Jutisi: Jurnal Ilmiah Teknik Informatika dan Sistem Informasi
Jl. Ahmad Yani, K.M. 33,5 - Kampus STMIK Banjarbaru
Loktabat – Banjarbaru (Tlp. 0511 4782881), e-mail: puslit.stmikbjb@gmail.com
e-ISSN: 2685-0893
p-ISSN: 2089-3787

Analisis Keamanan *Website* SMK Wongsorejo Gombong terhadap Serangan *SQL Injection* dengan PTES

Sidik Maulana Akhson[1*], Fahmi Fachri[2]
Teknik Informatika, Universitas Maarif Nahdlatul Ulama Kebumen, Kebumen, Indonesia
*e-mail *Corresponding Author: sidik20001@gmail.com*

Abstract

This study analyzes the security of the SMK Wongsorejo Gombong website against SQL Injection attacks using the Penetration Testing Execution Standard (PTES) method. The issue arose when several users experienced access difficulties and data loss on the school's website. By following the seven stages of PTES, the research identified vulnerabilities using OWASP ZAP 2.15.0 and SQLMap 1.8.11.11#dev. The findings revealed a high-risk SQL Injection vulnerability in the jawaban_id parameter. Exploitation successfully exposed the database structure, accessed user tables, and decrypted the admin password using brute force methods. This research contributes to the development of security testing procedures for educational information systems and provides improvement recommendations, including input validation, the use of PDO (PHP Data Objects) with parameterized queries, stored procedures, and escaping techniques.

Keywords: *Website Security; SQL Injection; Penetration Testing; OWASP ZAP; School Information System.*

Abstrak

Penelitian ini menganalisis keamanan *website* SMK Wongsorejo Gombong terhadap serangan *SQL Injection* menggunakan metode *Penetration Testing Execution Standard* (PTES). Permasalahan muncul ketika beberapa pengguna mengalami kesulitan akses dan hilangnya data pada *website* sekolah. Melalui tujuh tahapan PTES, penelitian mengidentifikasi kerentanan menggunakan OWASP ZAP 2.15.0 dan SQLMap 1.8.11.11#dev. Hasil penelitian menunjukkan adanya celah keamanan SQL *Injection* pada parameter jawaban_id dengan tingkat risiko tinggi. Eksploitasi berhasil mengungkap struktur *database*, mengakses tabel pengguna, dan mendekripsi *password* admin menggunakan metode *brute force.* Penelitian ini memberikan kontribusi dalam pengembangan prosedur pengujian keamanan untuk sistem informasi pendidikan dan menghasilkan rekomendasi perbaikan berupa penggunaan input validasi, PDO (*PHP Data Objects*) dengan *parameterized query, stored procedure* dan *escaping*.

Kata kunci: *Keamanan Website; SQL Injection; Penetration Testing; OWASP ZAP; Sistem Informasi Sekolah.*

1. Pendahuluan

Pertumbuhan pengguna internet secara global terus meningkat signifikan. Pada tahun 2022, jumlah pengguna internet mencapai 4,9 miliar orang atau 69% dari populasi dunia, dan diperkirakan akan meningkat menjadi 72,6% pada tahun 2025 seiring dengan semakin luasnya penggunaan perangkat pintar yang terhubung ke internet [1]. Peningkatan ini mendorong perkembangan layanan digital yang semakin kompleks, terutama di sektor pendidikan, di mana institusi pendidikan semakin mengandalkan sistem informasi berbasis web untuk mengelola data dan menyebarkan informasi secara efektif [2]. Peningkatan ini mengharuskan lembaga pendidikan untuk beradaptasi dengan perubahan teknologi guna meningkatkan efektivitas dan efisiensi dalam penyampaian informasi kepada seluruh pemangku kepentingan [3]. Perkembangan teknologi informasi yang pesat ini menuntut institusi pendidikan untuk tidak hanya fokus pada peningkatan fungsionalitas, tetapi juga mengantisipasi potensi ancaman siber yang dapat mengganggu operasional [4]. Salah satu metode serangan yang paling sering

dimanfaatkan adalah sql injection, yang memungkinkan peretas mengeksploitasi sistem database suatu website untuk memperoleh informasi sensitif secara tidak sah [5].

SMK Wongsorejo Gombong sebagai salah satu lembaga pendidikan yang telah mengadopsi *website* sebagai media utama untuk mendukung aktivitas administrasi dan komunikasi dengan seluruh pemangku kepentingan. Adaptasi terhadap perubahan teknologi ini bertujuan untuk meningkatkan efektivitas dan efisiensi penyampaian informasi [6]. Namun berdasarkan observasi awal, ditemukan beberapa indikasi permasalahan keamanan yang serius. Administrator *website* tidak dapat mengakses halaman admin, sementara beberapa pengguna juga mengalami kesulitan masuk ke halaman pengguna, meskipun kredensial telah dicadangkan. Selain itu, ditemukan hilangnya informasi pengunggah berita pada konten tertanggal 6 Mei 2024, sementara informasi serupa pada berita lain tetap tercatat dengan baik. Hal ini mengarah pada dugaan adanya eksploitasi celah keamanan dalam sistem, yang berpotensi disebabkan oleh serangan *sql injection*.

Menanggapi hal tersebut, diperlukan langkah analisis keamanan yang sistematis dan menyeluruh untuk mengidentifikasi serta mengevaluasi kemungkinan adanya celah pada sistem. Dalam penelitian ini digunakan pendekatan PTES (*Penetration Testing Execution Standard)* sebagai metode utama dalam menguji kerentanan sistem terhadap serangan siber, khususnya SQL Injection. Metode PTES dipilih karena, menurut penelitian yang dilakukan oleh Sunaringtyas & Prayoga, pendekatan ini terbukti memudahkan proses pengujian penetrasi. Selain itu, PTES juga dinilai efektif dalam meminimalkan potensi konflik antara penguji dan klien yang mungkin timbul akibat perbedaan pemahaman mengenai cakupan pengujian [7].

Berdasarkan permasalahan tersebut, penelitian ini bertujuan untuk menganalisis keamanan *website* sistem informasi SMK Wongsorejo Gombong, khususnya terhadap potensi serangan *sql Injection*, menggunakan pendekatan *Penetration Testing Execution Standard* (PTES). Dengan metode ini, penelitian akan mengevaluasi tingkat keamanan *website* terhadap ancaman siber serta memberikan rekomendasi perbaikan guna meningkatkan ketahanan sistem terhadap eksploitasi yang mungkin terjadi.

2. Tinjauan Pustaka

Beberapa penelitian terdahulu telah dilakukan oleh berbagai peneliti dalam mengevaluasi keamanan sistem informasi.

Penelitian terdahulu dengan judul "Analisis Pengujian Keamanan *Website* Pengelolaan Internet Desa Kragan Menggunakan Metode *Penetration Testing Execution Standard* (PTES)" yang ditulis oleh Burhani & Priyawati. Penelitian ini menggunakan metode *Penetration Testing Execution Standard* (PTES). Hasil penelitian menunjukkan terdapat 14 celah keamanan, seperti *Sql injection, Absence of Anti-CSRF Tokens*, dan *Missing Anti-clickjacking Header*, dengan tingkat risiko yang bervariasi. Meski demikian, *sql injection* tidak dapat dieksploitasi karena perlindungan dari *Secure Socket Layer* (SSL) dan *Web Application Firewall* (WAF) [8].

Penelitian terdahulu dengan judul "Analisis Kerentanan Keamanan Web Menggunakan Metode OWASP dan PTES di Web Pemerintahan Desa XYZ" yang ditulis oleh Fauzi. Penelitian ini menggunakan gabungan metode OWASP (*Open Web Application Security Project*) dan PTES (*Penetration Testing Execution Standard*) untuk mengidentifikasi dan menganalisis kerentanan keamanan web pada *website* Pemerintahan Desa XYZ. Hasil pengujian menunjukkan bahwa sebagian besar kerentanan berada pada kategori sedang hingga rendah, dengan beberapa kerentanan signifikan seperti *sql injection* yang memerlukan prioritas perbaikan [9].

Penelitian terdahulu dengan judul "Teknik Uji Penetrasi *Web Server* Menggunakan *Sql injection* dengan SQLMap di Kali Linux" oleh Hermawan. Penelitian ini menggunakan metode *penetration testing* dengan teknik *sql injection* untuk mengidentifikasi kerentanan pada *web server* menggunakan *tools* SQLMap di sistem operasi Kali Linux. Hasil penelitian menunjukkan bahwa serangan *sql injection* memungkinkan penyerang untuk mengeksploitasi *database* target, termasuk mengakses tabel, kolom, hingga data autentikasi seperti username dan password. Simulasi dilakukan pada *virtual machine* dengan dua komputer, di mana salah satu bertindak sebagai penyerang dan lainnya sebagai target. SQLMap secara otomatis mendeteksi dan mengeksploitasi kerentanan *sql injection* untuk menampilkan data penting dari *server* target [5].

Penelitian terdahulu dengan judul "Uji Penetrasi Injeksi sql terhadap Celah Keamanan *Database Website* menggunakan SQLmap" oleh Riyanti. Penelitian ini menggunakan *tools*

SQLmap di sistem operasi Kali Linux untuk mengidentifikasi dan mengeksploitasi kerentanan *sql injection* pada *database website*. Hasil penelitian menunjukkan bahwa SQLmap dapat mendeteksi berbagai kerentanan seperti tabel, kolom, dan data dalam *database* target. Penyerang memanfaatkan input yang tidak tervalidasi untuk menyisipkan kode sql berbahaya, sehingga memungkinkan manipulasi data. Selain itu, penelitian ini menekankan pentingnya validasi *input*, penggunaan *Web Application Firewall* (WAF), dan pemindaian keamanan berkala untuk memitigasi risiko serangan *sql injection* [10].

Penelitian ini berfokus pada SMK Wongsorejo Gombong sebagai objek kajian dan penggunaan *tools* terbaru, yang membedakannya dari penelitian sebelumnya. Pada penelitian ini, digunakan metode PTES (*Penetration Testing Execution Standard*) untuk menganalisis keamanan sistem informasi terhadap serangan *sql injection*, dengan dukungan *tools* terbaru dalam pengujian keamanan. Penggunaan teknologi uji terkini bertujuan untuk mengidentifikasi celah keamanan secara lebih akurat, mengungkap bagian sistem yang paling rentan terhadap eksploitasi *sql injection*, serta memberikan rekomendasi perbaikan yang relevan guna meningkatkan ketahanan sistem terhadap ancaman siber.

3. Metodologi

Pada tahap ini, peneliti mulai menjalankan proses pengujian dengan mengikuti metode *Penetration Testing Execution Standard* (PTES). Metode ini terdiri dari tujuh langkah yang akan dijalankan secara sistematis oleh peneliti, yakni *Pre-engagement Interaction, Intelligence Gathering, Threat Modeling, Vulnerability Analysis, Exploitation, Post Exploitation*, dan *Reporting* [11]. Berikut kerangka kerja dari PTES.

Gambar 1. Metode *Penetration Testing Execution Standard* (PTES)

Rangkaian aktivitas *pre-engagement* dimulai dengan melakukan analisis dan identifikasi terhadap *website* SMK Wongsorejo Gombong. Selanjutnya, peneliti mengajukan surat izin penelitian kepada SMK Wongsorejo Gombong. Setelah mendapat konfirmasi, dilakukan persiapan *tools* dan komponen pendukung yang diperlukan dalam pelaksanaan PTES terhadap *website* tersebut [12].

Pada fase *intelligence gathering*, dilakukan pengumpulan berbagai informasi penting yang akan digunakan dalam pelaksanaan PTES [13]. Informasi yang dibutuhkan meliputi informasi domain dan subdomain, detail *IP address*, kredensial email, serta konfigurasi DNS [14].

Selanjutnya, peneliti melakukan *threat modeling* dengan menyusun pendekatan pengujian berbasis model. Metode ini diimplementasikan untuk mempermudah proses identifikasi dan analisis kerentanan keamanan yang akan ditemukan selama pengujian sistem [15].

Pada tahap *vulnerability analysis*, peneliti mulai menganalisis informasi SMK Wongsorejo Gombong menggunakan alat OWASP ZAP. *Tools* ini dirancang untuk menganalisis berbagai kerentanan keamanan yang mungkin ada dalam sebuah *website* [16].

Selanjutnya tahap *exploitation* mencakup serangkaian percobaan penembusan sistem menggunakan metode *sql injection* untuk menyerang celah keamanan *website* SMK Wongsorejo Gombong [7]. peneliti menggunakan SQL MAP sebagai *tools* untuk melaksanakan serangan penetrasi [10].

Pada tahap *post exploitation*, peneliti melakukan evaluasi tingkat risiko kerentanan keamanan setelah serangkaian pengujian. Untuk menunjang analisis, disusun sebuah tabel yang menggambarkan penilaian risiko serangan yang telah teridentifikasi sebelumnya [17].

Sebagai tahap akhir dalam penelitian, peneliti melaksanakan proses *reporting* yang mencakup penyusunan dokumentasi berupa laporan detail dari seluruh rangkaian analisis dan pengujian yang telah dilaksanakan [18].

4. Hasil dan Pembahasan

Pada bagian ini membahas tahapan yang dilakukan dalam penelitian serta hasil yang diperoleh dari objek yang diuji. Selanjutnya, hasil tersebut dianalisis dan disusun dalam bentuk laporan, disertai dengan rekomendasi perbaikan sistem berdasarkan hasil pengujian untuk meningkatkan keamanannya di masa mendatang.

4.1 Hasil Uji Coba

1) *Pre-Engagement*

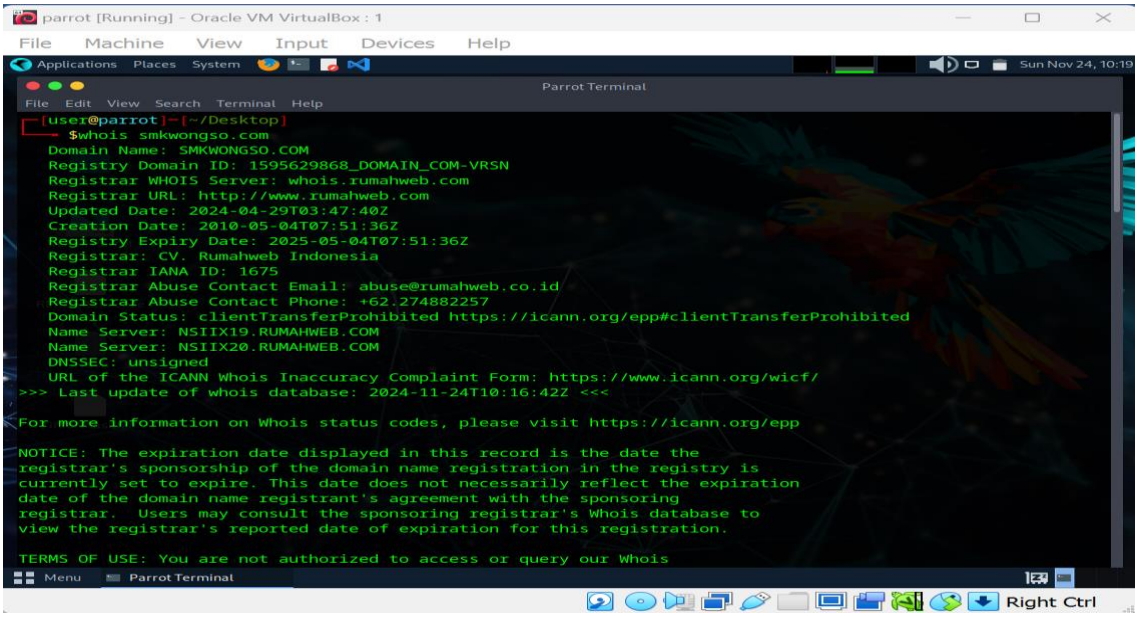
Pada tahapan ini peneliti melakukan *pre-engagement* untuk menganalisis keamanan *website* SMK Wongsorejo Gombong. Dalam tahap ini, peneliti terlebih dahulu meminta izin secara resmi kepada pihak sekolah untuk melakukan penelitian dan pengujian keamanan *website*. Setelah mendapatkan izin, peneliti melakukan identifikasi awal terhadap permasalahan kerentanan yang potensial pada *website* sekolah tersebut melalui observasi awal dan pengumpulan informasi yang tersedia secara publik.

Selanjutnya, peneliti melakukan koordinasi dan konfirmasi dengan tim IT SMK Wongsorejo Gombong mengenai ruang lingkup pengujian yang akan dilakukan. Dalam diskusi ini, disepakati batasan-batasan pengujian, metode yang akan digunakan, serta jadwal pelaksanaan analisis keamanan *website* untuk memastikan proses pengujian berjalan sesuai dengan prosedur dan tidak mengganggu operasional *website* sekolah.

2) *Intelligence Gathering*

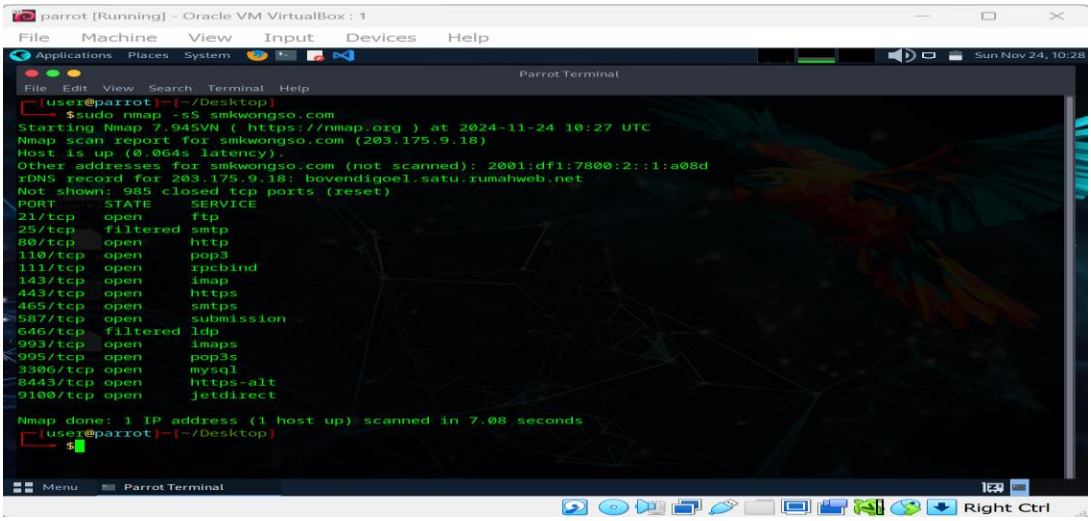
Pada tahap berikutnya, peneliti melaksanakan proses Intelligence Gathering atau pengumpulan informasi, dengan tujuan memperoleh sebanyak mungkin data penting yang nantinya akan dimanfaatkan dalam proses penetration testing selanjutnya. Dalam proses ini, peneliti mencari dan mengumpulkan berbagai informasi krusial seperti alamat domain beserta subdomain *website*, alamat email yang terkait, alamat IP *server*, informasi name *server*, serta *port-port* yang terbuka pada *website* SMK Wongsorejo Gombong.

Peneliti menggunakan beberapa *tools* pengujian untuk mendapatkan informasi detail mengenai domain target, peneliti menggunakan *tools whois* dengan menjalankan perintah "whois smkwongso.com".



Gambar 2. Hasil *Scan* dengan *Whois*

Tools ini berhasil mengumpulkan informasi penting seperti nama registrasi domain, tanggal registrasi domain, status domain, serta informasi name *server* yang digunakan. Informasi name *server* yang diperoleh dapat digunakan untuk mengidentifikasi penyedia hosting yang digunakan oleh *website* target.

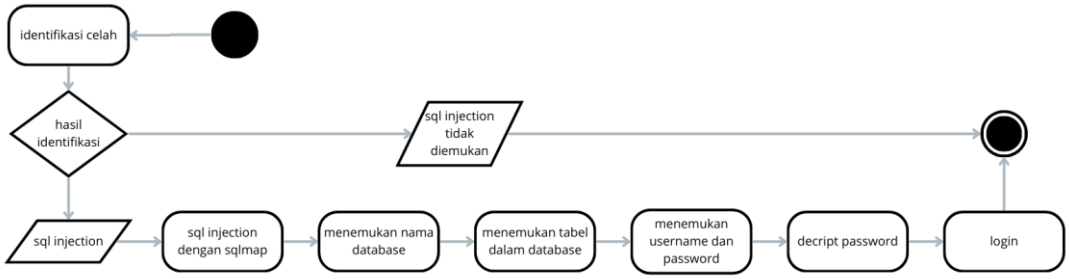


Gambar 3. Hasil *Scan* dengan tools NMAP

Selanjutnya untuk mengidentifikasi *port-port* yang terbuka pada *website* target, peneliti menggunakan tool NMAP dengan menjalankan perintah "sudo nmap -sS smkwongso.com". Hasil scanning menunjukkan terdapat beberapa *port*. Informasi *port* yang terbuka ini memberikan gambaran layanan apa saja yang berjalan pada *server* target dan dapat menjadi celah potensial yang perlu diperhatikan dari sisi keamanan.

3) *Threat Modeling*

Setelah melakukan tahap *Intelligence Gathering* dan mendapatkan berbagai informasi teknis tentang *website* tersebut, tahap selanjutnya adalah melakukan *threat modeling* untuk memudahkan peneliti memahami potensi kerentanan keamanan yang mungkin ditemukan selama pengujian dalam penelitian ini, Gambar 4 menampilkan sebuah model ancaman yang dirancang khusus untuk menggambarkan serangan *sql injection*. Pada tahap awal model ini, digambarkan situasi di mana penyerang mulai dengan mengirimkan perintah *sql injection* ke dalam sistem aplikasi web yang menjadi target.



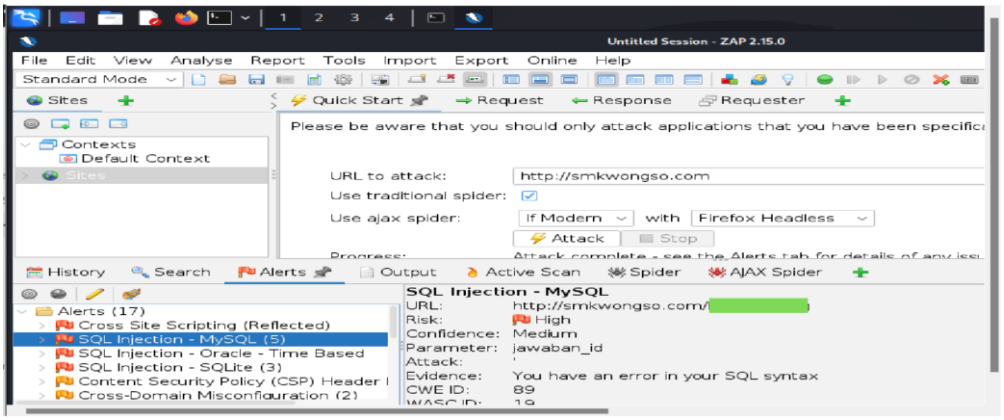
Gambar 4. Pemodelan Serangan

Kemudian, *database* aplikasi web yang telah menerima perintah *sql injection* memberikan akses kepada penyerang. Penyerang selanjutnya berhasil memperoleh *password* pengguna dari *database* tersebut. Setelah mendapatkan *password*, langkah berikutnya adalah mendekripsi *password* yang telah terenkripsi. Setelah proses dekripsi selesai, penyerang melakukan login ke *website* menggunakan *username* dan *password*. Akhirnya, penyerang dapat mengakses *server* aplikasi web dan melihat data pribadi yang tersimpan di dalamnya.

4) *Vulnerability Analysis*

Tahap berikutnya dalam penelitian adalah melakukan Analisis Kerentanan pada *website* system informasi SMK Wongsorejo Gombong. Untuk mengidentifikasi celah keamanan, peneliti menggunakan alat bantu OWASP ZAP versi terbaru 2.15.0. Pemilihan OWASP ZAP didasarkan pada beberapa keunggulan, yakni gratis, mudah digunakan, dan mampu menganalisis tingkat kerentanan secara komprehensif.

Melalui *tools* ini, peneliti dapat mendeteksi berbagai kerentanan keamanan *website* dengan klasifikasi risiko mulai dari tingkat rendah (*low risk*) hingga tingkat tinggi (*high risk*). Proses analisis kerentanan ini bertujuan untuk mengungkap potensi kelemahan yang mungkin terdapat pada *website* SMK Wongsorejo Gombong, sehingga dapat dilakukan tindakan pencegahan dan perbaikan yang tepat.

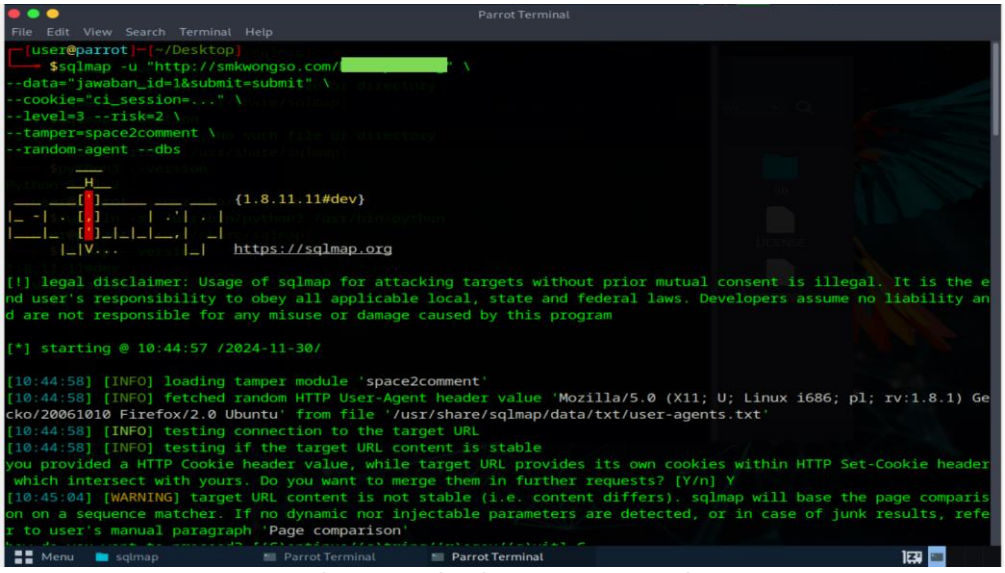


Gambar 5. Hasil Scan OWASP ZAP Versi 2.15.0

Tools ini memungkinkan peneliti untuk mendapatkan gambaran menyeluruh tentang kondisi keamanan *website*, termasuk mengidentifikasi berbagai celah potensial yang dapat membahayakan kerahasiaan, integritas, dan ketersediaan data.

5) *Exploitasion*

Exploitation merupakan tahapan yang dilakukan setelah ditemukannya celah keamanan, dengan tujuan untuk membuktikan sejauh mana kerentanan tersebut bisa dimanfaatkan. Pada penelitian ini, peneliti menguji kerentanan yang ada di dalam *website* smkwongso.com terhadap serangan *sql injection*. Pada tahap eksploitasi, peneliti menggunakan *tools* SQLMap untuk melakukan pentest terhadap *website* tersebut. SQLMap adalah *tools open-source* versi 1.8.11.11#dev yang dirancang untuk mengotomatisasi eksploitasi kerentanan SQL Injection pada aplikasi web. *Tools* ini digunakan untuk mengakses data sensitif yang ada di *database* target. Peneliti melakukan eksploitasi menyerang *database* dengan menjalankan perintah sqlmap -u "http://smkwongso.com/.../..." --data="jawaban_id=1&submit=submit" --cookie="ci_session=..." --level=3 --risk=2 --tamper=space2comment --random-agent --dbs.

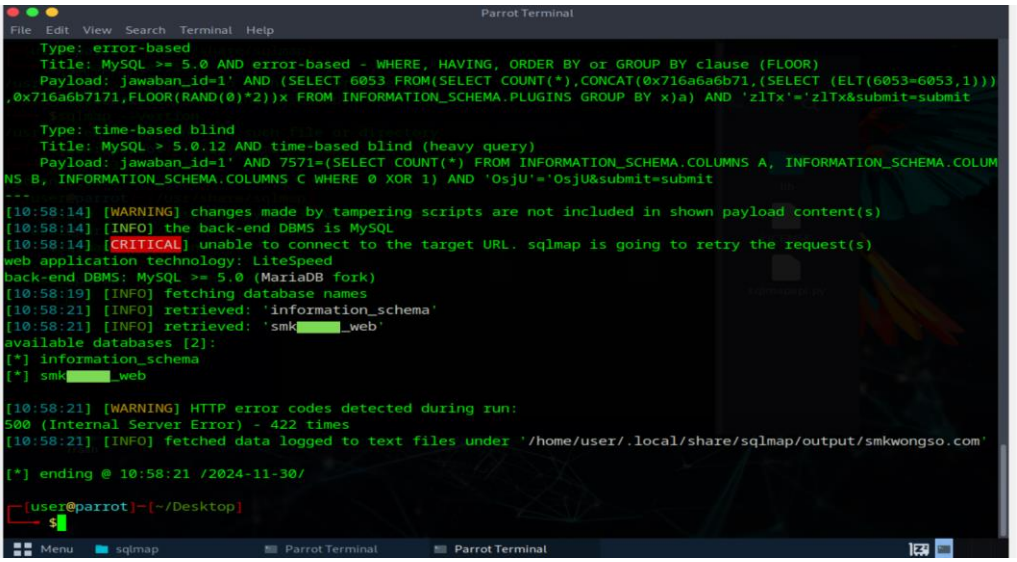


Gambar 6. Perintah Serangan *Database*

Perintah ini digunakan untuk mengidentifikasi *database* pada *website* target dengan memanfaatkan kerentanan *sql injection* yang terdeteksi sebelumnya pada parameter jawaban_id. Parameter tersebut ditemukan melalui tahap vulnerability analysis menggunakan OWASP ZAP. Dalam perintah ini, parameter --data="jawaban_id=1&submit=submit" digunakan untuk menguji kerentanan pada data yang dikirim melalui metode post. Selain itu, parameter --cookie="ci_session=..." menyisipkan *session cookie* yang valid, yaitu ci_session, yang digunakan oleh *framework CodeIgniter* untuk autentikasi. *Cookie* ini memungkinkan SQLMap mengakses area *website* yang memerlukan otorisasi.

Pengujian dilakukan dengan level=3 dan risk=2, yang memperluas cakupan eksplorasi SQLMap. Dengan level=3, SQLMap tidak hanya menguji parameter utama seperti *query string* dan data POST, tetapi juga mengevaluasi *header HTTP*, *referer*, dan *cookie*, sehingga meningkatkan kemungkinan deteksi kerentanan yang tersembunyi. Sementara itu, risk=2 memungkinkan SQLMap menggunakan *payload* yang lebih agresif dibandingkan dengan risk=1, termasuk eksploitasi berbasis waktu (*time-based blind SQL Injection*) dan teknik berbasis *union-based injection* untuk mengungkap lebih banyak data. Konfigurasi ini memberikan keseimbangan antara efektivitas deteksi dan keamanan sistem, memastikan eksploitasi dilakukan tanpa menyebabkan perubahan data yang merusak.

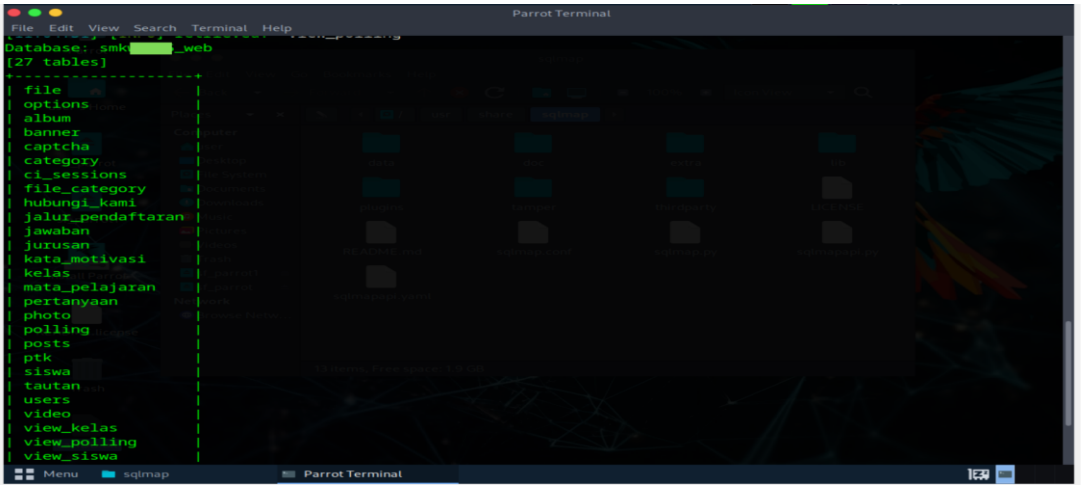
Peneliti juga menggunakan parameter --tamper=space2comment, yaitu sebuah *script* yang menggantikan spasi dalam *query sql* dengan komentar (/**/). Hal ini dilakukan untuk mengelabui sistem keamanan seperti *Web Application Firewall (WAF)* atau *Intrusion Detection System (IDS)* agar *payload sql injection* tidak terdeteksi. Parameter --random-agent digunakan untuk mengirimkan User-Agent secara acak ke URL target, menyerupai permintaan dari pengguna biasa, sehingga mengurangi kemungkinan terdeteksi oleh sistem keamanan berbasis pola. Terakhir, parameter --dbs memerintahkan SQLMap untuk menampilkan daftar *database* pada *server* target, memberikan wawasan tentang struktur *database* yang ada.



Gambar 7. Informasi database

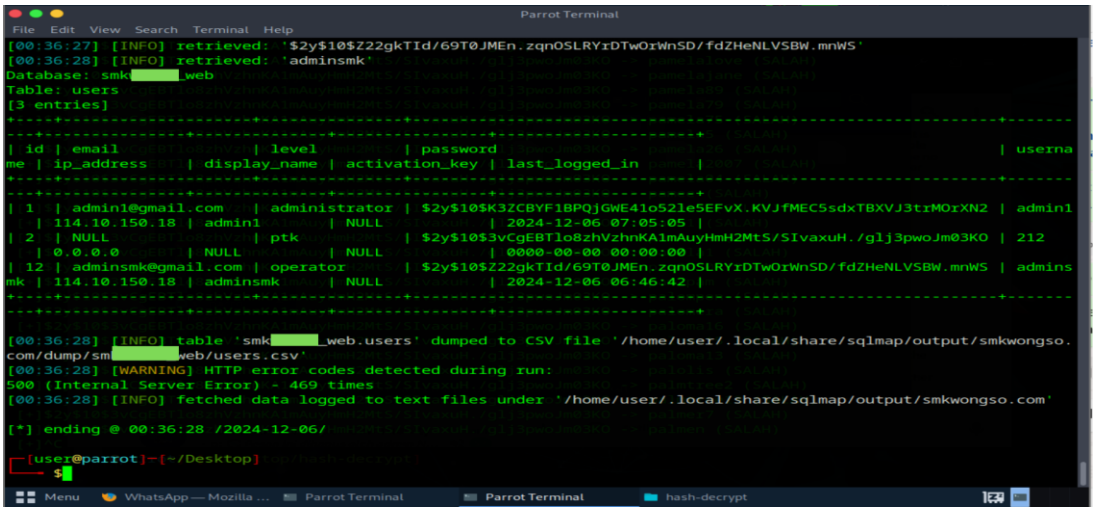
Gambar 7 adalah output dari perintah yang dijalankan, menunjukkan bahwa *database* yang digunakan pada *server* target adalah smkxyz_web. Informasi ini menunjukkan bahwa *server* berhasil merespons injeksi dan menampilkan nama *database* yang digunakan oleh aplikasi *website* tersebut.

Langkah selanjutnya peneliti mengakses tabel yang terdapat pada *database* tersebut. Peneliti menjalankan perintah sqlmap -u "http://smkwongso.com/.../..." \ --data="jawaban_id=1&submit=submit" \ --cookie="ci_session=..." \ --level=3 --risk=2 --tamper=space2comment --random-agent \ -D smkxyz_web --tables untuk menampilkan daftar tabel dalam *database* smkxyz_web. Perintah ini digunakan untuk mengakses *database* smkxyz_web dengan parameter -D untuk menentukan *database* yang akan diinjeksi, serta --tables untuk menampilkan daftar tabel.



Gambar 8. Informasi Table Database

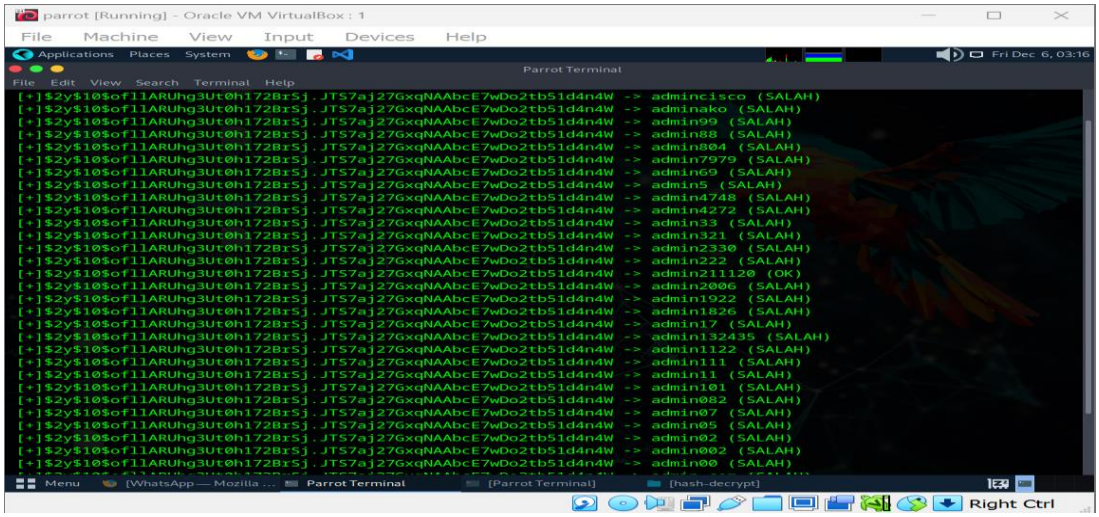
Gambar 8 menunjukkan daftar tabel yang berhasil diidentifikasi dalam basis data smkxyz_web. Informasi ini memberikan gambaran struktur data yang disimpan pada *server* target. Dengan daftar tabel yang telah diperoleh, peneliti dapat menentukan tabel mana yang berpotensi menyimpan data sensitif, seperti informasi pengguna dalam hal ini table *users*. Langkah eksplorasi selanjutnya adalah mengakses data pada table *users* tersebut untuk menguji sejauh mana kerentanan dapat dimanfaatkan. Tahapan berikutnya adalah mengakses isi kolom dalam tabel *users*. Peneliti menjalankan perintah perintah sqlmap -u "http://smkwongso.com/.../..." --data="jawaban_id=1&submit=submit" --cookie="ci_session=..." --level=3 --risk=2 --tamper=space2comment --random-agent -D smkxyz_web -T *users* --dump untuk mendapatkan data dari tabel *users*. Perintah ini menggunakan parameter -T untuk menentukan tabel *users*, serta --dump untuk menampilkan isi kolom *users*.



Gambar 9. Informasi table users

Gambar 9 menampilkan hasil dari perintah yang menunjukkan adanya tiga pengguna (*users*) yang tercatat dalam tabel *users*. Data yang berhasil diakses mencakup informasi sensitif seperti nama pengguna (*username*), alamat email, level pengguna, dan *hashed password*. Setelah mengidentifikasi isi kolom dalam tabel *users* pada basis data smkxyz_web, tabel ini menjadi fokus utama dalam tahap pengujian penetrasi. Hal ini disebabkan oleh pentingnya data yang terkandung di dalamnya, seperti nama pengguna dan kata sandi yang digunakan untuk mengakses situs web SMK Wongsorejo Gombong. Kata sandi yang tercatat dalam tabel *users* masih dalam bentuk terenkripsi, sebagaimana ditampilkan pada gambar 9. Untuk mengungkap kata sandi yang terenkripsi, peneliti menggunakan *tools* IndoExploit dan file kamus rockyou

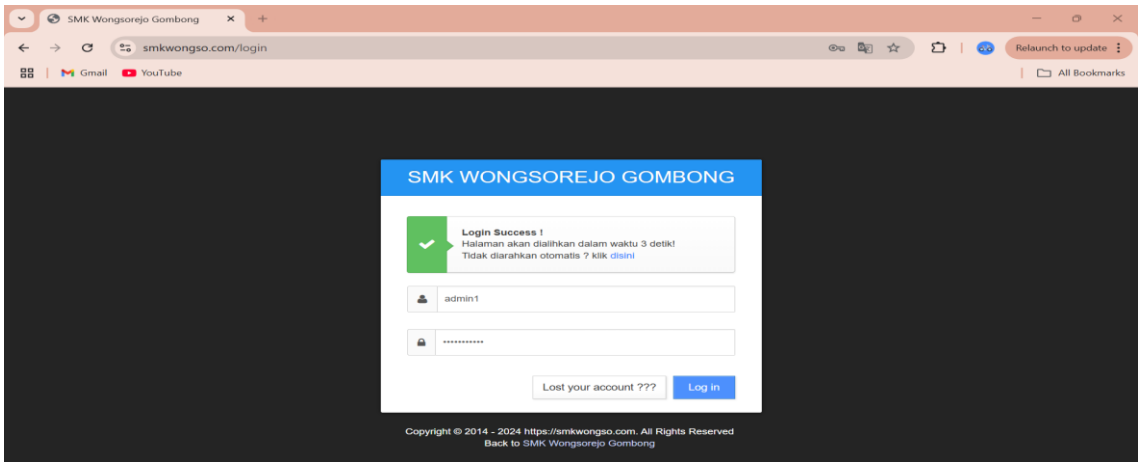
untuk melakukan proses dekripsi dengan metode *brute force*. Hasil dari proses dekripsi kata sandi untuk akun admin1 menggunakan metode ini ditunjukkan pada gambar berikutnya.



Gambar 10. Hasil Decript Password

Gambar 10 menampilkan hasil dari proses *brute force* yang dilakukan untuk mendekripsi kata sandi. Proses ini berhasil ditandai dengan indikator "OK" pada kata sandi admin211120. Hal ini menunjukkan bahwa alat bantu berhasil menemukan kata sandi terenkripsi yang sesuai dengan data yang ada di tabel *users*.

Setelah itu peneliti mencoba masuk kedalam *website* SMK Wongsorejo Gombong dengan *username*: admin1 dan *password*: admin211120. Berikut hasil dari melakukan sebuah exploitasi *website* SMK Wongsorejo Gombong.



Gambar 11. Tampilan berhasil login

Tools SQLMap berhasil melakukan eksploitasi terhadap *database website* smkwongso.com dan memberikan akses terhadap data yang ada di tabel *users*. Informasi ini dapat dimanfaatkan lebih lanjut untuk memperkuat sistem keamanan *website* agar kerentanan serupa tidak terjadi lagi.

6) *Post Exploitation*

Tahapan selanjutnya yang peneliti lakukan adalah melakukan *post exploitation*, yaitu tahapan yang dilakukan setelah mengeksploitasi kerentanan yang ada pada target. Pada tahap ini, dilakukan penilaian tingkat risiko terhadap sistem yang memiliki celah keamanan setelah dilakukan pengujian pada tahapan sebelumnya. Penilaian ini bertujuan untuk mengukur dampak dari serangan yang berhasil dilakukan terhadap *database website* smkwongso.com. Peneliti melakukan analisis terhadap potensi ancaman yang dapat terjadi akibat adanya

kerentanan *sql injection* dan menyusun tabel penilaian untuk melihat risiko serangan berdasarkan hasil eksploitasi.

Table 1. Tingkat Risiko

Attack	Risk	*Tools*
Sql Injection	high	Sqlmap & Owasp zap

Berdasarkan hasil pengujian di atas, ditemukan celah keamanan dengan tingkat risiko *high* pada *website* smkwongso.com berupa serangan *sql injection*. Serangan ini memiliki potensi besar untuk mengakses data sensitif, termasuk informasi pengguna dan admin yang tersimpan di *database*. Hasil eksploitasi menunjukkan bahwa serangan *sql injection* berhasil mengungkap struktur *database*, tabel, serta isi kolom yang berisi data penting, seperti *hash password*. Tingkat risiko dinilai tinggi karena *sql injection* memungkinkan pengambilalihan kontrol penuh terhadap *database*, sehingga membuka peluang terjadinya pencurian data penting.

7) *Reporting*

Tahapan terakhir dalam metode PTES adalah membuat laporan atau *reporting*, yang mencakup seluruh hasil analisis dan eksploitasi kerentanan pada sistem target. Dalam tahap ini, peneliti mendokumentasikan setiap proses penting yang dilakukan selama pengujian terhadap *website* smkwongso.com, termasuk hasil eksploitasi menggunakan *tools* seperti OWASP ZAP dan SQLMap. Laporan ini berfungsi sebagai referensi untuk memahami potensi ancaman dan langkah mitigasi yang diperlukan untuk memperbaiki kerentanan yang ditemukan.

Hasil eksploitasi menunjukkan bahwa terdapat beberapa kerentanan pada *website*, salah satunya adalah *sql injection*, yang ditemukan pada parameter jawaban_id melalui metode *post* pada halaman polling. Berdasarkan analisis dari laporan OWASP ZAP, kerentanan ini memiliki tingkat risiko tinggi, dengan potensi besar untuk mendapatkan akses tidak sah ke *database* dan memodifikasi data penting. Berikut adalah hasil eksploitasi yang dirangkum dalam tabel:

Table 2. Hasil Pentes

Attack	Status	*Tools*
Sql Injection	Successful	Sqlmap & Owasp zap

Sql injection pada *database* MySQL berhasil dieksploitasi dengan SQLMap menggunakan payload khusus dan parameter tertentu, seperti penggunaan --tamper=space2comment untuk menghindari deteksi WAF. Serangan ini memungkinkan akses terhadap tabel-tabel dalam *database*, seperti tabel *users*, yang berisi informasi sensitif, termasuk *username* dan *password* yang telah di-hash. Selain *sql injection*, OWASP ZAP mengidentifikasi beberapa kelemahan lain, termasuk *Missing Content Security Policy (CSP) Header*, yang meningkatkan risiko terhadap serangan *Cross-Site Scripting (XSS*). Pada kerentanan XSS, serangan dilakukan dengan menyisipkan skrip JavaScript pada *input*, sehingga dapat dijalankan di browser pengguna tanpa disadari. Berdasarkan hasil tersebut, peneliti merekomendasikan langkah-langkah perbaikan untuk meningkatkan keamanan *website* smkwongso.com.

4.2 Rekomendasi dan Pembahasan

Tabel 3 berikut adalah rekomendasi perbaikan yang telah dirangkum:

Table 3. Hasil Rekomendasi Perbaikan

Attack	*Recommendation*
Sql Injection	Menggunakan PDO (*PHP Data Objects*) dengan parameterized queries untuk memisahkan perintah SQL dan input pengguna. Melakukan validasi *input* secara ketat untuk memastikan hanya data yang sesuai format yang diterima.

Attack	*Recommendation*
	Mengimplementasikan *stored procedure* untuk memperkuat kontrol logika pemrosesan database di sisi *server*. Menerapkan *escaping* pada setiap input pengguna

Berdasarkan penelitian Naomi Augusta, rekomendasi untuk mencegah serangan SQL Injection adalah penerapan teknik pertahanan berlapis, seperti penggunaan PDO (*PHP Data Objects*) dengan *parameterized query* untuk memastikan pemisahan antara perintah SQL dan input pengguna. Teknik ini terbukti efektif karena mampu menghentikan eksploitasi yang sebelumnya berhasil dilakukan menggunakan SQLMap, dengan hasil menunjukkan penurunan eksploitasi hingga 100%. Selain itu, validasi input secara ketat juga disarankan untuk memastikan hanya data yang sesuai format yang diproses oleh sistem. Penggunaan stored procedure turut direkomendasikan untuk membatasi manipulasi *query* dari luar, serta penerapan *escaping* pada input pengguna yang ditampilkan kembali, sebagai bentuk perlindungan tambahan terhadap karakter berbahaya yang dapat dimanfaatkan dalam serangan injeksi [19].

Penelitian ini berkontribusi dengan mengembangkan temuan-temuan dari penelitian sebelumnya yang juga menggunakan metode *Penetration Testing Execution Standard* (PTES) untuk menguji kerentanannya terhadap SQL Injection. Penelitian sebelumya seperti yang dilakukan oleh Burhani & Priyawati [8] juga menggunakan PTES dalam menganalisis celah keamanan terhadap serangan SQL *Injection* pada aplikasi web, namun kurang memberikan detail tentang bagaimana eksploitasi celah tersebut dapat dilakukan secara lebih mendalam dengan menggunakan alat terbaru. Penelitian ini memperkenalkan alat pengujian versi terbaru dari penelitian sebelumnya , seperti OWASP ZAP versi 15 dan SQLMap versi *1.8.11.11*, yang memungkinkan peneliti untuk mengeksploitasi kerentanannya secara lebih efisien.

Dengan demikian, penelitian ini tidak hanya memperkuat hasil penelitian sebelumnya, tetapi juga memberikan pendekatan yang lebih lengkap untuk mengatasi masalah SQL Injection, serta menawarkan solusi yang dapat diterapkan untuk meningkatkan keamanan system.

5. Simpulan

Berdasarkan analisis keamanan *website* SMK Wongsorejo Gombong menggunakan metode *Penetration Testing Execution Standard (PTES),* dapat disimpulkan bahwa terdapat kerentanan keamanan yang serius pada *website* tersebut. Pengujian yang dilakukan melalui tujuh tahap *PTES (Pre-engagement Interaction, Intelligence Gathering, Threat Modeling, Vulnerability Analysis, Exploitation, Post Exploitation, dan Reporting)* berhasil mengidentifikasi adanya kerentanan *sql injection* dengan tingkat risiko tinggi.

Serangan *sql injection* pada parameter jawaban_id pada halaman xyz memungkinkan penyerang untuk mengakses basis data sistem informasi sekolah dan memperoleh informasi sensitif. Keberhasilan eksploitasi menggunakan SQLMap dan OWASP ZAP membuktikan bahwa *website* tersebut memiliki celah keamanan yang dapat disalahgunakan untuk mendapatkan akses tidak sah ke data pengguna, termasuk *username* dan *password* admin.

Rekomendasi peningkatan keamanan website meliputi penerapan PDO (*PHP Data Objects*) dengan parameterized queries untuk mencegah injeksi perintah SQL secara langsung, validasi input yang ketat untuk membatasi jenis data yang dapat diterima sistem, penggunaan stored procedure untuk memperkuat kontrol logika pemrosesan database di sisi server, serta penerapan *escaping* pada input yang ditampilkan ulang atau digunakan kembali dalam *query*. Implementasi keempat langkah ini menjadi penting dalam membatasi potensi eksploitasi SQL Injection dan meningkatkan ketahanan sistem informasi sekolah terhadap ancaman siber.

Daftar Referensi

[1] “Internet Users by Country 2024,” world population review. Accessed: Nov. 04, 2024. [Online]. Available: https://worldpopulationreview.com/country-rankings/internet-users-by-country

[2] N. Putri *et al.*, “Inovasi Pemanfaatan Teknologi Informasi Dalam Meningkatkan Efisiensi Manajemen Pendidikan Di Mis 05 Darussalam,” *Ar-Risalah Media Keislam. Pendidik. dan Huk. Islam*, vol. 22, no. 1, pp. 33–50, 2024, doi: 10.69552/ar-risalah.v22i1.2372.

[3] J. Santoso, & P. Selwen. "Penerapan Strategi Kepemimpinan Transformasional Dalam Meningkatkan Kinerja Organisasi Pendidikan". *Jurnal Ilmiah Kanderang Tingang*, vol. 14, no. 2, pp. 400-409, 2023.
[4] A. F. P. Dinarto, "Analisis Keamanan Aplikasi Website Menggunakan Metode Penetration Testing Berdasarkan Framework ISSAF Pada Perusahaan Daerah XYZ," *Innov. J. Soc. Sci. Res.*, vol. 4, pp. 4536–4549, 2024.
[5] R. Hermawan, "Teknik Uji Penetrasi Web Server Menggunakan SQL Injection dengan SQLmap di Kalilinux," *STRING (Satuan Tulisan Ris. dan Inov. Teknol.*, vol. 6, no. 2, pp. 210–216, 2021, doi: 10.30998/string.v6i2.11477.
[6] A. Gozali, "Layanan Bimbingan Dan Konseling Berbasis Teknologi Informasi Pada Masa PSBB (Pembatasan Sosial Berskala Besar)," *Coution J. Couns. Educ.*, vol. 1, no. 2, pp. 36–49, 2020, doi: 10.47453/coution.v1i2.117.
[7] S. U. Sunaringtyas and D. S. Prayoga, "Implementasi Penetration Testing Execution Standard Untuk Uji Penetrasi Pada Layanan Single Sign-On," *Edu Komputika J.*, vol. 8, no. 1, pp. 48–56, 2021, doi: 10.15294/edukomputika.v8i1.47179.
[8] L. F. Burhani and D. Priyawati, "Analisis Pengujian Keamanan Website Pengelolaan Internet Desa Kragan Menggunakan Metode Penetration Testing Execution Standard (Ptes)," *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 9, no. 1, pp. 307–319, 2024, doi: 10.29100/jipi.v9i1.4455.
[9] R. M. Fauzi, R. Hermawan, D. R. Adhy, and S. Maesaroh, "Analisis Kerentanan Keamanan Web Menggunakan Metode Owasp Dan Ptes Di Web Pemerintahan Desa Xyz," *Power Elektron. J. Orang Elektro*, vol. 13, no. 2, pp. 225–231, 2024, doi: 10.30591/polektro.v13i2.6711.
[10] A. Riyanti, B. M. Rahmanto, D. R. Hardianto, R. D. A. Yuristiawan, and A. Setiawan, "Uji Penetrasi Injeksi SQL terhadap Celah Keamanan Database Website menggunakan SQLmap," *J. Internet Softw. Eng.*, vol. 1, no. 4, pp. 1–9, 2024, doi: 10.47134/pjise.v1i4.2623.
[11] P. Prasetyo, Djumhadi, and W. N. Alimyaningtias, "Analisis Perbandingan Metode Ptes Dan Issaf Sebagai Uji Keamanan Router Di Zurich Hotel Balikpapan," *Forbis J. Forensic Bus. Inf. Syst.*, vol. 1, no. 1, pp. 8–13, 2024.
[12] Satria Galang Saputra, B. Parga Zen, and Abdurahman, "Analisis Keamanan Jaringan Wireless menggunakan Metode Penetration Testing Execution Standard (PTES)," *J. Sist. Inf. Galuh*, vol. 1, no. 2, pp. 43–51, 2023, doi: 10.25157/jsig.v1i2.3152.
[13] H. Herman, I. Riadi, Y. Kurniawan, and I. A. Rafiq, "Analisis Keamanan Website Menggunakan Information System Security Asessment Framework(ISSAF)," *J. Teknol. Inform. dan Komput.*, vol. 9, no. 1, pp. 126–136, 2023, doi: 10.37012/jtik.v9i1.1439.
[14] S. Andriyani, M. F. Sidiq, and B. P. Zen, "Analisis Celah Keamanan Pada Website Dengan Menggunakan Metode Penetration Testing Dan Framework Issaf Pada Website SMK Al-Kautsar," *J. Inform. Inf. Technol.*, vol. 8798, pp. 1–13, 2023.
[15] R. N. Dasmen, R. Rasmila, T. L. Widodo, K. Kundari, and M. T. Farizky, "Pengujian Penetrasi Pada Website Elearning2.Binadarma.Ac.Id Dengan Metode Ptes (Penetration Testing Execution Standard)," *J. Komput. dan Inform.*, vol. 11, no. 1, pp. 91–95, 2023, doi: 10.35508/jicon.v11i1.9809.
[16] D. Hariyadi and F. E. Nastiti, "Analisis Keamanan Sistem Informasi Menggunakan Sudomy dan OWASP ZAP di Universitas Duta Bangsa Surakarta," *J. Komtika (Komputasi dan Inform.*, vol. 5, no. 1, pp. 35–42, 2021, doi: 10.31603/komtika.v5i1.5134.
[17] F. A. Al Zulfi and D. F. Suyatno, "Pengujian Fungsionalitas dan Celah Keamanan Website Kampoeng Sinaoe Menggunakan Equivalence Partition, Boundary Value Analysis, Fuzzing, dan Penetration Testing," *J. Emerg. Inf. Syst. Bus. Intell.*, vol. 4, no. 3, pp. 139–146, 2023.
[18] Y. A. Pohan, "Meningkatkan Keamanan Webserver Aplikasi Pelaporan Pajak Daerah Menggunakan Metode Penetration Testing Execution Standar," *J. Sistim Inf. dan Teknol.*, vol. 3, pp. 1–6, 2021, doi: 10.37034/jsisfotek.v3i1.36.
[19] N. Augusta, A. I. Hadiana, and F. R. Umbara, "Sistem Keamanan Website Dengan Multi Metode Untuk Mencegah SQL Injection," *In Seminar Nasional Penelitian* (SEMNAS CORISINDO 2024), pp. 315–320, 2024.