Final Security Assessment Report

Project: FUTURE_CS_01 — Web Application Vulnerability Assessment

Author: Yehya Hamdy Shehata
Contact: yehya.hamdy1111@gmail.com — https://www.linkedin.com/in/yehya-shehata/
Portfolio: https://decisive-catshark.super.site/

Date of assessment: (5 October 2025)

Executive Summary

This document summarizes the findings from a hands-on vulnerability assessment performed as part of an internship exercise. The assessment targeted a web application deployed for learning purposes (test environment) and focused on common web vulnerabilities mapped to the OWASP Top 10. Four verified issues are included in this report: SQL Injection (authentication bypass), DOM XSS (search), Broken Access Control (credential brute-force), and Security Misconfiguration (exposed /ftp). Each finding includes impact, evidence references, remediation recommendations, and OWASP mapping.

Overall risk: The combination of an authentication bypass and exposed admin credentials constitutes a high-to-critical risk to the application. Immediate remediation is recommended for authentication-related issues and exposed sensitive files.

Scope

- **Target:** (insert application name / URL / environment)
- **Testing window:** (insert date range)
- **Tools used:** Burp Suite Community Edition, Browser (DevTools), Kali Linux (optional), manual testing techniques.
- **Excluded:** No destructive testing or interaction with production systems. All testing performed in a controlled/test environment.

Methodology

1. Reconnaissance: explored public endpoints, reviewed robots.txt, and identified input parameters (e.g., q in search).
2. Manual testing: used Burp Suite proxy to intercept and manipulate requests, attempted SQLi payloads and DOM payloads, and validated server responses.
3. Focused exploitation: where vulnerabilities were confirmed, collected PoC artifacts (screenshots, request/response captures, Burp saved items).
4. Documentation: sanitized and stored artifacts in PoC folders and compiled findings into this final report.

Findings (detailed)

1) SQL Injection — Login (Authentication Bypass)

- **Severity:** Critical / High
- **Description:** The login endpoint is vulnerable to SQL injection. A tautology payload injected into the username field allowed authentication bypass and retrieval of an admin email address.
- **Evidence (PoCs):** SQLi Login Page/PoCs/Request.txt, SQLi Login Page/PoCs/Response.txt, SQLi Login Page/PoCs/Screenshot_Browser_Admin.png, SQLi Login Page/PoCs/Screenshot_BurpSuite.png, SQLi Login Page/PoCs/Authentication_bypass (Burp saved item).
- **Impact:** Full account takeover, data exposure, unauthorized administrative actions.
- **OWASP mapping:** A03:2021 — Injection (SQLi).
- **Remediation:** Use parameterized queries / prepared statements, validate and sanitize inputs, apply least-privilege DB accounts, suppress detailed DB errors, and re-test after fixes.

---

2) DOM XSS — Search Parameter

- **Severity:** Medium — High (context dependent)
- **Description:** The q search parameter is reflected into the DOM unsafely, allowing execution of attacker-supplied JavaScript via a DOM XSS vector. Non-script

payloads (e.g., <img src=x onerror=alert("XSS")>) succeeded where direct <script> tags did not.

- **Evidence (PoCs):** DOM XSS Search/PoCs/DOM_XSS_Payload_Screenshot.png, DOM XSS Search/PoCs/Result_of_Payload_Screenshot.png.
- **Impact:** Client-side code execution in victim browsers, potential cookie/session theft, UI manipulation, and phishing.
- **OWASP mapping:** A03:2021 — Injection (client-side / script injection), guidance from XSS prevention rules.
- **Remediation:** Properly encode/sanitize data inserted into the DOM, avoid innerHTML with untrusted data, use textContent or safe DOM APIs, implement CSP, and add client-side XSS testing in CI.

---

3) Broken Access Control — Credential Brute-force

- **Severity:** High
- **Description:** After obtaining an admin email (via SQLi), a targeted brute-force attack using a ~100-entry public password list and Burp Suite Community Edition identified the admin password admin123.
- **Evidence (PoCs):** Broken Access Control/PoCs/Intruder_Grep-Match.png, Broken Access Control/PoCs/Intruder_Password_BruteForce.png, Broken Access Control/PoCs/Intruder_Sucess_Request.png, Broken Access Control/PoCs/Intruder_Sucess_Response.png. Passwordlist stored in Broken Access Control/Passwordlist/.
- **Impact:** Account compromise, privilege escalation, administrative access.
- **OWASP mapping:** A01:2021 — Broken Access Control; related to authentication failures (A07/A02).
- **Remediation:** Enforce rate-limiting and progressive delays, account lockouts/alerts, MFA for privileged accounts, strong password policies and denylists, and bot/credential-stuffing protections (CAPTCHA, WAF).

---

4)Security Misconfiguration — Exposed /ftp Directory

- **Severity:** High
- **Description:** robots.txt lists /ftp as disallowed, but the directory was publicly accessible. Directory listings and confidential files were retrievable without authentication.
- **Evidence (PoCs):** Security Misconfiguration/PoCs/robots_file_screenshot.png, Security Misconfiguration/PoCs/access_to_forbidden_folder.png, Security Misconfiguration/PoCs/access_to_forbidden_files.png, Security Misconfiguration/PoCs/acess_confidential_data.png.
- **Impact:** Sensitive data exposure, information leakage useful for further attacks, regulatory & reputational risk.
- **OWASP mapping:** A05:2021 — Security Misconfiguration.
- **Remediation:** Remove sensitive files from web-accessible locations, disable directory listing, enforce authentication/authorization for sensitive directories, fix file permissions, and avoid relying on robots.txt for protection.

Risk Summary & Prioritization

Finding	Likelihood	Impact	Priority
SQL Injection (login)	High	Critical	Immediate (Fix now)
Broken Access Control (brute-force)	Medium	High	High (within days)
Security Misconfiguration (/ftp)	Medium	High	High (within days)
DOM XSS (search)	Medium	Medium-High	Medium (patch soon)

Recommendation: Prioritize the SQLi fix and credential protections first, then lock down exposed directories and remediate DOM XSS.

OWASP Top 10 Checklist (brief)

- A01:2021 – Broken Access Control — Covered by finding #3.
- A03:2021 – Injection — Covered by finding #1 (SQLi) and client-side injection in finding #2.
- A05:2021 – Security Misconfiguration — Covered by finding #4.
- Further items: recommend broad scan for other OWASP categories.

Deliverables Included

- Final report (this document) — add to Final Report/ as PDF/Markdown.
- PoC artifacts (sanitized) — kept in each vulnerability folder under PoCs/.
- OWASP checklist and remediation recommendations — included in this report.
- (Optional) Video walkthrough — not included; can be added upon request.

Appendices

A — Reproduction notes (general)

- Use Burp Suite Community as proxy to capture traffic.
- Sanitize any session tokens before storing artifacts.
- Do not run tests against production systems without permission.

B — Artifact index

- SQLi Login Page/PoCs/*
- DOM XSS Search/PoCs/*
- Broken Access Control/PoCs/* + Passwordlist/
- Security Misconfiguration/PoCs/*

Conclusion

This assessment identified multiple high-impact vulnerabilities that together pose a critical threat to the application's confidentiality and integrity. Immediate remediation of SQL injection and credential protections is strongly recommended, followed by configuration hardening and client-side sanitization. The artifacts in the repository support these conclusions and can be used to validate fixes.

Contact & Next Steps

For help implementing remediations or running follow-up verification scans, contact:
Yehya Hamdy Shehata — yehya.hamdy1111@gmail.com

Suggested filename: Final_Report_FUTURE_CS_01.md
Suggested commit message: docs: add final security assessment report and artifacts

End of report