# 🛡️ Cyber Security Internship - Task 1 Report

|  |  |
| --- | --- |
| Intern Name | Jagdish Zate |
| Track Code | CS |
| Task Number | 01 |
| Internship Domain | Cyber Security |
| Task Title | Web Application Security Testing |
| GitHub Repository | https://github.com/JagdishZate400/FUTURE\_CS\_01.git |

## 🎯 1. Objective

The objective of this task is to conduct security testing on a sample web application to identify potential vulnerabilities such as SQL Injection, Cross-Site Scripting (XSS), and Authentication flaws. The outcome is a comprehensive security report documenting findings and mitigation strategies.

## 🛠 2. Tools and Technologies Used

• Kali Linux - Penetration testing environment  
• MySQL - Database backend for DVWA  
• DVWA - Target web application  
• Burp Suite - Web traffic interception and manipulation  
• SQLMap - Automated SQL Injection testing  
• OWASP ZAP - Scanning and spidering web application

## ⚙️ 3. Environment Setup

- DVWA was configured and hosted on Kali Linux using Apache and MySQL.  
- DVWA setup scripts were executed using terminal.  
- Successful login was tested using default credentials.  
- All screenshots are available in the attached document.

## 🧪 4. Testing Methodology

• SQL Injection:  
 - Used SQLMap on login and search inputs.  
 - Bypassed login using ' OR 1=1 --.  
• Cross-Site Scripting (XSS):  
 - Injected <script>alert('XSS')</script> in form inputs.  
 - Alert popup confirmed XSS vulnerability.  
• Authentication Flaws:  
 - Brute-force simulated using Burp Suite Intruder.  
 - No lockout or rate limiting mechanisms found.

## 🚨 5. Identified Vulnerabilities

• SQL Injection - High - SQLMap - Login and Search inputs  
• Reflected XSS - Medium - OWASP ZAP - Input fields in comments  
• Weak Authentication - High - Burp Suite - No account lockout or brute force protection

## 🔐 6. Recommended Mitigation

• SQL Injection:  
 - Use parameterized queries and input sanitization.  
• XSS:  
 - Implement output encoding and CSP headers.  
• Authentication Flaws:  
 - Enforce CAPTCHA, implement account lockout and session validation.

## 📈 7. Outcome & Learning

This task provided hands-on experience in web application testing. It strengthened skills in ethical hacking, secure development, and using industry-standard tools for vulnerability assessment.

## 📎 8. Deliverables

• GitHub Repository: https://github.com/JagdishZate400/FUTURE\_CS\_01.git  
• Screenshots Document: Attached  
• Security Report: This file  
• (Optional) Walkthrough Video: Insert link if available

## 🔗 9. Task Reference

• Task Page: https://futureinterns.com/cyber-security-task-1/  
• Internship Site: https://futureinterns.com  
• LinkedIn: https://linkedin.com/company/future-interns

## ✅ 10. Conclusion

The task successfully demonstrated the identification and mitigation of vulnerabilities in DVWA using professional cybersecurity tools. It showcases applied knowledge and practical skills in ethical hacking.

# 🖼️ Appendix: Screenshots

Note: Screenshots used during the task are included in the original Word document submitted.  
The key screenshots covered the following:  
  
1. DVWA Environment Setup  
2. MySQL Configuration  
3. SQL Injection Demo (Login Bypass)  
4. XSS Alert Execution  
5. Burp Suite Brute-force Simulation