Cybersecurity Internship Task Report

1. Setup

Installed and configured WebGoat locally, an intentionally vulnerable web application provided by OWASP.

OWASP ZAP was used as the proxy and vulnerability scanner. The browser was configured to use localhost:8080 for traffic interception.

2. Vulnerability Analysis with OWASP ZAP

WebGoat was scanned using OWASP ZAP. The following vulnerabilities were identified:

- SQL Injection (Login Module)
- Reflected Cross-Site Scripting (XSS)
- Cross-Site Request Forgery (CSRF)

Each vulnerability was tested manually to demonstrate exploitation.

3. SQL Injection

The login form of WebGoat was tested using the payload: ' OR '1'='1

Result: Login bypassed, proving SQL Injection is successful.

4. Reflected XSS

Reflected XSS payload used: <script>alert(1)</script>

The payload was passed via the query string and successfully triggered a popup alert.

5. CSRF

A crafted HTML form was used to simulate a CSRF attack. When executed, it triggered an unwanted state change (fund transfer).

6. OWASP ZAP Findings

Screenshots and findings from ZAP have been recorded showing vulnerability alerts, request-response pairs, and parameter evidence.

7. Conclusion

The exercise demonstrated how common vulnerabilities such as SQL Injection, XSS, and CSRF can be identified using automated tools (OWASP ZAP) and exploited manually.
Understanding these flaws is crucial in securing real-world applications.