AICOMS
Applied Information Technology and Computer Science
e-ISSN: 2964-7703
Vol. 4 No. 2, Desember-2025, pp. 37-45

Penerapan Pentesting pada EasyCart untuk Menghadapi Ancaman Keamanan Siber

Imam Sutanto[1], Mochamad Fahrul Reza [2]

[1] Universitas Esa Unggul; imam.sutanto@esaunggul.ac.id
[2] Universitas Esa Unggul; fahrulreza693@student.esaunggul.ac.id
* Korespondensi: fahrulreza693@student.esaunggul.ac.id

Info Artikel:
Dikirim: 06 Agustus 2025
Direvisi: 07 September 2025
Diterima: 13 November 2025

Intisari: Keamanan informasi pada aplikasi e-commerce merupakan aspek krusial dalam menjaga integritas, kerahasiaan, dan ketersediaan data pengguna. Metode yang digunakan adalah penetration testing dengan pendekatan black-box dan grey-box, mengacu pada standar Penetration Testing Execution Standard (PTES) serta kerangka kerja OWASP Top 10 tahun 2021.Pengujian dilakukan melalui tujuh tahapan PTES, yaitu: Pre-engagement Interactions, Intelligence Gathering, Threat Modeling, Vulnerability Analysis, Exploitation, Post-Exploitation, dan Reporting. Lingkungan pengujian dijalankan secara lokal dengan memanfaatkan alat bantu seperti Burp Suite, OWASP ZAP, Nikto, SQLMap, dan Nmap. Hasil pengujian mengidentifikasi sebanyak 20 kerentanan dengan tingkat risiko tinggi, sedang, dan rendah, termasuk di antaranya Cross-Site Scripting (XSS), SQL Injection, Broken Access Control, dan Security Misconfiguration.Rekomendasi mitigasi disusun berdasarkan kontrol ISO/IEC 27001:2022, khususnya Annex A.5 (kebijakan keamanan informasi), A.8 (manajemen aset), dan A.12 (keamanan operasional). Penelitian ini memberikan kontribusi terhadap pemahaman serta penerapan pengujian keamanan berbasis standar pada aplikasi simulatif, sekaligus menekankan pentingnya validasi input, konfigurasi sistem yang aman, dan pembaruan berkala sebagai langkah mitigasi terhadap ancaman siber.

Kata Kunci: Pentesting; OWASP Top 10; PTES; ISO/IEC 27001; EasyCart; Keamanan Web

Abstrak: *Information security in e-commerce applications is a crucial aspect in maintaining the integrity, confidentiality, and availability of user data. The method used is penetration testing with a black-box and grey-box approach, referring to the Penetration Testing Execution Standard (PTES) and the OWASP Top 10 framework for 2021. The testing was conducted through the seven PTES phases: Pre-engagement Interactions, Intelligence Gathering, Threat Modeling, Vulnerability Analysis, Exploitation, Post-Exploitation, and Reporting. The testing environment was run locally using tools such as Burp Suite, OWASP ZAP, Nikto, SQLMap, and Nmap. The testing results identified 20 vulnerabilities with high, medium, and low risk levels, including Cross-Site Scripting (XSS), SQL Injection, Broken Access Control, and Security Misconfiguration. Mitigation recommendations are based on ISO/IEC 27001:2022 controls, specifically Annex A.5 (information security policy), A.8 (asset management), and A.12 (operational security). This research contributes to the understanding and application of standards-based security testing in simulation applications, while emphasizing the importance of input validation, secure system configuration, and regular updates as mitigation measures against cyber threats.*

Keywords: Pentesting; OWASP Top 10; PTES; ISO/IEC 27001; EasyCart; Web Security

1. Pendahuluan

Perkembangan teknologi informasi telah memberikan dampak signifikan terhadap industri e-commerce, mendorong transformasi digital yang masif dalam sistem transaksi dan interaksi pelanggan. Di balik pertumbuhan tersebut, tantangan terhadap keamanan sistem informasi menjadi semakin kompleks dan mendesak untuk diatasi. Menurut laporan Check Point Research[1] ,terjadi lonjakan serangan siber terhadap situs e-commerce, terutama yang mengeksploitasi kelemahan pada input pengguna dan mekanisme autentikasi. Hal ini menjadikan keamanan data pengguna sebagai aspek krusial yang menuntut perhatian serius dari pengembang dan penyedia platform digital.
Untuk mengantisipasi ancaman tersebut, pengujian keamanan aplikasi secara sistematis melalui pendekatan penetration testing (uji penetrasi) menjadi langkah penting. PTES (Penetration Testing Execution Standard) menyediakan kerangka kerja metodologis yang dapat digunakan dalam proses pengujian ini, sementara OWASP Top 10 berperan sebagai referensi utama dalam pengklasifikasian jenis kerentanan yang umum ditemukan pada aplikasi web [2].Meskipun telah tersedia banyak pedoman teknis, masih diperlukan studi aplikatif yang dapat menjembatani standar tersebut dengan implementasi nyata pada proyek berbasis teknologi terkini.

Tujuan utama dari studi ini adalah menyusun panduan pengujian keamanan berbasis standar yang dapat diterapkan pada aplikasi web modern, serta merumuskan rekomendasi mitigasi berdasarkan kontrol keamanan informasi dalam ISO/IEC 27001:2022 [3]. Hasil penelitian diharapkan memberikan kontribusi praktis bagi pengembang dalam membangun sistem e-commerce yang lebih aman dan andal, serta menjadi referensi dalam penguatan kebijakan teknis dan manajemen risiko informasi di lingkungan digital.

2. Tinjauan Pustaka

2.1 *E-commerce*

E-commerce, atau perdagangan elektronik, adalah proses membeli dan menjual barang dan jasa melalui internet. Ini mencakup berbagai transaksi yang dilakukan secara online, seperti belanja online, pembayaran elektronik, lelang online, dan perbankan internet. E-commerce memungkinkan individu dan perusahaan untuk melakukan transaksi tanpa batasan geografis, memanfaatkan platform digital untuk menjangkau pasar yang lebih luas[4].

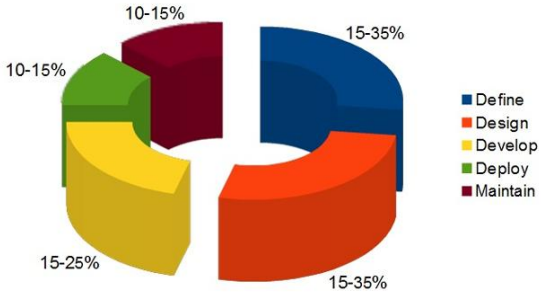
2.2 *Open Web Application Security Project*

Merupakan proyek Pengujian OWASP telah dikembangkan selama bertahun-tahun.Tujuan dari proyek ini adalah untuk membantu orang memahami apa, mengapa, kapan, di mana, dan bagaimana cara menguji aplikasi web.Proyek ini telah memberikan kerangka kerja pengujian yang lengkap,bukan hanya sebuah daftar periksa sederhana atau resep masalah yang harus ditangani.

Pembaca dapat menggunakan kerangka kerja ini sebagai template untuk membangun program pengujian mereka sendiri atau untuk memenuhi syarat proses orang lain.Panduan Pengujian menjelaskan secara rinci kerangka kerja pengujian secara umum dan teknik yang diperlukan untuk mengimplementasikan kerangka kerja tersebut dalam praktiknya[2].

2.3 *Penetration Testing*

Pengujian penetrasi telah menjadi teknik umum yang digunakan untuk menguji keamanan jaringan selama bertahun-tahun.Hal ini juga dikenal secara umum sebagai pengujian kotak hitam atau peretasan etis.Pengujian penetrasi pada dasarnya adalah "seni" menguji aplikasi yang sedang berjalan dari jarak jauh untuk menemukan kerentanan keamanan, tanpa mengetahui cara kerja aplikasi itu sendiri. Biasanya, tim uji penetrasi akan memiliki akses ke aplikasi seolah-olah mereka adalah pengguna. Penguji bertindak seperti penyerang dan mencoba menemukan dan mengeksploitasi kerentanan.Dalam banyak kasus, penguji akan diberikan akun yang valid pada system[5].



Gambar 1. Upaya pengujian dalam SDLC

2.4 *Penetration Testing Execution Standard*

Standar pelaksanaan pengujian penetrasi terdiri dari tujuh (7) bagian utama. Ini mencakup semua hal yang terkait dengan pengujian penetrasi - mulai dari komunikasi awal dan alasan di balik pentest, melalui pengumpulan intelijen dan fase pemodelan ancaman di mana penguji bekerja di belakang layar untuk mendapatkan pemahaman yang lebih baik tentang organisasi yang diuji, melalui penelitian kerentanan, eksploitasi, dan pasca eksploitasi, di mana keahlian keamanan teknis penguji ikut berperan dan digabungkan dengan pemahaman bisnis tentang keterlibatan, dan akhirnya ke pelaporan, yang menangkap keseluruhan proses, dengan cara yang masuk akal bagi pelanggan dan memberikan nilai terbaik baginya[6].

2.5 *ISO 27001*

ISO/IEC 27001:2022 merupakan standar internasional yang menetapkan persyaratan untuk sistem manajemen keamanan informasi (Information Security Management System/ISMS). Standar ini memberikan kerangka kerja dalam menetapkan, menerapkan, memelihara, dan terus meningkatkan keamanan informasi dalam suatu organisasi [7]. ISO/IEC 27001 digunakan sebagai acuan untuk menyusun rekomendasi keamanan berdasarkan hasil pengujian sistem EasyCart. Rekomendasi diarahkan agar sejalan dengan kontrol keamanan global yang telah distandarkan dalam Annex A.

Adapun kontrol yang menjadi referensi utama adalah:

1. A.5 – Policies for Information Security: Menyediakan arah dan dukungan dari manajemen untuk keamanan informasi;
2. A.8 – Access Control: Menjamin bahwa hanya pihak berwenang yang memiliki akses terhadap informasi dan system;
3. A.12 – Operations Security: Menjaga keberlangsungan dan keamanan sistem informasi selama proses operasional.

3. Metode Penelitian

Penelitian ini menerapkan pendekatan penetration testing berbasis PTES (Penetration Testing Execution Standard), diselaraskan dengan klasifikasi kerentanan OWASP Top 10 dan kontrol keamanan ISO/IEC 27001:2022. Setiap tahapan—dari intelligence gathering, threat modeling, hingga post-exploitation dan reporting—diintegrasikan untuk mengevaluasi sistem simulatif EasyCart dengan metode black-box dan grey-box testing.Pengujian dilakukan terhadap fitur utama EasyCart seperti autentikasi, input pengguna, dan transaksi. Tools yang digunakan meliputi Nmap, Burp Suite, OWASP ZAP, dan SQLMap. Temuan kerentanan dievaluasi berdasarkan tingkat risiko dan diberikan rekomendasi mitigasi sesuai kontrol ISO/IEC 27001 Annex A.

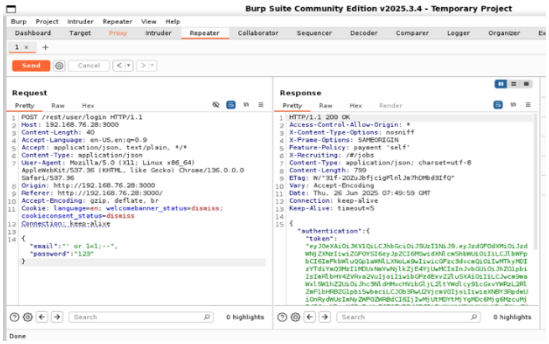
4. Hasil dan Pembahasan

Pengujian sistem simulatif EasyCart menghasilkan identifikasi lima jenis kerentanan aktif yang mencakup *SQL Injection*, *Broken Access Control*, Reflected XSS, CSRF, dan *Exposed Sensitive Files*. Setiap temuan ini diuji menggunakan pendekatan PTES (Penetration Testing Execution Standard), dengan tahapan eksplorasi, eksploitasi, dan verifikasi dampak secara sistematis. Hasil eksploitasi menunjukkan bahwa aktor ancaman dapat melakukan eskalasi hak akses, manipulasi data pengguna, hingga pengambilalihan sesi login secara tidak sah.

4.1 Hasil Temuan Kerentanan Sistem

4.1.1 *SQL Injection*

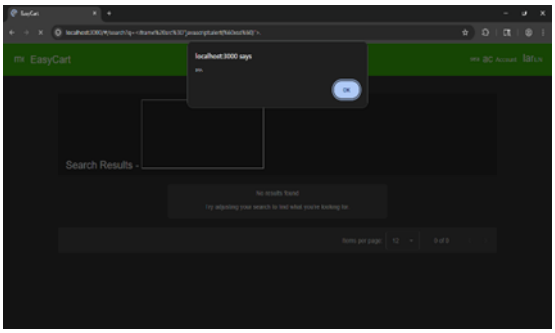
Kerentanan ini ditemukan pada halaman login admin, di mana pengguna dapat memasukkan payload seperti admin' OR '1'='1 sebagai input. Input ini tidak divalidasi atau disanitasi oleh sistem, sehingga *query* SQL *backend* berubah menjadi pernyataan yang selalu benar. Akibatnya, proses autentikasi gagal membedakan antara pengguna sah dan penyerang. *Output* dari eksploitasi ini adalah akses penuh ke dashboard administrator tanpa kredensial yang valid. Untuk mengatasi hal ini, sistem harus menerapkan prepared statements atau parameterized queries yang mencegah injeksi kode, serta memperkuat filter input di sisi server menggunakan framework ORM yang aman.



Gambar 2. Tampilan Request untuk login admin

4.1.2 *Cross-Site Scripting (XSS)*

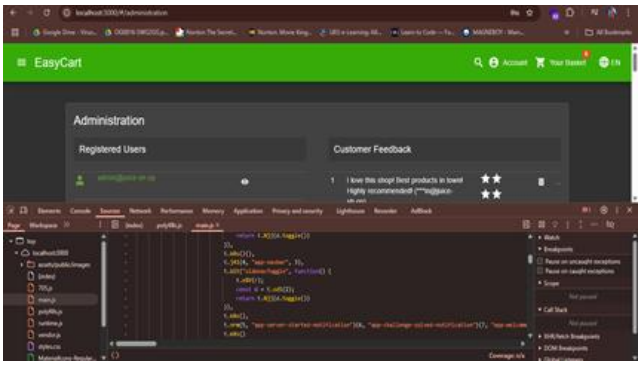
Temuan XSS muncul ketika pengguna memasukkan skrip JavaScript berbahaya ke dalam kolom input search (misalnya: `<script>alert('XSS')</script>`). Sistem tidak melakukan escaping atau encoding pada output HTML, sehingga browser mengeksekusi skrip tersebut secara langsung. Hasilnya, penyerang dapat mencuri cookie sesi korban, melakukan redirect berbahaya, atau mengubah tampilan antarmuka pengguna. Solusi teknis mencakup output encoding pada setiap data dinamis yang ditampilkan, implementasi Content-Security-Policy (CSP) untuk membatasi eksekusi skrip, serta validasi input di sisi server.



Gambar 3. Tampilan ketika input script xss

4.1.3 *Broken Access Control*

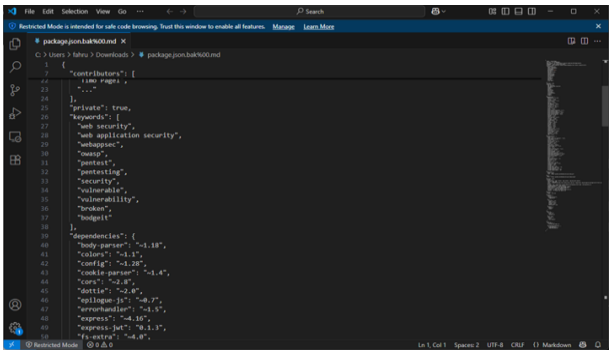
Kerentanan ini terjadi karena sistem tidak menerapkan verifikasi hak akses yang tepat pada endpoint manajemen peran. Penyerang dapat memodifikasi permintaan HTTP (seperti mengubah parameter role) untuk mengakses fitur atau data yang seharusnya hanya tersedia bagi admin. Proses otorisasi yang lemah menyebabkan eskalasi hak akses tanpa autentikasi tambahan. Output-nya adalah kemampuan penyerang untuk mengakses halaman superadmin dengan hanya login sebagai admin biasa. Untuk mitigasi, sistem harus menerapkan Role-Based Access Control (RBAC) secara konsisten dan memverifikasi hak akses di sisi server, bukan hanya pada antarmuka pengguna.



Gambar 4. Access Superadmin dengan login admin

4.1.4 *Security Misconfiguration*

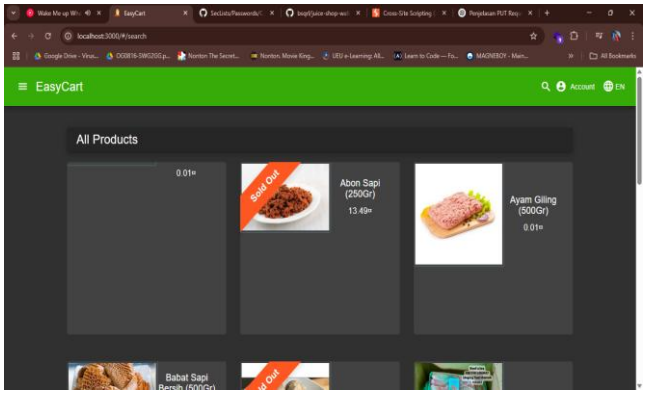
Sistem EasyCart menyimpan file konfigurasi cadangan seperti .bak atau .env.bak di direktori web yang dapat diakses publik. Ketika penyerang menebak atau meramban URL tertentu, file ini dapat diunduh dan dianalisis. Proses ini mengekspos variabel lingkungan penting seperti kredensial database dan struktur direktori sistem. Output dari eksploitasi ini adalah pengungkapan informasi sensitif yang dapat digunakan untuk serangan lanjutan. Solusi teknis mencakup penghapusan file tidak penting dari direktori produksi, konfigurasi server untuk menolak akses ke file dengan ekstensi tertentu, serta audit rutin terhadap direktori publik



Gambar 5. Hasil file .bak berhasil di akses

4.1.5 Cross Site Request Forgery

Kerentanan CSRF ditemukan karena sistem tidak mengimplementasikan token validasi pada setiap permintaan yang mengubah data (misalnya, menambah produk atau mengubah harga). Penyerang dapat membuat halaman palsu yang diam-diam mengirim permintaan POST atas nama pengguna sah yang sedang login. Proses eksploitasi memanfaatkan cookie sesi korban tanpa sepengetahuan mereka. Output-nya adalah manipulasi data yang valid tanpa interaksi pengguna langsung. Untuk menghindari serangan ini, sistem harus menyertakan CSRF token unik di setiap form dan validasi token tersebut pada sisi server, serta mengaktifkan atribut SameSite pada cookie untuk mencegah akses silang.



Gambar 6. Hasil Ketika menambahkan produk

4.2 Klasifikasi Risiko

Setiap kerentanan dievaluasi berdasarkan tingkat eksploitasi dan dampaknya terhadap sistem. Tabel berikut merangkum klasifikasi risiko

Tabel 1. Klasifikasi Risiko

Tingkat Risiko	Jumlah Temuan	Kerentanan
Critical	1	Sql Injection
High	1	Broken Access Control
Medium	2	XSS
Low	1	Cross Site Request Forgery
Information	1	Security Misconfigure (user dapat akses file tersembunyi)

4.3 Diskusi Temuan

Berdasarkan hasil pengujian yang telah dilakukan, terlihat bahwa sebagian besar kerentanan pada sistem EasyCart disebabkan oleh kelemahan dalam validasi input, kurangnya kontrol akses yang ketat, serta konfigurasi sistem yang tidak aman. *SQL Injection* dan XSS muncul karena input pengguna tidak disaring atau disanitasi dengan benar sebelum diproses oleh sistem *backend* maupun ditampilkan kembali ke browser. *Broken Access Control* menunjukkan bahwa sistem tidak memverifikasi peran pengguna secara menyeluruh, memungkinkan aktor ancaman untuk memanipulasi hak akses dengan mudah. Selain itu, absennya token CSRF dan file konfigurasi yang dapat diakses publik mengindikasikan lemahnya implementasi prinsip-prinsip keamanan dasar dalam pengembangan sistem.

Kerentanan-kerentanan ini tidak hanya menunjukkan adanya celah teknis, tetapi juga mencerminkan kurangnya penerapan *framework* keamanan seperti OWASP ASVS atau DevSecOps dalam siklus pengembangan perangkat lunak. Padahal, kerangka kerja seperti PTES, OWASP Top 10, dan ISO/IEC 27001 sudah menyediakan pedoman teknis dan kebijakan mitigasi yang dapat diadopsi secara sistematis. Oleh karena itu, perlu dilakukan evaluasi menyeluruh terhadap kebijakan keamanan, pelatihan pengembang, serta penerapan kontrol keamanan yang lebih kuat di sisi server. Penting juga untuk memastikan bahwa semua komponen sistem, termasuk konfigurasi dan dependency eksternal, diperiksa dan diperbarui secara berkala.

4.4 Validasi Hasil Pengujian

Untuk memastikan akurasi dan validitas dari setiap temuan kerentanan, proses validasi dilakukan secara sistematis dengan menggabungkan pendekatan pengujian otomatis dan manual. Tools otomatis seperti Burp Suite, OWASP ZAP, SQLMap, dan Nmap digunakan untuk melakukan pemindaian awal terhadap permukaan serangan pada aplikasi EasyCart. Setiap hasil pemindaian dari tools tersebut kemudian diverifikasi secara manual untuk menghindari kemungkinan false positive, dengan cara mengulangi eksploitasi secara langsung menggunakan parameter serangan yang sama dan mengamati respon sistem secara real-time.

Selain itu, pengujian manual dilakukan dengan memperhatikan skenario eksploitasi dunia nyata, di mana input dan perilaku pengguna disimulasikan menyerupai pola serangan aktual yang kerap terjadi pada aplikasi e-commerce. Hasil eksploitasi juga dibandingkan dengan referensi dari OWASP Testing Guide dan dokumentasi kerentanan umum (CVE) untuk menilai sejauh mana dampak serta keparahan dari kerentanan yang ditemukan.

Proses validasi ini bertujuan untuk memastikan bahwa setiap kerentanan yang dilaporkan benar-benar dapat dieksploitasi dalam kondisi lingkungan sistem saat itu, dan bukan hanya merupakan anomali dari hasil pemindaian alat. Validasi ganda ini meningkatkan reliabilitas hasil pengujian serta memperkuat justifikasi dalam penyusunan rekomendasi mitigasi berdasarkan standar ISO/IEC 27001:2022.

4.5 Rekomendasi Mitigasi

4.5.1 *SQL Injection*

Untuk mengatasi kerentanan SQL Injection, penggunaan *prepared statements* atau *parameterized queries* sangat direkomendasikan, sebagaimana dijelaskan dalam OWASP Top 10 tahun 2021[5]. Dengan pendekatan ini, input pengguna tidak akan dieksekusi sebagai bagian dari perintah SQL, melainkan diproses sebagai parameter terpisah yang aman. Selain itu, penerapan filter input berbasis whitelist serta penggunaan Object Relational Mapping (ORM) juga memperkecil peluang injeksi. ISO/IEC 27001:2022 Annex A.8 [8] mendukung penerapan kontrol terhadap akses dan integritas data untuk mencegah manipulasi melalui jalur input yang tidak tervalidasi. Penelitian dari Ray & Frankl (2007)[9] juga menunjukkan bahwa teknik ASSIST yang menggabungkan analisis statik dan instrumentasi runtime dapat secara efektif mencegah eksploitasi injeksi SQL.

4.5.2 *Cross-Site Scripting (XSS)*

Kerentanan XSS dapat diminimalkan dengan menerapkan *output encoding* pada setiap data yang ditampilkan di browser, sehingga karakter berbahaya tidak dapat dieksekusi sebagai skrip. Hal ini sesuai dengan pedoman dalam OWASP Web Security Testing Guide [5] serta didukung oleh kebijakan Content-Security-Policy (CSP) yang membatasi sumber skrip eksternal. Studi dari Ghafir et al. (2017) [10] juga mengklasifikasikan berbagai teknik mitigasi XSS, termasuk validasi input dan filtrasi konten dinamis. ISO/IEC 27001 Annex A.12 [8] merekomendasikan pengamanan operasional sistem, termasuk kontrol terhadap konten yang ditampilkan di sisi klien. Pendekatan modern menggunakan machine learning juga mulai digunakan untuk mendeteksi XSS secara otomatis dan efektif, sebagaimana diteliti oleh Hossen & Sabarimalai (2024) [11].

4.5.3 *Broken Access Control*

Untuk mencegah penyalahgunaan hak akses, sistem harus menerapkan *Role-Based Access Control (RBAC)* dan validasi peran secara eksplisit di sisi server. Setiap endpoint yang sensitif harus dilindungi dengan middleware atau guard yang memverifikasi otorisasi pengguna sebelum memberikan respon. Panduan OWASP dan standar ISO/IEC 27001:2022 Annex A.9 [8] menekankan pentingnya kontrol akses berbasis peran dan pembatasan akses minimal (least privilege). Dalam PTES [6], otorisasi merupakan bagian penting dalam tahap eksploitasi dan post-exploitation yang dapat menentukan keberhasilan atau kegagalan serangan. Selain itu, audit akses secara berkala dan penerapan log aktivitas dapat memperkuat deteksi dini terhadap penyalahgunaan hak akses..

4.5.4 *Security Misconfiguration*

Kesalahan konfigurasi sering kali membuka celah keamanan serius seperti eksposur file backup atau konfigurasi yang seharusnya bersifat internal. Untuk mitigasinya, sistem harus memblokir akses ke file dengan ekstensi sensitif seperti .bak, .env, atau .log, serta memastikan file tersebut tidak termasuk dalam direktori publik. PTES [6] dan ISO/IEC 27001 Annex A.12 [8] menggarisbawahi pentingnya keamanan konfigurasi operasional. Studi oleh Umapathi et al. (2017)[12] menyarankan penggunaan automated scanner dan audit konfigurasi saat proses deployment agar konfigurasi sistem tetap sesuai dengan baseline keamanan. Di sisi implementasi, penggunaan pipeline DevSecOps juga dapat membantu menjaga konsistensi konfigurasi saat pengembangan hingga produksi.

4.5.5 *Cross-Site Request Forgery (CSRF)*

Untuk mengatasi CSRF, sistem harus menyertakan token acak dan unik pada setiap form yang melakukan perubahan data, serta melakukan validasi token tersebut di sisi server. OWASP Top 10 [2] secara eksplisit menyebutkan pentingnya penggunaan anti-CSRF token sebagai mitigasi utama. Selain itu, pengaturan atribut cookie dengan *SameSite=Strict* atau *SameSite=Lax* juga mencegah pengiriman otomatis cookie oleh browser pada permintaan silang. ISO/IEC 27001:2022 Annex A.12 [8] turut mendukung pengamanan proses otorisasi transaksi, terutama yang berkaitan dengan modifikasi data pengguna. Studi oleh Agrawal (2023) [13] menyimpulkan bahwa kombinasi token validasi dan same-site cookie dapat secara signifikan mengurangi efektivitas serangan CSRF.

4.6 Penanganan Insiden Keamanan

Penanganan insiden keamanan merupakan bagian penting dalam siklus keamanan informasi, khususnya ketika serangan siber telah berhasil dilakukan terhadap sistem. Berdasarkan ISO/IEC 27035 dan ISO/IEC 27001:2022 Annex A.16, organisasi wajib memiliki prosedur sistematis dalam menghadapi dan memulihkan diri dari insiden [14]. Langkah pertama adalah isolasi sistem yang terdampak untuk mencegah penyebaran serangan ke sistem lain. Selanjutnya dilakukan identifikasi dan analisis log untuk menelusuri metode serangan, sumber serangan, dan dampak terhadap data atau layanan.

Apabila ditemukan adanya kebocoran atau modifikasi data, organisasi harus segera melakukan notifikasi kepada pihak terkait, termasuk pengguna dan otoritas regulasi jika diperlukan. Tahapan berikutnya adalah pemulihan sistem melalui perbaikan kerentanan, instalasi patch keamanan, dan pengembalian layanan secara bertahap. Setelah sistem kembali stabil, penting untuk melaksanakan post-incident review untuk mengevaluasi akar penyebab dan menyesuaikan kebijakan serta sistem pertahanan yang ada.

Standar PTES juga menekankan bahwa laporan insiden yang komprehensif harus disusun setelah proses pemulihan selesai, yang mencakup kronologi serangan, dampak yang terjadi, serta tindakan korektif yang telah diambil. Dengan implementasi prosedur tanggap insiden yang terstruktur, organisasi dapat meminimalisasi kerugian akibat serangan serta meningkatkan kesiapan menghadapi insiden serupa di masa mendatang.

5. kesimpulan

Penelitian ini menunjukkan bahwa sistem e-commerce simulatif EasyCart memiliki beberapa kerentanan kritis dan signifikan terhadap serangan siber, termasuk *SQL Injection*, *Cross-Site Scripting* (XSS), *Broken Access Control*, *Security Misconfiguration*, *dan Cross-Site Request Forgery* (CSRF). Pengujian dilakukan menggunakan pendekatan penetration testing berbasis standar PTES dengan referensi OWASP Top 10, serta hasil temuan divalidasi melalui kombinasi metode otomatis dan manual untuk meningkatkan akurasi eksploitasi.Hasil dari pengujian menegaskan bahwa kurangnya validasi input, absennya mekanisme proteksi terhadap akses yang tidak sah, dan konfigurasi sistem yang tidak aman merupakan akar permasalahan dari sebagian besar kerentanan yang ditemukan. Rekomendasi mitigasi telah disusun berdasarkan kontrol keamanan ISO/IEC 27001:2022, yang mencakup kebijakan keamanan informasi, manajemen aset, dan keamanan operasional.Dengan diterapkannya mitigasi teknis seperti prepared statements, output encoding, implementasi CSRF token, dan kontrol akses berbasis peran, sistem e-commerce dapat memiliki tingkat resiliensi yang lebih tinggi terhadap berbagai vektor serangan. Penelitian ini diharapkan dapat menjadi referensi bagi pengembang dan pengelola sistem dalam membangun aplikasi web yang aman, serta mendorong adopsi praktik DevSecOps dalam proses pengembangan perangkat lunak.

Daftar Pustaka

[1] checkpoint-team, “17th January– Threat Intelligence Report,” Check Point Research. Accessed: Jul. 28, 2025. [Online]. Available: https://research.checkpoint.com/2022/17th-january-threat-intelligence-report/

[2] OWASP Foundation., “OWASP Top 10 - 2021: The Ten Most Critical Web Application Security Risks.”

[3] ISO, “International Standard 27001 Information security, cybersecurity and privacy protection-Information security management systems-Requirements,” vol. 2022, pp. iii–5, 2022.

[4] A. Bloomenthal, “E-commerce Defined: Types, History, and Examples,” Investopedia. Accessed: Feb. 28, 2025. [Online]. Available: https://www.investopedia.com/terms/e/ecommerce.asp

[5] matteo mauidi and andrew Muller, “Owasp Web Security Testing Guide,” pp. 1–179, 2014.

[6] PTES, “High Level Organization of the Standard,” PTES, Penetration Testing Execution Standard. Accessed: Feb. 28, 2025. [Online]. Available: http://www.pentest-standard.org/index.php/Main_Page

[7] Mark Sharron, “ISO 27001 – Annex A Controls,” isms.online. Accessed: Jul. 18, 2025. [Online]. Available: https://www.isms.online/iso-27001/annex-a-controls/

[8] Sam Peters, “The Ultimate Guide to ISO 27001,” isms.online. Accessed: Aug. 05, 2025. [Online]. Available: https://www.isms.online/iso-27001/

[9] J. Ha *et al.*, “Improved error reporting for software that uses black-box components,” *Proc. ACM SIGPLAN Conf.*

Program. Lang. Des. Implement., pp. 101–111, 2007, doi: 10.1145/1250734.1250747.

[10] S. Gupta and B. B. Gupta, "Cross-Site Scripting (XSS) attacks and defense mechanisms: classification and state-of-the-art," *Int. J. Syst. Assur. Eng. Manag.*, vol. 8, pp. 512–530, 2017, doi: 10.1007/s13198-015-0376-0.

[11] V. Babaey and A. Ravindran, "GenXSS: An AI-Driven Framework for Automated Detection of XSS Attacks in WAFs," *Conf. Proc. - IEEE SOUTHEASTCON*, pp. 1519–1524, 2025, doi: 10.1109/SoutheastCon56624.2025.10971558.

[12] K. Sugata, T. Ogawa, and M. Haseyama, "Emotion estimation via tensor-based supervised decision-level fusion from multiple Brodmann areas," *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, pp. 999–1003, 2017, doi: 10.1109/ICASSP.2017.7952306.

[13] S. Agrawal, "Mitigating Cross-Site Request Forgery (CSRF) Attacks Using Reinforcement Learning and Predictive Analytics," *Appl. Res. Artif. Intell. Cloud Comput.*, vol. 6, no. 9, pp. 17–30, 2023.

[14] Darmanto, D., Muhammad, A. R., & Rustiarni, R. (2024). Analisis tingkat kesiapan keamanan informasi menggunakan indeks kami 4.2 pada Politeknik Negeri Ketapang. *Informasi Interaktif: Jurnal Informatika dan Teknologi Informasi*, *9*(1), 1-9.