Jurnal Impresi Indonesia (JII)
p-ISSN: 2828-1284 e-ISSN: 2810-062x
Vol. 5. No. 5. 2026

Analisis Kerentanan Aplikasi Berbasis Web Menggunakan *Framework OWASP WSTG*, Top 10 dan *Risk Rating*

Ilham Kusdi[1*], Dwi Asih Haryanti[2], Hustinawaty[3]
Universitas Gunadarma, Indonesia
Email: ilhamkusdi@gmail.com[*]

Keywords:
Web Application Security; OWASP; WSTG; OWASP Top 10; OWASP Risk Rating.

Abstract
Web-based application security is becoming increasingly important as the use of the internet and information technology in various aspects of life increases. This study aims to analyze the security of web-based applications using a combination of several Open Web Application Security Project (OWASP) frameworks, namely OWASP Web Security Testing Guide (WSTG) version 4.2, OWASP Top 10:2021, and OWASP Risk Rating Methodology. Data was collected through penetration testing using OWASP WSTG. The results showed that the analyzed web applications had vulnerabilities in several OWASP Top 10 categories, such as A07 - Identification and Authentication Failures, A05 - Security Misconfiguration, A03 - Injection, and A01 - Broken Access Control. Of the 26 gaps that were successfully identified, they were then grouped based on their risk level, namely seven Critical gaps, six High gaps, and 13 Medium gaps. This study shows that OWASP WSTG version 4.2 can be used to effectively identify vulnerabilities in web-based applications, both technical vulnerabilities and application business logic. OWASP Top 10:2021 can help in classifying and prioritizing the most common types of vulnerabilities. The OWASP Risk Rating Methodology can provide a quantitative approach in assessing the level of risk

Kata Kunci:
Keamanan Aplikasi Web; OWASP; WSTG; OWASP Top 10; OWASP Risk Rating.

Abstrak
Keamanan aplikasi berbasis web menjadi semakin penting seiring dengan meningkatnya penggunaan internet dan teknologi informasi dalam berbagai aspek kehidupan. Penelitian ini bertujuan untuk menganalisis keamanan aplikasi berbasis web menggunakan kombinasi beberapa framework Open Web Application Security Project (OWASP), yaitu OWASP Web Security Testing Guide (WSTG) versi 4.2, OWASP Top 10:2021, dan OWASP Risk Rating Methodology. Data dikumpulkan melalui pengujian penetrasi menggunakan OWASP WSTG. Hasil penelitian menunjukkan bahwa aplikasi web yang dianalisis memiliki kerentanan pada beberapa kategori OWASP Top 10, seperti A07 - Identification and Authentication Failures, A05 - Security Misconfiguration, A03 - Injection, dan A01 - Broken Access Control. Dari 26 celah yang berhasil diidenfifikasi kemudian dikelompokkan berdasarkan tingkat risikonya yaitu tujuh celah Critical, enam celah High, dan 13 celah Medium. Penelitian ini menunjukkan bahwa OWASP WSTG versi 4.2 dapat digunakan untuk mengidentifikasi kerentanan pada aplikasi berbasis web baik kerentanan teknis ataupun logika bisnis aplikasi secara efektif. OWASP Top 10:2021 dapat membantu dalam mengklasifikasikan dan memprioritaskan jenis kerentanan yang paling umum. OWASP Risk Rating Methodology dapat memberikan pendekatan kuantitatif dalam menilai tingkat risiko.

PENDAHULUAN

Dalam era digital yang semakin maju, internet dan aplikasi berbasis web telah menjadi bagian dari kehidupan sehari-hari. Namun, seiring dengan semakin meningkatnya penggunaan aplikasi web, ancaman terhadap keamanan informasi juga semakin meningkat (Khera et al., 2019; Paramarta et al., 2021; Stiawan et al., 2021; Thion & Coulondre, 2022). Salah satu tantangan utama dalam pengelolaan aplikasi web adalah memastikan setiap potensi celah keamanan terdeteksi dan ditangani sebelum dieksploitasi oleh pihak yang tidak bertanggung jawab (Armando & Rosalina, 2024; Dharmawangsa et al., 2023; Kamal et al., 2024; OWASP, 2026a). Banyak organisasi masih menghadapi kendala dalam hal sumber daya, pengetahuan, dan prosedur standar untuk melakukan pengujian keamanan secara berkala dan sistematis (OWASP, 2026b, 2026c; Priambodo et al., 2023; Setiawan et al., 2023).

Open Web Application Security Project (OWASP) adalah organisasi nirlaba yang berfokus pada peningkatan keamanan perangkat lunak terutama yang berbasis web (Umar et al., 2024; Widyaningrum et al., 2024; Yusuf et al., 2025). OWASP menyediakan berbagai alat dan framework yang dirancang untuk membantu pengembang dan profesional keamanan dalam mengidentifikasi dan mengatasi kerentanan keamanan dalam aplikasi web (Albahar, 2017; Deepa & Thilagam, 2016; Hydara et al., 2015). Beberapa framework utama yang disediakan oleh OWASP adalah OWASP *Web Security Testing Guide* (WSTG), OWASP *Top 10*, dan OWASP *Risk Rating Methodology*.

Urgensi penelitian ini menjadi sangat penting mengingat maraknya serangan siber yang menargetkan institusi pendidikan di Indonesia. Data dari Pusat Respons Insiden Keamanan Siber Nasional (CIDC) tahun 2024 mencatat peningkatan serangan pada sistem informasi akademik sebesar 45% dibandingkan tahun sebelumnya. Penelitian (Rafeli et al., 2022) pada website XYZ dan (Yamin et al., 2022) pada aplikasi SISAKTI Udayana juga mengonfirmasi tingginya risiko kerentanan pada sistem akademik. Tanpa pengujian keamanan yang sistematis, data sensitif mahasiswa dan dosen berisiko tinggi untuk bocor atau disalahgunakan.

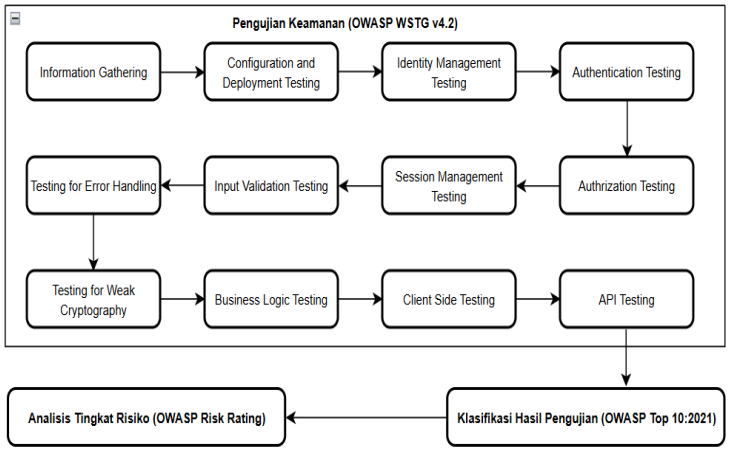
Sejumlah penelitian terdahulu telah berupaya memetakan solusi terhadap permasalahan ini menggunakan kerangka kerja *Open Web Application Security Project* (OWASP). Penelitian ini bertujuan untuk menganalisis kerentanan aplikasi berbasis web dengan menggunakan kombinasi framework OWASP WSTG v4.2, OWASP Top 10:2021 dan OWASP *Risk Rating Methodology*. Hasil dari penelitian ini diharapkan dapat memberikan kontribusi nyata dalam pengembangan strategi pengamanan aplikasi web yang lebih efektif dan adaptif terhadap ancaman siber yang terus berubah di era teknologi digital. Tujuan penelitian ini adalah menganalisis kerentanan aplikasi berbasis web menggunakan kombinasi framework OWASP, dengan manfaat teoritis memperkaya kajian keamanan siber dan manfaat praktis memberikan rekomendasi mitigasi bagi pengelola sistem informasi akademik.

METODE PENELITIAN

Tahap penelitian ini disusun secara sistematis untuk memastikan bahwa proses analisis kerentanan aplikasi web dapat dilakukan secara terstruktur, terukur, dan sesuai

dengan kaidah ilmiah. Gambar 1 menunjukkan tahap penelitian yang dilakukan disini sebagai berikut:

1. Pengujian Keamanan (OWASP WSTG v4.2)
2. Klasifikasi Hasil Pengujian (OWASP Top 10:2021)
3. Analisis Tingkat Risiko (OWASP *Risk Rating Methodology*)



Gambar 1. Tahap Penelitian

Pengujian Keamanan dengan OWASP WSTG v4.2

Pengujian keamanan dilakukan menggunakan OWASP WSTG v4.2 [4] yang mencakup 12 skenario dengan total 97 butir pengujian sebagai berikut:

1. *Information Gathering* adalah tahap awal dalam proses pengujian keamanan aplikasi web. Tujuannya adalah untuk mengumpulkan sebanyak mungkin informasi mengenai aplikasi target sebelum melanjutkan ke tahap pengujian yang lebih detail. Hal ini cukup penting karena dapat membantu memberikan pemahaman mengenai struktur, teknologi yang digunakan, serta potensi kelemahan dari aplikasi tersebut. Skenario ini terdari dari 10 butir pengujian.
2. *Configuration and Deployment Management Testing* merupakan bagian penting dari pengujian keamanan aplikasi web yang berfokus pada pengujian konfigurasi sistem dan proses deployment untuk memastikan bahwa tidak ada kelemahan yang dapat dimanfaatkan oleh penyerang. Skenario ini terdari dari 11 butir pengujian.
3. *Identity Management Testing* merupakan bagian penting dari pengujian keamanan aplikasi web yang berfokus pada pengelolaan identitas pengguna. Tujuannya adalah untuk memastikan bahwa sistem mengelola identitas pengguna dengan aman dan sesuai dengan prinsip-prinsip keamanan. *Identity Management* mencakup seluruh proses yang berkaitan dengan pembuatan akun pengguna, pengelolaan peran dan hak akses, *provisioning* dan *deprovisioning* akun, validasi identitas pengguna. Skenario ini terdari dari 5 butir pengujian.
4. *Authentication Testing* merupakan serangkaian pengujian yang berfokus pada mekanisme autentikasi dalam aplikasi web yang bertujuan untuk memastikan sistem autentikasi aman dari penyalahgunaan, tahan terhadap serangan *brute force*, *credential stuffing*, dan *session*

hijacking, menerapkan *best practice* dalam manajemen sesi dan pengelolaan identitas pengguna. Skenario ini terdari dari 10 butir pengujian.

5. *Authorization Testing* merupakan serangkaian pengujian yang bertujuan untuk memastikan bahwa mekanisme otorisasi dalam aplikasi web berfungsi dengan benar dan tidak dapat dilewati oleh pengguna yang tidak berwenang. *Authorization* adalah proses pengecekan hak akses yang dilakukan setelah pengguna berhasil melewati tahap autentikasi. Skenario ini terdari dari 4 butir pengujian.
6. *Session Management Testing* merupakan serangkaian pengujian yang bertujuan untuk mengevaluasi keamanan pengelolaan sesi dalam aplikasi web. Tujuan utamaya adalah untuk memastikan bahwa sesi tidak mudah ditebak atau dicuri, sesi tidak bisa digunakan kembali setelah logout, sesi memiliki timeout yang sesuai, sesi tidak bisa diambil alih (*session hijacking*) atau dipalsukan (*session fixation*). Skenario ini terdari dari 9 butir pengujian.
7. *Input Validation Testing* merupakan serangkaian pengujian yang bertujuan untuk mengevaluasi bagaimana aplikasi web memeriksa, memfilter, dan memvalidasi input yang diterima dari pengguna atau sumber luar sebelum diproses lebih lanjut. Pengujian ini berfokus pada cara aplikasi memproses input, apakah validasi dilakukan dengan baik, baik di sisi klien maupun server, serta bagaimana aplikasi merespons input yang tidak valid, berbahaya, atau tidak terduga. Validasi yang lemah atau tidak ada bisa mengakibatkan serangan seperti *SQL Injection*, *Cross-Site Scripting* (XSS), *Command Injection*, *Buffer Overflow*, dan lain-lain. Skenario ini terdari dari 19 butir pengujian.
8. *Testing For Error Handling* merupakan serangkaian pengujian yang dilakukan untuk mengevaluasi bagaimana aplikasi web menangani kesalahan (*errors*) yang terjadi selama proses operasionalnya. Prioritas utama adalah memastikan aplikasi menampilkan pesan kesalahan yang aman, hanya memberikan informasi yang relevan kepada pengguna yang berwenang, serta tidak mengungkapkan data sensitif yang dapat disalahgunakan oleh pihak tidak bertanggung jawab. Skenario ini terdari dari 2 butir pengujian.
9. *Testing For Weak Cryptography* merupakan serangkaian pengujian yang bertujuan untuk mengidentifikasi penggunaan algoritma, protokol, atau konfigurasi kriptografi yang lemah dalam aplikasi web. *Weak Cryptography* mengacu pada penggunaan algoritma, protokol, atau implementasi kriptografi yang rentan terhadap serangan karena kelemahan desain, panjang kunci yang tidak memadai, atau penerapan yang salah. Skenario ini terdari dari 4 butir pengujian.
10. *Business logic Testing* merupakan serangkaian pengujian yang dilakukan untuk memastikan bahwa logika bisnis sebuah aplikasi berjalan sesuai dengan aturan dan kebijakan yang telah ditetapkan, serta tidak dapat disalahgunakan oleh penyerang. Pengujian ini tidak menitikberatkan pada kerentanan teknis seperti injeksi atau XSS, melainkan pada kesalahan dalam alur proses, aturan bisnis, dan validasi yang berpotensi menimbulkan kerugian atau penyalahgunaan. Skenario ini terdari dari 9 butir pengujian.
11. *Client Side Testing* adalah serangkaian pengujian yang fokus pada aplikasi yang berjalan di sisi klien, seperti web browser, untuk memastikan bahwa data yang dikirimkan dari dan diterima oleh aplikasi web aman, dan bahwa aplikasi tidak rentan terhadap serangan yang dapat dilakukan melalui sisi klien. Serangan di sisi klien dapat mempengaruhi autentikasi, otorisasi, validasi data, serta memanipulasi elemen antarmuka

pengguna dan data yang dikirim ke server. Meskipun pengujian umumnya difokuskan pada sisi server, sisi klien tetap menjadi area rawan serangan karena sebagian besar interaksi pengguna berlangsung melalui browser. Skenario ini terdari dari 13 butir pengujian.

12. *API Testing* merupakan serangkaian pengujian yang dilakukan untuk mengevaluasi keamanan *Application Programming Interface* (API) yang digunakan oleh aplikasi web atau sistem. API berfungsi sebagai antarmuka yang memungkinkan komunikasi dan pertukaran data antara aplikasi, layanan, atau perangkat lunak yang berbeda. Skenario ini hanya terdari dari 1 butir pengujian.

Klasifikasi Hasil Pengujian Berdasarkan OWASP Top 10

Pada tahap ini dilakukan klasifikasi temuan yang telah divalidasi berdasarkan OWASP Top 10:2021 [5] sebagai berikut:

1. *Broken Access Control* (A01:2021) adalah kelemahan keamanan yang terjadi ketika sistem gagal membatasi akses pengguna sesuai dengan izin yang telah ditentukan. Situasi ini dapat menyebabkan data terbuka, diubah, atau dihapus secara ilegal, serta memungkinkan pengguna menjalankan fungsi bisnis di luar batas yang sah.
2. *Cryptographic Failures* (A02:2021) kelemahan keamanan yang timbul ketika sistem gagal melindungi data sensitif secara efektif menggunakan teknik kriptografi. Kondisi ini berpotensi menyebabkan kebocoran atau pencurian data yang sangat penting.
3. *Injection* (A03:2021) adalah kelemahan keamanan yang terjadi ketika data yang tidak terpercaya dimasukkan ke dalam interpreter sebagai bagian dari perintah atau kueri. Kondisi ini memungkinkan pelaksanaan perintah yang tidak diinginkan oleh pihak penyerang.
4. *Insecure Design* (A04:2021) adalah kerentanan keamanan yang muncul akibat desain atau arsitektur yang kurang efektif atau tidak ada sama sekali. Ini berbeda dengan kelemahan implementasi, karena desain yang tidak aman tidak dapat diperbaiki hanya dengan implementasi yang sempurna.
5. *Security Misconfiguration* (A05:2021) adalah kelemahan keamanan yang terjadi ketika sistem tidak dikonfigurasi dengan benar atau tidak memiliki pengaturan keamanan yang memadai. Ini dapat menyebabkan aplikasi rentan terhadap berbagai jenis serangan.
6. *Vulnerable and Outdated Components* (A06:2021) adalah kelemahan keamanan yang terjadi ketika sistem menggunakan komponen yang rentan atau sudah usang. Ini dapat menyebabkan aplikasi rentan terhadap berbagai jenis serangan.
7. *Identification and Authentication Failures* (A07:2021) adalah kelemahan keamanan yang terjadi ketika sistem gagal mengkonfirmasi identitas pengguna, mengotentikasi, atau mengelola sesidengan benar. Hal ini dapat menyebabkan berbagai serangan terkait otentikasi.
8. *Software and Data Integrity Failures* (A08:2021) adalah kelemahan keamanan yang terjadi Ketika sistem gagal melindungi integritas perangkat lunak dan data. Ini dapat menyebabkan aplikasi rentan terhadap berbagai jenis serangan.
9. *Security Logging and Monitoring Failures* (A09:2021) adalah kerentanan yang disebabkan karena sistem tidak melakukan pencatatan maupun mekaukan pemantauan aktivitas keamanan pada aplikasi. Hal ini dapat menyebabkan ketidakmampuan untuk mendeteksi, mengeskalasi, dan merespons pelanggaran keamanan secara efektif.

10. *Server-Side Request Forgery* (A10:2021) adalah kelemahan keamanan yang terjadi ketika aplikasi web mengambil sumber daya jarak jauh tanpa memvalidasi URL yang diberikan oleh pengguna. Ini memungkinkan penyerang untuk memaksa aplikasi mengirim permintaan yang dibuat ke tujuan yang tidak diharapkan, bahkan ketika dilindungi oleh firewall, VPN, atau daftar kontrol akses jaringan.

Analisis Tingkat Risiko Menggunakan OWASP *Risk Rating Methodology*

Analisis risiko dengan OWASP *Risk Rating Methodology* memiliki tahapan berikut:

1. *Identifying a Risk* adalah mengidentifikasi risiko keamanan yang ditemukan selama pengujian [6]. Risiko diambil dari daftar temuan kerentanan yang telah diverifikasi pada tahap sebelumnya.
2. *Factors for Estimating Likelihood* adalah menilai peluang terjadinya risiko berdasarkan aspek seperti tingkat keahlian penyerang, motivasi, dan kesempatan [6].
3. *Factors for Estimating Impact* adalah menilai dampak dari risiko berdasarkan faktor-faktor seperti kerugian finansial, kerusakan reputasi, dan pelanggaran privasi [6].
4. *Determining Severity of the Risk* adalah menghitung tingkat risiko berdasarkan kombinasi kemungkinan dan dampak [6].

Likelihood and Impact Levels	
0 to <3	LOW
3 to <6	MEDIUM
6 to 9	HIGH

Gambar 2. *Likelihood and Impact Levels*

Gambar 2 menunjukkan tingkat dari kemungkinan terjadi (*likelihood*) dan dan dampak (*impact*) dari risiko, dimana untuk nilai 0 sampai dengan 2 dapat dikategorikan sebagai rendah (*LOW*), nilai 3 sampai dengan 5 dapat dikategorikan sebagai sedang (*MEDIUM*), sedangkan untuk nilai 6 sampai dengan 9 dapat dikategorikan sebagai tinggi (*HIGH*).

5. *Deciding What to Fix* adalah memprioritaskan risiko berdasarkan tingkat risiko untuk menentukan tindakan mitigasi yang diperlukan [6].

Overall Risk Severity				
Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood			

Gambar 3. *Overall Risk Severity*

Gambar 3 menunjukkan tingkat risiko berdasarkan kombinasi tingkat kemungkinan dan dampaknya yang secara garis besar dapat dikelompokkan menjadi *Note*, *Low*, *Medium*, *High* dan *Critical*.

HASIL DAN PEMBAHASAN

Hasil pengujian website Sistem Informasi Akademik (SIAKAD) yang dipilih menjadi objek pengujian dengan menggunakan OWASP WSTG versi 4.2, dari 12 skenario, 97 total butir pengujian yang diujikan mendapatkan hasil 79 butir pengujian dapat dilakukan, dan 18 butir pengujian dilewati.

Hasil Pengujian Berdasarkan OWASP WSTG v4.2

Dari pengujian tersebut berhasil diidentifikasi 26 kerentanan sebagai berikut:

1. *Application Error Disclosure* adalah kerentanan yang terjadi ketika aplikasi menampilkan pesan kesalahan yang terlalu detail kepada pengguna.
2. *Arbitrary File Upload* adalah kerentanan yang memungkinkan pengguna mengunggah file tanpa validasi yang memadai terhadap jenis, ukuran, atau isi file.
3. *Broken Access Control* adalah kerentanan yang terjadi ketika mekanisme pembatasan hak akses tidak diterapkan atau dikonfigurasi dengan benar, sehingga pengguna dapat mengakses data atau fungsi yang seharusnya tidak diizinkan.
4. *Brute Force Login* adalah kerentanan yang terjadi ketika sistem tidak membatasi jumlah percobaan login, sehingga penyerang dapat mencoba berbagai kombinasi nama pengguna dan kata sandi secara berulang hingga berhasil masuk.
5. *Business Logic Data Validation Bypass* adalah kerentanan yang terjadi ketika validasi data pada proses bisnis tidak dilakukan dengan benar atau dapat dilewati, sehingga penyerang dapat mengirimkan input yang tidak sesuai aturan bisnis untuk mendapatkan keuntungan atau merusak sistem.
6. *Cross Site Request Forgery* adalah kerentanan yang memungkinkan penyerang memaksa pengguna yang telah terautentikasi untuk melakukan aksi yang tidak diinginkan pada aplikasi, tanpa sepengetahuan pengguna.
7. *Default Credentials* adalah kerentanan yang terjadi ketika sistem masih menggunakan nama pengguna dan kata sandi bawaan (*default*) dari vendor atau instalasi awal, tanpa diganti dengan kredensial yang lebih aman.
8. *Directory Listing Enabled* adalah kerentanan yang terjadi ketika server mengizinkan penampilan daftar isi direktori melalui browser.
9. *Directory Traversal* adalah kerentanan yang memungkinkan penyerang mengakses file atau direktori di luar folder yang diizinkan dengan memanipulasi jalur (*path*) pada input, misalnya menggunakan karakter.
10. *Exposed Database Service* adalah kerentanan yang terjadi ketika layanan basis data dapat diakses langsung dari jaringan publik tanpa pengamanan yang memadai, seperti autentikasi kuat atau firewall.
11. *HTML Injection* adalah kerentanan yang terjadi ketika aplikasi menerima dan menampilkan input pengguna tanpa validasi atau penyaringan yang tepat, sehingga penyerang dapat menyisipkan kode HTML berbahaya ke dalam halaman.
12. *Improper Session Invalidation* adalah kerentanan yang terjadi ketika sesi pengguna tidak diakhiri dengan benar setelah logout atau setelah batas waktu berakhir.

13. *Insecure Cookie Attributes* adalah kerentanan yang terjadi ketika cookie tidak dikonfigurasi dengan atribut keamanan yang tepat, seperti *Secure*, *HttpOnly*, atau *SameSite*.
14. *Missing HTTP Security Headers* adalah kerentanan yang terjadi ketika server tidak mengirimkan header keamanan penting dalam respon HTTP, seperti *Content-Security-Policy*, *X-Frame-Options*, atau *Strict-Transport-Security*.
15. *No Rate Limiting* adalah kerentanan yang terjadi ketika sistem tidak membatasi jumlah permintaan dalam jangka waktu tertentu.
16. *No Session Timeout* adalah kerentanan yang terjadi ketika sesi pengguna tetap aktif tanpa batas waktu atau tanpa mekanisme kedaluwarsa.
17. *Privilege Escalation* adalah kerentanan yang memungkinkan pengguna memperoleh hak akses yang lebih tinggi dari yang seharusnya, misalnya dari pengguna biasa menjadi administrator.
18. *Reflected XSS* adalah kerentanan yang terjadi ketika input berbahaya dari pengguna langsung dikembalikan atau direfleksikan dalam respons halaman tanpa proses validasi atau penyaringan.
19. *Session Fixation* adalah kerentanan yang terjadi ketika aplikasi mengizinkan penggunaan atau penggantian ID sesi yang ditentukan oleh penyerang.
20. *SQL Injection* adalah kerentanan yang terjadi ketika input pengguna dimasukkan ke dalam perintah SQL tanpa validasi atau penyaringan yang tepat.
21. *Stored XSS* adalah kerentanan yang terjadi ketika input berbahaya dari pengguna disimpan di server, misalnya dalam database, dan kemudian ditampilkan kembali kepada pengguna lain tanpa validasi atau penyaringan.
22. *Unnecessary Services Exposed* adalah kerentanan yang terjadi ketika layanan atau port yang tidak diperlukan tetap aktif dan dapat diakses dari jaringan publik.
23. *Using Outdated Components* adalah kerentanan yang terjadi ketika aplikasi menggunakan komponen perangkat lunak, *library*, *framework*, atau *plugin* yang sudah usang dan tidak diperbarui.
24. *Vulnerable JS Library* adalah kerentanan yang terjadi ketika aplikasi menggunakan pustaka JavaScript yang memiliki celah keamanan atau versi yang sudah usang.
25. *Weak Password Policy* adalah kerentanan yang terjadi ketika sistem menerapkan kebijakan kata sandi yang lemah, seperti mengizinkan kata sandi pendek, tanpa kombinasi huruf, angka, atau simbol, serta tanpa mekanisme penggantian berkala.
26. *Weak TLS Configuration* adalah kerentanan yang terjadi ketika konfigurasi protokol TLS (*Transport Layer Security*) tidak mengikuti standar keamanan yang baik, misalnya menggunakan versi TLS yang usang, *cipher suite* lemah, atau tidak mengaktifkan fitur seperti *Perfect Forward Secrecy*.

Klasifikasi Hasil Pengujian Berdasarkan OWASP Top 10:2021

Selanjutnya dilakukan klasifikasi berdasarkan OWASP Top 10:2021 pada setiap celah yang ditemukan. Ringkasan hasil klasifikasi OWASP Top 10:2021 celah dapat dilihat pada Tabel 1 sebagai berikut:

Tabel 1. Hasil Klasifikasi OWASP Top 10:2021

No	Nama Celah	Klasifikasi OWASP Top 10:2011
01	*Application Error Disclosure*	*A05:2021 – Security Misconfiguration*
02	*Arbitrary File Upload*	*A05:2021 – Security Misconfiguration*
03	*Broken Access Control*	*A01:2021 – Broken Access Control*
04	*Brute Force Login*	*A07:2021 – Identification and Authentication Failures*
05	*Business Logic Data Validation Bypass*	*A04:2021 – Insecure Design*
06	*Cross Site Request Forgery*	*A07:2021 – Identification and Authentication Failures*
07	*Default Credentials*	*A07:2021 – Identification and Authentication Failures*
08	*Directory Listing Enabled*	*A05:2021 – Security Misconfiguration*
09	*Directory Traversal*	*A05:2021 – Security Misconfiguration*
10	*Exposed Database Service*	*A05:2021 – Security Misconfiguration*
11	*HTML Injection*	*A03:2021 – Injection*
12	*Improper Session Invalidation*	*A07:2021 – Identification and Authentication Failures*
13	*Insecure Cookie Attributes*	*A07:2021 – Identification and Authentication Failures*
14	*Missing HTTP Security Headers*	*A05:2021 – Security Misconfiguration*
15	*No Rate Limiting*	*A07:2021 – Identification and Authentication Failures*
16	*No Session Timeout*	*A07:2021 – Identification and Authentication Failures*
17	*Privilege Escalation*	*A01:2021 – Broken Access Control*
18	*Reflected XSS*	*A03:2021 – Injection*
19	*Session Fixation*	*A07:2021 – Identification and Authentication Failures*
20	*SQL Injection*	*A03:2021 – Injection*
21	*Stored XSS*	*A03:2021 – Injection*
22	*Unnecessary Services Exposed*	*A05:2021 – Security Misconfiguration*
23	*Using Outdated Components*	*A06:2021 – Vulnerable and Outdated Components*
24	*Vulnerable JS Library*	*A06:2021 – Vulnerable and Outdated Components*
25	*Weak Password Policy*	*A07:2021 – Identification and Authentication Failures*
26	*Weak TLS Configuration*	*A02:2021 – Cryptographic Failures*

Analisis Tingkat Risiko Menggunakan OWASP *Risk Rating Methodology*

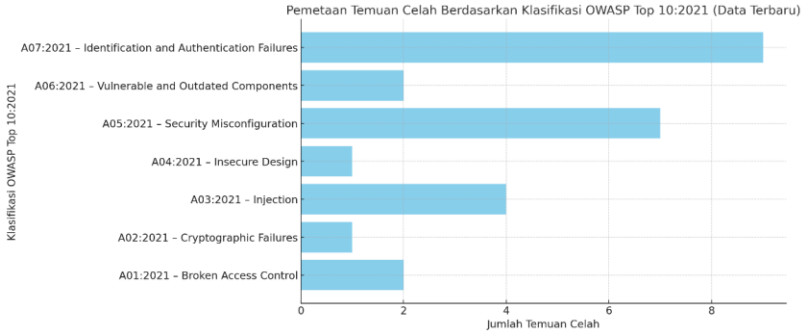
Langkah selanjutnya adalah mengevaluasi tingkat bahaya dari setiap kerentanan yang ditemukan menggunakan OWASP *Risk Rating Methodology*. Ringkasan hasil analisis tingkat risiko masing-masing temuan celah dapat dilihat pada Tabel 2 sebagai berikut:

Tabel 2. Ringkasan Hasil Analisis *Risk Rating*

No	Nama Celah	Klasifikasi OWASP Top 10:2011	Risiko
01	*Application Error Disclosure*	*A05:2021 – Security Misconfiguration*	*Medium*
02	*Arbitrary File Upload*	*A05:2021 – Security Misconfiguration*	*Critical*
03	*Broken Access Control*	*A01:2021 – Broken Access Control*	*Critical*
04	*Brute Force Login*	*A07:2021 – Identification and Authentication Failures*	*Critical*
05	*Business Logic Data Validation Bypass*	*A04:2021 – Insecure Design*	*High*
06	*Cross Site Request Forgery*	*A07:2021 – Identification and Authentication Failures*	*Medium*
07	*Default Credentials*	*A07:2021 – Identification and Authentication Failures*	*Critical*
08	*Directory Listing Enabled*	*A05:2021 – Security Misconfiguration*	*Medium*
09	*Directory Traversal*	*A05:2021 – Security Misconfiguration*	*Medium*
10	*Exposed Database Service*	*A05:2021 – Security Misconfiguration*	*Medium*
11	*HTML Injection*	*A03:2021 – Injection*	*Medium*
12	*Improper Session Invalidation*	*A07:2021 – Identification and Authentication Failures*	*High*
13	*Insecure Cookie Attributes*	*A07:2021 – Identification and Authentication Failures*	*Medium*
14	*Missing HTTP Security Headers*	*A05:2021 – Security Misconfiguration*	*Medium*

No	Nama Celah	Klasifikasi OWASP Top 10:2011	Risiko
15	*No Rate Limiting*	*A07:2021 – Identification and Authentication Failures*	*Medium*
16	*No Session Timeout*	*A07:2021 – Identification and Authentication Failures*	*Medium*
17	*Privilege Escalation*	*A01:2021 – Broken Access Control*	*Critical*
18	*Reflected XSS*	*A03:2021 – Injection*	*High*
19	*Session Fixation*	*A07:2021 – Identification and Authentication Failures*	*Medium*
20	*SQL Injection*	*A03:2021 – Injection*	*Critical*
21	*Stored XSS*	*A03:2021 – Injection*	*High*
22	*Unnecessary Services Exposed*	*A05:2021 – Security Misconfiguration*	*Medium*
23	*Using Outdated Components*	*A06:2021 – Vulnerable and Outdated Components*	*Critical*
24	*Vulnerable JS Library*	*A06:2021 – Vulnerable and Outdated Components*	*Medium*
25	*Weak Password Policy*	*A07:2021 – Identification and Authentication Failures*	*High*
26	*Weak TLS Configuration*	*A02:2021 – Cryptographic Failures*	*High*

Berdasarkan hasil pengujian yang dilakukan menggunakan framework OWASP, yaitu OWASP WSTG versi 4.2 ditemukan 26 celah keamanan yang diklasifikasikan ke dalam berbagai kategori OWASP Top 10:2021 yang dapat dilihat pada Gambar 4 dibawah ini.



Gambar 4. Klasifikasi Celah Berdasarkan OWASP Top 10

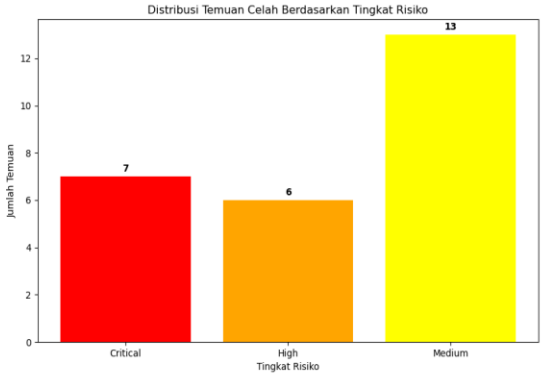
Grafik tersebut menggambarkan jumlah celah yang ditemukan dalam setiap kategori OWASP, yang telah dikelompokkan juga berdasarkan tingkat kerentanannya. Distribusi celah berdasarkan kategori OWASP Top 10 dapat dilihat pada Tabel 3 berikut.

Tabel 3. Distribusi Celah Berdasarkan OWASP Top 10

Kategori OWASP	Jumlah	Persentase
A01:2021 – Broken Access Control	2	7.69%
A02:2021 – Cryptographic Failures	1	3.85%
A03:2021 – Injection	4	15.38%
A04:2021 – Insecure Design	1	3.85%
A05:2021 – Security Misconfiguration	7	26.92%
A06:2021 – Vulnerable and Outdated Components	2	7.69%
A07:2021 – Identification and Authentication Failures	9	34.62%

Sementara berdasarkan OWASP *Risk Rating Methodology*, dari 26 celah yang teridenfifikasi dapat dikelompokkan berdasarkan tingkat risikonya yaitu 7 celah *Critical*, 6 celah *High*, dan 13 celah *Medium*. Gambar 5 menunjukkan ringkasan tingkat risiko celah

keamanan berdasarkan OWASP *Risk Rating Methodology*, sedangkan Tabel 4 menunjukkan distribusi celah berdasarkan tingkat risikonya.



Gambar 5. Ringkasan Tingkat Risiko Celah Keamanan

Hasil pengujian dapat disimpulkan sebagai berikut:

a. Lebih dari 60% dari total celah yang ditemukan berada pada kategori A07 (*Identification and Authentication Failures*) dan A05 (*Security Misconfiguration*). Hal ini menunjukkan kelemahan sistem dalam pengelolaan identitas, sesi, dan konfigurasi keamanan.
b. Sebagian besar celah (50%) berada pada tingkat risiko *Medium*. Sedangkan celah dengan tingkat risiko *Critical* tersebar di berbagai kategori, termasuk diantaranya A01, A03, A05, A06, dan A07. Meskipun demikian, celah *Critical* ini harus tetap menjadi prioritas utama untuk mitigasi karena potensi dampaknya yang besar, disusul dengan celah *High*.

Tabel 4. Distribusi Celah Berdasarkan Tingkat Risiko

Tingkat Risiko	Jumlah	Persentase
Critical	7	26.92%
High	6	23.08%
Medium	13	50.00%

KESIMPULAN

Penelitian ini berhasil membuktikan bahwa integrasi framework OWASP WSTG v4.2, OWASP Top 10:2021, dan OWASP *Risk Rating Methodology* merupakan instrumen yang efektif dalam mengidentifikasi serta memetakan profil risiko keamanan aplikasi web secara sistematis. Penerapan metode ini memiliki keunggulan dalam memberikan cakupan pengujian teknis yang komprehensif serta penilaian risiko kuantitatif yang objektif, yang dibuktikan dengan penemuan 26 celah keamanan yang terukur. Namun, hasil evaluasi juga menyingkap kelemahan signifikan pada sistem yang diuji, di mana kategori Identification and Authentication Failures menjadi titik lemah dominan (34,62%) dan 26,92% dari total kerentanan berada pada tingkat risiko *Critical*. Secara keseluruhan, kombinasi ketiga pendekatan OWASP ini memberikan jawaban atas permasalahan penelitian dengan menyediakan kerangka kerja yang terstruktur untuk menentukan prioritas mitigasi keamanan aplikasi secara akurat.

Peneliti selanjutnya disarankan untuk memperluas cakupan pengujian pada arsitektur *Application Programming Interface* (API) dan aplikasi seluler dengan mengintegrasikan framework spesifik seperti OWASP ASVS atau OWASP *Mobile Security*. Mengingat dinamika ancaman siber, penggunaan metodologi versi terbaru seperti OWASP WSTG v5 sangat dianjurkan untuk menjaga relevansi parameter pengujian terhadap teknologi terkini. Selain itu, pengembangan penelitian dapat diarahkan pada kombinasi alat pemindaian otomatis dan analisis manual yang lebih mendalam untuk mengatasi kesulitan deteksi celah logika bisnis yang kompleks. Eksplorasi pada aspek-aspek tersebut diharapkan mampu menutupi keterbatasan penelitian ini dalam memetakan vektor serangan yang lebih luas dan heterogen.

REFERENSI

Albahar, M. A. (2017). Cyber attacks and terrorism: A twenty-first century conundrum. *Science and Engineering Ethics*, *25*(4), 993–1008. https://doi.org/10.1007/s11948-016-9864-0

Armando, Y., & Rosalina. (2024). Penetration Testing Tangerang City Web Application With Implementing OWASP Top 10 Web Security Risks Framework. *Jurnal Informatika Dan Sains*, *6*(2).

Deepa, G., & Thilagam, P. S. (2016). Securing web applications from injection and logic vulnerabilities: Approaches and research challenges. *Information and Software Technology*, *74*, 160–180. https://doi.org/10.1016/j.infsof.2016.02.005

Dharmawangsa, I. D. G., Sasmita, G. M. A., & Pratama, I. P. A. E. (2023). Penetration Testing Berbasis OWASP Testing Guide Versi 4.2 (Studi Kasus: X Website). *Jurnal Jitter*, *15*(2).

Hydara, I., Sultan, A. B. M., Zulzalil, H., & Admodisastro, N. (2015). Current state of research on cross-site scripting (XSS) — A systematic literature review. *Information and Software Technology*, *58*, 170–186. https://doi.org/10.1016/j.infsof.2014.07.010

Kamal, M., Nasrullah, M., Rosadi, R., Anggito, Y., & Roy, S. C. (2024). Evaluation of Information Security at the XYZ Foundation Using OWASP Top 10 2021 Framework. *Journal of Advances in Information and Industrial Technology*, *6*(2).

Khera, Y., Kumar, D., Sujana, & Garg, N. (2019). Analysis and impact of vulnerability assessment and penetration testing. *Procedia Computer Science*, *163*, 154–160. https://doi.org/10.1016/j.procs.2019.12.097

OWASP. (2026a). *OWASP Risk Rating Methodology*. https://owasp.org/www-community/OWASP_Risk_Rating_Methodology

OWASP. (2026b). *OWASP Top 10:2021*. https://owasp.org/Top10/

OWASP. (2026c). *OWASP Web Security Testing Guide v4.2*. https://owasp.org/www-project-web-security-testing-guide/v42/

Paramarta, V., Destriana, R., & Sensuse, D. I. (2021). SQL injection and XSS vulnerability detection and prevention in web applications: A systematic review. *Journal of Physics: Conference Series*, *1869*(1), 12100. https://doi.org/10.1088/1742-6596/1869/1/012100

Priambodo, D. F., Rifansyah, A. D., & Hasbi, M. (2023). Penetration Testing Web XYZ Berdasarkan OWASP Risk Rating. *Jurnal Teknika*, *11*(1).

Rafeli, A. I., Seta, H. B., & Widi, I. W. (2022). Pengujian Celah Keamanan Menggunakan Metode OWASP Web Security Testing Guide (WSTG) pada Website XYZ. *Jurnal Informatika*, *9*(3).

Setiawan, H., Erlangga, L. E., Siddiq, S., & Gunawan, Y. A. (2023). Analisis Kerawanan Pada Aplikasi Website Menggunakan Standar OWASP Top 10 Untuk Penilaian Risk

Rating. *Jurnal Ilmiah Kriptografi*, *17*(1).
Stiawan, D., Abdullah, A. H., & Idris, M. Y. (2021). Penetration testing and vulnerability assessment: A comprehensive approach to network security. *Egyptian Informatics Journal*, *22*(2), 219–231. https://doi.org/10.1016/j.eij.2020.08.004
Thion, R., & Coulondre, S. (2022). A novel approach to automated security testing using OWASP methodology. *Computers & Security*, *114*, 102597. https://doi.org/10.1016/j.cose.2021.102597
Umar, R., Riadi, I., & Elfatiha, M. I. A. (2024). Security analysis of web-based academic information system using OWASP framework. *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, *5*(4).
Widyaningrum, B. N., Rani, D. M., & Ramadhani, L. K. (2024). Analisis Metode OWASP V4.2 dalam Pengujian Keamanan Sistem Informasi Rumah Sakit. *Jurnal Teknik Elektromedik Polbitrada*, *5*(2).
Yamin, R. T. N. Y., Suarjaya, I. M. A. D., & Pratama, I. P. A. E. (2022). Penetration Testing on the SISAKTI Application at Udayana University Using the OWASP Testing Guide Version 4. *Jurnal Ilmiah Merpati*, *10*(3).
Yusuf, M. F., Hikmah, I. R., Amiruddin, & Sunaringtyas, S. U. (2025). Security Testing of XYZ Website Application Using ISSAF and OWASP WSTG v4.2 Methods. *Jurnal Teknika*, *12*(2).