Pengujian Kelemahan Keamanan Aplikasi Web Menggunakan Peretasan Etis

Nurdin[1], Salman[2], Marsellus O Kadang[3]
[1,2]Prodi Teknik Informatika, [3]Prodi Sistem Informasi, [1,2,3]Universitas Dipa Makassar
Jl. Perintis Kemerdekaan KM.9 Makassar
Email :[1]nurdin@undipa.ac.id, [2]salmanhannake@undipa.ac.id, [3]mkadang2000@gmail.com

Abstrak

Di era digital, segalanya menjadi terhubung melalui jaringan, sehingga ketika berbagai layanan disediakan oleh aplikasi web, orang menjadi rentan terhadap peretasan. Menurut laporan ancaman keamanan 2019 dari Symantec, rata-rata 4.800 situs web rentan terhadap pencurian informasi digital. Makalah ini bertujuan untuk mengidentifikasi celah dan kelemahan dalam jaringan serta aplikasi web menggunakan pengujian penetrasi guna melindungi institusi dari ancaman siber. Banyak metode pemindaian yang disarankan oleh berbagai penulis untuk mengidentifikasi kelemahan. Namun, dalam penelitian ini, analisis dan penilaian kerentanan dilakukan menggunakan alat Nikto, Zed Attack Proxy (ZAP) dari OWASP, Netcraft, Sparta, dan Network Mapper (NMAP), yang diuji melalui platform Kali Linux dan mesin pencari. Alat ZAP dan Nikto diuji pada sepuluh domain berbeda untuk mengidentifikasi kelemahan keamanan. Hasil analisis menunjukkan bahwa alat ZAP menemukan serangan tingkat rendah. Dari hasil perbandingan antara alat Nikto dan ZAP, Nikto mengidentifikasi lebih banyak kelemahan dibandingkan ZAP.

Kata kunci: Pengujian Penetrasi, Kerentanan Keamanan, Nikto.

Abstract

In the digital age, everything is becoming networked, so when various services are provided by web applications, people become vulnerable to hacking. According to Symantec's 2019 security threat report, an average of 4,800 websites are vulnerable to digital information theft. This paper aims to identify gaps and weaknesses in networks and web applications using penetration testing to protect institutions from cyber threats. Many scanning methods are suggested by various authors to identify weaknesses. However, in this study, vulnerability analysis and assessment was performed using Nikto, Zed Attack Proxy (ZAP) tool from OWASP, Netcraft, Sparta, and Network Mapper (NMAP), which were tested through Kali Linux platform and search engine. The ZAP and Nikto tools were tested on ten different domains to identify security flaws. The analysis results showed that the ZAP tool found low-level attacks. From the comparison results between Nikto and ZAP tools, Nikto identified more weaknesses than ZAP.

Keywords*: Penetration Testing, Security Vulnerabilities, Nikto*

1. Pendahuluan

Dalam kehidupan kita sehari-hari, semua *domain* dari dari aplikasi perbankan hingga organisasi pemerintah dan aplikasi seluler menggunakan layanan *web* untuk mengirim dan menerima informasi. Aplikasi *web* adalah yang yang paling rentan terhadap peretasan [1]. Taktik yang paling utama peretas adalah mengidentifikasi celah dalam jaringan infrastruktur, mencuri *data* rahasia dan kata sandi dan meretas informasi dari organisasi yang dapat menyebabkan kerugian finansial. Kejahatan dunia maya yang dirilis oleh laporan keamanan RSA 2019, menyebutkan 43% penipuan meningkat di media sosial melalui aplikasi *web* [2].

Peretasan etis atau pengujian penetrasi atau serangan white-hat adalah alat penting untuk menguji sistem komputer dan aplikasi jaringan atau *web* untuk menemukan kelemahan keamanan. Para peretas biasanya menyerang situs *web* terbuka dengan Serangan sisi klien atau serangan sisi server dengan cara yang berbeda metode untuk masuk ke jaringan untuk menemukan kerentanan. Hal ini dapat dicapai dengan metode HTTP seperti metode untuk mendapatkan (melalui URL), metode *posting*, metode menempatkan dan menghapus atau *web cookie* (Halaman Beranda), dan ancaman dilakukan melalui pengujian otomatis atau manual.

Aktivitas kejahatan siber semakin meningkat dari hari ke hari dan mengeksploitasi situs *web* karena tidak adanya keamanan dalam infrastruktur jaringan [3]. Perlindungan *data* adalah tingkat prioritas tertinggi saat ini sehingga pekerjaan yang menonjol adalah menemukan kelemahan keamanan dalam jaringan dan aplikasi *web*. Tujuan utama dari makalah ini adalah untuk menemukan bagaimana peretas mengidentifikasi celah dalam infrastruktur jaringan untuk menyerang aplikasi *web*. Dengan demikian, analisis kerentanan dan teknik penilaian *web* digunakan untuk mengumpulkan informasi dan ancaman dunia maya yang terkait dengannya. Makalah ini membantu mengamankan aplikasi *web* di masa depan.

Etika kebajikan yang ditemukan kurang dalam perusahaan sementara individu dipengaruhi oleh peretasan manusia; oleh karena itu dianalisis teori etika menggunakan pengujian penetrasi dalam rekayasa sosial dengan pertimbangan etika kebajikan [4]. Peretas telah mengidentifikasi kelemahan keamanan dalam jaringan dan meluncurkan serangan karena internet sangat kuat dalam peralatan, staf, kode, elemen jaringan, dan *firmware* [5] Studi kasus dilakukan di atas kertas untuk beberapa alat. Makalah ini telah menggunakan alat yang berbeda seperti NMAP, *Metasploit*, dan *meterpreter* di Kali Linux untuk menemukan kelemahan.

Teknik *Penetration Testing* (PT) dan *Vulnerability Assessment* (VA) untuk menemukan celah keamanan dalam sebuah organisasi. Disarankan untuk Meluncurkan patch keamanan untuk meminimalisir ancaman dan langkah pencegahan terhadap *Owasp Top* 10 [6]. Terdapat dua model untuk mengidentifikasi kelemahan aplikasi *web* dengan pemindai skrip *python* dan meminimalkan kerentanan dengan *ModSecurity*. *ModSecurity* adalah sebuah *firewall* teknik *web*. Menggunakan teknik ini untuk menemukan lebih banyak kelemahan dan membedakan ke dalam tingkat serangan yang rendah, sedang atau tinggi [7].

Penelitian tentang faktor dan komponen penting yang dipertimbangkan untuk pengujian penetrasi dan memperkenalkan beberapa alat dan proses dalam peraturan TI [8]. Dipresentasikan sebuah pengujian dorongan untuk kata sandi yang lebih kuat untuk saran etika dalam proses otentikasi dari berbagai literatur [9]. Penelitian tentang pentingnya kursus peretasan yang beretika untuk melindungi jaringan komputer [10].

Penggunaan alat analisis *OWASP* untuk mengukur tingkat kerentanan dalam pengembangan keamanan aplikasi *web* [11]. Penggunaan otomatisasi dan pengujian manual untuk memeriksa kerentanan pada aplikasi *web*. Analisis komparatif dilakukan secara otomatis oleh *OWASP ZAP, Acunetix* dan *Burp Suite* [12] Pengujian manual dilakukan dengan alat *Vulnerability Assessment* and *Penetration Testing* (VAPT) dan hasilnya adalah akurasi 100%, dan pengujian manual memberikan hasil yang lebih baik daripada pengujian otomatisasi.

Terdapat berbagai serangan kejahatan siber seperti ransomware dan spear-phishing di situs *web* dan mengevaluasi hasil pengujian menggunakan pemindai kerentanan *web Vulscan* dan *OWASP ZAP*. Pemindai ini akan mendeteksi kerentanan skrip lintas situs (XSS) dan injeksi *SQL*. *XSS* adalah serangan sisi klien di mana penyerang menyuntikkan skrip ke halaman *web* pada halaman yang ditargetkan dan kerentanan injeksi *SQL* dilakukan pada server basis *data* ketika input tidak diambil dengan benar dari basis *data* [13].

Proses *Software Development Life Cycle* (SDLC) penilaian kerentanan aplikasi *web* merupakan aspek utama dalam fase keamanan. Oleh karena itu, kerentanan dalam aplikasi *web* seperti entitas eksternal XML (XEE) dan pemalsuan permintaan lintas situs (CSRF) dan risiko Pemalsuan Permintaan Situs Server (SSRF) telah ditemukan. Metode penilaian manual dan otomatis digunakan untuk memprediksi kelemahan dengan pembelajaran mesin dan analisis program hibrida [14].

Model keadaan berhingga yang mengidentifikasi kelemahan keamanan pada aplikasi *web* dan mengimplementasikan algoritma pembuatan jalur dan algoritma pembuatan tes berbasis penelusuran kedalaman pertama untuk melakukan inisiasi tes fungsional dalam *domain* perbankan [15]. Diperkenalkan dua fase seperti eksploitasi aplikasi dan pengintaian untuk menemukan kelemahan keamanan pada aplikasi *web* [16].

Diberikan rekomendasi *Reinforcement Learning* (RL) untuk mempelajari aktivitas yang rumit dan menyarankan intelligent automated penetration testing system (IAPTS) untuk mendapatkan informasi, meniru pengujian dan belajar dari pengetahuan [17]. Algoritma *PERSEUS, GIP, PEGASUS* digunakan untuk memecahkan masalah proses keputusan Markov yang diamati sebagian (POMDP). Terdapat enam vektor intrusi situs *web* Bangladesh menggunakan alat pengujian kotak putih dan kotak hitam dan mengidentifikasi aplikasi *web* yang menghadapi kelemahan keamanan yang serius [18].

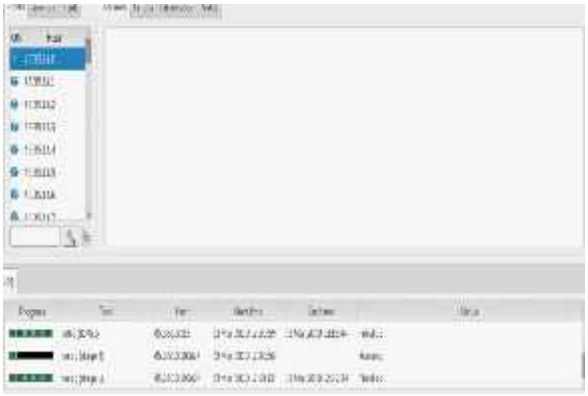
Disarankan berbagai alat dan teknik analisis untuk pengumpulan informasi, kelemahan infrastruktur jaringan dan penilaian situs *web* seperti nama *domain*, penemuan sub*domain*, pemetaan rute, ekstraksi iframe dan pelacakan alamat IP dengan menggunakan jejak teknis [19]. Pengujian penetrasi pada platform Kali Linux untuk meretas ponsel *Bluetooth*, serangan *Man-in-the-Middle*, pengujian

penetrasi ponsel dan mengendus lalu lintas menggunakan alat pemindai *port*, dan hasil kerentanan diberikan oleh diagram grafis [20].

Di sini *Sparta, Network Mapper* (NMAP), *Netcraft, Ze*NMAP, Virus total, alat pelacakan *IP* dianalisis untuk menemukan kerentanan dalam jaringan serta aplikasi *web*.

A. *Sparta*

Ini adalah alat analisis kerentanan infrastruktur jaringan, implementasi GUI python oleh ahli penetrasi, dalam tahap pencacahan dan pemindaian. Alat ini memindai *port* terbuka dan tertutup aplikasi *web* dan menemukan *port* terbuka yang lebih rentan terhadap serangan topi hitam dan maju untuk menjalankan alat tambahan terhadap layanan yang terdeteksi seperti *smbenum, snmpcheck, Nikto,* dan sebagainya. Penyerang topi hitam umumnya memiliki pengetahuan yang luas tentang melewati kebijakan keamanan dan membobol jaringan komputer untuk mendapatkan keuntungan finansial, kredensial *login* dan pengakuan pribadi, rencana politik untuk suatu perubahan sosial, tetapi tidak berwenang untuk melakukan segala jenis aktivitas pengujian penetrasi [21]. Alat pemindaian ini mengidentifikasi *port* terbuka. *Port* yang terbuka adalah jalan masuk yang bisa dimasuki oleh penyerang.



Gambar 1 Laporan Analisis oleh alat *Sparta*

B. *Network Mapper (*NMAP)

NMAP (Network Mapper) digunakan untuk evaluasi keamanan dan lokasi jaringan serta untuk menemukan log jaringan, memantau *host* dan menangani program peningkatan sumber daya. Ini lebih membantu untuk memindai jaringan yang besar. Perintah NMAP: NMAP -T4 -A - v Nama *Domain Name System* (DNS) digunakan untuk menghasilkan informasi seperti nama aplikasi, versi OS, dll. NMAP mencakup penampil hasil (ZeNMAP), alat *debugging* (Ncat) dan alat analisis respons (Nping). Analisis ZeNMAP digunakan untuk mengumpulkan *data* tentang pengelolaan layanan dan *port* yang kurang aman dan menemukan jejak *port* dalam nama *host*. Skrip ini menemukan kerentanan pada alamat IP target. Gambar 2. Laporan analisis dengan hasil NMAP diberikan berikut ini.

NMAP scan re*port* for www.undipa.ac.id (216.70.123.73)
Starting NMAP 7.70(https://NMAP.org) at 2019-07-09 03:19 EDT
PORT STATE SERVICE VERSION
80/tcp opentcpwrapped 113/tcp closed ident
443/tcp open tcpwrapped Public Key type: rsa
Public Key bits: 2048
Signature Algorithm: sha256WithRSAEncryption Device type: VoIP
phone|firewall|*web*cam|specialized OS CPE: cpe:/h:grandstream:gxp1105
cpe:/h:firebrick:fb2700

Gambar 2 Laporan Analisis dengan alat NMAP

C. *Ze*NMAP

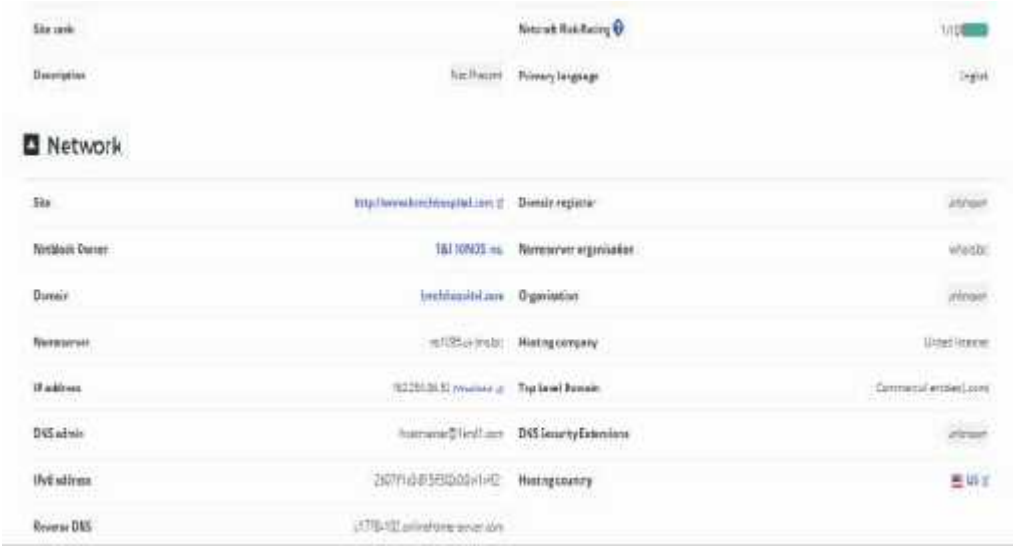
ZeNMAP juga disebut pemeta jaringan. Ini adalah pemindai keamanan NMAP dengan antarmuka pengguna grafis (GUI) lintas *platform* untuk penemuan jaringan. Alat ZeNMAP

menemukan *port* yang terbuka, jumlah HOP dan waktu perjalanan pulang pergi (RTT). Hasilnya ditunjukkan pada Gambar 3.

Gambar 3 Laporan Analisis oleh alat *Ze*NMAP

D. *Netcraft*

Netcraft menyediakan informasi keamanan lengkap mengenai situs *web* apa pun; https://searchdns.netcraft.com digunakan untuk membuat laporan keamanan tentang situs *web* target apa pun. Dalam pengujian otomatis ini, catatan riwayat *host*ing, informasi jaringan, dan pelacak *web* yang terkait dengan situs ini dan teknologinya, jaringan periklanan, dan informasi *domain* dikumpulkan.



Gambar 4 Laporan Analisis oleh alat *Netcraft*

E. Pelacak Alamat *IP*

www.ipfingerprints.com digunakan untuk mengetahui lokasi geografis dengan aktivitas pemindaian teknis pasif untuk mendapatkan informasi pribadi, *Email* atau URL [3]. Alat IP memiliki lima opsi seperti menemukan lokasi IP, situs di server, WHOis *Lookup*, memeriksa *port* yang terbuka, tes *ping*. Serangan siber dapat dilakukan dengan menggunakan alat ini.

Gambar 5 Laporan Analisis Pelacak Alamat IP

F. Total Virus

Layanan total virus mendeteksi URL untuk kode berbahaya dan *file* yang mencurigakan. *https://www.virustotal.com* digunakan untuk menemukan *sub-domain*.

Gambar 6 Laporan Analisis Berdasarkan Total Virus

Tabel 1 Hasil Perbandingan Alat yang Berbeda

Alat	Serangan
Sparta	Rentang IP yang diperiksa, serangan kamus mungkin terjadi
NMAP	Menemukan sumber *host* lain atau serangan baru
Netcraft	Mendeteksi serangan *Phishing*
*Ze*NMAP	Menemukan *port* terbuka yang mengekspos jaringan mereka ke serangan siber.
Total Virus	Malware yang berbeda dapat disisipkan menggunakan kerentanan ini.
Pelacak *IP*	Serangan DDoS

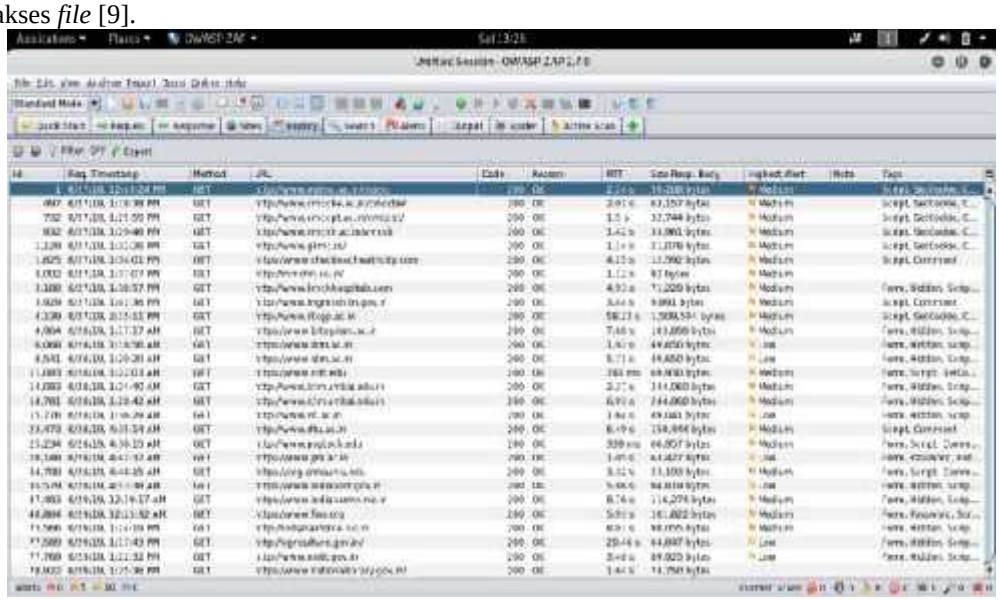
Dari hasil perbandingan berbagai alat yang terlibat dalam mengidentifikasi serangan, dapat diketahui bahwa serangan siber bisa saja terjadi.

2. Metode Penelitian

Percobaan dilakukan dengan menggunakan Intel(R) Pentium(R) CPU N3710 1.60 GHz dengan RAM 4GB. *Data* dikumpulkan dari Rumah Sakit, Sekolah Tinggi Teknik, Organisasi Pemerintah, Sekolah, Perusahaan Kesehatan, Organisasi Bisnis, Olahraga, Bank, Organisasi Keuangan, Industri TI, dan kemudian dilakukan analisis kerentanan dan penilaian terhadap 100 situs *web* dengan menggunakan nama *host*name/*host* ID. Proses pemindaian dilakukan pada platform kali Linux dengan menggunakan pengujian penetrasi pada sepuluh situs *web* teratas dari setiap *domain.*

Analisis dan penilaian kerentanan adalah langkah-langkah untuk menemukan celah keamanan dalam jaringan, sistem komputer, atau aplikasi *web* institusi dengan pemahaman pengetahuan yang sesuai tentang infrastruktur jaringan, dan mengetahui kemungkinan ancaman lingkungan. Pengujian otomatis seperti OWASP ZAP dan alat *Nikto* digunakan untuk mendeteksi kelemahan dalam infrastruktur jaringan dan aplikasi *web*.

OWASPs Zap digunakan sebagai pemindai keamanan untuk aplikasi *web*. Zap memiliki berbagai modul seperti *Proxy* untuk menangkap, *fuzzer* untuk mengidentifikasi kerentanan, Spider untuk menemukan aplikasi *web*, Scanner untuk serangan aktif dan pasif, dan metode *Dictionary* untuk mengakses *file* [9].



Gambar 7 Laporan analisis oleh alat OWASP ZAP

Tahap pengujian terdiri dari lima langkah. Seperti langkah awal, *Host* Name/*Host* ID harus diberikan sebagai input untuk menyerang. Selanjutnya, proses scanning dilakukan untuk mengidentifikasi kelemahan pada infrastruktur jaringan. Setelah kerentanan tersebut ditemukan, maka proses analisis risiko dilakukan. Terdapat empat kategori risiko yaitu; rendah, tinggi, informational dan medium. Sebagai langkah terakhir, hasilnya disimpulkan.

Dalam ZAP, empat mode serangan seperti mode standar, mode terproteksi, mode serangan, dan mode aman digunakan untuk mengidentifikasi kerentanan di *web*. Di sini pengujian pemindaian didemonstrasikan pada *mode* standar untuk mengidentifikasi risiko menggunakan metode *Get*. Kerentanan yang ditemukan pada alat ini dijelaskan Gambar 7.

Setelah proses pemindaian, ZAP mengidentifikasi risiko tingkat menengah dan rendah dan kerentanannya dijelaskan di bawah ini:

OWASP ZAP mendeteksi risiko tingkat menengah (tinggi) seperti

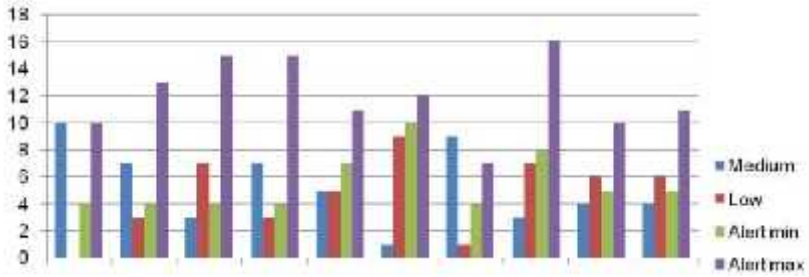
- Penulisan ulang URL-Pihak ketiga dapat melampirkan ID sesi
- Pengungkapan kesalahan aplikasi-Informasi sensitif dapat berisiko di tangan penyerang yang mengarah ke serangan ransomware
- *Header X Frame Options* (XFO) tidak disetel tanpa sepengetahuan pengguna, penyerang akan mengendalikan aktivitas komputer (serangan *clickjacking*)

- Teknik injeksi SQL digunakan untuk menyerang kelemahan lapisan *database* dari sebuah aplikasi dan digunakan untuk mengubah informasi dengan bantuan *query DELETE, INSERT, ALTER* untuk menghapus, menambah, dan mengubah *data* dari *database*.

Risiko tingkat rendah (menengah) seperti

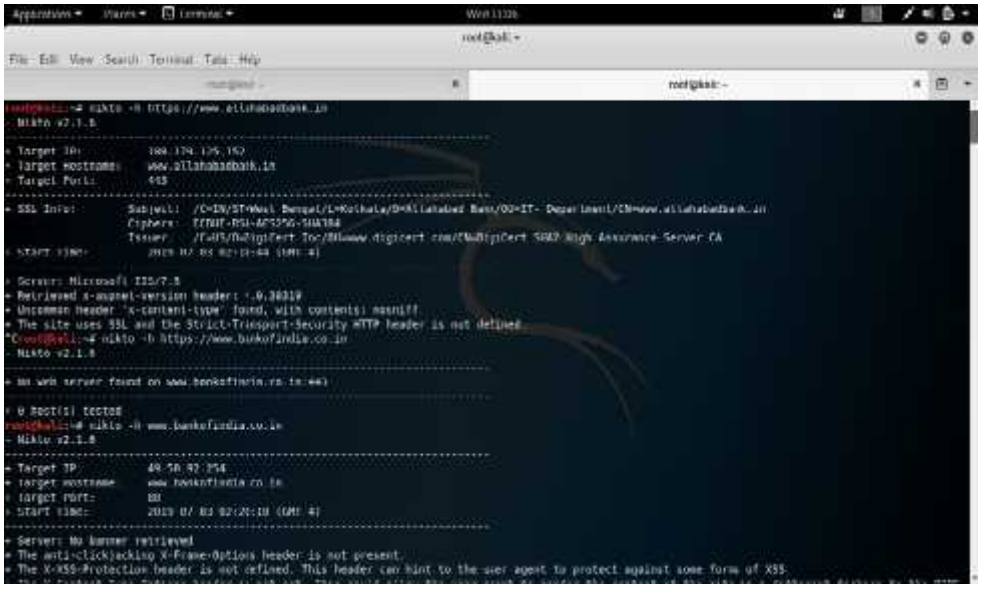
- Pemalsuan Permintaan Lintas Situs (CSRF) - Melalui kode *JavaScript*, *file* berbahaya dapat disertakan melalui berbagai tautan yang dilampirkan ke *file* sumber dan mengeksekusi tindakan yang tidak diinginkan
- *Cookie* tanpa bendera aman dan bendera HTTP saja Serangan *man-in-the-middle* terjadi dan karena itu tidak memiliki otentikasi, integritas *data*, dan kerahasiaan saat memberikan layanan.
- *Model X-Content* tidak ada ancaman serangan *sniffing* jenis *media* menyebabkan kerentanan keamanan
- Serangan skrip lintas situs (XSS) - *Browser web* tidak mengizinkan keamanan XSS. Ini adalah sistem kelemahan keamanan dalam program *web*
- Halaman yang aman mencakup konten campuran Koneksi HTTP yang tidak aman menyebabkan *file* video, gambar, dan *style sheet* dalam dokumen.
- Pragma HTTP dan kontrol *cache* tidak mengimplementasikan *header* HTTP dengan benar.

Dari lingkungan pengujian, kerentanan dan perlakuan telah terdeteksi dengan metode pemindaian dan serangan tingkat menengah dan rendah telah ditemukan dari sepuluh *domain*. Kelemahan keamanan tidak hanya pada satu *domain* tertentu, tetapi semua kerentanan *domain* juga telah terdeteksi. Hasil dari ZAP diberikan pada Gambar 8.



Gambar 8 Grafik Penilaian Hasil oleh Alat OWASP

Nikto adalah pemindai otomatis analisis keamanan jaringan *command-line* perangkat lunak gratis [20]. Digunakan untuk memindai situs *web* dan server Anda dengan segera untuk mengetahui kesalahan konfigurasi dan kerentanan keamanan. Untuk mengidentifikasi kerentanan pada aplikasi *web*, gunakan sintaks *Nikto -h* nama *host* di baris perintah.



Gambar 9 Laporan analisis oleh alat *Nikto*

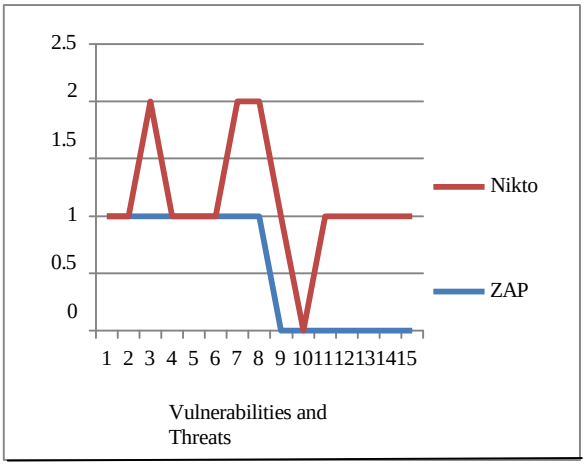
Penilaian melalui alat *Nikto* mengidentifikasi kerentanan seperti

- Kebocoran server laporkan urusannya
- *Header* perlindungan XSS dirancang
- Pembajakan *anti*-klik tidak ada
- Opsi *X-Frame* tidak efektif
- *Header* yang tidak memadai dalam keamanan trans*port*asi.
- *Header* versi *x-aspnet* yang diambil

Dari penilaian alat *Nikto*, hanya situs *web* teratas dari beberapa *domain* yang mengeksploitasi *header* sementara *domain* yang tersisa tidak menggunakan *file header* dan peringatan informasi yang ditemukan. Peringatan dan risiko dapat ditemukan dengan menggunakan alat pemindaian *OWASP ZAP*. Peringatan maksimum telah terdeteksi pada *domain* perbankan jika dibandingkan dengan *domain* lainnya. Hasilnya diberikan pada Gambar 10.

3. Hasil dan Pembahasan

Pada Gambar 10, *OWASP ZAP* berhasil menemukan risiko tingkat menengah dan rendah dan menemukan kerentanan di angka 19 (menengah) dan 81 (rendah). Tidak ada yang tinggi.



Gambar 10 Hasil perbandingan OWASP ZAP dan *Nikto Tool*

Gambar 10 menyebutkan berbagai kerentanan dan ancaman yang disebutkan di bawah ini;

- ID sesi dalam penulisan ulang URL
- Pengungkapan kesalahan aplikasi
- Opsi bingkai X *Header* tidak disetel
- Penyertaan *file* sumber *JavaScript* lintas *domain*
- *Header* tipe konten tidak ada
- Perlindungan XSS *browser web* tidak diaktifkan
- *Header* opsi jenis konten X tidak ada
- *Cookie* tanpa bendera HTTP saja dan tanpa bendera aman
- *Header X-XSS-Protection* tidak didefinisikan
- *Header* yang tidak umum ditemukan
- SSL dan keamanan-transportasi-ketat tidak didefinisikan
- *Header* HTTP keamanan tidak didefinisikan
- Server membocorkan inode melalui *ETag*
- Diambil x-didukung oleh *header*

Garis merah menunjukkan kelemahan yang terdeteksi oleh alat *Nikto* dan garis biru menunjukkan kerentanan alat ZAP. Alat *Nikto* mengidentifikasi kerentanan seperti halnya alat OWASP. Alat *Nikto* menemukan beberapa informasi tambahan seperti server, sandi, dan informasi *Secure Socket Layer* (SSL). Protokol kriptografi SSL digunakan dalam jaringan komputer untuk keamanan komunikasi. Kerentanan dan ancaman yang ditemukan oleh alat *Nikto* lebih tinggi daripada yang ditemukan oleh alat OWASP ZAP, dan sebagai perbandingan, jelas bahwa beberapa kerentanan yang terlewatkan oleh alat OWASP ZAP ditemukan oleh alat *Nikto*. Kurangnya keamanan dapat menyebabkan peretas tingkat lanjut untuk mengeksploitasi kelemahan tersebut. Di masa depan, tingkat risiko yang tinggi mungkin saja terjadi dan oleh karena itu mengidentifikasi celah pada tahap awal dalam jaringan dan aplikasi *web* sangat diperlukan. “Mencegah lebih baik daripada mengobati” adalah moto terbaik untuk mengamankan dunia maya dari penyerang.

4. Kesimpulan

Garis merah menunjukkan kelemahan yang terdeteksi oleh alat Nikto dan garis biru menunjukkan kerentanan alat ZAP. Alat Nikto mengidentifikasi kerentanan seperti halnya alat OWASP. Alat Nikto menemukan beberapa informasi tambahan seperti server, sandi, dan informasi Secure Socket Layer (SSL). Protokol kriptografi SSL digunakan dalam jaringan komputer untuk keamanan komunikasi. Kerentanan dan ancaman yang ditemukan oleh alat Nikto lebih tinggi daripada yang ditemukan oleh alat OWASP ZAP, dan sebagai perbandingan, jelas bahwa beberapa kerentanan yang terlewatkan oleh alat OWASP ZAP ditemukan oleh alat Nikto. Kurangnya keamanan dapat menyebabkan peretas tingkat lanjut untuk mengeksploitasi kelemahan tersebut. Di masa depan, tingkat risiko yang tinggi mungkin saja terjadi dan oleh karena itu mengidentifikasi celah pada tahap awal dalam jaringan dan aplikasi web sangat diperlukan. “Mencegah lebih baik daripada mengobati” adalah moto terbaik untuk mengamankan dunia maya dari penyerang.

Daftar Pustaka

[1] Joseph. Muniz and Aamir. Lakhani, Web Penetration Testing with Kali Linux : a Practical Guide to Implementing Penetration Testing Strategies on Websites, Web Applications, and Standard Web Protocols with Kali Linux.
[2] “Computer and Information Security Handbook.” [Online]. Available: http://www.elsevierdirect.com
[3] A. Afifah Rodhiyatun Nisa, Ananditto Daffa Wijayanto, Arya Prabudi Jaya Priana, and A. Setiawan, “Analisis Log Server untuk mendeteksi Serang DDoS pada Keamaan Jaringan di Website,” Journal of Internet and Software Engineering, vol. 1, no. 3, p. 17, Jun. 2024, doi: 10.47134/pjise.v1i3.2612.
[4] K. Bednar and S. Spiekermann, “The Power of Ethics: Uncovering Technology Risks and Positive Value Potentials in IT Innovation Planning,” Business and Information Systems Engineering, vol. 66, no. 2, pp. 181–201, Apr. 2024, doi: 10.1007/s12599-023-00837-4.
[5] D. Suhaila, Muhammad Karim Bachtiar, and Tedi Kurniawan, “Ananlisis Vulnerabilitas dan Pengujian Terhadap Google Gruyere,” Journal of Internet and Software Engineering, vol. 1, no. 3, p. 10, Jun. 2024, doi: 10.47134/pjise.v1i3.2574.

[6] F. Tinambunan et al., "PENGUJIAN SISTEM INFORMASI AKADEMIK UNIVERSITAS X MELALUI PENDEKATAN PENETRATION TESTING BERDASARKAN OWASP TOP 10," 2024.
[7] M. Kluban, M. Mannan, and A. Youssef, "On detecting and measuring exploitable javascript functions in real-world applications," ACM Transactions on Privacy and Security, vol. 27, no. 1, Feb. 2024, doi: 10.1145/3630253.
[8] R. Rodin, V. Anjelika, W. Wilawati, and W. F. Ninsik, "Prospek industri penerbitan di Kabupaten Rejang Lebong: Studi pada penerbit CV Andhra Grafika," Daluang: Journal of Library and Information Science, vol. 4, no. 1, pp. 32–43, Jun. 2024, doi: 10.21580/daluang.v4i1.2024.17776.
[9] S. Dargaoui, M. Azrour, A. El Allaoui, A. Guezzaz, A. Alabdulatif, and A. Alnajim, "Internet of Things Authentication Protocols: Comparative Study," Computers, Materials and Continua, vol. 79, no. 1. Tech Science Press, pp. 65–91, 2024. doi: 10.32604/cmc.2024.047625.
[10] M. Binhammad, S. Alqaydi, A. Othman, and L. H. Abuljadayel, "The Role of AI in Cyber Security: Safeguarding Digital Identity," Journal of Information Security, vol. 15, no. 02, pp. 245–278, 2024, doi: 10.4236/jis.2024.152015.
[11] F. Putra Utama, R. Muhamad, and H. Nurhadi, "Uncovering the Risk of Academic Information System Vulnerability through PTES and OWASP Method."
[12] S. Sri Rahayu, D. Aris Firmansyah, S. Susanti, and U. Adhirajasa Reswara Sanjaya Bandung, "Analisis Penggunaan Tools Automation Testing pada Aplikasi : Systematic Literature Review," Remik: Riset dan E-Jurnal Manajemen Informatika Komputer, vol. 8, no. 1, 2024, doi: 10.33395/remik.v8i1.13241.
[13] T. Olmayan et al., "Non-Technical Cyber-Attacks and International Cybersecurity: The Case of Social Engineering 1." [Online]. Available: http://dergipark.org.tr/tr/pub/jss
[14] F. Hussain, R. Rahman, Z. S. Attarbashi, W. H. N. Fadaq, and M. Mustafa, "Understanding Human Behavior in Phishing Attacks Across Diverse User Groups: An Ethical Hacking Analysis," in 2024 IEEE 1st Karachi Section Humanitarian Technology Conference, Khi-HTC 2024, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/KHI-HTC60760.2024.10482040.
[15] M. Adri Ramadhan, D. Saputra, D. Iskandar Mulyana, S. Tinggi Ilmu Komputer Cipta Karya Informatika, and D. Jakarta, "PENCEGAHAN SERANGAN BERBASIS KATA SANDI: STUDI KOMPREHENSIF TENTANG IMPLEMENTASI HASH PADA APLIKASI WEB PREVENTION OF PASSWORD-BASED ATTACKS: A COMPREHENSIVE STUDY OF HASH IMPLEMENTATION IN WEB APPLICATIONS," Journal of Information Technology and Computer Science (INTECOMS), vol. 7, no. 3, 2024.
[16] M. S. H. Tamsir Ariyadi, Irwansyah, "Analisis Keamanan Jaringan Wifi Mahasiswa UBD Dari Serangan Packet Sniffing," JURNAL ILMIAH INFORMATIKA (JIF), vol. 12, no. 01, pp. 53–58, 2024.
[17] M. A. Ghaly and S. A. Hannan, "International Journal of INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING Protecting Software Defined Networks with IoT and Deep Reinforcement Learning." [Online]. Available: www.ijisae.org
[18] V. Gustina DM and A. Ananda, "Kecerdasan Buatan untuk Security Orchestration, Automation and Response: Tinjauan Cakupan," Jurnal Komputer Terapan, vol. 10, no. 1, pp. 36–47, Jun. 2024, doi: 10.35143/jkt.v10i1.6247.
[19] F. Prasetyo et al., "PERTAHANAN TINGKAT SERVER TERHADAP SERANGAN DNS SPOOFING DI JARINGAN MODERN," 2024. [Online]. Available: https://jurnal.umj.ac.id/index.php/just-it/index
[20] A. Javed, A. Ehtsham, M. Jawad, M. N. Awais, A.-H. Qureshi, and H. Larijani, "Implementation of Lightweight Machine Learning-Based Intrusion Detection System on IoT Devices of Smart Homes," Future Internet, vol. 16, no. 6, p. 200, Jun. 2024, doi: 10.3390/fi16060200.
[21] M. Ahmed, H. R. Kambam, Y. Liu, S. Jaidka, and K. Petrova, "Impact and Significance of Human Factors in Digital Information Security." [Online]. Available: http://innove.org/ijist/.