Web Application Security Testing
Task-1

Internship Report:

Submitted by:
Priyanshu Shekhar Choudhary

Role: Cyber Security Intern
Organization: Future Interns
Month: July 2025

1. Task Overview

As part of my cybersecurity internship at Future Interns, I was assigned

Task 1: Web Application Security Testing. The task involved testing a sample web application (DVWA) for common vulnerabilities such as SQL Injection, Cross-Site Scripting (XSS), and Authentication Flaws.

- Objective: Identify and test vulnerabilities in DVWA using ethical hacking techniques.
- Skills Gained: Web application security testing, ethical hacking, vulnerability exploitation.

2. Tools and Environment Used

- Kali Linux (via VMware Workstation)
- DVWA (Damn Vulnerable Web Application)
- Burp Suite Community Edition
- Firefox (configured with Burp Proxy)

3. Vulnerabilities Tested

3.1 SQL Injection

- ✔ Injected malicious payload into the User ID parameter: ‘1 --
- ✔ Intercepted the request using Burp Suite Proxy
- ✔ Bypassed the query logic and retrieved admin user data
- ✔ Successfully validated SQL Injection vulnerability in DVWA

3.2 Cross-Site Scripting (XSS)

- ✔ Used reflected XSS payload: `<script>alert("XSS")</script>`
- ✔ Script executed successfully in the browser
- ✔ Verified vulnerability using both Burp Suite and Firefox
- ✔ Confirmed that input validation was not enforced

3.3 Authentication Flaws

- ✔ Performed brute-force attack using Burp Suite Intruder
- ✔ Fixed username: `admin`, tested multiple passwords
- ✔ Successfully logged in using `**password**` as valid credentials
- ✔ Demonstrated lack of brute-force protection mechanism

4. Learning Outcome

- Gained hands-on experience in testing web application vulnerabilities
- Understood how to intercept, modify, and replay HTTP requests
- Learned to use Burp Suite effectively for SQLi, XSS, and brute force
- Developed a solid understanding of OWASP Top 10 security flaws

5. Conclusion

This task gave me an in-depth understanding of how real-world web application attacks are carried out. Using DVWA as a test environment, I was able to practically apply my learning to identify, exploit, and analyze vulnerabilities. The experience has been extremely valuable in building my foundation as a cybersecurity professional. I look forward to the upcoming tasks.

6. Awareness and Prevention

To build secure web applications, it's essential to proactively defend against known vulnerabilities. The following best practices help reduce the attack surface and improve system integrity:

6.1 SQL Injection (SQLi) Prevention

- Use prepared statements (parameterized queries) instead of raw SQL
- Implement proper input validation and sanitization
- Use ORM frameworks to abstract direct SQL queries
- Avoid displaying raw database errors to users

6.2 Cross-Site Scripting (XSS) Prevention

- Escape all user inputs when rendering HTML (`&`, `<`, `>`, `'`, `"`)
- Use Content Security Policy (CSP) headers to restrict script execution
- Validate and sanitize all form inputs on both client and server sides
- Avoid using `innerHTML` or unsafe DOM manipulation in JavaScript

6.3 Authentication Flaws / Brute-Force Attack Prevention

- Enforce strong password policies (length, complexity, rotation)
- Implement rate limiting or account lockout after failed login attempts
- Use CAPTCHA or multi-factor authentication (MFA)
- Monitor login activity and log suspicious behavior