BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 6, No 4, June 2026 | Hal 1593-1602
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v6i4.1186

Security Assessment of E-Commerce Website Using NIST SP 800-115 Based on OWASP Top 10

Arif Setyo Wibowo*, Henni Endah Wahanani, Andreas Nugroho Sihananto

Faculty of Computer Science, Informatic, Universitas Pembangunan Nasional "Veteran" Jawa Timur, Surabaya, Indonesia
Email: 1,*21081010057@student.upnjatim.ac.id, 2henniendah@upnjatim.ac.id, 3andreas.nugroho.jarkom@upnjatim.ac.id
Correspondence Author Email: 21081010057@student.upnjatim.ac.id

Abstract−The rapid growth of e-commerce platforms in Indonesia has increased the risk of cyber threats targeting sensitive user data, including personal information and payment details. PT. XYZ, a mattress company that recently launched its first e-commerce website, has attracted 42,222 visitors and generated revenue of Rp994,878,300 within its first six months, yet has never undergone any form of security testing. This raises serious concerns, as undetected vulnerabilities may expose the platform to identity theft, data breaches, and unauthorized access. This study aims to identify existing security vulnerabilities, determine the severity level of each finding, and provide concrete remediation recommendations before those vulnerabilities are exploited. The assessment was conducted using the NIST SP 800-115 framework across four phases: Planning, Discovery, Attack, and Reporting, with vulnerability classification based on OWASP Top 10 (2021). The Discovery phase utilized Google Dorking, WHOIS, wfuzz, Wappalyzer, Nmap, Burp Suite, and OWASP ZAP to gather intelligence and identify weaknesses. The Attack phase successfully exploited six confirmed vulnerabilities: Clickjacking, CSP Header Not Set, Vulnerable JS Library, Cross-Domain Misconfiguration, Source Code Disclosure, and Username Enumeration and Brute Force, mapped to OWASP categories A05, A06, and A07, with risk levels ranging from Medium to High. This research contributes by demonstrating that newly deployed platforms are not inherently secure and that integrating NIST SP 800-115 with OWASP Top 10 provides a structured approach to identifying real security vulnerabilities in e-commerce systems.

Keywords: OWASP TOP 10 2021; E-Commerce; Penetration Testing; NIST SP 800-115; Security Testing

1. INTRODUCTION

E-commerce is a trading system based on digital technology that enables buying and selling activities, exchange of goods or services conducted through the internet network without the need for face-to-face interaction [1]. This convenience has driven more and more businesses to shift to digital platforms, and e-commerce in Indonesia itself continues to grow rapidly in line with the increasing number of internet users and the changing digital lifestyle of society [2]. This growth is driven not only by advances in technological infrastructure, but also by the growing public trust in digital transactions, which are considered more practical, efficient, and accessible anytime and anywhere.

On the other hand, e-commerce platforms store large amounts of sensitive user data, ranging from names and addresses to payment information, which makes them a prime target. BSSN recorded 610.63 million cyber attack incidents throughout 2024 [3], reflecting the enormous number of threats lurking on digital platforms every day. E-commerce platforms are among the most vulnerable because they store a combination of valuable data simultaneously, ranging from user identities to financial transaction information, making them targets for various threats such as identity theft, malware attacks, phishing, and data breaches that can harm both consumers and service providers [4]. The impact of such security incidents is not only financial, but also damages company reputation and reduces consumer trust in the long term.

One of the main causes of successful attacks is vulnerabilities that go undetected from the early stages of system development. OWASP reported that nearly 94% of web applications tested had at least one form of access control weakness [5]. This fact confirms that security testing is no longer an optional activity, but a fundamental necessity for every platform that manages user data. Ironically, many companies that are new to the digital world still neglect the security aspect due to budget constraints or a lack of awareness of the existing risks.

PT. XYZ is a mattress company that has recently entered the digital world by launching its first e-commerce website. In the first six months, the platform successfully attracted 42,222 visitors with revenue of Rp994,878,300, yet to this day has never undergone any form of security testing. This situation is a serious concern given that e-commerce platforms that have not implemented adequate security mechanisms are potentially vulnerable to various cyber threats such as identity theft and customer data breaches [6]. Without a structured security assessment, critical vulnerabilities may have existed since the early stages of development and have never been identified to this day.

To address this problem, NIST SP 800-115, or the Technical Guide to Information Security Testing and Assessment published by the National Institute of Standards and Technology, provides a structured security testing framework through four main phases, namely Planning, Discovery, Attack, and Reporting [7]. This approach was selected because NIST SP 800-115 provides a method that can be directly applied in the execution of penetration testing, while also producing concrete improvement recommendation steps for the company [8]. To make the testing more targeted and focused on the most relevant threats, vulnerability categorization is carried out using OWASP Top 10 because it is more relevant to web application security compared to other frameworks, with a focus on the most commonly found vulnerabilities in the world [9].

Several previous studies have examined the security of e-commerce web applications using penetration testing. The first study, titled "Analisis Keamanan Aplikasi Berbasis Web Menggunakan Metode Penetration Testing (Studi Kasus: E-Commerce As-Sakinah Mart)" [10], analyzed the security level of the As-Sakinah Mart e-commerce web application using penetration testing with tools such as Nmap, Gobuster, SQLMap, and Burp Suite. The results showed

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 6, No 4, June 2026 | Hal 1593-1602
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v6i4.1186

that the website had met the OWASP Top 10-2021 standard for common technical vulnerabilities, but a critical Business Logic Vulnerability (CWE-840) with a CVSS Base Score of 7.7 (High) was found, which allowed attackers to manipulate product prices in the shopping cart during checkout. The study recommended implementing server-side price validation, cross-checking transaction data, and conducting regular security audits.

The second study, titled "Analisa Celah Keamanan dalam Pengembangan Website E-Commerce (Studi Kasus: Mataharimu.com)" [11], conducted penetration testing on Mataharimu.com using an External Testing approach through four stages: Audit, Scanning, Access, and Analysis. The study found three security vulnerabilities, namely Cross-Site Scripting (XSS), HTML Injection, and Directory Listing, all of which were caused by the absence of input sanitization and improper server configuration. The study recommended implementing the htmlspecialchars() function for all form inputs and configuring the .htaccess file to restrict unauthorized directory access.

The third study, titled "Analisis Kerentanan Aplikasi Web E-Commerce Berdasarkan Standar OWASP Top 10: Studi Kasus pada Situs Kopi Lampung Nusantara" [12], analyzed security vulnerabilities using OWASP ZAP with active and passive scanning. A total of 21 vulnerabilities were identified across four risk levels. The only high-risk finding was PII Disclosure, which exposed personal user data through unprotected GET requests. Medium-risk findings included Absence of Anti-CSRF Tokens and CSP Header Not Set, while low-risk findings were dominated by cookie misconfigurations. The study recommended implementing encryption, Anti-CSRF Tokens, CSP headers, and the comprehensive use of HTTPS across all pages.

The fourth study, titled "Penerapan Pentesting pada EasyCart untuk Menghadapi Ancaman Keamanan Siber" [13], conducted a systematic security evaluation on the EasyCart e-commerce application using a penetration testing method based on the Penetration Testing Execution Standard (PTES) with OWASP Top 10 as a reference. The testing successfully identified 12 vulnerabilities consisting of 5 high risks, 5 medium risks, and 2 low risks, covering SQL Injection, Broken Access Control, Cross-Site Scripting (XSS), and Identification and Authentication Failures. The study also conducted mitigation validation through a second round of testing that proved the critical vulnerabilities had been closed, with recommendations including the implementation of parameterized queries, input validation, strengthened authentication, and the addition of a Web Application Firewall (WAF).

The fifth study, titled "Analisis Keamanan Website Menggunakan Information System Security Assessment Framework (ISSAF)" [14], conducted penetration testing on a website using the ISSAF (Information Systems Security Assessment Framework) methodology. The study found an SQL Injection vulnerability in the cek_login.php file through the email parameter using the POST method, which allowed attackers to access the database, upload a backdoor, and gain full control over the server. The study recommended implementing strict input filtering and sanitization on all SQL queries and proper server configuration to prevent further exploitation.

Based on the five studies reviewed, penetration testing has been consistently proven effective in uncovering vulnerabilities across various e-commerce platforms, and OWASP Top 10 has been widely adopted as the primary categorization framework. However, several gaps remain unaddressed. Prior studies have predominantly relied on non-NIST methodologies such as PTES [13] and ISSAF [14], or applied OWASP-based testing without a structured procedural framework [11], [12], leaving the application of NIST SP 800-115 on e-commerce platforms underrepresented. More critically, none of the reviewed studies assessed a platform that had never undergone any prior security testing despite already being in active commercial operation with real user traffic and transaction data**.** This study fills that gap by conducting a structured security assessment on PT. XYZ e-commerce platform using NIST SP 800-115 and OWASP Top 10, supported by tools such as Nmap, Wappalyzer, WHOIS, Google Dorking, OWASP ZAP, Wfuzz, and Burp Suite, with the aim of identifying all existing security vulnerabilities, determining the severity level of each vulnerability, and providing concrete remediation recommendations that can be immediately implemented before those vulnerabilities are exploited by irresponsible parties.

2. RESEARCH METHODOLOGY

2.1 Research Flow

The research flow applied to the security testing of PT. XYZ e-commerce website is shown in Figure 1. The diagram presents the stages of the security assessment structured based on the NIST SP 800-115 methodology and mapped against the OWASP Top 10 2021 vulnerability categories. Each stage in the diagram reflects a direct application of the NIST SP 800-115 phases, from Planning through Reporting, while the OWASP Top 10 2021 framework serves as the classification standard used throughout the Discovery and Attack stages to categorize and prioritize identified vulnerabilities.

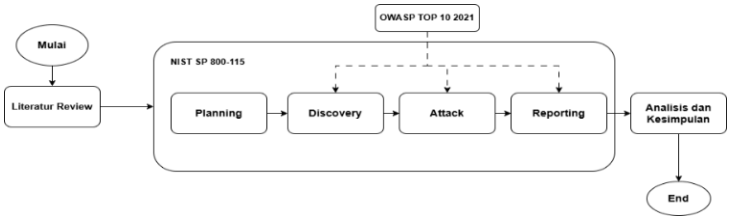

Figure 1. Research Flow

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 6, No 4, June 2026 | Hal 1593-1602
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v6i4.1186

Figure 1 illustrates the research flow. The first stage is a literature review to identify the theoretical foundation and references related to security assessment, NIST SP 800-115, and OWASP Top 10 2021. This is followed by the planning stage, where the testing scope, target system, rules of engagement, and official authorization from PT. XYZ are established to ensure a controlled assessment process.

The next stage is discovery, which consists of Information Gathering and Vulnerability Analysis. Information Gathering collects technical details such as domain names, IP addresses, and technologies used by the target system, while Vulnerability Analysis identifies and classifies potential security weaknesses in accordance with OWASP Top 10 2021. The process continues with the attack stage, where identified vulnerabilities are exploited in a controlled manner, prioritized based on OWASP risk categories. The research concludes with reporting, which documents all findings including OWASP classifications, risk levels, exploitation evidence, and security improvement recommendations. The final stage is analysis and conclusion, which summarizes the overall assessment results to address the research objectives.

2.2 Research Object

The object of this research is an e-commerce website owned by a company operating in the mattress industry. In this study, PT. XYZ has established rules to maintain the confidentiality of the company's identity. This platform is the first e-commerce website launched by PT. XYZ and, as of the time this research was conducted, has never undergone any form of security testing.

2.3 OWASP Top 10 2021

The OWASP Top 10 2021 is a list of the ten most critical web application security vulnerability categories published by the Open Web Application Security Project (OWASP), based on incident data from thousands of web applications worldwide [15]. Compared to the previous 2017 version, several significant changes were introduced in the 2021 edition, including the addition of three new categories: Insecure Design (A04), Software and Data Integrity Failures (A08), and Server-Side Request Forgery (A10), as shown in figure 2.

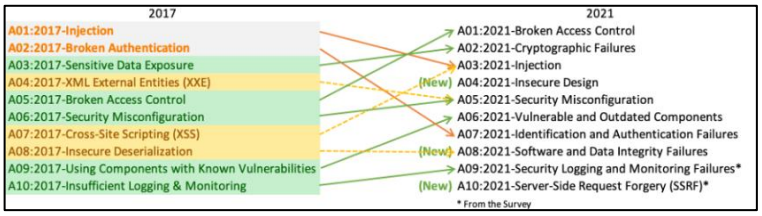

Figure 2. OWASP TOP 10 2021

This list is used as a reference framework in this study to classify and prioritize vulnerabilities discovered during the testing process on PT. XYZ website platform. The ten categories are presented in Table 1 as follows:

Table 1. OWASP TOP 10 Category

Kode	Number
A01	Broken Access Control
A02	Cryptographic Failures
A03	Injection
A04	Insecure Design
A05	Security Misconfiguration
A06	Vulnerable and Outdated Components
A07	Identification and Authentication Failures
A08	Software and Data Integrity Failures
A09	Security Logging and Monitoring Failures
A10	Server-Side Request Forgery (SSRF)

2.4 NIST SP 800-115

NIST SP 800-115, or the Technical Guide to Information Security Testing and Assessment, is a guide published by the National Institute of Standards and Technology (NIST) that provides a structured framework for information security testing [7]. This guide defines a security testing methodology that can be directly applied in the execution of penetration testing. NIST SP 800-115 consists of four main phases that complement one another, as shown in Figure 3.

Figure 3 presents the sequential flow of the NIST SP 800-115 phases, beginning with Planning and concluding with Reporting. The first phase is Planning, which involves defining the testing scope, objectives, rules of engagement, and official authorization from the target party. The second phase is Discovery, which encompasses the collection of technical information about the target system and the analysis of existing vulnerabilities. The third phase is Attack, where the identified vulnerabilities are exploited in a controlled manner to demonstrate the real-world impact of the security weaknesses. The fourth phase is Reporting, which contains documentation of all findings, exploitation evidence, risk

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 6, No 4, June 2026 | Hal 1593-1602
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v6i4.1186

levels, and security improvement recommendations. In addition, there is an Additional Discovery mechanism that allows the process to return to the Discovery phase if new information is found during the Attack phase.

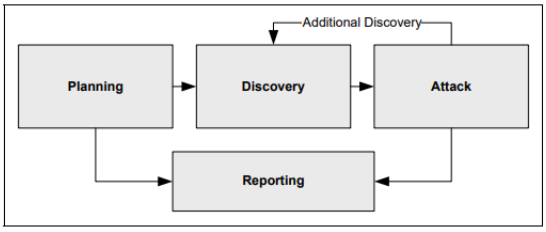

Figure 3. NIST SP 800-115 Phases

3. RESULT AND DISCUSSION

This section elaborates on the complete security assessment process performed against PT. XYZ e-commerce website, detailing the procedures, tools, findings, and evidence gathered at each phase of the NIST SP 800-115 framework, with all identified vulnerabilities classified in accordance with the OWASP Top 10 (2021) categories.

3.1 Planning

The Planning phase is the initial stage that establishes the legal and technical foundation of the entire security assessment process. This phase involves defining the testing scope, objectives, methodology, and Rules of Engagement (RoE) agreed upon with PT. XYZ. The details of the testing plan are presented in Table 2.

Table 2. Testing Plan

Parameter	Details
Target	Website E-Commerce PT. XYZ (xyz.com)
Testing Type	Black-Box Penetration Testing
Method	NIST SP 800-115 + OWASP Top 10 (2021)
Scope	Web application, HTTP headers, API endpoints, directories, open ports
Rules of Engagement	Excessive brute-force, user data access, and production system disruption are prohibited
Authorization	Official authorization letter from PT. XYZ dated per agreement

The planning phase produced a pre-engagement document covering the testing scope, action boundaries, and official authorization from PT. XYZ. The clear establishment of RoE serves as the ethical foundation of the assessment, ensuring that all testing activities are legally and professionally accountable.

3.2 Discovery Intelligence Gathering

The Discovery phase begins with Intelligence Gathering, a process of systematically collecting information about the target based on NIST SP 800-115 Section 3 regarding Information Gathering Techniques. In this sub-phase, various OSINT (Open Source Intelligence) techniques were employed to map the attack surface without performing active exploitation. The tools used along with their findings and security implications are summarized in Table 3.

Table 3. Intelligence gathering results

Tool	Finding	Security Implication
Google Dorking	Customer login page found via query site:xyz.com inurl:login	Login page publicly exposed through search engines
WHOIS Lookup	Domain xyz.com registered under PT Infinys System Indonesia, IP: 148.135.xxx.xxx, active until 2027	Registrar identity and infrastructure information can be leveraged by attackers
Wappalyzer	Framework: Laravel, Web Server: LiteSpeed, CDN: Cloudflare, Library: jQuery 1.12.4 and Bootstrap 4.1.3	Outdated library versions potentially carry recorded CVEs (prototype pollution, XSS)
wfuzz (Directory Fuzzing)	Found /back-auth (administrator login page) and /up (internal monitoring endpoint) publicly accessible without CAPTCHA	Admin panel exposed and vulnerable to brute force and enumeration attacks

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 6, No 4, June 2026 | Hal 1593-1602
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v6i4.1186

Tool	Finding	Security Implication
Nmap	Port 21 (FTP) open without encryption; Port 8888 running sun-answerbook (uncommon)	Open FTP risks data interception; uncommon port expands the attack surface
Burp Suite	Cookie lacks HttpOnly flag; Security headers (X-Frame-Options, CSP, HSTS) not found in server response	XSRF-TOKEN potentially stolen via XSS; server configuration does not follow security best practices

Table 3 summarizes the results of the Intelligence Gathering results reveal that the target system has several weak security conditions even before active exploitation is performed. The most critical finding is the administrator login page /back-auth, which is publicly accessible without any protection, along with the complete absence of security headers in the server response. The information obtained in this phase serves as the foundation for Vulnerability Analysis in the subsequent sub-phase.

3.3 Discovery Vulnerability Analysis

Vulnerability Analysis was conducted using OWASP ZAP as an automated scanner to systematically identify vulnerabilities. The scan was performed across all endpoints within the testing scope defined in the Planning phase. The results of the OWASP ZAP scan are shown in figure 4, which successfully identified 48 alerts of varying severity levels.

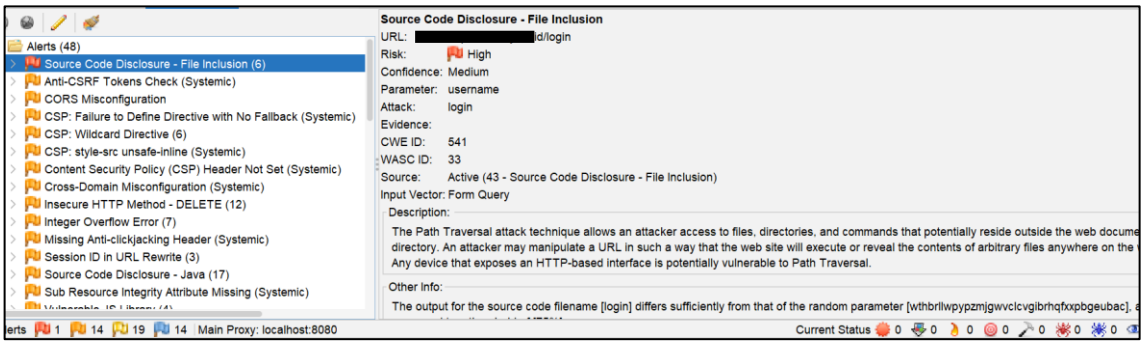

Figure 4. OWASP ZAP Scan Results

As shown in Figure 4, the automated scan produced 48 alerts across varying severity levels. Two additional findings from the manual scan of /back-auth, namely the exposed administrator login page and the username enumeration indication, were added as these were not captured by the automated scanner, bringing the total to 1 High, 16 Medium, and 20 Low findings. Each alert includes details such as risk level, confidence level, CWE-ID, vulnerability description, and remediation recommendations. Informational alerts were excluded as they do not represent directly exploitable vulnerabilities. All remaining findings were mapped to OWASP Top 10 2021 categories and serve as the basis for the exploitation phase.

3.4 Attack

The Attack phase was conducted in accordance with NIST SP 800-115 Section 5.2, where all exploitation activities were carried out in a controlled manner with written authorization. Six test scenarios were executed covering all vulnerabilities identified during the Vulnerability Analysis phase.

Clickjacking testing was performed using Burp Suite Clickbandit by injecting the script into the browser's DevTools Console. The xyz.com page was successfully loaded within an iframe without any resistance, confirming the absence of X-Frame-Options and Content-Security-Policy frame-ancestors headers, as shown in Figure 5.

Figure 5. Burp Clickbandit Script in DevTools Console

The test was concluded by pressing the finish button, which displayed a confirmation that the Clickjacking attack was successfully executed, as shown in Figure 6.

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 6, No 4, June 2026 | Hal 1593-1602
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v6i4.1186

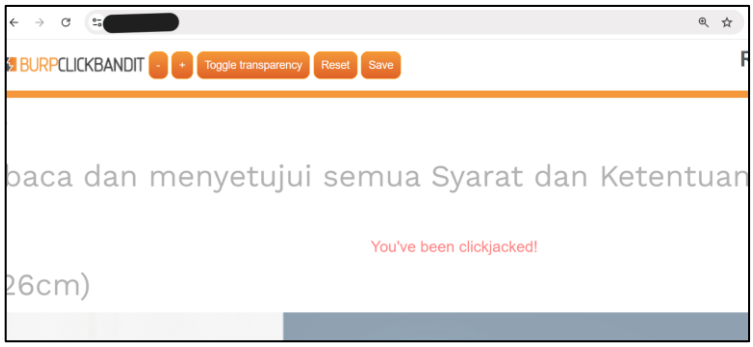

Figure 6. Burp Clickbandit Confirmation Result

CSP Header Not Set verification was conducted by comparing response headers between xyz.com and Shopee using browser DevTools. The target server returned no Content-Security-Policy, X-Frame-Options, or Strict-Transport-Security headers, leaving the application susceptible to malicious script injection.

The third test verified the Vulnerable JS Library vulnerability. Vulnerable JS Library testing confirmed through DevTools Console that the target uses jQuery 1.12.4 and Bootstrap 4.1.3, both carrying documented CVEs including CVE-2020-11022 and CVE-2019-8331, as shown in Figure 7. Direct exploitation was constrained by Laravel and Cloudflare protection layers, though these remain bypassable through advanced techniques.

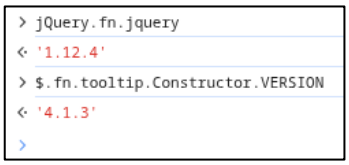

Figure 7. jQuery and Bootstrap Version Confirmation

Cross-Domain Misconfiguration was verified using Burp Suite Repeater by sending a request to /api/get-harga with the Origin header modified to an untrusted domain. The server returned Access-Control-Allow-Origin: *, confirming that all origins are permitted to access the API without restriction, directly exposing the endpoint to data exfiltration threats.

Source Code Disclosure was demonstrated by accessing a URL identified during the Vulnerability Analysis phase using an inappropriate HTTP method. The server displayed a Laravel debug error page exposing PHP 8.3.21, Laravel 11.46.1, internal middleware paths including app/Http/Middleware/TrackVisitor.php, and application source code at lines 27 through 34, providing attackers sufficient information to map the system architecture and search for relevant exploits.

Username Enumeration and Brute Force testing on /back-auth was conducted using OWASP ZAP Fuzzer and a Python script. Two wordlists containing common administrator usernames and email addresses were used as payloads against the username parameter on the login form. Manual verification revealed that an unregistered username returned the message "*User tidak ditemukan*", while the valid username superadmin@gmail.com with an incorrect password returned "*Password Salah*", confirming that the account exists in the system. This difference in error responses implicitly discloses valid account information to an attacker, as shown in Figure 8.

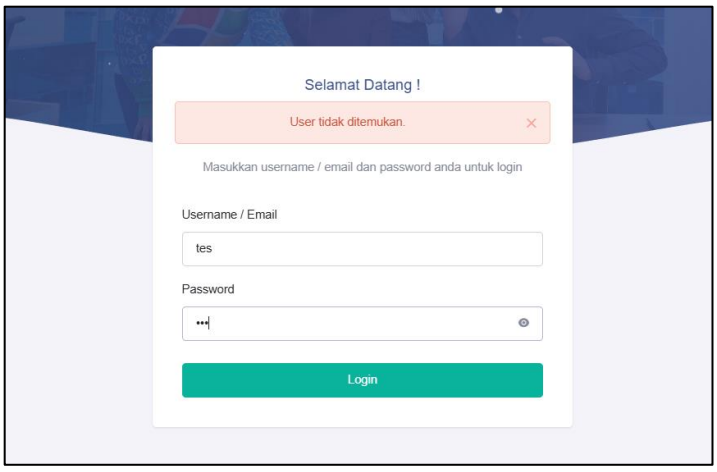

Figure 8. Error Message for Invalid Username

The second condition tested a valid username with an incorrect password, which returned a different error message confirming account existence, as shown in Figure 9.

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 6, No 4, June 2026 | Hal 1593-1602
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v6i4.1186

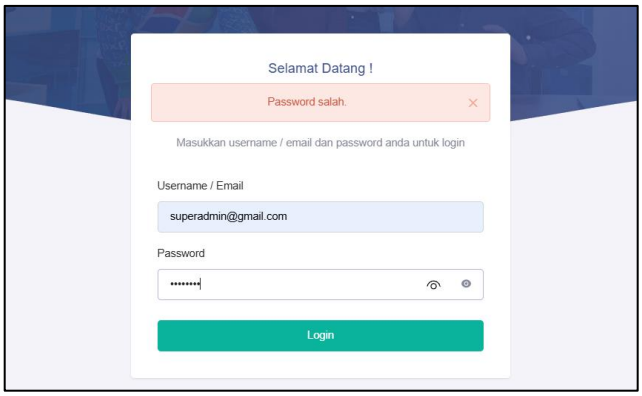

Figure 9. Error Message for Valid Username

Subsequent rate limiting testing via consecutive login attempts all returned HTTP 200 with no 429 response, as shown in Figure 10, proving the complete absence of login attempt restrictions. This combination of username exposure, no rate limiting, and no CAPTCHA makes this the highest-risk finding of the assessment.

Figure 10. Rate Limiting Test Results Using Python Script

3.5 Reporting

The Reporting phase documents all findings, exploitation evidence, impact analysis, and remediation recommendations based on the testing results from all previous phases. The report is intended as a basis for decision-making by PT. XYZ in improving system security. The complete remediation recommendations for each identified vulnerability are presented in Table 4.

Table 4. Remediation recommendations based on findings

Vulnerability	OWASP Top 10 Category	Remediation Recommendation	Priority
Username Enumeration & Brute Force	A07 - Identification and Authentication Failures	Standardize login error messages to be generic, implement rate limiting (5x/minute per IP) using Laravel RateLimiter, and add Google reCAPTCHA to the /back-auth page	High
Clickjacking	A05 - Security Misconfiguration	Domain xyz.com registered under PT Add SecurityHeaders middleware to insert X-Frame-Options: DENY header and frame-ancestors 'none' directive in every server response	Medium
CSP Header Not Set	A05 - Security Misconfiguration	Implement a complete Content-Security-Policy with the default-src 'self' directive, and enable the HttpOnly and Secure flags on session cookies in config/session.php	Medium
Vulnerable JS Library	A06 - Vulnerable and Outdated Components	Update jQuery to version 3.7.1 and Bootstrap to version 5.3.3 via NPM or CDN with integrity and crossorigin attributes for SRI	Medium
Cross-Domain Misconfiguration	A05 - Security Misconfiguration	Replace the CORS wildcard (*) configuration with an approved origin list (xyz.com, www.xyz.com) in config/cors.php of Laravel	Medium
Source Code Disclosure	A05 - Security Misconfiguration	Disable debug mode (APP_DEBUG=false, APP_ENV=production) in the .env file and create a custom error page that does not expose technical information	Medium

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 6, No 4, June 2026 | Hal 1593-1602
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v6i4.1186

3.6 Discusion

The results of this security assessment on PT. XYZ e-commerce website confirm that web application platforms, particularly those that have never undergone prior security testing, carry significant and exploitable vulnerabilities from the outset. The six vulnerabilities identified in this study, namely Clickjacking, CSP Header Not Set, Vulnerable JS Library, Cross-Domain Misconfiguration, Source Code Disclosure, and Username Enumeration & Brute Force, reflect a pattern of inadequate security configuration that is consistent with findings reported across multiple recent studies using comparable methodologies.

The most critical finding in this study is Username Enumeration & Brute Force on the /back-auth administrator login page, classified under A07 – Identification and Authentication Failures with a High severity rating. This vulnerability arose from the combination of differential error messages between valid and invalid usernames, the complete absence of rate limiting confirmed by zero HTTP 429 responses across consecutive login attempts, and the lack of CAPTCHA protection. A penetration testing study on a government office website that had previously experienced a breach in 2023 using the same NIST SP 800-115 framework found two critical vulnerabilities: CSRF with a CVSS score of 9.3 classified as Critical under A05 – Security Misconfiguration, and XSS with a CVSS score of 8.2 classified as High under A03 – Injection [16]. The parallel between both studies confirms that unprotected session-handling and input validation surfaces consistently yield the highest-severity findings in web penetration assessments. An earlier vulnerability assessment of two e-government websites using NIST SP 800-115 combined with OWASP similarly demonstrated that combining both frameworks with automated scanning tools produces consistent threat level categorizations across different web targets, even when scanner tools differ in speed and type of findings [17]. This supports the methodological choice in this study to use NIST SP 800-115 as the procedural structure while applying OWASP Top 10 as the classification standard.

A dominant pattern observable in this study's findings is that four out of six identified vulnerabilities fall under A05 – Security Misconfiguration, including Clickjacking due to missing X-Frame-Options header, CSP Header Not Set, Cross-Domain Misconfiguration from wildcard CORS configuration, and Source Code Disclosure caused by Laravel debug mode being enabled in a production environment. This concentration under a single OWASP category indicates a systemic issue rather than isolated developer errors: the underlying framework and server were deployed without applying basic security hardening measures. This observation is strongly consistent with a VAPT study based on OWASP Top Ten that identified Clickjacking as a successfully exploited vulnerability under A05 – Security Misconfiguration, caused by the absence of X-Frame-Options and Content Security Policy (CSP) configurations, and also found Missing Anti-Clickjacking Header as a medium-severity finding using Acunetix and OWASP ZAP as scanning tools [18]. Both studies demonstrate that missing security header configurations are a widespread and recurring problem in web platforms, and that fixing them through server-side configuration changes can address multiple OWASP findings simultaneously without requiring significant architectural changes.

Regarding the Vulnerable JS Library finding, this study confirmed through browser DevTools that the target uses jQuery 1.12.4 and Bootstrap 4.1.3, both carrying documented CVEs including CVE-2020-11022 and CVE-2019-8331, classified under A06 – Vulnerable and Outdated Components. A closely related finding appears in penetration testing research on the Astonprinter.com website using NIST SP 800-115, which identified Vulnerable JS Library as a medium-severity vulnerability among 20 total exploitable vulnerabilities found, with the two highest-severity findings being DNS Server Spoofed Request Amplification DDoS and Path Traversal [19]. Both studies confirm that outdated JavaScript libraries represent a persistent medium-severity risk in web systems whose dependency stacks are not regularly audited, and that the NIST SP 800-115 framework is effective in surfacing this category of vulnerability through its structured Discovery phase.

Regarding the overall effectiveness of the methodology, a comparative study evaluating OWASP, PTES, and NIST SP 800-115 across DVWA and OWASP Juice Shop environments found a clear trade-off between the three frameworks: OWASP proved most efficient with an average of 85 minutes and 59 total vulnerabilities detected, PTES demonstrated the greatest technical depth with 63 vulnerabilities and the highest severity scores, while NIST SP 800-115 detected 49 vulnerabilities and excelled specifically in compliance and risk management integration [20]. This study's approach of pairing NIST SP 800-115 as the procedural backbone with OWASP Top 10 as the vulnerability classification system is designed precisely to capture the regulatory accountability strength of NIST while compensating for its lower raw detection count through the addition of OWASP's application-focused classification categories. The black-box approach adopted here further confirms that meaningful vulnerabilities including High-severity findings can be discovered without internal access to source code or system credentials, which aligns with the broader conclusion that NIST SP 800-115 provides a reliable and replicable structure regardless of testing modality [16], [18], [20].

Beyond comparing vulnerability types, a notable distinction between this study and prior work lies in the target profile. The studies referenced in this discussion each assessed platforms that were already in active operation: an e-government website that had experienced a confirmed breach [16], two local government websites selected specifically because one had previously shown negative security indicators [17], and a commercial website evaluated after deployment [18]. This study, in contrast, targets a newly launched e-commerce platform that had never undergone any form of security testing. The discovery of six exploitable vulnerabilities including a High-severity finding within a platform that had already attracted 42,222 visitors and generated Rp994,878,300 in revenue demonstrates that newly deployed commercial platforms are not inherently safer than mature platforms, and that the absence of prior testing does not imply the absence

of vulnerabilities. This underscores the necessity of embedding security testing into the development lifecycle rather than treating it as a reactive measure post-deployment, a principle consistently supported across the referenced studies [16], [17], [18], [19], [20].

4. CONCLUSION

This research conducted a security assessment on PT. XYZ e-commerce website using NIST SP 800-115 and OWASP Top 10 (2021) across four phases: Planning, Discovery, Attack, and Reporting. Six vulnerabilities were successfully identified and exploited in a controlled manner, consisting of Clickjacking, CSP Header Not Set, Vulnerable JS Library, Cross-Domain Misconfiguration, and Source Code Disclosure at Medium severity, as well as Username Enumeration & Brute Force at High severity. A notable pattern from the findings is that four out of six vulnerabilities fall under A05 - Security Misconfiguration, pointing to a common underlying issue where the server and framework were not configured with security in mind from the start. This also means that fixing the missing security headers alone would address multiple vulnerabilities at once without requiring significant architectural changes. The combination of NIST SP 800-115 and OWASP Top 10 proved effective throughout this research. NIST provided the procedural structure that kept the assessment organized and ethical, while OWASP gave a reliable classification system that made the findings easy to communicate regardless of the audience's technical background. This research has its limitations. The black-box approach and strict Rules of Engagement meant that some exploitation scenarios could not be fully pursued, and the scope did not cover the internal network, payment gateway, or mobile application layer. Future research would benefit from a grey-box or white-box approach, PCI-DSS integration for payment security, and broader infrastructure coverage.

REFERENCES

[1] Munir Azhari and Breda Kustanto, "Penerapan E-Commerce Dalam Meningkatkan Daya Saing Ekonomi Di Era Teknologi Informasi," *Aliansi : Jurnal Manajemen dan Bisnis*, vol. 19, no. 2, pp. 193–210, Dec. 2024, doi: https://doi.org/10.46975/ka212950.

[2] D. I. Maharani, "Pages 201-210 Perekonomian di Era Transformasi Digital," *Jurnal Simki Economic*, vol. 7, no. 1, pp. 201–210, Jan. 2024, doi: https://doi.org/10.29407/jse.v7i1.493.

[3] Nouvan, "Jumlah Serangan Siber di Indonesia Naik, Tembus 610 Juta pada 2024," Dataloka. Accessed: Jun. 29, 2026. [Online]. Available: https://dataloka.id/politik/4259/jumlah-serangan-siber-di-indonesia-naik-tembus-610-juta-pada-2024/

[4] Nia Ramadhani and Muhammad Irwan Padli Nasution, "Tantangan Dan Solusi Keamanan Siber Dalam Transaksi E-Commerce," *JURNAL PENELITIAN SISTEM INFORMASI (JPSI)*, vol. 2, no. 2, pp. 134–144, May 2024, doi: https://doi.org/10.54066/jpsi.v2i2.1930.

[5] OWASP (Open Worldwide Application Security Project), "A01 Broken Access Control - OWASP Top 10:2021," OWASP.

[6] R. Ernitasari, S. Ramdhani, and M. Gowon, "Keamanan E-Commerce: Cara Menjaga Data Pribadi Tetap Aman," *JURNAL SIBER MULTI DISIPLIN*, no. 3, pp. 213–223, Dec. 2025, doi: https://doi.org/10.38035/jsmd.v3i3.

[7] K. Scarfone, M. Souppaya, A. Cody, and A. Orebaugh, "Technical Guide to Information Security Testing and Assessment ," *NIST Special Publication (SP) 800-115*, Sep. 2008, doi: https://doi.org/10.6028/NIST.SP.800-115.

[8] H. Jati Setyadi and M. Rivani Ibrahim, "Penetration Testing Website E-Journals Metode NIST SP 800-115 dan OWASP," *METIK JURNAL*, vol. 9, no. 1, pp. 72–81, Jun. 2025, doi: https://doi.org/10.47002/metik.v9i1.1030.

[9] R. Al Ridwan Bintang Firdaus and T. Ismardiko Widyawan, "Pengujian Kerentanan Website Menggunakan Metode Penetration Testing Dengan OWASP (Studi Kasus : Pemerintah Kabupaten Semarang)," *CyberSecurity dan Forensik Digital*, vol. 8, no. 2, pp. 106–115, Dec. 2025, doi: https://doi.org/10.14421/csecurity.2025.8.2.5372.

[10] M. Yuwan, S. Nacikit, M. Rahman, and A. E. Wardoyo, "Analisis Keamanan Aplikasi Berbasis Web Menggunakan Metode Penetration Testing (Studi kasus: E-Commerce As-Sakinah Mart)," *Informatics for Educators And Professionals : Journal of Informatics*, vol. 10, no. 1, pp. 25–33, Apr. 2025, doi: https://doi.org/10.51211/itbi.v10i1.3324.

[11] M. Fierza and E. Erlangga, "Analisa Celah Keamanan Dalam Pengembangan Website E-Commerce (Studi Kasus: Mataharimu.Com)," *Jurnal Ilmiah Computing Insight*, vol. 3, no. 2, p. 20, Aug. 2022, doi: https://doi.org/10.30651/comp_insight.v3i2.14535.

[12] A. R. Supriyatna, I. Asrowardi, S. D. Putra, and E. Subyantoro, "Analisis Kerentanan Aplikasi Web E-commerce Berdasarkan Standar OWASP Top 10: Studi Kasus pada Situs Kopi Lampung Nusantara," *EXPERT: Jurnal Manajemen Sistem Informasi dan Teknologi*, vol. 14, no. 2, p. 95, Dec. 2024, doi: https://doi.org/10.36448/expert.v14i2.4034.

[13] M. Fahrul Reza and I. Sutanto, "Penerapan Pentesting pada EasyCart untuk Menghadapi Ancaman Keamanan Siber," *IKRAITH-INFORMATIKA*, vol. 10, no. 1, pp. 96–105, Nov. 2025, doi: https://doi.org/10.37817/ikraith-informatika.v9i3.

[14] H. Herman, I. Riadi, Y. Kurniawan, and I. A. Rafiq, "Analisis Keamanan Website Menggunakan Information System Security Asessment Framework(ISSAF)," *Jurnal Teknologi Informatika dan Komputer*, vol. 9, no. 1, pp. 126–136, Mar. 2023, doi: https://doi.org/10.37012/jtik.v9i1.1439.

[15] F. T. Vierino, H. E. Wahanani, and A. Junaidi, "Evaluating Web Application Security Using OWASP Top 10 and NIST SP 800-115," *bit-Tech*, vol. 8, no. 3, pp. 3754–3764, Apr. 2026, doi: https://doi.org/10.32877/bt.v8i3.3702.

[16] Ilham Akbar, Khairunnisak Nur Isnaini, and Banu Dwi Putranto, "Penetration Testing Through NIST SP 800-115 and OWASP TOP 10 With Risk Analysis Using CVSS on the XY Diskominfo Website," *Journal of Innovation Information Technology and Application (JINITA)*, vol. 7, no. 2, pp. 314–326, Dec. 2025, doi: https://doi.org/10.35970/jinita.v7i2.2907.

[17] E. Z. Darojat, E. Sediyono, and I. Sembiring, "Vulnerability Assessment Website E-Government dengan NIST SP 800-115 dan OWASP Menggunakan Web Vulnerability Scanner," *JURNAL SISTEM INFORMASI BISNIS*, vol. 12, no. 1, pp. 36–44, Sep. 2022, doi: https://doi.org/10.21456/vol12iss1pp36-44.

[18] D. Rohmaniah, W. Miftahul Ashari, and A. Dwi Putra, "Enhancing Website Security Using Vulnerability Assessment and Penetration Testing (VAPT) Based on OWASP Top Ten," *Journal of Applied Informatics and Computing (JAIC)*, vol. 9, no. 2, pp. 404–411, Mar. 2025, doi: https://doi.org/10.30871/jaic.v9i2.9069.

[19] A. Agustinus and I. Sembiring, "Website Vulnerability Testing Using The Penetration Testing Method Referring To Nist Sp 800 – 155 (Case Study (Astonprinter.com Domain))," *Jurnal Teknik Informatika (Jutif)*, vol. 5, no. 6, pp. 1651–1662, Dec. 2024, doi: https://doi.org/10.52436/1.jutif.2024.5.6.3859.

[20] M. Bunan Imtias, K. Umam, H. Mustofa, and M. Hadi Subowo, "Comparative Analysis of Penetration Testing Frameworks: OWASP, PTES, and NIST SP 800-115 for Detecting Web Application Vulnerabilities," *Journal of Applied Informatics and Computing (JAIC)*, vol. 9, no. 6, p. 3689, Dec. 2025, doi: https://doi.org/10.30871/jaic.v9i6.9846.