# Cyber Security Internship Report

## Cyber Security Internship – Task 7

# Web Application Vulnerability Testing – Experiment Report

**Submitted by: Hetal Antala**

**Date : 26 / 01 / 2026**

## 1. Introduction

Web applications are widely used to deliver services over the internet, making them a common target for cyberattacks. Vulnerabilities in web applications can lead to data breaches, unauthorized access, and system compromise. This task focuses on identifying and understanding common web application vulnerabilities using a deliberately vulnerable application and industry-standard security testing tools.

## 2. What is Web Application Vulnerability Testing?

Web application vulnerability testing is the process of identifying security weaknesses in a web application that could be exploited by attackers. It helps organizations detect flaws in input validation, authentication, and session

handling before they are exploited in real-world scenarios.

---

## 3. OWASP Top 10

The OWASP Top 10 is a list of the most critical web application security risks published by the Open Web Application Security Project (OWASP). It serves as a standard awareness document for developers and security professionals.

Common OWASP Top 10 vulnerabilities include:

- SQL Injection
- Cross-Site Scripting (XSS)
- Broken Authentication
- Security Misconfiguration
- Cross-Site Request Forgery (CSRF)

---

## 4. Tools Used

- **Damn Vulnerable Web Application (DVWA)** – Vulnerable test environment
- **Burp Suite Community Edition** – Web traffic interception and analysis
- **Mozilla Firefox** – Web browser
- **Docker** – Isolated environment for running DVWA

---

## 5. Environment Setup

DVWA was deployed using Docker to ensure a stable and isolated testing environment. The browser was configured to route traffic through Burp Suite, allowing interception and analysis of HTTP requests and responses.

---

## 6. SQL Injection Vulnerability

SQL Injection occurs when user input is improperly handled and directly included in SQL queries, allowing attackers to manipulate database operations.

### Steps Performed

- Navigated to the SQL Injection module in DVWA
- Entered malicious SQL payload in the input field
- Intercepted the request using Burp Suite

### Observation

The application returned database records without proper authentication, confirming the presence of an SQL Injection vulnerability.

---

## 7. Cross-Site Scripting (XSS)

Cross-Site Scripting (XSS) allows attackers to inject malicious scripts into web pages viewed by other users.

### Steps Performed

- Navigated to the Reflected XSS module in DVWA
- Injected a JavaScript payload into the input field
- Observed the execution of the script in the browser

### Observation

The injected script executed successfully, displaying a browser alert, confirming the XSS vulnerability.

---

## 8. Burp Suite Request Analysis

Burp Suite was used to intercept and analyze HTTP requests and responses between the browser and the DVWA application. This helped in understanding how malicious payloads are sent to the server and how the application responds to them.

---

## 9. Mitigation Techniques

To prevent web application vulnerabilities, the following security measures should be implemented:

- Use prepared statements and parameterized queries
- Validate and sanitize user inputs
- Encode output data properly
- Implement CSRF protection tokens
- Apply proper authentication and session management

---

## 10. Importance of Web Application Security Testing

- Protects sensitive user data
- Prevents unauthorized access
- Reduces risk of data breaches
- Improves application reliability and trust

---

## 11. Conclusion

This task provided practical exposure to web application vulnerability testing using DVWA and Burp Suite. By identifying SQL Injection and Cross-Site Scripting vulnerabilities, this experiment strengthened understanding of real-world web security issues and their mitigation techniques. The task enhanced hands-on skills essential for a career in cybersecurity and penetration testing.

---