Digital Transformation Technology (Digitech) | e-ISSN : 2807-9000
Volume 3, **Number** 1, Maret 2023
https://doi.org/10.47709/digitech.v3i1.2476

Analisis Keamanan Pada Web Aplikasi *Open Journal System* Terhadap Serangan *Cross Site Scripting* (XSS) Menggunakan Metode *Vulnerability Assessment*

Yunanri.W*
Prodi. Informatika, Fakultas Rekayasa Sistem, Universitas Teknologi Sumbawa, Indonesia
yunanri.w@uts.ac.id

Histori Artikel:
Diajukan: 24 Juni 2023
Disetujui: 05 Juli 2023
Dipublikasi: 06 Juli 2023

Kata Kunci:
Analisis; XSS; Serangan; OJS; Vulnerability

Digital Transformation Technology (Digitech) *is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0).*

Abstrak
Pesatnya perkembangan teknologi informasi telah membawa dampak positif di berbagai bidang, salah satunya adalah teknologi internet. Banyak instansi yang membangun aplikasi *web.* Tanpa memperhatikan *web* aplikasi yang dibangun aman atau ada gangguan, salah satu *web* aplikasi tersebut adalah *Open Journal System* yang digunakan untuk mempublikasikan karya ilmiah. Penelitian ini bertujuan untuk mengetahui adanya *vulnerabilitie*s pada *cross-site scripting* pada aplikasi *Open journal system* jinteks menggunakan *tools* OWASP. Pengujian ini dilakukan untuk mengamankan aplikasi yang digunakan sebagai rekomendasi tindak lanjut dalam mengamankan aplikasi. OJS jinteks Metode yang digunakan terdiri dari beberapa tahap seperti penemuan jaringan atau pengumpulan informasi, kerentanan pemindaian, pelaporan, dan perbaikan. Pengujian yang telah dilakukan berhasil menemukan kerentanan berupa 2 *medium,* 7 *low,* dan 3 *informational*. Hasil pengujian menunjukkan bahwa tidak ada kerentanan dalam skrip lintas situs pada aplikasi JINTEKS OJS.

PENDAHULUAN

Internet sebagai jaringan komunikasi global dapat dijadikan sebagai media sumber informasi terkini, seperti ilmu pengetahuan, hiburan, bisnis, teknologi dan sumber informasi lainya mencakup berbagai bidang kehidupan. Universitas Teknologi Sumbawa (UTS) mempunyai beberapa aplikasi *web* yang berisi sistem informasi dan dokumen yang dipublikasi bagi pengguna. Salah satu aplikasi *web* tersebut adalah *open journal system* jinteks. Aplikasi *web* menjadi alternatif bagi institusii dalam mempromosikan kepada masyarakat umum, aplikasi *web* juga mudah untuk diakses banyak orang yang tidak kenal tempat maupun waktu. Dengan adanya kemudahan tersebut banyak instansi membangun aplikasi web tampa memperhatikan apakah aplikasi yang dibangun sesuai atau tidak dengan standar keamanan, apakah sistem yang dibangun ada gangguan atau sudah aman(Octavriana et al., 2021)(W et al., 2018).

Banyak sekali dijumpai kerentanan pada aplikasi *web.* Salah satu kerentanan tersebut adalah *Cross Site Scripting* (XSS). Kerentanan XSS memungkinkan penyerang mengirimkan kode atau skrip yang dapat dieksekusi ke *server,* yang akan diarahkan ke *browser* pengguna. Dengan adanya kerentanan XSS tersebut membuat *hacker* mudah untuk melakukan penyerangan sehingga dari serangan tersebut *hacker* dapat memanipulasi data, mengubah tampilan *web,* melakukan pencurian data dan melakukan pemindahan hak pengelolan akses dari aplikasi *web* tersebut. Hilangnya data dan diambilnya informasi penting pada sebuah perusahaan dan organisasi untuk kepentingan individu membuat kerugian pada pihak lain, seperti mengakses data pribadi suatu perusahaan atau lembaga untuk melakukan transaksi secara eksesif sehingga menguras uang yang dimiliki korban, menggunakan data dan akses untuk melakukan penipuan dan sebagainya(Riadi et al., 2020).

Adapun *tool* yang digunakan untuk mencari kerentanan *cross site scripting* pada aplikasi OJS jinteks adalah OWASP ZAP. *Tool* ini dapat memberikan rekomendasi penanganan kerentana *cross site scripting*(W et al., 2018)(Riadi et al., 2020). Melakukan analisis deteksi vulnerability pada web server menggunakan OWASP scanner yang berhasil menemukan kerentanan yang dapat menyebabkan file lokal dapat dimanipulasi dengan menggunakan serangan croos site scripting (XSS), penelitian ini juga dilengkapi dengan solusi penanganan kerentanannya (W et al., 2016).

Pengujian kerentanan terhadap web server SIMAK dengan melakukan penetration testing menemukan dua kelemahan yaitu Apache Server Etag Header Information Disclosure dengan status medium atau middle risk dan Unix Operating System Unsupported Version Detection yang berstatus critical atau high risk (Wahyudi, 2019).

Penelitian ini bertujuan untuk mengetahui adanya *vulnerabilitie*s pada *cross-site scripting* pada aplikasi *Open journal system* jinteks menggunakan *tools* OWASP

STUDI LITERATUR

2.1 Website

Website adalah sebuah layanan di suatu domain internet yang terdiri dari 1 atau lebih halaman untuk tujuan tertentu yang dapat diakses oleh orang di dunia maya (Waryanto, 2018). World Wide Web (WWW) adalah suatu program yang ditemukan oleh Sir Timothy Jhon Tim Berners-Lee pada tahun 1991 (Hidayatullah dan Khairul, 2017). Awalnya Berners-Lee hanya ingin bagaimana menemukan suatu cara untuk memudahkan menyusun arsip – arsip risetnya. Maka dari itu Berners-Lee mengembangkan suatu sistem untuk keperluan pribadinya. Sistem yang dikembangkan adalah software yang bernama Enquire. Dengan sistem tersebut Berners-Lee berhasil menciptakan jaringan yang dapat menautkan arsip – arsip riset sehingga memudahkan dirinya ketika ingin mencari sebuah informasi yang diinginkan. Sehingga hal inilah yang menjadi dasar atau awal terbentuknya website yang sekarang sudah berkembang pesat.

2.2. Masalah Keamanan

Masalah kemanan menjadi aspek yang paling penting dari sebuah sistem multimedia, masalah keamanan sering dianggap sepele dan kurangnya perhatian dari perancang sistem multimedia (Ariyus, 2009).
Sangat fatal apabila sistem informasi yang didalamnya terdapat rahasia tercuri. untuk itu keamanan dari sistem informasi ini harus dijaga dan diperhatikan betul untuk mencegah terjadinya kejahatan komputer.

2.3. Mikrotik

Mikrotik adalah perangkat yang digunakan untuk router network dengan sitem operasi linux base, untuk menggunakannya biasanya bisa melalui aplikasi bernama winbox. Mikrotik didesain khusus untuk memudahkan pengguna dalam berbagai keperluan jaringan komputer seperti rancang bangun sistem jaringan dari skala kecil hingga kompleks, fitur mikrotik juga dibilang banyak sehingga tambah mempermudah penggunanya. Fiturnya meliputi Ipsec, caching DNS client, routing static, firewall dan NAT, web proxy, UpnP, SNMP, MNDP, monitoring atau accounting, tools dan masih banyak fitur lainnya.

2.4. Cross-Site Scripting (XSS)

Cross-Site Scripting (XSS) merupakan kejahatan keamanan website dengan memanfaatkan celah keamanan pada form input website (Fogie, Grossman, Hansen, Rager, DAN Petkov. 2007). Ketika penyerang menemukan sebuah celah xss pada sebuah website, penyerang akan memanfaatkan hal tersebut dengan memasukan sebuah script salah satunya untuk menjebak korban yang apabila korban masuk pada jebakan tersebut maka website dapat diambil alih kendali.

Serangan xss terbagi dalam 2 kategori, diantaranya:
Persistent:
Serangan ini biasanya disebut dengan stored xss, biasanya ditemukan pada halaman situs dimana client di ijinkan memasukan script contohnya pada halaman kotak pencarian (Wang dkk. 2007, 2011, Van Acker dkk. 2012).

Non-Persistent:

Serangan XSS ini biasanya disebut reflected XSS, serangan dimana penyerang dapat memasukan script yang dapat disimpan pada database sebuah situs, di mana script yang dimasukkan dikembalikan ke server aplikasi web korban contohnya tampilan kesalahan pesan yang dapat ditampilkan pada browser client lain (Avancini and Ceccato 2011, Athanasopoulos dkk. 2010).

2.5. Penelitian lain nya yang relefan:

a. OWASP merekomendasikan perusahaan-perusahaan untuk memperhatikan sepuluh ancaman keamanan aplikasi *web* yaitu *broken access control, security misconfiguration, insecure deserialization, injectionn, sensitive data exposure, XML external entities, broken authentication, cross site scripting, using components with known vulnerabilities, and insufficient logging and monitoring* (Yunus, 2019).

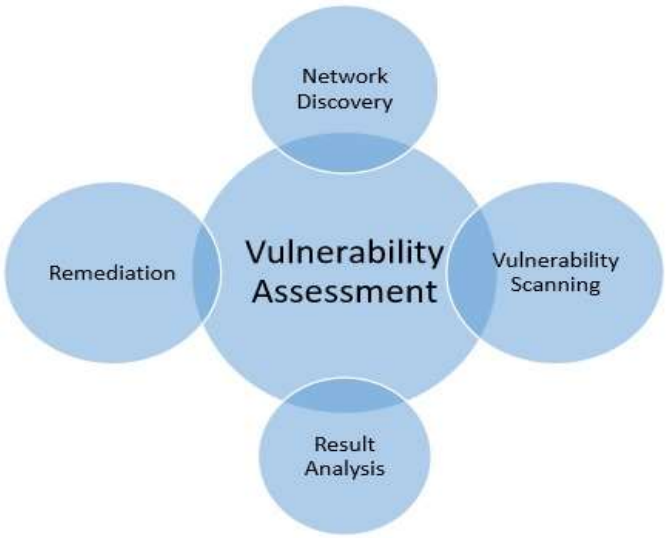
b. OWASP mengembangkan *tool* yang digunakan untuk mengamankan aplikasi *web,* salah satunya ialah *Zad Attack Proxy* (ZAP). ZAP merupakan aplikasi untuk menemukan kerentanan dalam suatu aplikasi *web* dengan cara menyediakan scanner otomatis (Syarifudin, 2018).

c. Kelebihan dari ZAP ini di antaranya bersifat mudah diinstal*, community based, open source, intercepting proxy, traditional & ajax spider, active scanner, growing add ons, forced browsing, fuzzer, dynamic, smart card support, SSL cerificates, integrared, dan web socket support* (Sunardi et al., 2019).

d. Penelitian serupa telah dilakukan oleh peneliti sebelumnya diantaranya deteksi kerentanan yang dilakukan dengan membandingkan dua *tool* yaitu ZAP dan Archni yang berhasil mendapatkan bukti dalam bentuk deskripsi, *URL,* metode, parameter, informasi, dan bukti (Sunardi et al., 2019).

METODE

Vulnerability Assessment merupakan salah satu Langkah untuk mengidentifikasi, mendeteksi dan memcari celah keamanan *cross site scripting* (XSS). Vulnerability Assessment merupakan metode untuk mengumpulkan informasi sebagai acuan rekomendasi dalam melakuakn *countermeasure* pada *system* OJS yang telah dibangun(Alazmi & De Leon, 2022)(Liu et al., 2019)(Kishimoto et al., 2020).



Gambar 1. Metode vulnerability Assessment, Audit sistem dan jaringan

Meruapakan. alur pengujian dalam penelitian ini. *Network discovery* atau *information gathering* pada tahapan ini menemukan struktur rancang bangun dari keamanan jaringan pada suatu target sasaran yang dituju. sebagai barometer metodelogi, *scanning vulnerability* bertujuan untuk mencari celah kerentanan pada *website* mengguanakan *tool* Owasp ZAP, *result analysis* merupakan kesimpulan akhir berupa tabel dari jumlah nilai dari. penelitian yang telah dilakukan, dan remediation adalah solusi yang akan diberikan sebagai upaya memperbaiki sistem yang memiliki kerentanan(Sivanesan et al., 2018).

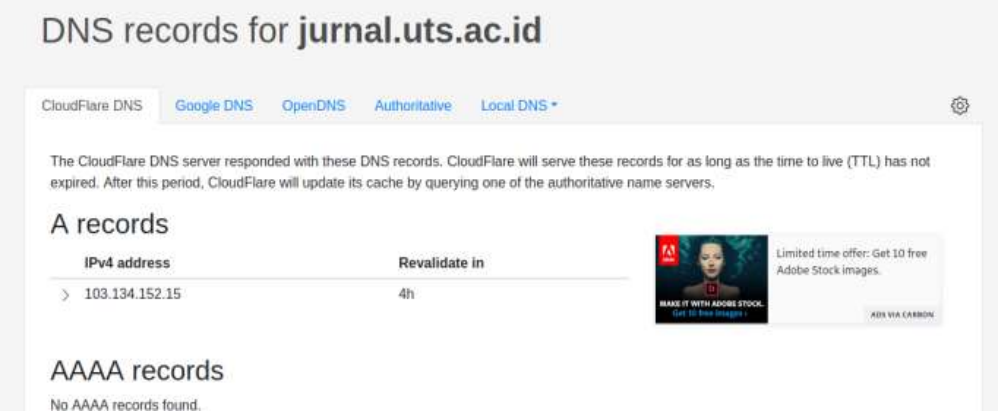
1. Network Discover merupakan tahapan mencari informasi terkait tentang: Open port, Kerentanan jaringan, Ip Addres.
2. Vulnerability Scanning: tahapan mencari kerentanan pada file sistem pada web aplikasi, webserver.
3. Result Analysis: tahapan melakukan laporan dari 2 tahapan sebelumnya.
4. Remediation: tahapan ini melakukan evaluasi kembali apakah sistem keseluruhan dapat bekerja sesuai yang di harapkan.

HASIL

Aplikasi *Open Journal System* (OJS) Jinteks memiliki 12 kerentanan, dimana kerentanan tersebut diperoleh menggunakan *tool* ZAP. Adapun alur pengujian yang digunakan untuk menemukan kerentanan pada aplikasi OJS Jinteks diantaranya sebegai berikut:

A. Network Discovery:

Nslookup merupakann *tool* yang digunakan untuk mengetahui *Ip Address* dari *Domain website* JINTEKS

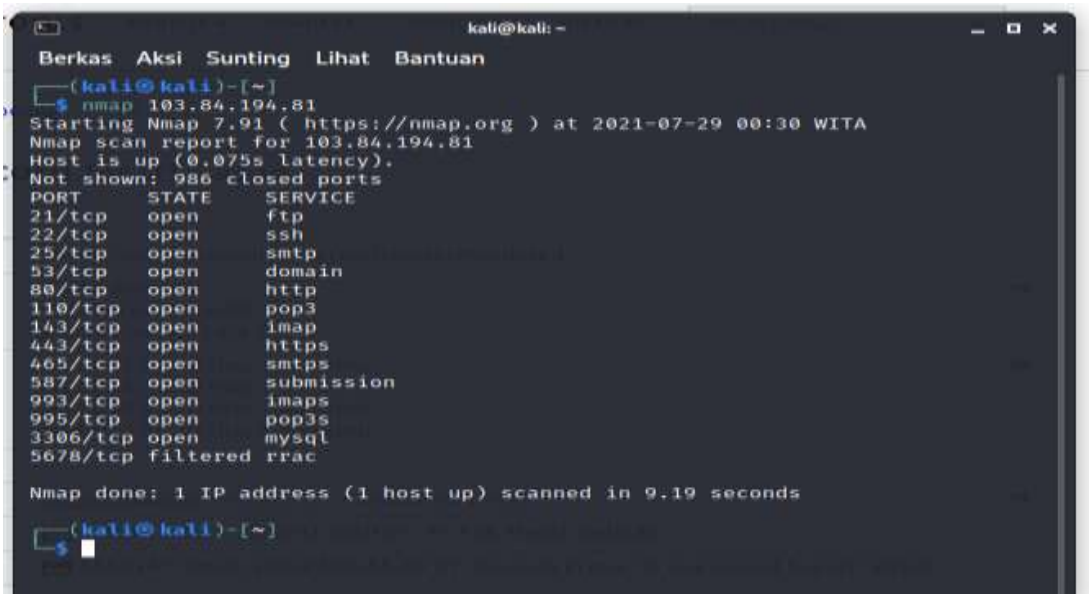


Gambar 2. Hasil s*canning tool* Nslookup

Gambar 2. Menjelaskan menemukan informasi berupa *DNS* dimana Dns itu sendiri sebuah sistem yang bertugas menyimpan semua informasi data domain dalam jaringan. Dengan adanya *DNS*, *domain* atau hostname yang ada akan ditranslate dan diterjemahkan dalam alamat *IP* sehingga dapat diakses. *DNS* ini ditemukan tahun 1983 oleh Paul Mackapetris. Sebelum menggunakan *DNS,* mapping domain dahulu menggunakan *file hosts.txt. File hosts.txt* tersebut memiliki kekurangan yaitu saat suatu *IP address* berubah, maka file juga harus berubah sehingga agak rumit. Berbeda dengan *DNS* dimana perubahan bersifat dinamis. Jadi jika ada perubahan pada suatu *host*, maka yang lainnya akan mengikuti, semuanya akan bersifat dinamis(Taha & Karabatak, 2018).

B. Scanning Port.

Nmap *Tool* Merupakan suatu prosedur aplikasi yang dibangun untuk menyelidiki *server* atau *port host* yang terbuka dengan cara ketik nmap dan *IP* dari *domain* OJS jinteks.

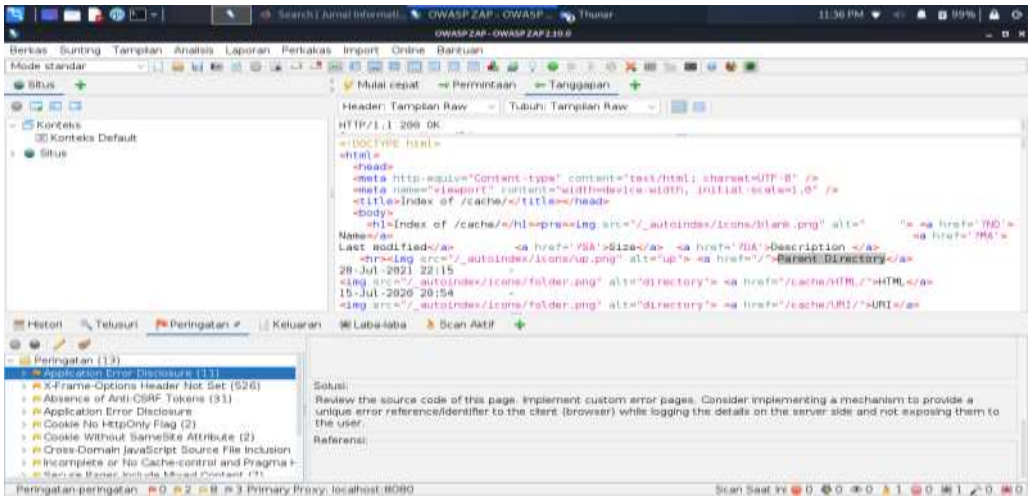


Gambar 3. Hasil scanning jaringan pada OJS Jinteks

Hasil pengujian yang telah dilakukan menggunakan NMAP didapatkan informasi penting mengenai *port-port* yang terbuka atau *open*. Adapun *Port* yang berstatus *open* pada OJS jinteks diantaranya *port* 21, 22, 25, 53, 80, 110, 143, 443, 465, 587, 993, 995, 3306, 5678. Dengan adanya celah ini *hacker* dengan mudah akan masuk dan melakukan penyerangan(Mokbal et al., 2019).

2. Vulnerability Scanning.

Pengujian mencari celah kerentanan pada OJS jinteks. Sebelum ditemukan kerentanan *cross site scripting* pada *owasp zap*, terlebih dahulu lakukan pengujian pada OJS jurnal jinteks. Adapun hasil pengujian secara. Manual pada. alamat *website* tersebut dengan memasukkan *script* pada tautan *web* tersebut adalah(Hakim et al., 2020):



Gambar 4. Hasil *Scanning tool Open Web Aplication Security Project* (OWASP)

Ada banyak *script* yang dapat digunakan untuk menguji serangan *Cross Site, Scripting, (*XSS) script tersebut. Dapat ditemukan pada *link github XSS*. dari beberapa *script* tersebut, dapat memilih sesuai kebutuhan yang digunakan. Salah satu *script* tersebut adalah

"><imgsrc=x onerror=prompt(document.domain)>.

Berdasarkan hasil sebelum dan sesudah pengujian kerentanan pada OJS jurnal jinteks menggunakan *tool* OWASP dengan *domain jurnal.uts.ac.id* tidak menampilkan adanya kerentanan *cross site scripting*. OWASP adalah suatu aplikasi yang digunakan untuk menemukan *vulnerabilities* pada suatu aplikasi *website.* menyediakan *scanner* secara automatis(Kascheev & Olenchikova, 2020)(Perumal & Kola Sujatha, 2021)(Lei et al., 2020).

PEMBAHASAN

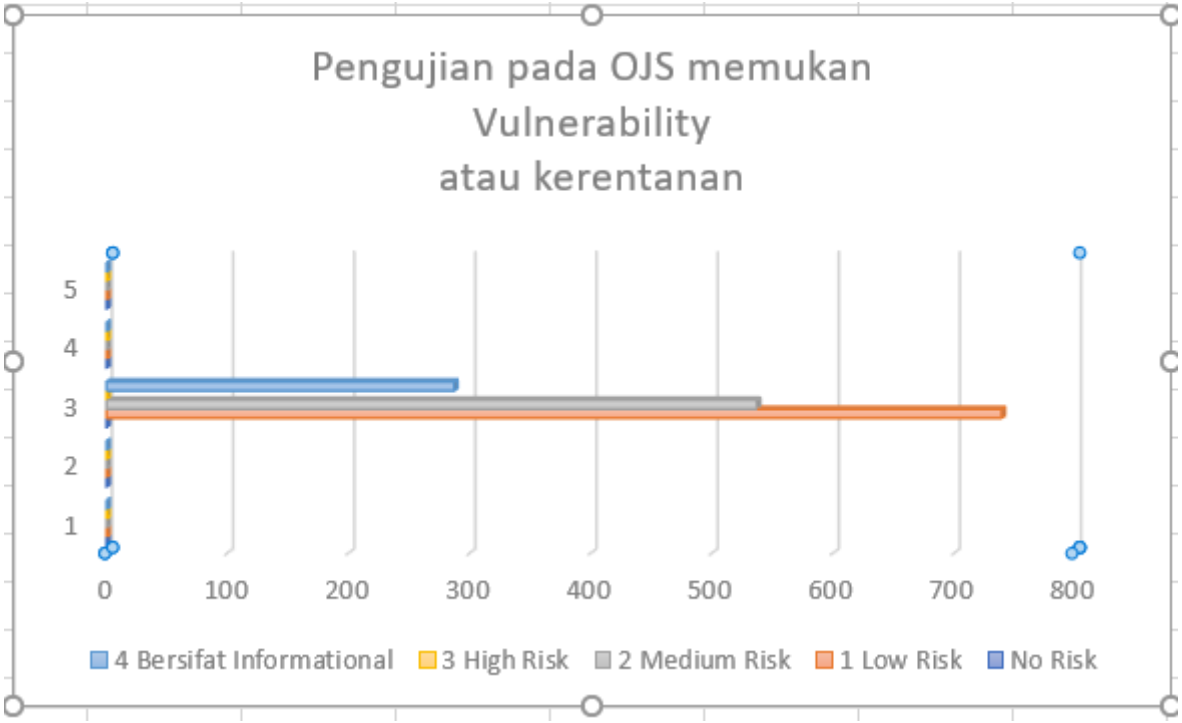
3. Result Analysis

Adapun hasil yang didapatkan dari pengujian tersebut mendeteksi 12 sub *file vulnerability* diantaranya:

Tabel 1. Pengujian yang telah dilakukan pada OJS jinteks mendeteksi 12 sub file vulnerability, medium, low, informational.(W et al., 2018)

No	***Alert***	***Risk***			**Keterangan**
		Low	*Medium*	*High*	
1	*Applic Ation error disclosure*		√		Untuk medium *risk* pada *website* jurnal jinteks ini berada pada tahap mengkhawatirkan untuk itu harus segera diperbaiki oleh *admin* pengelola *website* jurnal jinteks. kerentanan *medium* berjumlah 537.
2	*X-Frame-Options Header Not Set*		√		

3	*Absence of Anti-CSRF Tokens*	√		Sementara pada *Low Risk* masih berada pada keadaan kerusakan ringan. Yang Berjumlah 739.
4	*Cookie No Http flag*	√		
5	*Cookie without Same Site Attribute*	√		
6	*Cross-Domain Java Script Source File Inclusion*	√		
7	*Incomplete or No Cache-Control and Program HTTP Header Set*	√		
8	*Secure Pages Include Mixed Content*	√		
9	*X-Content-Type-Options Header Missing*	√		
10	*Information disclosure – sensitive information in URL*	*(Information)*		Untuk kerentanan informasi tambahan berjumlah 287.
11	*Information disclosure-suspicious comments*	*(Information)*		
12	*imestamp disclosure*	*(Information)*		



Gambar 5. Grafik data adanya kerentanan yang harus di waspadai

Data grafik diatas menunjukan adanya *Vulnerability* pada OJS Jinteks, antara lain: *Medium risk* 537 , *low risk* 739 dan bersifat informasi pendukung berjumlah 287. Dengan data diatas perlu adanya Tindakan lanjut oleh pengelola jurnal untuk memperbaiki sistem OJS nya(Habibi & Surantha, 2020)(Wang et al., 2019)(Singh et al., 2020)(Chen et al., 2021)(Zubarev & Skarga-Bandurova, 2019)(Xu et al., 2022).

KESIMPULAN

Adapun kelebihan dari *tool* OWASP adalah dapat melihat *source code* yang ditandai khusus oleh *tool* OWASP. *Tool* OWASP berhasil melakukan pengujian kerentanan sistem OJS jurnal jinteks. Pengujian yang dilakukan telah berhasil mengidentifikasi 3 tingkat kerentanan, yaitu *medium, low,* dan *informational*. Tingkat kerentanan diperoleh dari noktifikasi *alert* yang ditampilkan oleh *tool* OWASP. Pengujian pada *website* jurnal jinteks dengan *tool* OWASP diperoleh kerentanan *medium* 537, kerentanan *low* 739, dan kerentanan *informational* 287. total celah atau *vulnerability* yang ditemukan berjumlah 1563, analisis keamnan *web* mengnakan teknik *footprinting* dan *vulnerability scanning*. Hasil pengujian yang telah dilakukan berjalan dengan baik.

REFERENSI

Alazmi, S., & De Leon, D. C. (2022). A Systematic Literature Review on the Characteristics and Effectiveness of Web Application Vulnerability Scanners. *IEEE Access*, *10*, 33200–33219. https://doi.org/10.1109/ACCESS.2022.3161522

Chen, H. C., Nshimiyimana, A., Damarjati, C., & Chang, P. H. (2021). Detection and prevention of cross-site scripting attack with combined approaches. *2021 International Conference on Electronics, Information, and Communication, ICEIC 2021*, *500*. https://doi.org/10.1109/ICEIC51217.2021.9369796

Habibi, G., & Surantha, N. (2020). XSS attack detection with machine learning and n-gram methods. *Proceedings of 2020 International Conference on Information Management and Technology, ICIMTech 2020*, *August*, 516–520. https://doi.org/10.1109/ICIMTech50083.2020.9210946

Hakim, A. S., Cahyanto, T. A., & Azizah, H. (2020). Serangan cross-site scripting (XSS) berdasarkan base metric CVSS V.2. *Jurnal Smart Teknologi*, *2*(1).

Kascheev, S., & Olenchikova, T. (2020). The Detecting Cross-Site Scripting (XSS) Using Machine Learning Methods. *Proceedings - 2020 Global Smart Industry Conference, GloSIC 2020*, 265–270. https://doi.org/10.1109/GloSIC50886.2020.9267866

Kishimoto, K., Taniguchi, Y., & Iguchi, N. (2020). A Practical Exercise System Using Virtual Machines for Learning Cross-Site Scripting Countermeasures. *2020 IEEE International Conference on Consumer*

Electronics - Taiwan, ICCE-Taiwan 2020, 2020–2021. https://doi.org/10.1109/ICCE-Taiwan49838.2020.9258195

Lei, L., Chen, M., He, C., & Li, D. (2020). XSS Detection Technology Based on LSTM-Attention. *2020 5th International Conference on Control, Robotics and Cybernetics, CRC 2020*, 175–180. https://doi.org/10.1109/CRC51253.2020.9253484

Liu, M., Zhang, B., Chen, W., & Zhang, X. (2019). A Survey of Exploitation and Detection Methods of XSS Vulnerabilities. *IEEE Access*, *7*, 182004–182016. https://doi.org/10.1109/ACCESS.2019.2960449

Mokbal, F. M. M., Dan, W., Imran, A., Jiuchuan, L., Akhtar, F., & Xiaoxi, W. (2019). MLPXSS: An Integrated XSS-Based Attack Detection Scheme in Web Applications Using Multilayer Perceptron Technique. *IEEE Access*, *7*, 100567–100580. https://doi.org/10.1109/ACCESS.2019.2927417

Octavriana, T., Joni, K., & Ibadillah, A. F. (2021). Optimalisasi Jaringan Internet Dengan Load Balancing Pada High Traffic Network. *Jurnal Teknik Informatika*, *14*(1), 28–39. https://doi.org/10.15408/jti.v14i1.15018

Perumal, S., & Kola Sujatha, P. (2021). Stacking Ensemble-based XSS Attack Detection Strategy Using Classification Algorithms. *Proceedings of the 6th International Conference on Communication and Electronics Systems, ICCES 2021*, *vi*, 897–901. https://doi.org/10.1109/ICCES51350.2021.9489177

Riadi, I., Umar, R., & Lestari, T. (2020). Analisis Kerentanan Serangan Cross Site Scripting (XSS) pada Aplikasi Smart Payment Menggunakan Framework OWASP. *JISKA (Jurnal Informatika Sunan Kalijaga)*, *5*(3), 146–152. https://doi.org/10.14421/jiska.2020.53-02

Singh, M., Singh, P., & Kumar, P. (2020). An Analytical Study on Cross-Site Scripting. *2020 International Conference on Computer Science, Engineering and Applications, ICCSEA 2020*. https://doi.org/10.1109/ICCSEA49143.2020.9132894

Sivanesan, A. P., Mathur, A., & Javaid, A. Y. (2018). A Google Chromium Browser Extension for Detecting XSS Attack in HTML5 Based Websites. *IEEE International Conference on Electro Information Technology*, *2018-May*, 302–304. https://doi.org/10.1109/EIT.2018.8500284

Taha, T. A., & Karabatak, M. (2018). A proposed approach for preventing cross-site scripting. *6th International Symposium on Digital Forensic and Security, ISDFS 2018 - Proceeding*, *2018-Janua*, 1–4. https://doi.org/10.1109/ISDFS.2018.8355356

W, Y., Riadi, I., & Yudhana, A. (2018). Analisis Deteksi Vulnerability Pada Web Server Open Journal System Menggunakan OWASP Scanner. In *Jurnal Rekayasa Teknologi Informasi (JURTI)* (Vol. 2, Issue 1, p. 1). https://doi.org/10.30872/jurti.v2i1.1319

Wang, G., Xie, S., Zhang, X., Gao, J., Wei, F., Zhao, B., Wang, C., & Lv, S. (2019). An Effective Method to Safeguard Cyber Security by Preventing Malicious Data. *IEEE Access*, *7*, 166282–166291. https://doi.org/10.1109/ACCESS.2019.2951234

Xu, G., Xie, X., Huang, S., Zhang, J., Pan, L., Lou, W., & Liang, K. (2022). JSCSP: A Novel Policy-Based XSS Defense Mechanism for Browsers. *IEEE Transactions on Dependable and Secure Computing*, *19*(2), 862–878. https://doi.org/10.1109/TDSC.2020.3009472

Zubarev, D., & Skarga-Bandurova, I. (2019). Cross-Site Scripting for Graphic Data: Vulnerabilities and Prevention. *Conference Proceedings of 2019 10th International Conference on Dependable Systems, Services and Technologies, DESSERT 2019*, 154–160. https://doi.org/10.1109/DESSERT.2019.8770043