SECURITY AUDIT & PENETRATION TESTING REPORT

Project: CyberShield Intern Portal V3

Date: December 30, 2025

Tester: Syed Zunair Hussain

1. Executive Summary

This report documents the final penetration testing phase of the CyberShield V3 application. The objective was to verify the effectiveness of the security controls implemented during the development phase. Testing focused on the OWASP Top 10 vulnerabilities, specifically SQL Injection (SQLi), Cross-Site Scripting (XSS), and Cross-Site Request Forgery (CSRF), alongside infrastructure defenses like Rate Limiting.

2. Methodology

The test was conducted using a "Gray Box" approach.

Tools Used: Burp Suite Professional (Interception & Repeater), Manual Code Review, and Browser DevTools.

Environment: Node.js Express server with SQLite3 (In-memory).

3. Vulnerability Analysis & Test Results

3.1 SQL Injection (SQLi)

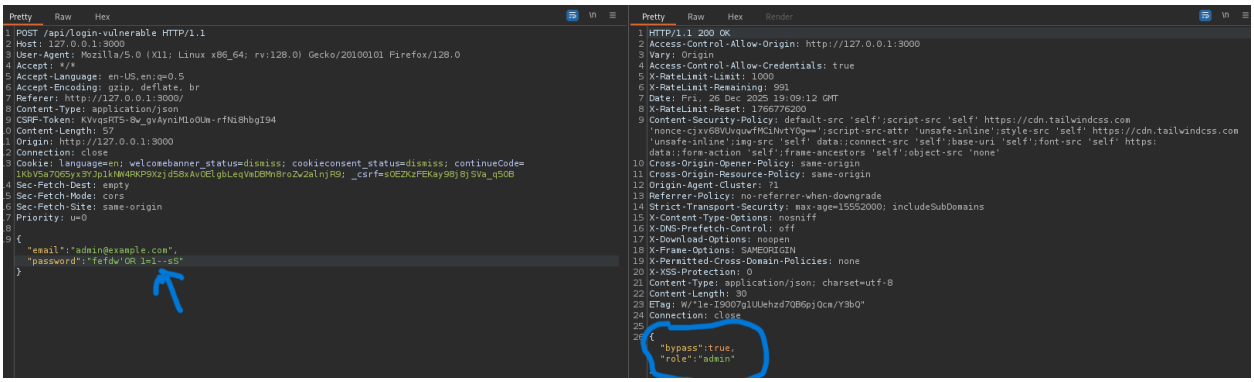
Attack Vector: Manipulating the email field in the authentication form to bypass login logic.

Initial Finding: The /api/login-vulnerable endpoint was confirmed to be vulnerable. Using the payload admin@example.com' -- allowed access without a password.

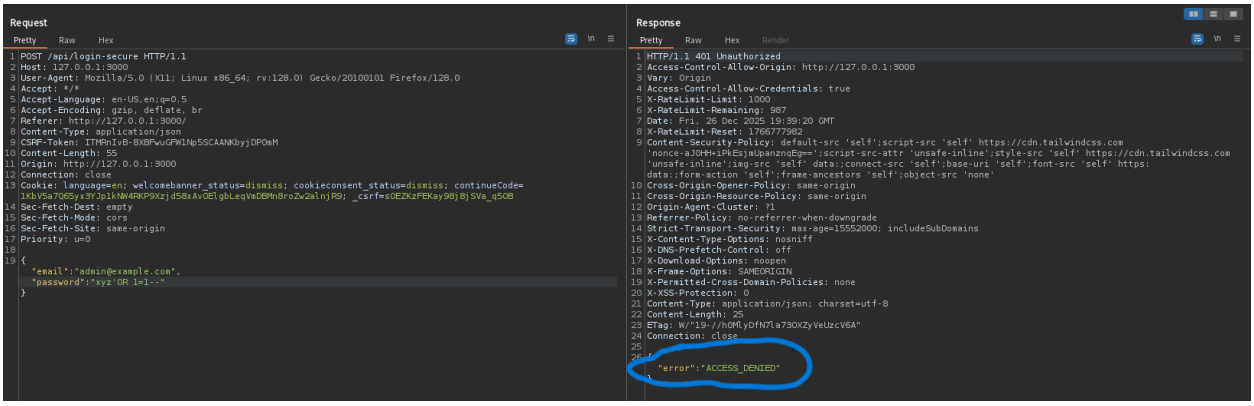
Security Fix: Implemented Parameterized Queries in the /api/login-secure endpoint.

Result: PASSED. The secure endpoint treats input as literal data. SQLi attempts now result in a 401 Unauthorized response.

Here is the image of successful sqli injection on vulnerable path:



Here Is the image of secure path:



3.2 Cross-Site Scripting (XSS)

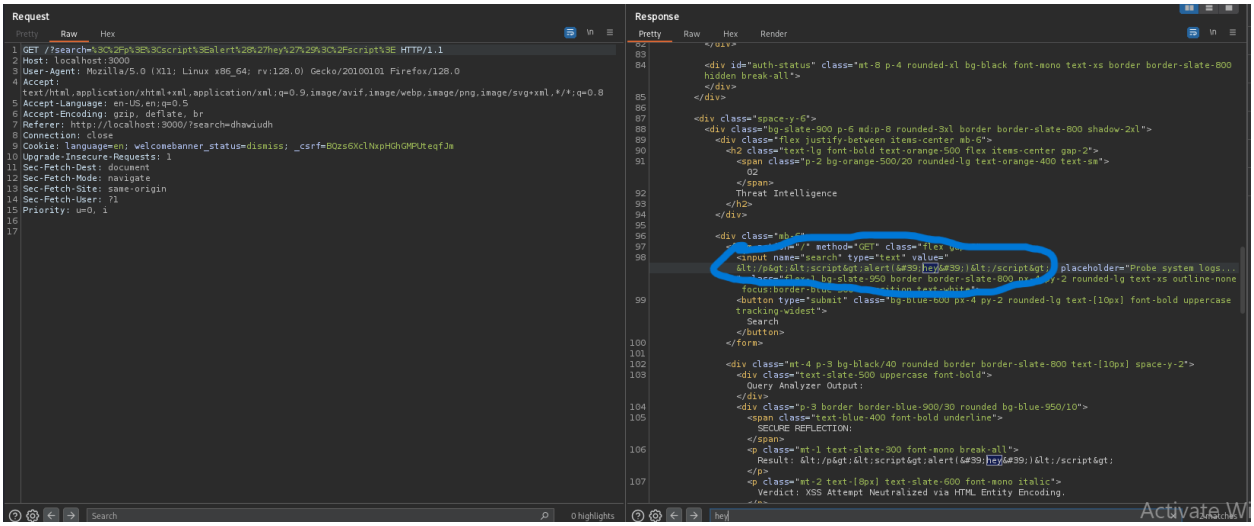
Attack Vector: Injecting <script> tags into the Search query parameter.

Finding: The search feature reflects user input back to the page.

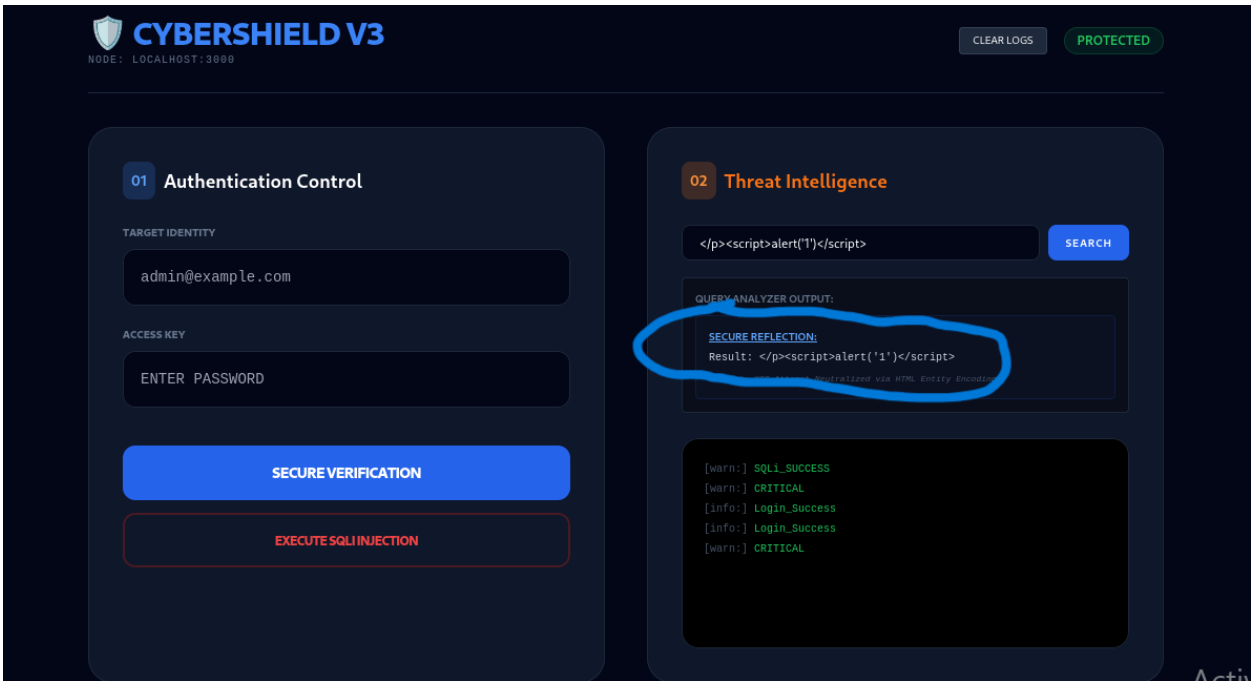
Security Fix: 1. Implemented a custom escapeHTML function to encode special characters. 2. Deployed Content Security Policy (CSP) via Helmet.js to block unauthorized inline scripts.

Result: PASSED. Malicious scripts are rendered as plain text and do not execute in the browser.

Here is the image of user input being sanitized :



Here is the image of user payload being shown as text :



3.3 Cross-Site Request Forgery (CSRF)

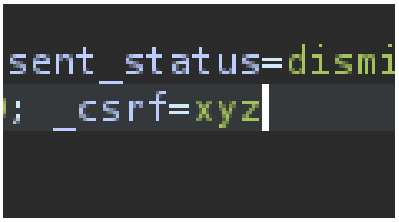
Attack Vector: Attempting to trigger the /api/clear-logs action from an external source or without a valid session token.

Finding: Without protection, a simple POST request could wipe system logs.

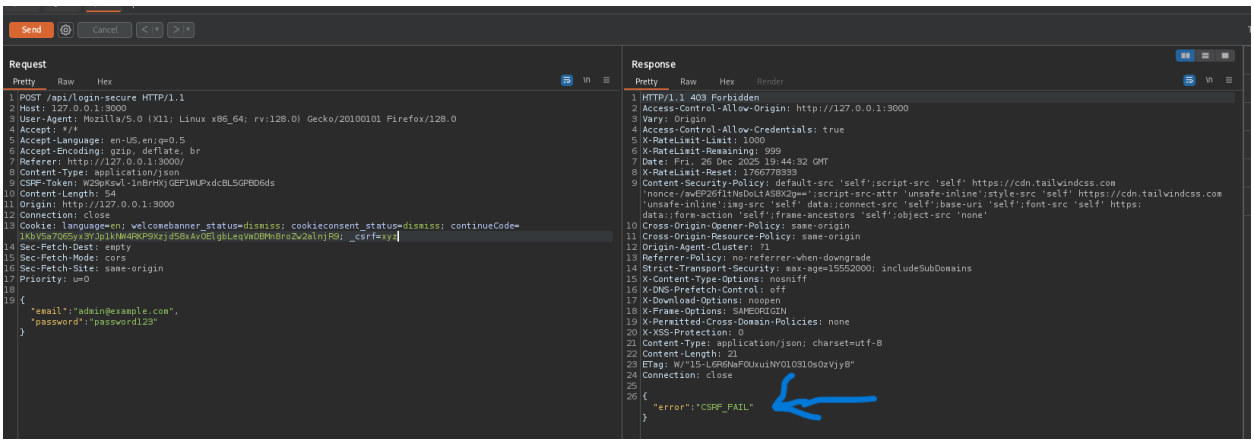
Security Fix: Integrated csurf middleware. The server now requires a valid CSRF-Token in the request header, matched against a secure, HttpOnly cookie.

Result: PASSED. Requests intercepted via Burp Suite and modified (token removed) were rejected with a 403 Forbidden error.

Here is the image of request with random csrf token :



The response:

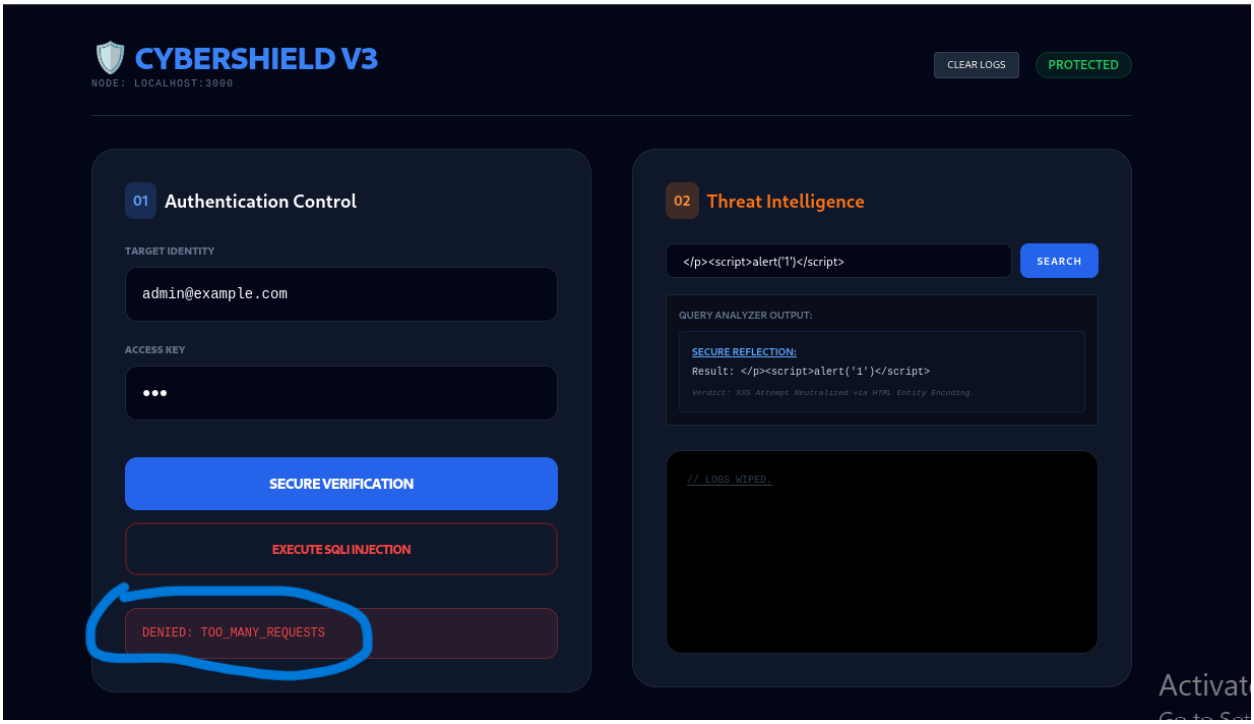


3.4 Rate Limiting & Brute Force Defense

Attack Vector: Automated rapid-fire requests to the login endpoint.

Security Fix: Implemented express-rate-limit targeting the /api/ route to prevent automated attacks.

Result: PASSED. Excessive requests triggered a 429 Too Many Requests status, protecting the server from resource exhaustion and brute force.



4. Security Improvements Summary

Below is a summary of the hardened components implemented during this audit:

Database Security: Verified Parameterized Queries (Prepared Statements) for all user-input fields.

Header Protection: Helmet.js active (HSTS, CSP, and X-Content-Type protection).

Authentication: Bcrypt password hashing implemented with 10 salt rounds.

Session Management: CSRF protection and HttpOnly cookies enabled for all session-sensitive routes.

Audit Logging: Winston Logger active, creating a persistent trail of all access and attack attempts.

5. Conclusion

The CyberShield V3 application has successfully passed the final penetration test. All critical vulnerabilities identified in the initial assessment have been mitigated through secure coding practices and middleware implementation. The application is now considered "Hardened" and ready for deployment.