dan dapat mencegah petugas menyalahgunakan wewenang mereka pada saat perhitungan suara secara manual serta menjaga akuntabilitas pemilu dan transparansi data rekapitulasi suara pemilu itu sendiri[14].

2.2 *Penetration Testing*

Pada aspek keamanan siber, keamanan sistem informasi atau perangkat lunak sangat penting. Untuk melindungi *website* dari pengakses data yang ilegal dan serangan pihak yang tidak bertanggung jawab, harus dilakukannya pengujian atau *self-test* pada sistem itu sendiri, salah satu upayanya yaitu dapat menggunakan metode *penetration testing*. *Penetration testing* adalah serangkaian langkah-langkah dan teknik evaluasi yang digunakan untuk menguji keamanan situs *website*. Proses deteksi keamanan ini dilakukan dengan melakukan simulasi penyerangan untuk mengetahui lokasi dan celah kerawanan pada situs *website* kemudian hasil dari deteksi tersebut ditutup dan diperbaiki. *Penetration testing* ini dilakukan untuk mencegah sistem terkontaminasi. Pada dasarnya tujuan dari melakukan *penetration testing* ini untuk menguji dan melindungi keamanan dari sistem informasi atau *website*, dengan melakukan pengujian ini dapat membantu menemukan kerentanan pada suatu sistem dan memeriksa dapatkah penyerang melakukan eksploitasi sehingga mendapatkan akses yang tidak sah. *Penetration testing* dapat menggunakan OWASP Top 10 menjadi parameter pengujiannya. [15].

2.3 OWASP Top 10

OWASP atau "*Open Web Application Security Project*" adalah sebuah organisasi nirlaba yang berfokus pada peningkatan keamanan aplikasi *website* dan perangkat lunak. OWASP ini tidak terafilisi dengan perusahaan manapun, dan merupakan organisasi non-profit, semua yang terasosiasi dengan OWASP merupakan sukarelawan, maka dari itu OWASP ini bersifat terbuka yang artinya bebas diakses oleh pengembang. Tujuan utama OWASP untuk meningkatkan pemahaman mengenai risiko keamanan bagi setiap pengembang *website* dan perangkat lunak. Demi mendukung hal tersebut OWASP telah menyediakan sumber daya, alat yang dapat digunakan dan panduan yang konprehensif untuk membantu para developer, arsitek dan tim keamanan dalam mengatasi kerentanan dalam *website* dan perangkar lunak. Implementasi menggunakan standar OWASP ini dapat membuat sistem lebih siap menghadapi ancaman keamanan[16].

OWASP telah menghadirkan 10 daftar teratas kerentanan yang dapat mengancam keamanan suatu *website* yang disebut OWASP Top 10. Hal ini bertujuan untuk memberikan pengetahuan mengenai celah keamanan dan kerentanan yang paling biasa ditemukan. OWASP Top 10 ini sendiri diciptakan untuk meningkatkan pengetahuan masyarakat mengenai keamanan suatu *website* atau perangkat lunak untuk menentukan jenis kerentanan keamanannya. OWASP Top 10 ini setiap beberapa tahun selalu melakukan pembaruan daftar yang tertera berdasarkan kombinasi data pengujian keamanan dan survei profesional dalam industri. OWASP Top 10 baru saja merilis daftar-daftar

kerentanannya versi terbaru yaitu tahun 2021. Daftar-daftar kerentanan dan ancaman tersebut dapat dilihat pada tabel di bawah ini[17].

Tabel 1. Kerentanan OWASP Top 10

OWASP Top 10-2021	
A01	*Broken Access Control*
A02	*Cryptographic Failures*
A03	*Injection*
A04	*Insecure Design*
A05	*Security Misconfiguration*
A06	*Vulnerable and Outdated Components*
A07	*Identification and Authentication Failures*
A08	*Software and Data Integrity Failures*
A09	*Security Logging and Monitoring Failures**
A10	*Server-Side Request Forgery (SSRF)**

2.3.1 *Broken Access Control*

Jenis kerentanan umum yang pertama yaitu *broken access control* ini lebih sering ditemukan pada kategori aplikasi. Pada kerentanan ini terjadi apabila *user* yang tidak sah atau tidak berkewenangan tetapi dapat mengakses bagian tertentu dari suatu aplikasi atau *website* yang sepatutnya tidak dapat diakses oleh sembarang *user*. Hal ini dapat menjadi penyebab terjadinya kasus pencurian data atau penggunaan *website* secara ilegal. Contoh kasusnya pada saat *user* dapat mengakses halaman administratif dengan memanipulasikan parameter URL.

2.3.2 *Cryptographic Failures*

Dalam kerentanan dari sisi *cryptographic failures* ini berfokus pada API yang berfungsi untuk menghubungkan layanan dari sumber eksternal. Kerentanan ini terjadi pada saat kesalahan penerapan teknik kriptografi, seperti algoritma atau pengelolaan pada data sensitif yang lemah dan pengaturan sandi yang tidak kuat. Hal tersebut membuat data sensitif akan rentan terhadap pencurian atau dimanipulasi oleh penyerang. Contohnya pada saat kata sandi yang disimpan tanpa hashing, dimana kata sandi pengguna disimpan hanya dalam bentuk teks biasa sehingga dapat mudah dibaca dan dicuri ketika terjadi pelanggaran atau serangan terhadap basis data.

2.3.3 *Injection*

Jenis kerentanan *injection* terjadi karena penyerang atau peretas melakukan serangan dengan menyisipkan kode berbahaya yang mereka buat seperti SQL *injection*, XSS, LDAP dan CRLF ke dalam input program yang di eksekusi aplikasi. Hal ini menyebabkan penyerang dapat mengambil alih kontrol *website* atau aplikasi dan mengakses data sensitif. Kerentanan ini dapat diatasi dengan menggunakan *prepared statements* atau dengan menerapkan validasi input yang ketat.

2.3.4 *Insecure Design*

Pada daftar kerentanan ini menunjukkan bahwa pada OWASP juga terdapat daftar kerentanan mengenai desain yang tidak aman. Kerentanan pada bagian desain ini muncul akibat desain yang tidak aman dan rentan terkena serangan. Dikarenakan kurangnya ketelitian pada saat tahapan desain aplikasi atau *website* yang dikakukan pengembang. Salah satu contoh kasus mengenai kerentanan yang disebabkan oleh desain ini iyalah ketika *user* kehilangan akses ke akun mereka atau data pribadi yang terekspose. Hal ini dapat diatasi salah satunya dengan memperluas penggunaan desain, pola dan pemodelan yang aman atau dengan menerapkan pengembangan berbasis risiko sejak awal.

2.3.5 *Security Misconfiguration*

Kerentanan mengenai konfigurasi kemanan ini sangat penting dikarenakan menunjukkan perubahan perangkat lunak yang dapat dikonfigurasi. Dengan kesalahan konfigurasi ini dapat membuat suatu sistem atau *website* rentan terhadap serangan. Hal ini dapat memberikan penyerang akses ilegal atau informasi yang sensitif. Contoh kasusnya ketika sistem dapat memperlihatkan pesan kesalahan rinci yang memaparkan konfigurasi sistem.

2.3.6 *Vulnerable and Outdated Components*

Kerentanan ini dapat timbul dari penggunaan komponen atau perangkat lunak yang tua atau tidak pernah diperbarui sehingga menjadi rentan terhadap kerentanan yang dapat diketahui. Biasanya terkait dengan penggunaan *framework* yang digunakan dalam membangun aplikasi atau *website*. Maka dari itu penyerang disini dapat masuk dan mengedit kode lalu masuk ke sistem dan dapat mengakibatkan *ransomware* atau pencurian data. Kerentanan ini dapat diatasi salah satunya dengan melakukan audit dan analisis sebelum meriliskan aplikasi atau *website*.

2.3.7 *Indentification and Authentication Failures*

Kegagalan dalam identifikasi dan penerapan autentifikasi ini biasanya terjadi karena kegagalan autentifikasi dua faktor yaitu penggunaan kata sandi yang tidak kuat atau gagalnya proses login dari *user*. Hal ini dapat menimbulkan celah serangan, contohnya jika tidak terdapat pembatasan pada saat percobaan *login*

maka sistem tersebut akan sangat rentan terhadap serangan *brute force*.

2.3.8 *Software and Data Integrity Failures*

Kerentanan ini merupakan kerentanan yang cukup serius dalam sebuah aplikasi atau *website*. Kerentanan ini mengacu pada integritas suatu perangkat lunak dan data yang tidak terjamin yang dapat menyebabkan kesalahan dalam proses validasi *input* atau kurangnya perlindungan pada saat memanipulasikannya. Hal ini dapat membuat penyerang bisa mengubah dan merusak perangkat lunak atau data yang digunakan oleh suatu *website* tersebut.

2.3.9 *Security Logging and Monitoring Failures*

Kerentanan pada aplikasi atau *website* yang disebabkan oleh kelalaian dalam pencatatan dan pemantauan pada aktivitas keamanan *website* yang tidak memadai. Hal ini menyebabkan keterlambatan dan kesulitan dalam menelusuri serangan yang terdapat pada aplikasi atau *website* sehingga dapat membuka celah bagi serangan siber.

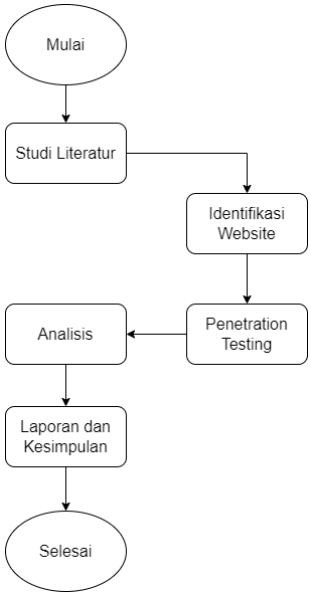
2.3.10 *Server-Side Request Forgery*

Pada kerentanan ini penyerang dapat melakukan serangan dengan cara server yang dimanipulasi agar penyerang dapat melakukan permintaan ke *database* yang tidak seharusnya dapat diakses oleh pengguna yang tidak berkewenangan. Hal ini membuat data yang sensitif dapat diakses atau dapat menyebabkan kerusakan pada sistem bahkan dapat mengambil alih server[18].

3. METODE PENELITIAN

Dalam penelitian ini membutuhkan sebuah metode atau alur kerja agar penelitian dapat dilakukan secara sistemasis dan dapat mencapai tujuan sesuai dengan yang diinginkan.

Berdasarkan gambar 1, tahapan dalam metode penelitian ini dimulai dari mencari studi literatur, kemudian mengidentifikasi *website* yang akan diuji, kemudian melakukan *penetration testing* pada *website* tersebut dan hasilnya akan di analisis sehingga dapat dilaporkan dalam bentuk laporan dan dibuatkan kesimpulan.



Gambar 1. Metode Penelitian

3.1 Studi Literatur

Pada tahap pertama penulis mencarian referensi dari jurnal dan penelitian sebelumnya, yang bertujuan untuk memahami konsep, teori dan metode yang relevan dalam melakukan pengujian penetrasi pada sebuah *website*. Pada penelitian ini penulis mencari teori pendukung mengenai *website* yang akan di uji, mengenai metode pengujian yang digunakan serta memahami teori OWASP top 10 sebagai alat acuan menganalisis keamanan pada suatu *website*.

3.2 Identifikasi *Website*

Pada tahap ini penulis mengidentifikasi *website* yang akan diuji kerentanannya. Dalam penelitian kali ini *website* yang diuji ialah *website* pengelolaan voting di Indonesia. *Website* ini terdapat banyak sekali fitur-fitur dan subdomain di dalamnya. Mulai dari fitur cek DPT, fitur yang digunakan untuk mencatat hasil rekapitulasi pemilu di seluruh Indonesia dan masih banyak lagi fitur-fitur lainnya. Selain *website* ini kaya dengan berbagai fiturnya *website* ini juga mengandung banyak data dari data yang sensitif hingga data yang memang dengaja ditampilkan dalam bentuk informasi sebagai bentuk ketransparansi lembaganya. Maka dari itu menjaga agar *website* ini agar tetap aman merupakan hal yang sangat penting, hal itulah menjadi alasan penulis memilih *website* pengelolaan voting ini sebagai *website* yang akan dilakukan uji kerentanannya.

3.3 *Penetration Testing*

Dalam melakukan penetrasi testing ini diperlukan beberapa tahapan sebagai berikut:

Gambar 2. Tahapan *Penetration Testing*

3.3.1 *Scanning*

Pada tahap pertama dalam melakukan penetrasi testing yaitu *scanning* atau pemindaian. Tahap ini bertujuan untuk mengumpulkan informasi tentang *website* target. Penulis disini melakukan proses *scanning* menggunakan *subfinder* dimana alat ini dapat membantu menemukan subdomain-subdomain pada website target sehingga penulis dapat mengetahui spesifikasi target yang akan di uji.

3.3.2 *Exploitation*

Tahap berikutnya dalam melakukan penetrasi testing adalah tahap exploitasi. Pada tahap ini penulis melakukan serangan terhadap target-target subdomain yang telah ditemukan pada tahap *scanning* dan melakukan pengujian kerentanan terhadap target tersebut. Dalam melakukan exploitasi ini menggunakan beberapa *tools* yang sering digunakan pada proses penetrasi testing yaitu Nuclei, Ex-Param, *Dirsearch*, dan JSRecon. Keempat alat tersebut memiliki pendekatan yang menyeluruh dalam pengujian ini, masing-masing alat tersebut memiliki fungsi fokus ke arah yang berbeda-beda. Dimulai dari *dirsearch* yang digunakan untuk menemukan file yang sensitif, nuclei dapat menemukan *insecure headers* seperti *header* HTTP yang tidak dikonfigurasi dengan benar dan semacamnya, ex-param berfokus pada parameter-paramter yang tersembunyi yang dapat membuka akses ke data yang penting, sedangkan JSRecon dapat mendeteksi API *key* yang terekspose dalam file JavaScript. Dengan menggunakan keempat alat ini akan mendapatkan hasil pengujian yang lebih komprehensif dan efisien sehingga mendapatkan celah kerentanan yang lebih beragam dan terdeteksi lebih mendalam.

3.3.3 Laporan Pengujian

Tahap terakhir dalam melakukan penetrasi testing ialah membuat laporan pengujian yang berisikan analisis terhadap hasil pengujian yang telah dilakukan. Proses pembuatan laporan ini meliputi kerentanan-kerentanan apa saja yang ditemukan sehingga dapat berdampak bagi *website* yang diuji.

3.4 Analisis

Setelah melakukan penetrasi testing yang menghasilkan sebuah temuan kerentanan-kerentanan pada beberapa subdomain dari *website* pengelolaan voting di Indonesia. Informasi-informasi tersebut yang akan dianalisis pada tahap ini. Dalam menganalisis kerentanan yang terjadi penulis memanfaatkan pendekatan OWASP Top 10 untuk mengidentifikasi kerentanan dan menentukan tingkat keamanan pada kerentanan tersebut.

3.5 Laporan dan Kesimpulan

Tahapan terakhir dalam penelitian adalah mengolah data berdasarkan hasil penelitian yang telah dilakukan oleh penulis menjadi sebuah laporan dan kesimpulan. Laporan yang memuat semua langkah-langkah dalam penelitian sehingga bisa mendapatkan sebuah kesimpulan yang sesuai dengan topik penelitian yang diambil.

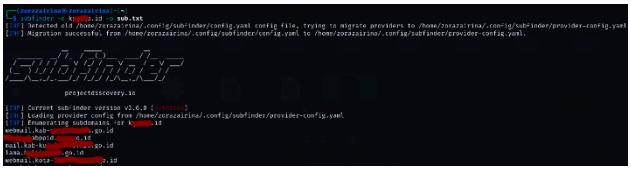
4. HASIL DAN PEMBAHASAN

Setelah melakukan beberapa tahapan dari mulai mencari referensi pada tahapan tinjauan pustaka dan merancang tahapan-tahapan dalam melakukan uji penetrasi untuk penelitian ini selanjutnya masuk ke tahap uji coba yang akan menghasilkan pembahasan dan kesimpulan. Pada *website* sistem pengelolaan voting ini penulis menggunakan 10 subdomain dari *website* tersebut untuk diuji kerentanannya. Berikut merupakan 10 *website* subdomain yang akan diuji:

Tabel I. Daftar *Website*

No	*Subdomain Website*
1	https://kab-kota***u.kp*.go.id
2	https://helpdesk.kab-s****g.kp*.go.id
3	https://absen.su****.kp*.go.id
4	https://ka*-beng*****tara.kp*.go.id
5	https://open***a.kp*.go.id
6	https://kab-la*****timur.kp*.go.id
7	https://cekpemilih.kab-*****imur.kp*.go.id
8	https://sms.kab-ma***.kp*.go.id
9	https://cekpemilih.kab-ba***.kp*.go.id
10	https://di***id.kp*.go.id

Pada tabel I yang merupakan target subdomain *website* yang akan diuji. Subdomain-subdomain tersebut didapatkan melalui proses scanning dengan menggunakan *tools subfinder* yang digunakan untuk mencari subdomain *website* yang masih aktif. Proses tersebut seperti ditujukan pada gambar 3. di bawah.

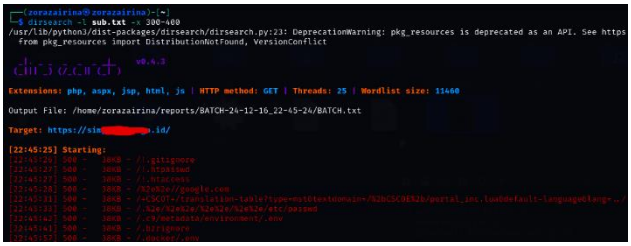


Gambar 3. *Scanning Subfinder*

4.1 *Penetration Testing*

Setelah melakukan *scanning* untuk mendapatkan subdomain dari *website* pengelolaan voting Indonesia, langkah selanjutnya yaitu mengeksekusi 10 target subdomain yang telah didapatkan. Pada proses penetrasi testing ini memerlukan beberapa langkah seperti berikut:

4.1.1 *Scanning*

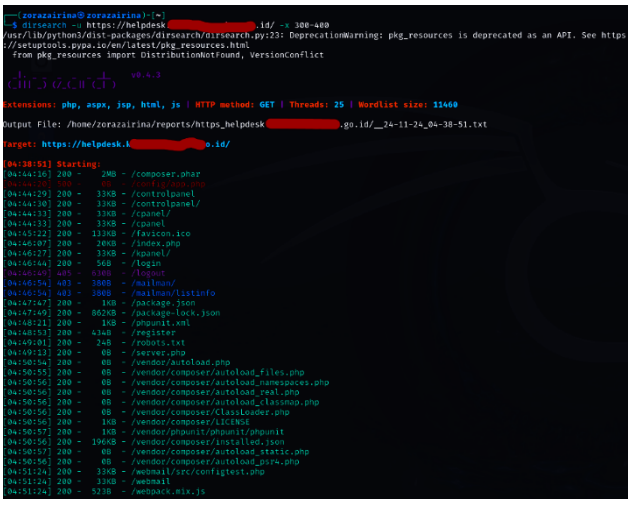


Gambar 4. *Scanning* Menggunakan *Dirsearch*

Gambar 4. di atas merupakan proses scanning menggunakan *dirsearch*. Pada tahap *scanning* ini target yang digunakan yaitu file dari sub.txt yang telah dilakukan *scanning* sebelumnya menggunakan *subfinder* untuk mencari subdomain-subdomain pada *website* pengelolaan voting ini, lalu penggunaan sintax ”-x 300-400” digunakan agar scanning dapat mengecualikan kode status HTTP direntang 300-400. *Scanning* dilakukan pada file sub.txt dari subdomain pertama hingga terakhir secara berurutan.

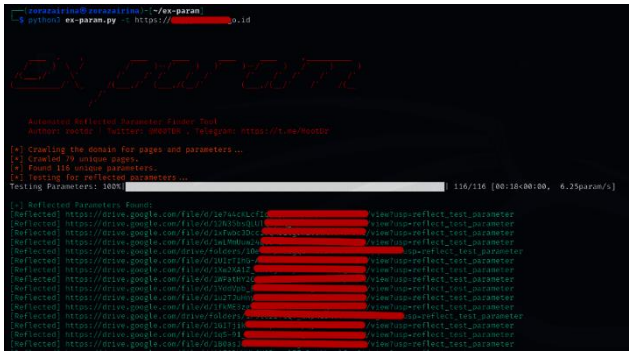
4.1.2 *Exploitation*

Penulis melakukan percobaan penetrasi testing tahap exploitasi dengan menggunakan beberapa metode yaitu *dirsearch*, nuclei, ex-param dan JSRecon. Dengan melakukan exploitasi target-target yang sudah sebutkan di atas sehingga mendapatkan hasil kerentanan yang dapat dianalisis berdasarkan OWASP Top 10 sebagai berikut:



Gambar 5. Kerentanan A01- *Broken Access Control*

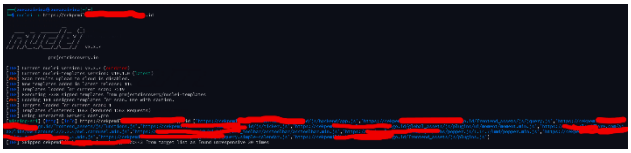
Pada gambar 5. di atas merupakan hasil dari percobaan exploitasi menggunakan *tools dirsearch*. Setelah melakukan pengujian penetrasi testing pada salah satu subdomain *website* di atas dapat ditemukan kerentanan bahwa file arsip dari composer dapat terekspose secara publik di direktori web *server*. Hal ini dapat berpotensi memberikan penyerang informasi penting mengenai sistem, dependensi dan konfigurasi proyek PHP yang digunakan. Terlebih lagi hal ini membuat penyerang dapat melihat source code dari *website* tersebut, maka dari itu kerentanan ini dikategorikan sebagai *sensitive data exposure*. Efek dari kerentanan tersebut dapat mengakibatkan pencurian data dan *defacing website* hal tersebut dapat memberikan dampak penurunan reputasi *website*. Kerentanan ini memiliki dampak yang tinggi sehingga masuk dalam kategori *high* dan dalam pengkategorian OWASP Top 10 2021 ini termasuk ke dalam A01-*broken access control* sedangkan kategori dalam CWE masuk ke dalam CWE-548 *exposure of information through directory listing*. Dalam pengujian ini didapatkan 5 hasil exploitasi lainnya yang memiliki kerentanan yang berindeks masuk ke kategrori yang sama yaitu A01-*broken access control*. Pencegahan kerentanan seperti ini dalam dilakukan dengan memperketan file *permission*, mengatur *access control* di server dan memastikan semua *library* diperbarui.



Gambar 6. Kerentanan A03-*Injection*

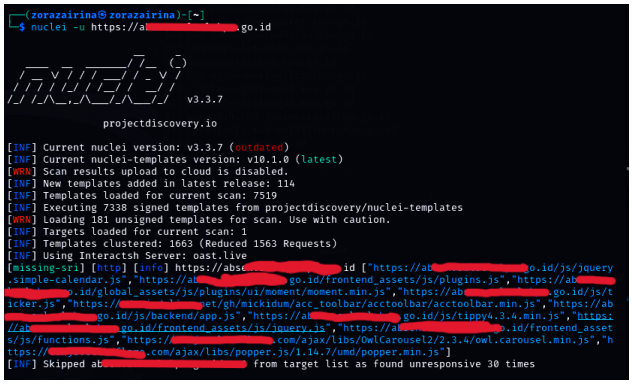
Pada gambar di atas merupakan proses dari exploitasi menggunakan *tools* ex-param. Gambar di atas bertujuan untuk menguji dan mencari parameter yang rentan terhadap exploitasi, apakah ada parameter yang dipantulkan dalam respons server. Setelah mengexploitasi target ditemukan kerentanan berupa 18 *reflected parameters*, parameter yang ditemukan berupa URL yang mengarah ke suatu folder atau file di *google drive* yang dapat dimanfaatkan kerentanannya jika tidak divalidasi atau disanitasi dengan benar. Dalam pengujian ini ditemukan 79 unik page dan 116 parameter yang berupa URL *goolge drive*, yang berisikan dokumen-dokumen penting dari istansi terkait. Maka dari itu kerentanan yang ditemukan

merupakan *reflected cross-site scripting* (XSS), hal ini dapat mengakibatkan penyerang dapat meyisipkan skrip berbahaya ke dalam parameter URL yang akan dieksekusi ketika link tersebut diakses. Kerentanan ini juga dapat mengakibatkan penyerang untuk bisa mencuri data sensitif seperti cookie sesi lalu mengarahkan ke situs berbahaya atau *phishing* dan dapat memanipulasi tampilan halaman. Kerentanan ini masuk ke dalam kategori OWASP Top 10 *injection* karena bisa melibatkan *input* yang tidak aman, dalam klasifikasi CWE dapat dikategorikan sebagai *improper neutralisation of input during web page generation* (*Cross-Site Scripting*). Agar kerentanan seperti ini tidak lagi terjadi pengembang dapat memvalidasi dan sanitasi *input user* dimana semua *input* yang diterima baik dalam bentuk URL, atau parameter lainnya harus disanitasi dan periksa dengan baik agar *inputan* yang bisa masuk harus sesuai dengan data yang diinginkan.



Gambar 7. Kerentanan A05-*Security Misconfiguration*

Pada gambar 7 di atas merupakan hasil dari exploitasi menggunakan metode nuclei. Setelah melakukan exploitasi pada targer sehingga mendapatkan sebuah celah kerentanan pada *website* tersebut. Pada gambar di atas memperlihatkan bahwa terjadinya *missing* file atau *script* pada JavaScript dari *website* seperti backend/app.js dan lain-lain, yang artinya beberapa ases dari *frontend* pada target tidak dapat diakses atau ditemukan, hal itu dapat menandakan bahwa adanya *misconfiguration* di server. Kerentaan tersebut dapat mengakibatkan *website* bisa saja tidak bekerja sebagaimana mestinya dan juga file yang hilang dapat dimanfaatkan penyerang untuk menemukan celah keamanan lain seperti path traversal atau directory listing. Kerentanan yang ditemukan termasuk CWE-552 *files or directories accessible to external parties*, dan pada OWASP Top 10 termasuk dalam kategori A05 *security misconfiguration*. Sehingga kerentanan ini termasuk ke dalam tingkatan medium severity. Agar dapat mencegah terjadinya kerentanan seperti *misconfiguration* di atas para pengembang dapat melakukan audit *endpoint* dimana pengembang harus memastikan *endpoint* yang dibutuhkan saja yang aktif serta pengembang harus mengamankan aset statis seperti javascript atau lainnya diperbarui secara teratur dan disimpan dengan aman.



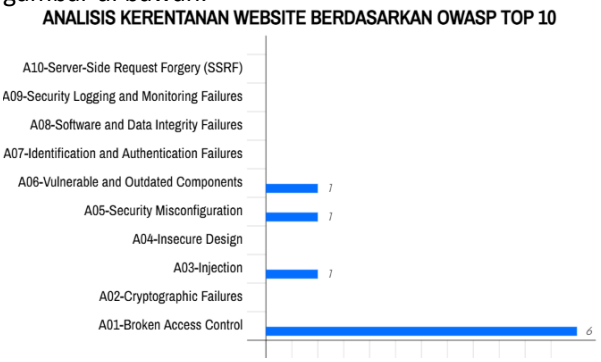
Gambar 8. Kerentanan A06- *Vulnerable and Outdated Components*

Pada gambar 8 di atas merupakan gambar exploitasi menggunakan metode nuclei dengan target dari salah satu subdomain *website*. Gambar di atas menunjukkan adanya potensi celah kerentanan dari penggunaan komponen atau javascript eksternal atau belum diperbarui seperti jquery.js, moment.min.js, dan owl.carousel.min.js. Apabila komponen atau pustaka-pustaka tersbut tidak diperbarui secara berkala akan memiliki celah keamanan yang mudah diketahui oleh penyerang. Serangan-serangan yang dapat berpotensi dilayangkan oleh penyerang seperti *cross-site scripting* (XSS), validasi *input* yang buruk sehingga dapat memanipulasi data atau eksekusi kode jarak jauh atau *remote code execution* (RCE) dan manipulasi *interface*. Maka kerentanan ini masuk ke dalam kategori OWASP Top 10 *vulnerable and outdated components* dan juga dalam klasifikasi berdasarkan CWE masuk dalam CWE-937 *use of components with known vulnerabilities*. Tidak hanya mengakibatkan kelemahan-kelemahan tadi, kerentanan ini juga dapat berdampak serius terutama apabila *website* ini digunakan untuk fungsi kritis seperti data pemilu atau absensi yang melibatkan data sensitif. Dengan melakukan beberapa hal seperti rutin melakukan audit dan pembaruan pustaka yang digunakan, menerapkan *content security policy* (CSP) dan memvalidasi semua *input user* sebelum di proses untuk mencegah XSS kerentanan ini dapat dicegah.

Setelah melakukan semua percobaan pengujian terhadap 10 subdomain yang menjadi target pada penelitian, dan selanjutnya telah dilakukan analisis dari hasil pengujian mengenai kerentanan-kerentanan yang didapatkan. Penulis melakukan penetrasi testing pada penelitian ini dengan 4 alat yaitu *dirsearch*, nuclei, ex-param dan terakhir JSRecon. Alat-alat tersebut memiliki tujuan dan fungsi yang berbeda-beda. *Dirseacrh* digunakan untuk menemukan file/direktori tersembunyi dari *website* atau sistem, nuclei biasa digunakan untuk mendeteksi kerentanan secara otomatis, ex-param untuk menemukan parameter

tersembunyi pada URL yang dapat dieksploitasi lebih lanjut sedangkan JSRecon digunakan untuk mengekstrak informasi penting dari file JS.

Kerentanan-kerentanan yang ditemukan kemudian dianalisis. Analisis dilakukan berdasarkan kerentaan OWASP top 10 2021 dan CWE (*Common Weakness Enumeration*). Setelah itu mengukur *severity* keamanan dalam sembuah kerentanan apakah dikategorikan *low, medium, high*, atau bahkan tingkat yang tertinggi yaitu *critical*. Sehingga dapat ditemukan cara penanggulangan dari masing-masing kerentanan tersebut. Pada penelitian ini berfokus pada analisis kerentanan berdasarkan OWASP Top 10 2021, maka dari itu berdasarkan subdomain yang telah diuji sehingga penulis mendapatkan beberapa kerentanan yang divisualisasikan dalam bentuk grafik batang pada gambar di bawah.



Gambar 9. Grafik OWASP Top 10

Gambar di atas merupakan hasil dari grafik percobaan pengujian penetrasi pada 10 subdomain target pada *website* pengelolaan voting di Indonesia. Maka hasil dari percobaan tersebut 9 dari 10 target yang diuji coba memiliki hasil kategori kerentanan berdasarkan OWASP Top 10 yang berbeda-beda. Dalam melakukan percobaan ini ditemukan target paling banyak memiliki kerentanan dengan kategori OWASP A01-*broken access control* dengan jumlah 6 target. Dilanjutkan dengan penemuan kerentanan dengan kategori OWASP A03-*injection*, A05-*security misconfiguration*, dan A06-*vulnarable and outdated components* yang masing-masing berjumlah 1. Analisis ini membuktikan bahwa masih perlu adanya perbaikan-perbaikan pada *website* ini, terlebih lagi pada *broken access control* yang dimana *website* ini memiliki system kontrol akses yang tidak efektif.

Hasil pengujian menunjukkan bahwa penerapan OWASP Top 10 sebagai kerangka analisis dapat mengidentifikasi berbagai kerentanan yang mengancam dalam *website* sistem pengelolaan voting di Indonesia sehingga memiliki dampak yang signifikan terhadap peningkatan keamanan *website* tersebut, terutama terkait pencegahan serangan yang dapat memanipulasikan hasil pemilu. Dengan menggunakan metode OWASP Top 10 dan didukung oleh alat-alat penetrasi seperti Nuclei, Ex-Param, *Dirsearch*, dan JSRecon beberapa celah dalam *website* ini dapat ditemukan yang sebelumnya belum terdeteksi oleh pengujian otomatis. Salah satu temuan kerentanan yaitu SQL *injection*, kerentanan ini dapat memungkinkan penyerang untuk mengakses dan memanipulasikan data hasil pemilu yang dapat merusak kepercayaan publik terhadap integritas pemilu.

4.2 Rekomendasi

Tabel II. Daftar Rekomendasi

Vulnerabilities	**Rekomendasi**
A01-*Broken Access Control*	• Menambahkan autentikasi berbasis token (JWT atau Oauth) pada *endpoint*. • Menerapkan kebijakan otoritas berbasis *Role-Based Access Control* pada *endpoint*. • Menonaktifkan *endpoint* yang tidak digunakan
A03-*Injection*	• Menerapkan sanitasi *input* dengan menggunakan pustaka validasi data seperti validator.js sebelum memproses server untuk menghindari serangan XSS. • Menggunakan *parameterized queries* dan *prepared statements* untuk memitigasi SQL *injection*
A05-*Security Misconfiguration*	• Menonaktifkan direktori listing dalam konfigurasi server *website* contohnya pada *options-indexes* pada apache. • Memastikan bahwa file log.env, config.php atau file konfigurasi lainnya tidak dapat diakses publik. • Memberikan minimum hak untuk akun server dan database.
A06-*Vulnarable And Outdated Components*	• Memperbarui Pustaka pihak ketiga ke versi terkini dan lakukan audit secara berkala menggunakan alat

Vulnerabilities	**Rekomendasi**
	seperti OWASP *depedency-check* dan npm audit. • Menghindari menggunakan Pustaka pihak ketiga yang tidak memiliki dukungan aktif atau informasi yang jelas.

Rekomendasi jangka panjang agar *website* sistem pengelolaan voting ini tetap terjaga keamanannya diperlukan strategi yang komprehensif yang didukung oleh kebijakan dari pemerintah. Pemerintah perlu mengembangkan kebijakan yang mendorong penerapan standar keamanan siber lebih ketat pada setiap *website* atau aplikasi milik instasi pemerintahan, serta perlunya pengujian keamanan secara rutin atau pengujian sebelum sistem dipergunakan. Terutama pada *website* sistem pengelolaan voting ini dapat dilakukan diwaktu menjelang periode pemilu. Dengan melakukan langkah ini Indonesia tidak hanya dapat meningkatkan keamanan sistemnya saja tetapi juga dapat memperkuat kepercayaan masyarakat terhadap proses demokrasi.

5. KESIMPULAN DAN SARAN

Setelah melakukan pengujian keamanan pada *website* pengelolaan voting di Indonesia berdasarkan parameter OWASP Top 10 tahun 2021 dengan mengambil 10 target sample subdomain dari *website* tersebut. Maka berdasarkan hasil pengujian melalui *penetration testing* dengan berbagai metode seperti *dirsearch*, nuclei, ex-param, dan JSRecon telah berhasil ditemukan beberapa kerentanan yaitu 9 dari 10 *website* target memiliki kerentanan diantaranya, 6 target subdomain *website* termasuk dalam OWASP A01-*broken access control*, 1 OWASP A03-*injection*, 1 OWASP A05-*security misconfiguration*, dan 1 lagi termasuk dalam OWASP A06-*vulnarable and outdated components*. Hasil temuan tersebut mengindikasikan adanya kerentanan dalam konfigurasi keamanan yang lemah, otoritasi akses dan penggunaan komponen perangkat lunak yang tidak aman atau kadaluwarsa. Dengan adanya penelitian ini diharapkan *website* dari intansi melakukan perbaikan sistem sesuai dengan rekomendasi berdasarkan OWASP Top 10 di atas dan dengan lebih memperhatikan peningkatan prosedur keamanan, perlindungan data serta secara rutin melakukan pengujian penetrasi dan menerapkan *framework* keamanan berbasis standar internasional.

DAFTAR PUSTAKA

[1] B. Wicaksono, R. Yuliana Rachmawati Kusumaningsih, C. Iswahyudi, J. Informatika, and I. Akprind, "PENGUJIAN CELAH KEAMANAN APLIKASI BERBASIS WEB MENGGUNAKAN TEKNIK PENETRATION TESTING DAN DAST (DYNAMIC APPLICATION SECURITY TESTING)," *Jurnal JARKOM*, vol. 8, no. 1, 2020, [Online]. Available: http://bagusw.win.

[2] D. M. D. Warouw, "PENTINGNYA WEBSITE SEBAGAI MEDIA INFORMASI DESTINASI WISATA DI DINAS KEBUDAYAAN DAN PARIWISATA KABUPATEN MINAHASA Oleh YUNICE ZEVANYA SURENTU," *ActaDiurnaKomunikasi*, vol. 2, no. 4, 2020.

[3] M. Labodo and T. Ilham, "PENGUATAN DEMOKRASI: PARTAI POLITIK DAN (SISTEM) PEMILU SEBAGAI PILAR DEMOKRASI," *MasyarakatIndonesia*, vol. 42, no. 1, pp. 115–126, 2016.

[4] F. Wisnaeni, "DAMPAK PANDEMI COVID-19: MODERNISASI DAN DIGITALISASI KOMISI PEMILIHAN UMUM REPUBLIK INDONESIA (KPU-RI)," *JurnalIlmiahGaluhJustisi*, vol. 8, no. 2, 2020.

[5] L. M. Gultom and M. Harahap, "ANALISIS CELAH KEAMANAN WEBSITE INSTANSI PEMERINTAHAN DI SUMATERA UTARA," *Jurnal Teknovasi*, vol. 02, no. 2, pp. 1–7, 2015, [Online]. Available: www.binjaikota.go.id

[6] A. M. Tania, D. Setiyadi, and F. N. Khasanah, "Copyright@2018. P2M STMIK BINA INSANI Keamanan Website Menggunakan Vulnerability Assessment," *INFORMATICS FOR EDUCATORS AND PROFESSIONALS*, vol. 2, no. 2, pp. 171–180, 2018.

[7] B. H. Setiawan and U. F. Najicha, "PERLINDUNGAN DATA PRIBADI WARGA NEGARA INDONESIA TERKAIT DENGAN KEBOCORAN DATA," *JurnalKewarganegaraan*, vol. 6, no. 1, pp. 976–982, 2022.

[8] S. Nurul, S. Anggrainy, and S. Aprelyani, "FAKTOR-FAKTOR YANG MEMPENGARUHI KEAMANAN SISTEM INFORMASI: KEAMANAN INFORMASI, TEKNOLOGI INFORMASI DAN

NETWORK (LITERATURE REVIEW SIM)," *JEMSI:JurnalEkonomiManajemenSistemInformasi*, vol. 3, no. 5, 2022, doi: 10.31933/jemsi.v3i5.

[9] A. W. Kuncoro, J. Informatika, F. Rahma, and M. E. Jurusan Informatika, "Analisis Metode Open Web Application Security Project (OWASP) pada Pengujian Keamanan Website: Literature Review," *AUTOMATA*, vol. 3, no. 1, 2022, [Online]. Available: https://www.sciencedirect.com

[10] E. Z. Darojat, E. Sediyono, and I. Sembiring, "Vulnerability Assessment Website E-Government dengan NIST SP 800-115 dan OWASP Menggunakan Web Vulnerability Scanner," *JURNAL SISTEM INFORMASI BISNIS*, vol. 12, no. 1, pp. 36–44, Sep. 2022, doi: 10.21456/vol12iss1pp36-44.

[11] Y. Thurfah Afifa Rosaliah and B. Hananto, "Pengujian Celah Keamanan Website Menggunakan Teknik Penetration Testing dan Metode OWASP TOP 10 pada Website SIM xxx," *Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA) Jakarta-Indonesia*, vol. 2, no. 2, pp. 33–46, 2021.

[12] 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, Feb. 2023, doi: 10.34148/teknika.v12i1.571.

[13] N. A. Prasetiyo, R. B. Huwae, and A. H. Jatmika, "AUDIT DAN ANALISIS WEBSITE PEMERINTAH MENGGUNAKAN PENGUJIAN PENETRASI SQL INJECTION DAN CROSS SITE SCRIPTING (XSS) (Audit and Analysis of Government Websites Using SQL Injection and Cross-Site Scripting (XSS) Penetration Testing)," *Jurnal Teknologi Informasi, Komputer, dan Aplikasinya (JTIKA)*, vol. 6, no. 2, pp. 525–533, 2024, [Online]. Available: http://jtika.if.unram.ac.id/index.php/JTIKA/

[14] M. hardiyanti, A. P. Pratama, D. A. Saputra, M. M. Sholehah, and A. R. M. R, "URGENSI SISTEM E-VOTING DAN SIREKAP DALAM PENYELENGGARAAN PEMILU 2024," *JurnalEquitable*, vol. 7, no. 2, 2022.

[15] S. Hidayatulloh and D. Saptadiaji, "Penetration Testing pada Website Universitas ARS Menggunakan Open Web Application Security Project (OWASP)," *JurnalAlgoritma*, vol. 18, no. 1, pp. 77–86, 2021, [Online]. Available: http://jurnal.itg.ac.id/

[16] R. R. Yusuf and T. N. Suharsono, "PENGUJIAN KEAMANAN DENGAN METODE OWASP TOP 10 PADA WEBSITE EFORM HELPDESK," *Prosiding Seminar Sosial Politik, Bisnis, Akuntansi dan Teknik (SoBAT)*, vol. 5, pp. 405–413, 2023.

[17] E. Nurelasari, D. Gumilang, and A. Farabi, "ANALISIS KEAMANAN SISTEM WEBSITE MENGGUNAKAN METODE OPEN WEB APPLICATION SECURITY PROJECT (OWASP) PADA SIMANTEP.ID," *Jurnal Mahasiswa Teknik Informatika*, vol. 8, no. 3, pp. 3049–3054, 2024.

[18] A. F. M. Ramadhan and S. A. Ilmananda, "Analisis Ancaman KeamananPada Sistem informasi Akademik Kampus Menggunakan Metode OWASP ZAP," *JurnalMahasiswateknikInformatika*, vol. 8, no. 4, pp. 7985–7991, 2024.