Penetration Testing untuk Menguji Sistem Keamanan Website Menggunakan OWASP ZAP

Hifzul Wathoni Maulana, Tomi Tri Sujaka, Muhammad Azwar, Husain, I Putu Hariyadi
Universitas Bumigora, Mataram, Indonesia

Correspondence : *e-mail*: 2101020041@universitasbumigora.ac.id

Abstrak

Di era digital yang semakin berkembang maka keamanan website merupakan aspek krusial dalam melindungi data dan menjaga integritas sistem. Penelitian ini bertujuan untuk menguji keamanan website terhadap serangan parameter injection menggunakan metode penetration testing. Pengujian dilakukan dengan mengacu pada standar OWASP Top 10 denan menggunakan alat OWASP Zed Attack Proxy. Proses pengujian dilakukan dalam empat tahap, yaitu perencanaan, pemindaian, eksploitasi, dan pelaporan. Eksploitasi dilakukan terhadap dua jenis serangan utama, yaitu SQL Injection dan Cross Site Scripting, menggunakan alat SQLMap dan Burp Suite. Hasil pengujian menunjukkan bahwa terdapat dua jenis kerentanan pada sistem, masing-masing memiliki tingkat risiko tingg dan medium. Langkah mitigasi dilakukan dengan mengimplementasikan teknik prepared statement, validasi dan sanitasi input, serta enkripsi data menggunakan algoritma Advanced Encryption Standard serta melakukan konfigurasi tambahan untuk mencegah akses yang tidak sah. Penelitian ini menghasilkan rekomendasi teknis yang dapat digunakan oleh pengembang untuk meningkatkan keamanan website dari ancaman cracker.

Kata kunci: *Penetration testing, parameter injection, SQL injection, Cross Site Scripting, OWASP ZAP.*

Abstract

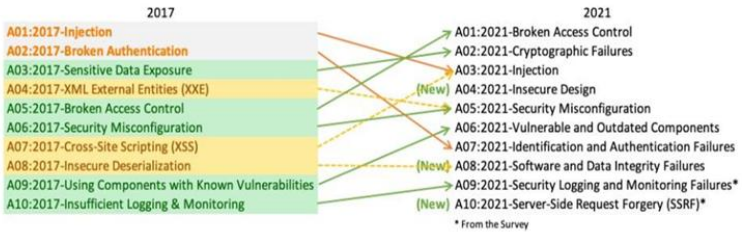
In the increasingly developing digital era, website security is a crucial aspect in protecting data and maintaining system integrity. This study aims to test website security against parameter injection attacks using the penetration testing method. Testing is carried out by referring to the OWASP Top 10 standard using the OWASP Zed Attack Proxy tool. The testing process is carried out in four stages, namely planning, scanning, exploitation, and reporting. Exploitation is carried out against two main types of attacks, namely SQL Injection and Cross Site Scripting, using the SQLMap and Burp Suite tools. The test results show that there are two types of vulnerabilities in the system, each with a high and medium risk level. Mitigation steps are taken by implementing prepared statement techniques, input validation and sanitation, and data encryption using the Advanced Encryption Standard algorithm and performing additional configurations to prevent unauthorized access. This study produces technical recommendations that can be used by developers to improve website security from cracker threats.

Keywords*: Penetration testing, parameter injection, SQL injection, Cross Site Scripting, OWASP ZAP.*

1. Pendahuluan

Di era digital yang semakin berkembang, penggunaan website terus meningkat di berbagai sektor seperti instansi, pendidikan, organisasi, hingga individu. Hal ini menjadikan aspek keamanan sebagai kebutuhan penting [1]. Seiring pertumbuhan pengguna internet, risiko serangan siber juga meningkat, termasuk pencurian data dan *deface* halaman website oleh pihak tidak bertanggung jawab [2].

Badan Siber dan Sandi Negara (BSSN) mencatat sebanyak 9.749 kasus serangan web *deface*, dengan 84% menyerang halaman utama dan 34% di antaranya menargetkan sektor perguruan tinggi [3]. Serangan yang sering terjadi adalah *SQL Injection* dan *Cross Site Scripting (XSS). SQL Injection* memanfaatkan celah input form *login*, seperti penggunaan queri admin or 1=1 untuk masuk tanpa otorisasi [4]. Sedangkan serangan *XSS* mengeksploitasi input tanpa validasi sehingga memungkinkan eksekusi skrip berbahaya [5]. Salah satu standar pengujian keamanan yang banyak digunakan adalah *OWASP Top 10*, yaitu daftar sepuluh besar kerentanan keamanan aplikasi web yang disusun oleh komunitas *Open Web Application Security Project (OWASP)* yang mengidentifikasi dan mengklasifikasikan sepuluh kerentanan keamanan paling kritis yang sering ditemukan dalam aplikasi web [6]. Daftar ini diperbarui secara berkala untuk mencerminkan tren terbaru dalam keamanan siber dan memberikan panduan bagi pengembang dan profesional keamanan untuk mengatasi risiko yang ada. [7].

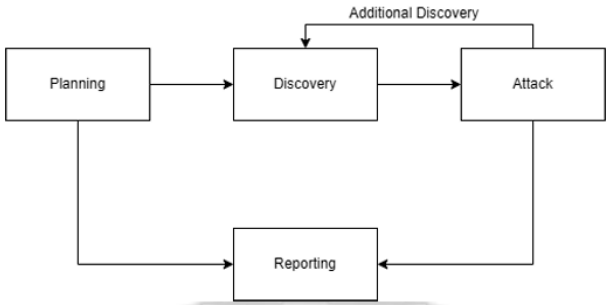


Gambar 1 OWASP TOP 10 List 2021

Gambar 1 di atas menunjukkan sepuluh jenis kerentanan dari *OWASP Top 10* yang diperbarui secara berkala, termasuk rilis pada tahun 2017 dan 2021 [8]. Dalam penelitian ini, acuan yang digunakan adalah daftar *OWASP Top 10* versi terbaru tahun 2021 yang mencakup *SQL Injection* dan *Cross-Site Scripting (XSS).* Pengujian sistem keamanan perlu dilakukan secara berkala. Salah satu alat yang digunakan dalam pengujian penetrasi adalah *OWASP Zed Attack Proxy (ZAP),* yang dapat mengidentifikasi berbagai kerentanan termasuk *SQL Injection* dan *Cross-Site Scripting (XSS)* [9].

2. Metode Penelitian

Dalam penelitian ini menggunakan metode *penetration testing* yang mencakup *planning*, *discovery, attack dan reporting* kemudian akan dilakukan mitigasi untuk mengatasi kerentan yang ditemukan [10].



Gambar 2 Metode *Penetration Testing*

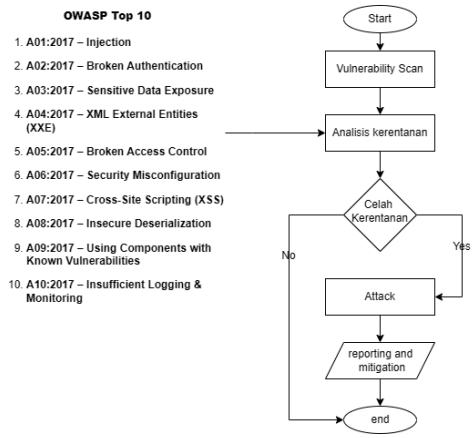
2.1 Planning and Discovery

Dalam penelitian ini website yang dipakai oleh peneliti yaitu website sekolah yaitu https://multisite.my.id yang berfungsi sebagai media informasi dan komunikasi. Website ini menyediakan fitur-fitur informasi sekolah dan form untuk komunikasi. Dalam penelitian ini website tersebut dijadikan objek untuk mengidentifikasi permasalahan yang akan diteliti, merumuskan pertanyaan penelitian, dan menentukan metode yang akan digunakan. Dengan demikian, penelitian menjadi terfokus dan sesuai dengan tujuan yang ditetapkan [11].

Table 1 Kebutuhan Penelitian

Kategori	Keterangan
Lenovo Thinkpad T460	Perangkat
Intel(R) Core(TM) i5-6300U CPU @ 2.40GHz 2.50 GHz	Processor
8 GB	RAM
256 GB	SSD
Windows 10 Pro	Sistem Operasi
Windows SubSystem for Linux (WSL) with Kali Linux	Sistem Pendukung
OWASP ZAP, SqlMap, BurpSuite	Alat untuk penetration testing

Pada tahap ini, peneliti hanya melakukan pengujian terhadap website milik sekolah, yaitu https://multisite.my.id/, sesuai dengan panduan *OWASP Top 10*. Adapun langkah-langkah yang dilakukan penulis dalam melaksanakan pengujian penetrasi akan dijelaskan melalui bagan alur pada Gambar berikut.



Gambar 3 *Flowchart Penetration Testing*

2.1.1 Vulnerability Scan

Pada penelitian ini, tools yang digunakan adalah *OWASP ZAP* yang bertujuan untuk mencari dan mengidentifikasi berbagai celah kerentanan yang terdapat pada sistem. Proses *Vulnerability scan* ini merupakan proses penting untuk menganailisis dan mengidentifikasi celah keamanan secara menyeluruh sehingga celah kerentanan yang ada dapat diketahui dan akan dilakukan sejauh mana kerentanan tersebut bisa di manfaatkan oleh *cracker*.

Table 2 Report Hasil Scanning

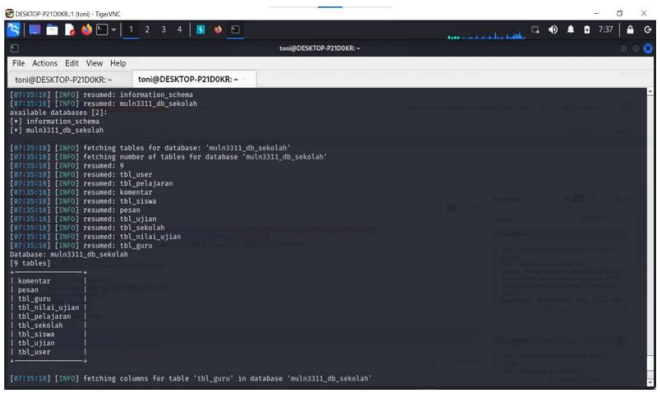
Alert	Risk Level
Cross-Site Scripting	High
SQL Injection - MySQL	High
Absence of Anti – CSRF Token	Medium
Content Security Policy (CSP) Header Not Set	Medium
Strict-Transport-Security Header Not Set	Medium
X-Content-Type-Options Header Missing	Medium
Vulnerable JS Library	Medium

Kerentanan yang memiliki risk level high merupakan masalah keamanan yang sangat berbahaya yang bisa dimanfaatkan oleh *cracker* untuk mengakses data sensitif atau merusak integritas pada website. Oleh karena itu peneliti akan melakukan eksploitasi terhadap kerentanan yang *risk level high* yaitu *SQL Injection* dan *Cross Site Scripting (XSS)* dan selanjutnya akan dilakukan mitigasi untuk menutup celah kerentanan tersebut.

2.2 Attack

2.2.1 Attacking SQL Injection – MySQL

Pengujian dilakukan dengan menjalankan *SQLMap* terhadap *URL* target menggunakan metode --dbs untuk mengekstraksi daftar nama database. *SQLMap* berhasil mendeteksi dan mengeksploitasi celah *SQL Injection* dapat dilihat pada gambar dibawah.

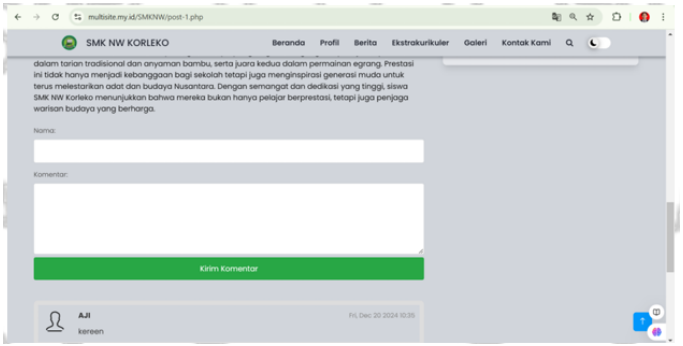


Gambar 4 Hasil Penyerangan *SQL Injection – MySQL*

Dari hasil eksploitasi, gambar diatas menunjukkan bahwa *SQLMap* berhasil mengeksploitasi parameter input yang rentan terhadap serangan *SQL Injection* dan memperoleh akses ke basis data bernama muln3111_db_sekolah. Setelah proses injeksi berhasil, *SQLMap* menampilkan struktur database tersebut yang terdiri dari beberapa tabel, antara lain tbl_user, tbl_guru, tbl_siswa, tbl_ujian, tbl_pelajaran, tbl_sekolah, komentar, dan pesan.

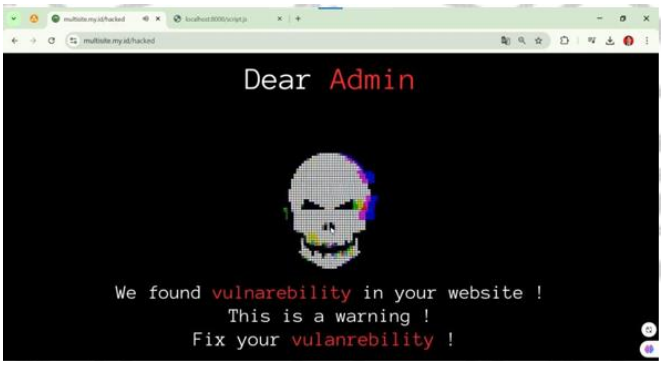
2.2.2 Attacking Cross-Site Scripting

Pengujian dilakukan dengan membuat script yang berisi file untuk memodifikasi halaman website. Script tersebut kemudian dihosting pada server lokal dan dipanggil menggunakan tag <script> yang disispkan kedalam input yang rentan.



Gambar 5 Tampilan UI Normal

Setelah halaman dimuat ulang, skrip yang disisipkan dijalankan secara otomatis oleh browser. Akibatnya, antarmuka website yang semula normal mengalami perubahan yang signifikan. Dapat dilihat pada gambar dibawah ini.



Gambar 6 Tampilan UI Setelah Disispkan Script

Hasil *deface* halaman website setelah *script XSS* dijalankan pada gambar 6 menunjukkan perubahan antarmuka, ini menunjukkan bahwa input pengguna tidak melalui proses validasi yang memadai, sehingga memungkinkan modifikasi visual halaman secara langsung.

3. Hasil dan Pembahasan

3.1 Reporting dan Mitigasi

Tahap pelaporan merupakan tahap pengujian yang bertujuan untuk mendokumentasikan seluruh temuan dan hasil pengujian, termasuk jenis kerentanan yang ditemukan, potensi serangan yang dapat terjadi, serta melakukan mitigasi untuk perbaikan sistem.

Table 3 Hasil *Reporting* dan *Mitigation*

OWASP TOP 10	Vulnerability	Risk Level	Mitigation
A03 - Injection	*SQL Injection - MySQL*	High	Menggunakan *prepared statement* untuk mencegah eksekusi *SQL* secara ilegal. Menambahkan enkripsi untuk

			data sensitif menggunakan algoritma *AES* untuk menjaga agar data yang berhasil dieksploitasi tetap aman.
A03 - Injection	*Cross-Site Scripting*	High	Melakukan *validasi input*, *sanitasi input* dengan *filter_input, strip_tags*, dan menggunakan *regex*.
A07 – Identification and Authentication Failures	*Absence of Anti – CSRF Token*	Medium	Mengaktifkan *CSRF protection* pada framework CodeIgniter melalui konfigurasi *config.php*. Gunakan token untuk validasi permintaan pengguna.
A05 – Security Misconfiguration	*Content Security Policy (CSP) Header Not Set*	Medium	Tambahkan header CSP seperti: *Content-Security-Policy*: default-src 'self'; untuk membatasi sumber konten eksternal dan mencegah serangan seperti *XSS*.
A05 – Security Misconfiguration	*Strict-Transport-Security Header Not Set*	Medium	Tambahkan header: *X-Content-Type-Options: nosniff* untuk mencegah *MIME type sniffing* yang berbahaya.
A05 – Security Misconfiguration	*X-Content-Type-Options Header Missing*	Medium	Tambahkan header: *Strict-Transport-Security: max-age=31536000; includeSubDomains* untuk memastikan semua komunikasi terjadi melalui *HTTPS*.
A06 – Vulnerable and Outdated Components	*Vulnerable JS Library*	Medium	Tambahkan atribut *integrity* dan *crossorigin*="anonymous" pada link *CDN*. Pastikan semua *library* pihak ketiga menggunakan versi terbaru dan aman. Serta Terapkan *CSP* dengan *script-src* yang spesifik hanya ke domain yang dipercaya. Validasi eksternal script secara manual atau menggunakan *proxy* keamanan.

4. Kesimpulan

Hasil dari penelitian ini menunjukkan bahwa terdapat berbagai kerentanan yang ditemukan pada sistem keamanan website yang diuji, terutama kerentanan dengan resiko *high* yang bisa mengakibatkan kebocoran data maupun merusak integritas website kerentanan itu adalah *SQL Injection* dan *Cross-Site Scripting.* Kerentanan ini berhasil dieksploitasi dengan menggunanakan *SqlMap*, *BurpSuite* dan *Script*, ini menunjukkan bahwa celah keamanan masih bisa dimanfaatkan oleh pihak yang tidak bertangung jawab. Hasil dari *vulnerability scan* juga menemukan beberapa konfigurasi keamanan yang belum diterapkan secara optimal, seperti tidak adanya *Absence of Anti – CSRF Token, Content Security Policy (CSP), Strict-Transport-Security dan X-Content-Type-Options*. Langkah mitigasi diterapkan untuk mengatasi kerentanan yang ditemukan dengan menerapkan prepared statement dan mengimplementasikan algoritma AES untuk mengamankan dan mencegah serangan *SQL Injection*, menerapkan validasi dan sanitasi input untuk mencegah serangan *Cross-Site Scripting* serta menerapkan konfigurasi tambahan juga dilakukan seperti mengaktifkan header keamanan dan pembatasan akses terhadap sumber daya external yang tidak sah.

Daftar Pustaka

[1] R. Al Ihsan and B. A. Sekti, “Pentingnya Keamanan Data Dalam Era Digital : Refleksi Terhadap Serangan Hacker Pada Pusat Data Nasional Indonesia,” pp. 2–6, 2023.

[2] E. Susanto, Lady Antira, K. Kevin, E. Stanzah, and A. A. Majid, “Manajemen Keamanan Cyber Di Era Digital,” *J. Bus. Entrep.*, vol. 11, no. 1, p. 23, 2023, doi: 10.46273/jobe.v11i1.365.

[3] Tenri, “Situs Web Perguruan Tinggi Terbanyak Alami Serangan Web Defacement,” *cyberthread*, 2021. https://cyberthreat.id/read/10636/Situs-Web-Perguruan-Tinggi-Terbanyak-Alami-

Serangan-Web-Defacement (accessed Jan. 11, 2025).

[4] H. Pranata, P. S. Ramadhan, and Tugiono, "Implementasi Pendeteksi Serangan SQL (Structured Query Language) Dengan Metode Randomization of Query," *J. CyberTech*, vol. 4, no. 5, pp. 1–9, 2021, [Online]. Available: www.cbronline.com.

[5] A. A. Chandra, A. Turmudi Zy, and A. Nugroho, "Penerapan Teknik Penetration Testing Terhadap Cross Site Scripting (Xss) Dalam Pengembangan Website," *Rabit J. Teknol. dan Sist. Inf. Univrab*, vol. 9, no. 2, pp. 262–270, 2024, doi: 10.36341/rabit.v9i2.4822.

[6] H. Hermanto and H. Haeruddin, "Peningkatan Sistem Keamanan Website Menggunakan Metode OWASP," *J. Ilmu Komput. dan Bisnis*, vol. 13, no. 1, pp. 94–104, 2022, doi: 10.47927/jikb.v13i1.277.

[7] Reza. Aditama; Edi. Negara, "Pemindai Kerentanan Terhadap Website Jago Masak Dengan Metode Pengujian Penetrasi OWASP ZAP," *J. Mantik*, vol. 6, no. 3, pp. 3406–3412, 2022.

[8] A. F. Hasibuan and D. Handoko, "Analisis Keretanan Website Dengan Aplikasi Owasp Zap," *J. Ilmu Komput. dan Sist. Inf.*, vol. 2, no. 2, pp. 257–270, 2023, [Online]. Available: https://jurnal.unity-academy.sch.id/index.php/jirsi/article/view/51.

[9] D. Ending Narhudin, B. Irawan, and A. Bahtiar, "Evaluasi Keamanan Website Menggunakan Metode Owasp: Penilaian Terhadap Serangan Injeksi Sql Dan Cross-Site Scripting (Xss)," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 8, no. 1, pp. 675–680, 2024, doi: 10.36040/jati.v8i1.8700.

[10] S. Hidayatulloh and D. Saptadiaji, "Penetration Testing pada Website Universitas ARS Menggunakan Open Web Application Security Project (OWASP)," *J. Algoritm.*, vol. 18, no. 1, pp. 77–86, 2021, doi: 10.33364/algoritma/v.18-1.827.

[11] A. Andriyadi, R. R. N. Fikri, and E. F. Saputri, "Evaluasi Sistem Informasi Perpustakaan Institut Informatika Darmajaya Dengan Whitebox Testing," *J. Innov. ...*, vol. 3471, no. 8, pp. 743–746, 2022.