DVWA Web Application Vulnerability Assessment Report

Vulnerability Assessment Report

Target: http://150.1.7.104/dvwa/
Report Date: October 3, 2025

---

1. Executive Summary

Scope & Objectives:
This assessment focused on identifying security vulnerabilities within the Damn Vulnerable Web Application (DVWA). The objective was to evaluate the application's resilience against common web-based attacks and provide a prioritized remediation plan.

Business Impact:
The vulnerabilities identified, if exploited in a production environment, could lead to a full compromise of the application and its underlying data. This would result in unauthorized access to sensitive information, damage to organizational reputation, loss of customer trust, and potential non-compliance with data protection regulations.

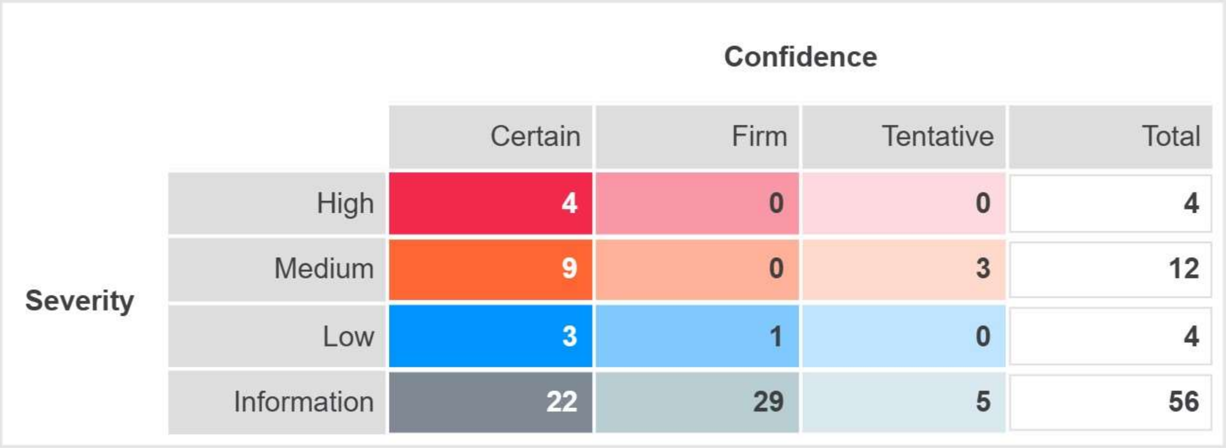
Non-Technical Language:
The assessment found several significant security weaknesses in the web application. These weaknesses could allow an attacker to steal user data, deface the website, or take complete control of the application server. Immediate action is required to address the most critical issues.

Timestamp:

- Assessment Performed: October 3, 2025
- Report Generated: October 3, 2025

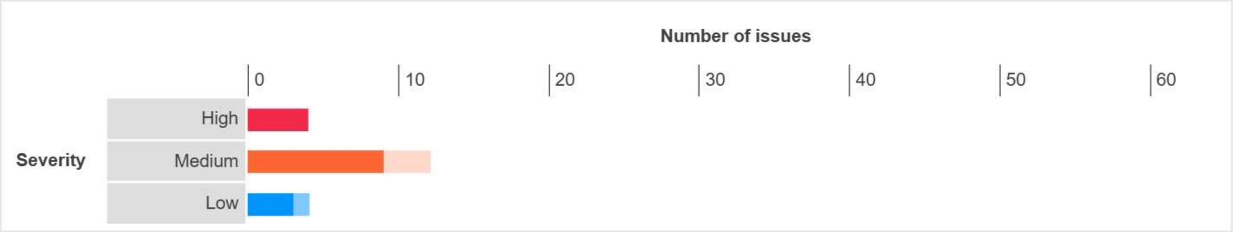
High-Level Remediation Summary:
The most urgent priorities are to fix vulnerabilities that allow attackers to execute malicious code on the server or steal user sessions. This involves implementing proper input validation, updating software, and enforcing secure cookie settings.

The table below shows the numbers of issues identified in different categories. Issues are classified according to severity as High, Medium, Low or Information. This reflects the likely impact of each issue for a typical organization. Issues are also classified according to confidence as Certain, Firm or Tentative. This reflects the inherent reliability of the technique that was used to identify the issue.



		Confidence			
		Certain	Firm	Tentative	Total
Severity	High	4	0	0	4
	Medium	9	0	3	12
	Low	3	1	0	4
	Information	22	29	5	56

The chart below shows the aggregated numbers of issues identified in each category. Solid colored bars represent issues with a confidence level of Certain, and the bars fade as the confidence level falls.



2. Assessment Overview

Methodology & Tools Used:

- Tool: Burp Suite Professional (Scanner and Proxy modules).
- Framework: OWASP Web Security Testing Guide (WSTG).
- Approach: Automated scanner + manual validation.

Scope of Target Systems:

- In-Scope: The DVWA application at http://150.1.7.104/dvwa/ (all modules and functionalities).

Exclusions:

- Denial-of-Service (DoS) testing not performed.
- Underlying host OS and network infrastructure were out of scope.

3. Findings

Identified Vulnerabilities:

- SQL Injection (Blind)
- Cross-Site Scripting (XSS) – Reflected
- Cross-Site Request Forgery (CSRF)
- Weak Session Cookie Security Attributes

Evidence:

- SQL Injection (Blind): The page vulnerabilities/sqli_blind/ was vulnerable. Payload 1' AND SLEEP(5)-- - caused a 5-second delay.
- Cross-Site Scripting (XSS): The page vulnerabilities/xss_r/ was vulnerable. Payload <script>alert('XSS')</script> executed in the browser.
- Cross-Site Request Forgery (CSRF): Password change functionality lacked anti-CSRF tokens.

4. Risk Assessment

Vulnerability	**Classification**	**Impact vs Likelihood**	**Critical Assets/Hosts**
SQL Injection (Blind)	High	High Impact: Full database disclosure. Likely: Easily exploitable.	150.1.7.104/dvwa/
Cross-Site Scripting (XSS)	Medium	Medium Impact: Session theft/defacement. Likely: Common & easy.	150.1.7.104/dvwa/
Cross-Site Request Forgery	Medium	Medium Impact: Unauthorized actions. Possible: Needs user interaction.	150.1.7.104/dvwa/
Weak Cookie Security	Low	Low Impact: Session hijacking risk. Unlikely: MITM required.	150.1.7.104/dvwa/

---

5. Recommendations

Critical/High Priority:

1. Remediate SQL Injection vulnerabilities.
   - Use parameterized queries (prepared statements).
   - Reference: OWASP SQL Injection Prevention Cheat Sheet.

Medium Priority:
2. Fix XSS vulnerabilities.

- Implement context-sensitive output encoding.
- Reference: OWASP XSS Prevention Cheat Sheet.

3. Implement Anti-CSRF Tokens.
   - Add synchronized tokens to all state-changing requests.
   - Reference: OWASP CSRF Prevention Cheat Sheet.

Low Priority:
4. Harden session cookies.

- Add HttpOnly, Secure, and SameSite attributes.
- Reference: OWASP Session Management Cheat Sheet.

6. Conclusion

Summary of Key Actions:
Immediate focus: SQL Injection → then XSS → then CSRF.

Timeline for Reassessment:
A follow-up test should be done within 2–4 weeks after fixes.

Closing Note:
Addressing these findings is crucial to strengthening the security posture. Continuous security testing is recommended.

7. Appendix

References:

- OWASP Top Ten – https://owasp.org/www-project-top-ten/
- SQL Injection Prevention Cheat Sheet – https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html
- XSS Prevention Cheat Sheet – https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html
- **The Link to the BrupSuite Vulnerability Assessment Report** - https://drive.google.com/file/d/1rcildNZJMSHg6NUVaIBOHaQ0LOSzhBAV/view?usp=sharing

Glossary:

- **SQL Injection (SQLi):** Exploits improper handling of input in database queries.
- **Cross-Site Scripting (XSS):** Allows attackers to inject scripts into web pages.
- **Cross-Site Request Forgery (CSRF):** Tricks victims into performing unintended actions.