【联合注入添加临时虚拟用户】GXYCTF2019 BabySQli WriteUp

原创

Lxxx 于 2021-06-30 20:54:29 发布 22 收藏

分类专栏： 网络安全

本文链接：https://blog.csdn.net/qq_43661593/article/details/118369321

网络安全 专栏收录该内容

15 篇文章 0 订阅

订阅专栏

联合注入添加临时虚拟用户：

在查询过程中如果数据库中没有对应的结果,会临时创建一个虚拟用户

举例

这是查询前的表

```
mysql> select * from wish;
+----+--------------+--------------------------------------------------------------------------+------------+--------+----------+
| id | name         | content                                                                  | time       | color  | password |
+----+--------------+--------------------------------------------------------------------------+------------+--------+----------+
|  1 | 张三         | 天天开心、心想事成、大吉大利、一帆风顺。                                 | 1490240257 | red    | 111      |
|  2 | PHP爱好者    | 祝愿PHP越来越好！                                                        | 1490241675 | yellow |          |
|  3 | 匿名         | 争取毕业月薪过万！                                                       | 1490251234 | blue   | 000000   |
|  4 | 小明         | 考上清华大学                                                             | 1490252675 | green  | 123      |
+----+--------------+--------------------------------------------------------------------------+------------+--------+----------+
4 rows in set (0.00 sec)

mysql>
```

接着我们尝试查询一个不存在的数据.

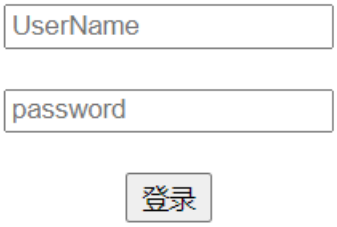
```
mysql> select 'd','ddd','dddd','dddd',5555;
+---+-----+------+------+------+
| d | ddd | dddd | dddd | 5555 |
+---+-----+------+------+------+
| d | ddd | dddd | dddd | 5555 |
+---+-----+------+------+------+
1 row in set (0.00 sec)
```

可能这样还不够直观,通过联合查询将直观性体现出来.(联合查询时注意列数)

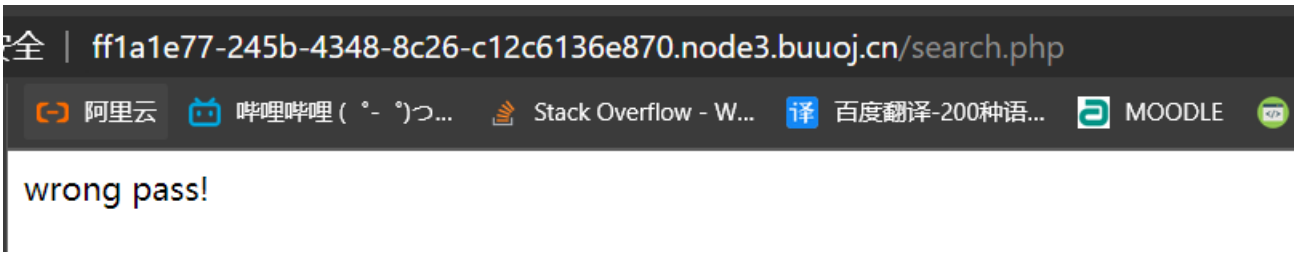
```
mysql> select * from wish union select 'd','dd','ddd','dddd',5555;
ERROR 1222 (21000): The used SELECT statements have a different number of columns
mysql> select * from wish union select 'd','dd','ddd','dddd','dddddd',5555;
+----+--------------+-----------------------------------------------------------+------------+--------+----------+
| id | name         | content                                                   | time       | color  | password |
+----+--------------+-----------------------------------------------------------+------------+--------+----------+
| 1  | 张三         | 天天开心、心想事成、大吉大利、一帆风顺。                  | 1490240257 | red    | 111      |
| 2  | PHP爱好者    | 祝愿PHP越来越好！                                         | 1490241675 | yellow |          |
| 3  | 匿名         | 争取毕业月薪过万！                                        | 1490251234 | blue   | 000000   |
| 4  | 小明         | 考上清华大学                                              | 1490252675 | green  | 123      |
| d  | dd           | ddd                                                       | dddd       | dddddd | 5555     |
+----+--------------+-----------------------------------------------------------+------------+--------+----------+
5 rows in set (0.00 sec)
```

WriteUp:

首先，拿到题目，是一个登录框页面

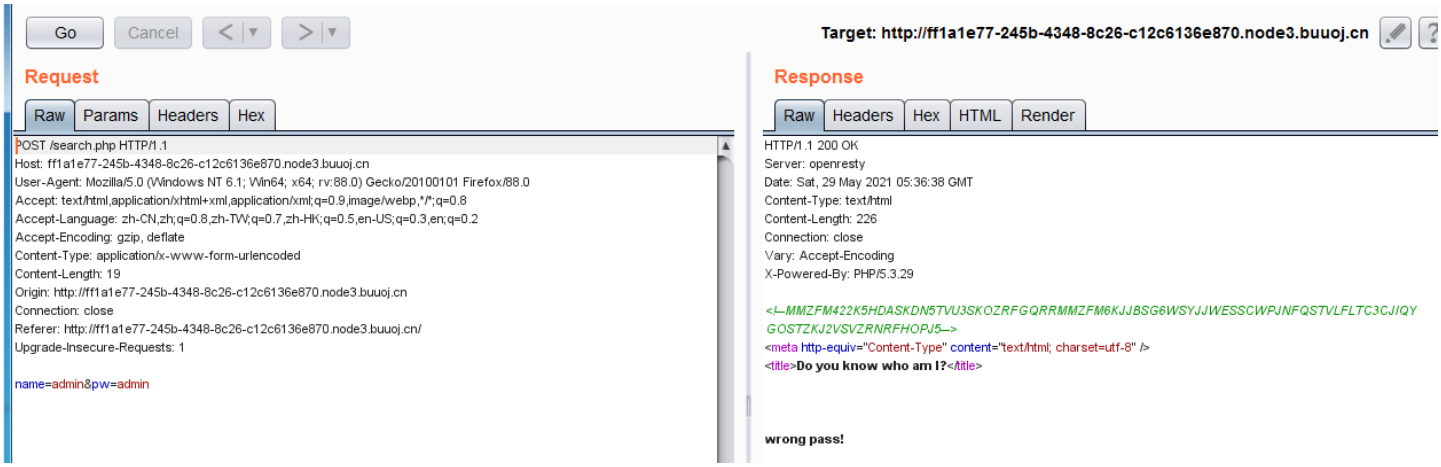


输入`admin`、`admin`，回显如下



可以发现，在URL中没有直接显示，应该是利用`POST`传输表单

用`burpsuite`截一下包

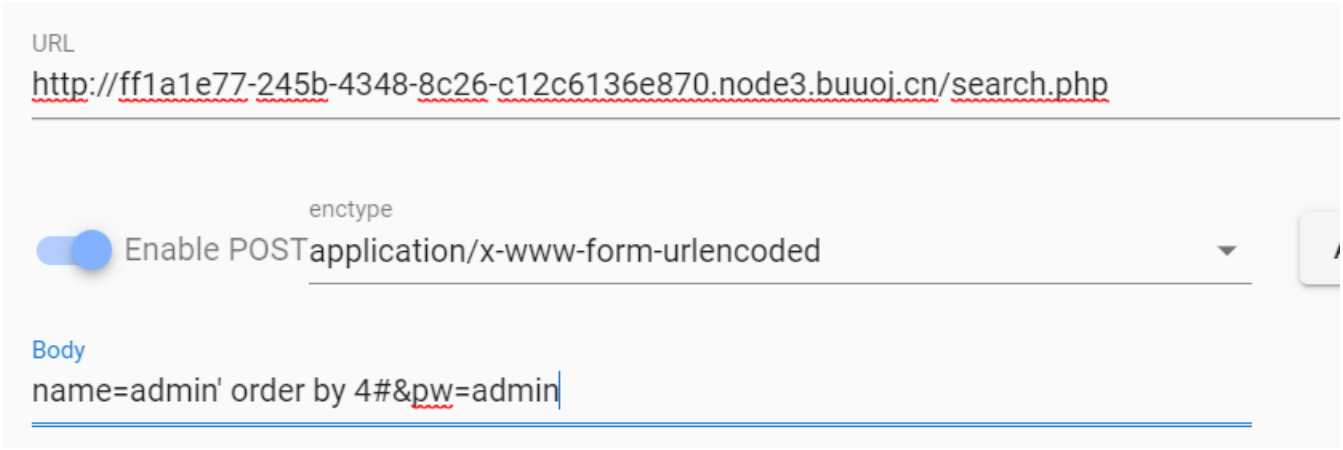


看到回显里有一段注释，猜测是使用`base`加密的一段文字

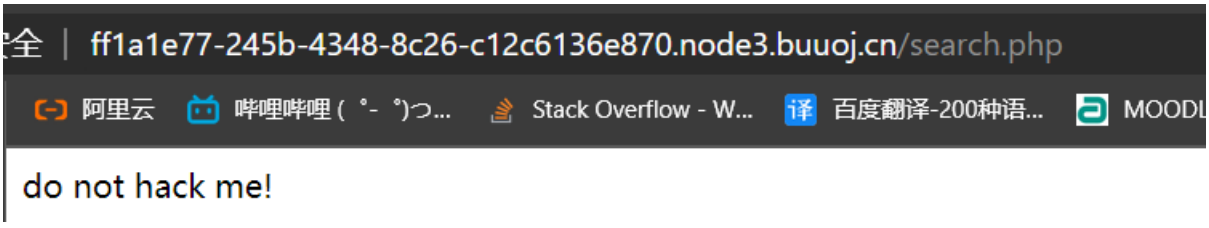
```
MMZFM422K5HDASKDN5TVU3SKOZRFGQRRMMZFM6KJJBSG6WSYJJWESSCWPJNFQSTVLFLTC3CJIQYGOSTZKJ2VSVZRNRFHOPJ5
经过base32解码后
c2VsZWN0ICogZnJvbSB1c2VyIHdoZXJlIHVzZXJuYW1lID0gJyRuYW1lJw==
看到有等号，再使用base64解码
select * from user where username = '$name'
```

可以看到，`hint`里应该是存在联合注入的漏洞

先用常规方法，利用`order by`查看当前字段



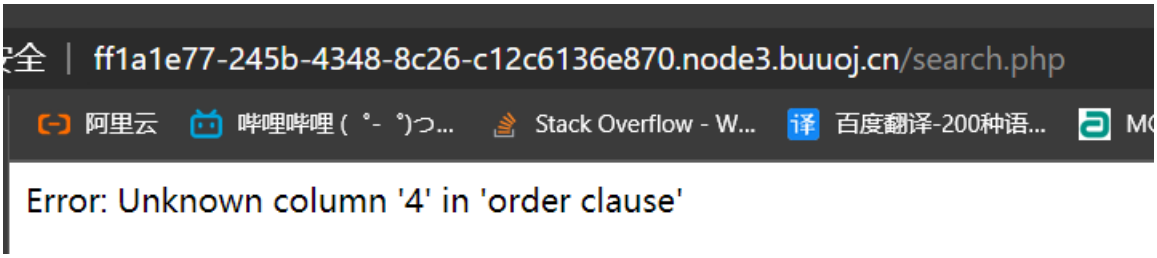
回显如下：



可以看到，是被过滤掉了

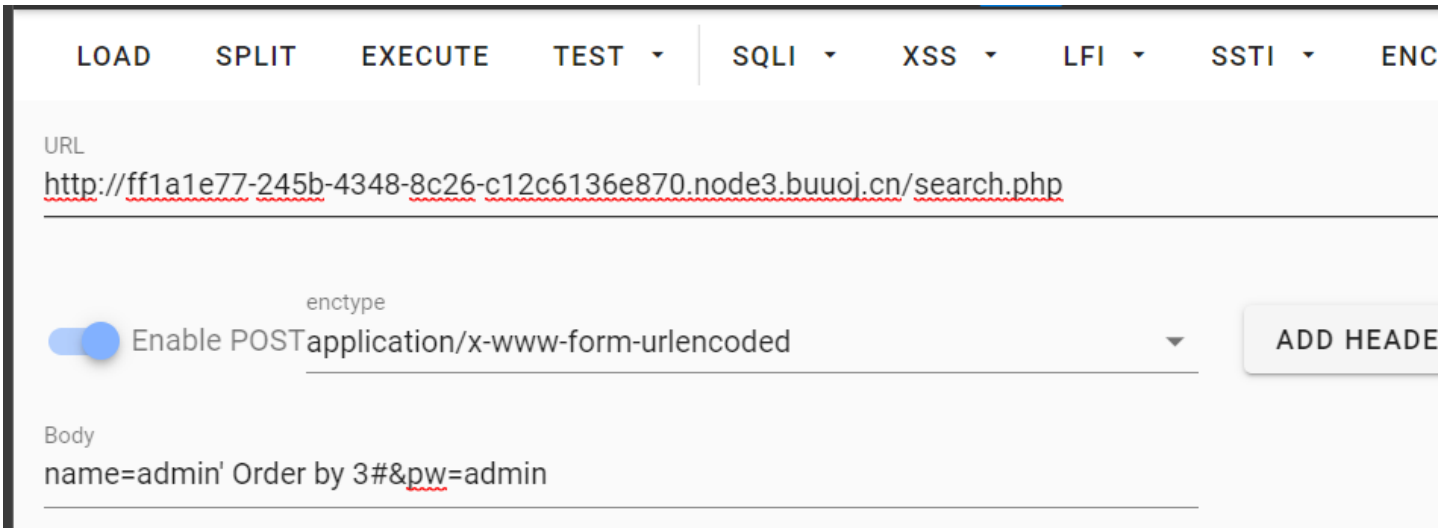
尝试使用大小写绕过，POST传参如下：

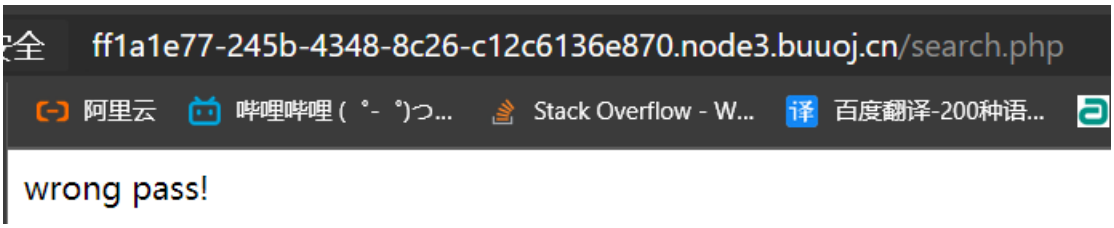
```
name=admin' Order by 4#&pw=admin
```



可以看到数据表中没有4个字段

最终确定数据表中一共有3个字段

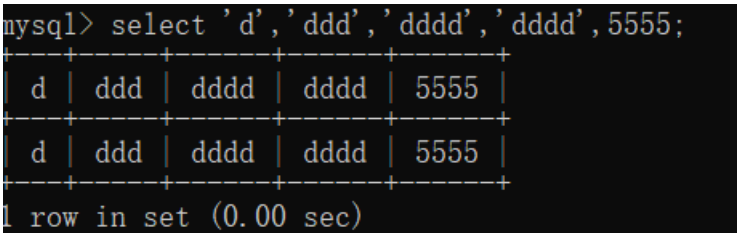




一般来说，登录框中的三个字段，第一个一般是`id`，第二个是用户名，第三个是密码

这时候可以使用联合注入，创建一个虚拟表，如下：

```
select 'd','ddd','dddd','dddd',55551
```

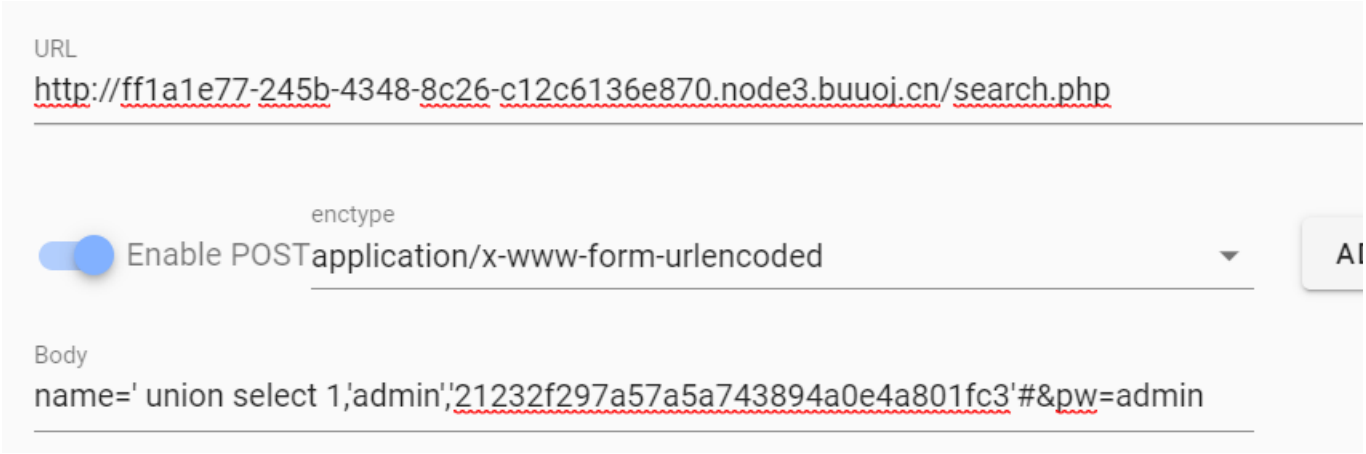


而根据题目给的`hint`：`select * from user where username = '$name'`

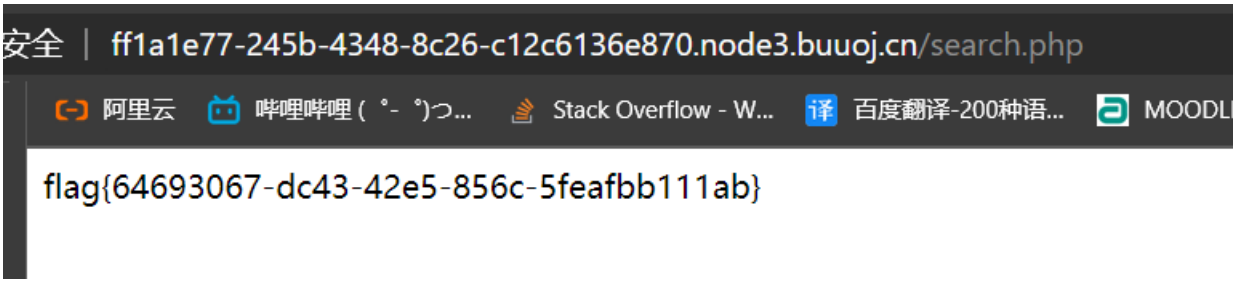
这时候我们采用联合注入，先用`'`闭合掉，然后添加`1`，`admin`，`password`

注意，验证密码是否正确，服务器通常会先读取密码，然后进行`md5`加密，最终将加密后的值与数据表中的值进行比对（这里的话我们是使用了临时虚拟表，所以第三栏的`password`就修改成输入值后经过`md5`加密的值

payload如下：



得到flag：



`flag{64693067-dc43-42e5-856c-5feafbb111ab}`