第一届BMZCTF公开赛-MISC-Writeup

原创

末 初 于 2020-12-28 13:24:47 发布 5270 收藏 11

分类专栏： CTF_MISC_Writeup

本文链接：https://blog.csdn.net/mochu7777777/article/details/111306837

文章目录

签到题

你猜猜flag

Snake

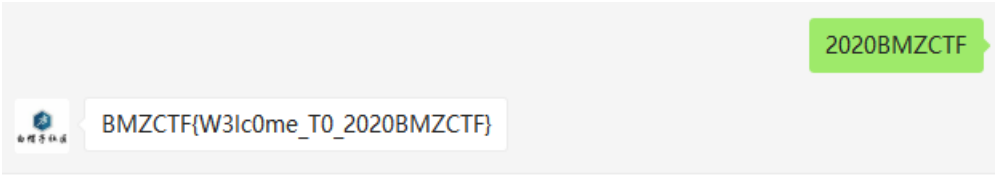
Tiga

Hack K

前言

> 首先恭喜 白帽子社区团队 成功举办第一届BMZCTF公开赛，我是本次比赛MISC赛题 `Snake` 、 `Tiga` 的出题人末初
> 以下是我写的这次BMZCTF公开赛的MISC赛题的Writeup，如果有什么写的不对的还请师傅们留言斧正
> 另外师傅们对于 `Snake` 、 `Tiga` 赛题有疑问的可以留言，我会及时回复

签到题

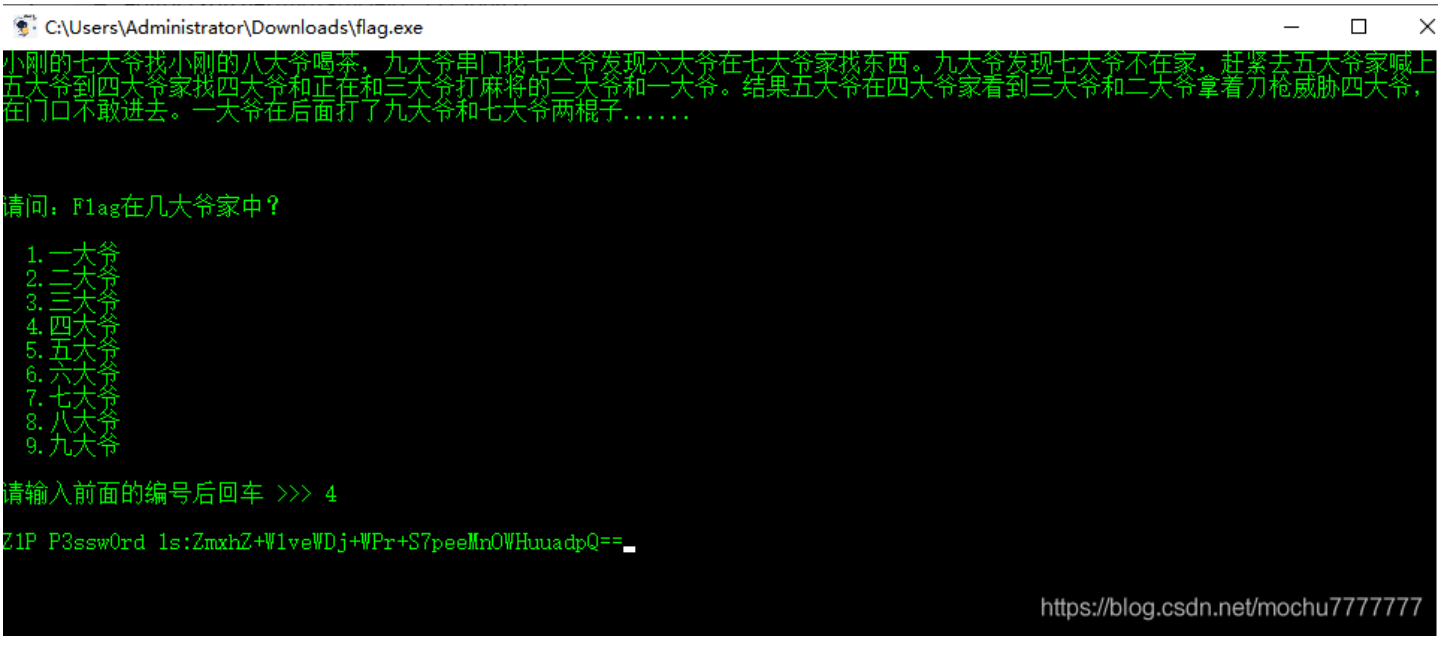



```
BMZCTF{W3lc0me_T0_2020BMZCTF}
```

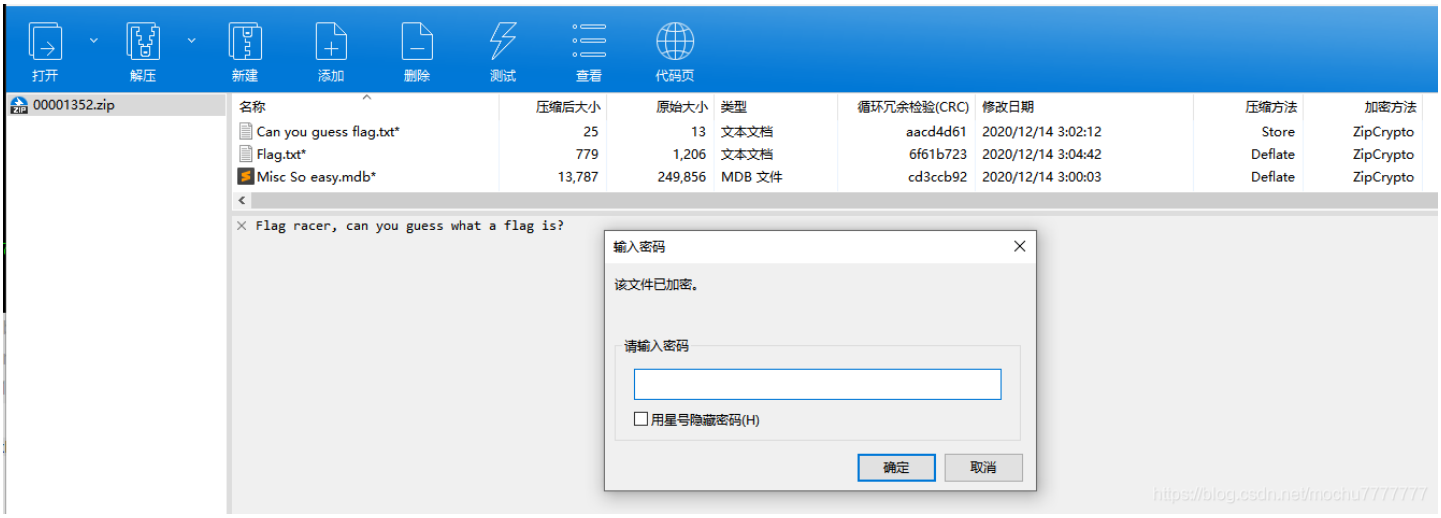
你猜猜flag

`flag.exe`， `binwalk`分析发现有附加zip数据， `foremost`分离出来

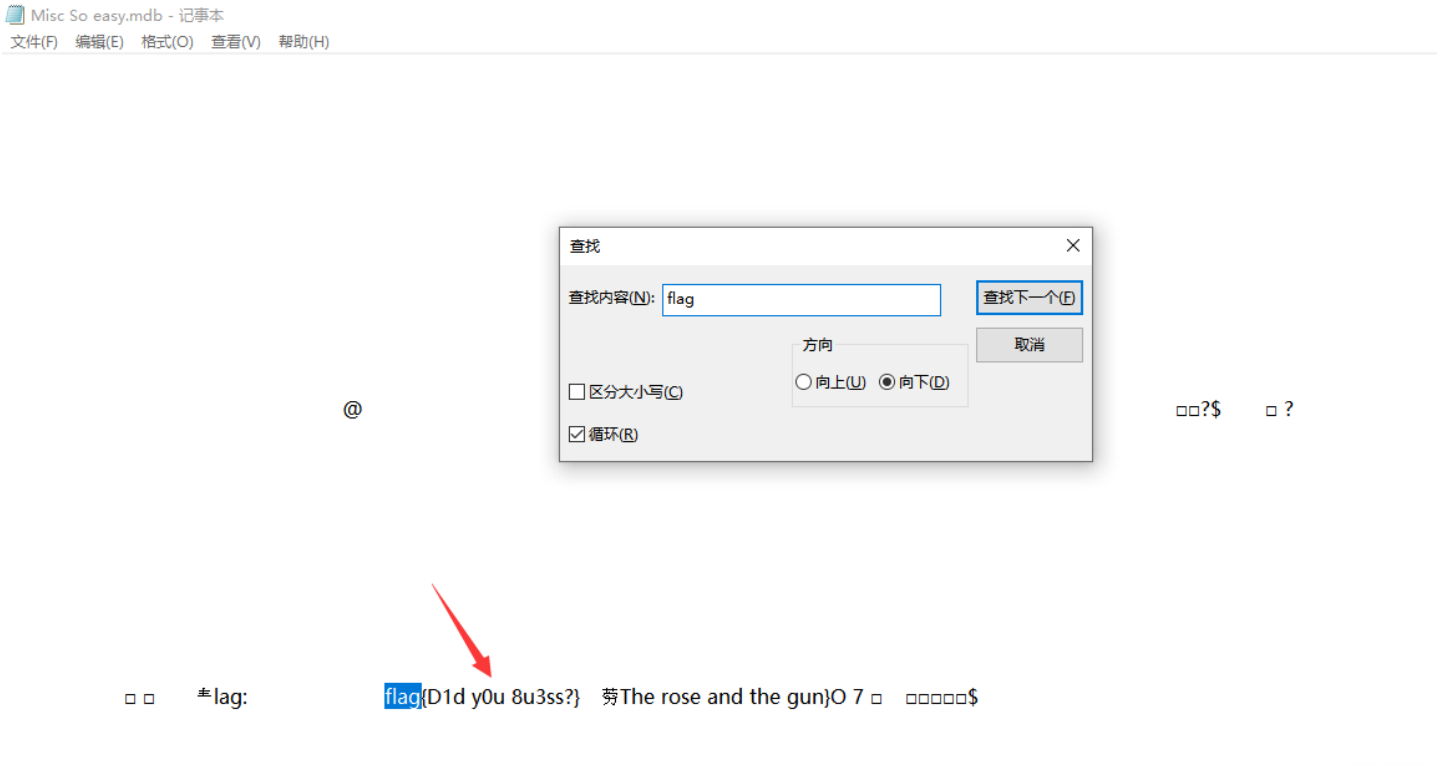
`flag.exe`只有输入`4`的时候输出的zip密码与其他的号码不同



分离出来的zip压缩包密码为：`ZmxhZ+W1veWDj+WPr+S7peeMnOWHuuadpQ==`



解压，再`mdb`文件中找到flag




```
flag{D1d y0u 8u3ss?}
```

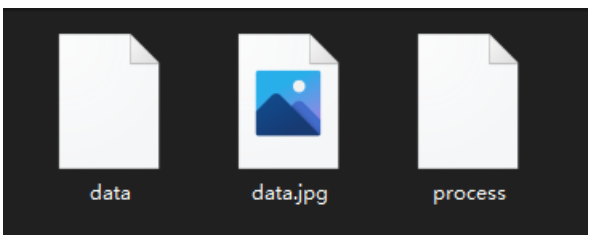
Snake

难度：中下
考察知识点：Short Ook!、程序逆向还原、Steghide、Serpent加密、二进制数据转二维码

注释里面是`Short ook!`编码

得到`snake.zip`的密码：`doyoulikesnake?`

解压得到三个文件



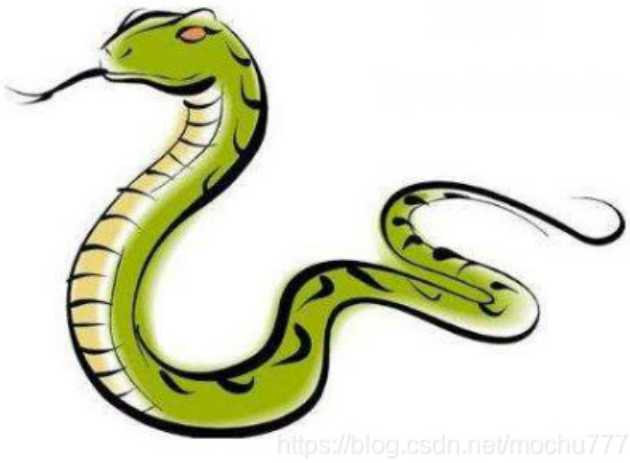
`process`是个`py`程序，添加后缀，然后打开

```
data_jpg = open('data.jpg','wb')
def file_encode():
    with open('snake.jpg','rb') as handle:
        i = 1
        while True:
            bytedata = handle.read(1)
            if(bytedata == b''):
                exit()
            process_data = data_encode(bytedata)
            data_write(process_data)
            i = i + 1

def data_encode(bytedata):
    data = int.from_bytes(bytedata,byteorder='big')
    if (data % 2 == 0):
        data = (data + 1) ^ 128
    else:
        data = (data - 1) ^ 128
    data = bytes([data])
    return data

def data_write(process_data):
    data_jpg.write(process_data)


if __name__ == '__main__':
    file_encode()
    data_jpg.close()
```

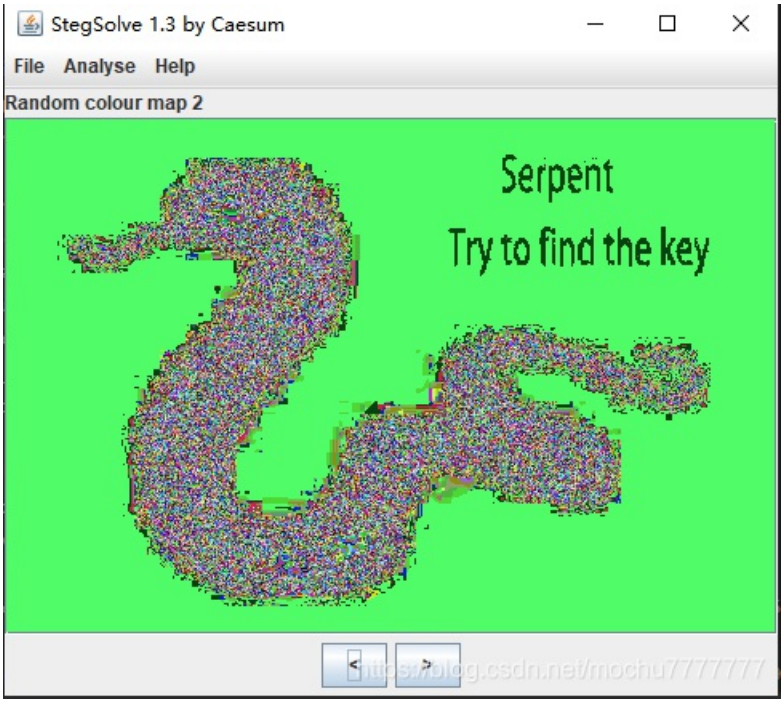
通过程序可以得知`data.jpg`是`process.py`处理`snake.jpg`得到的，逆向编写处理代码

```
with open('snake.jpg','wb') as flag:
    with open('data.jpg','rb') as f:
        for i in f.read():
            if (i % 2 == 0):
                i = (i + 1) ^ 128
            else:
                i = (i - 1) ^ 128
            i = bytes([i])
            flag.write(i)
```

得到`snake.jpg`



使用`stegsolve`打开，发现提示



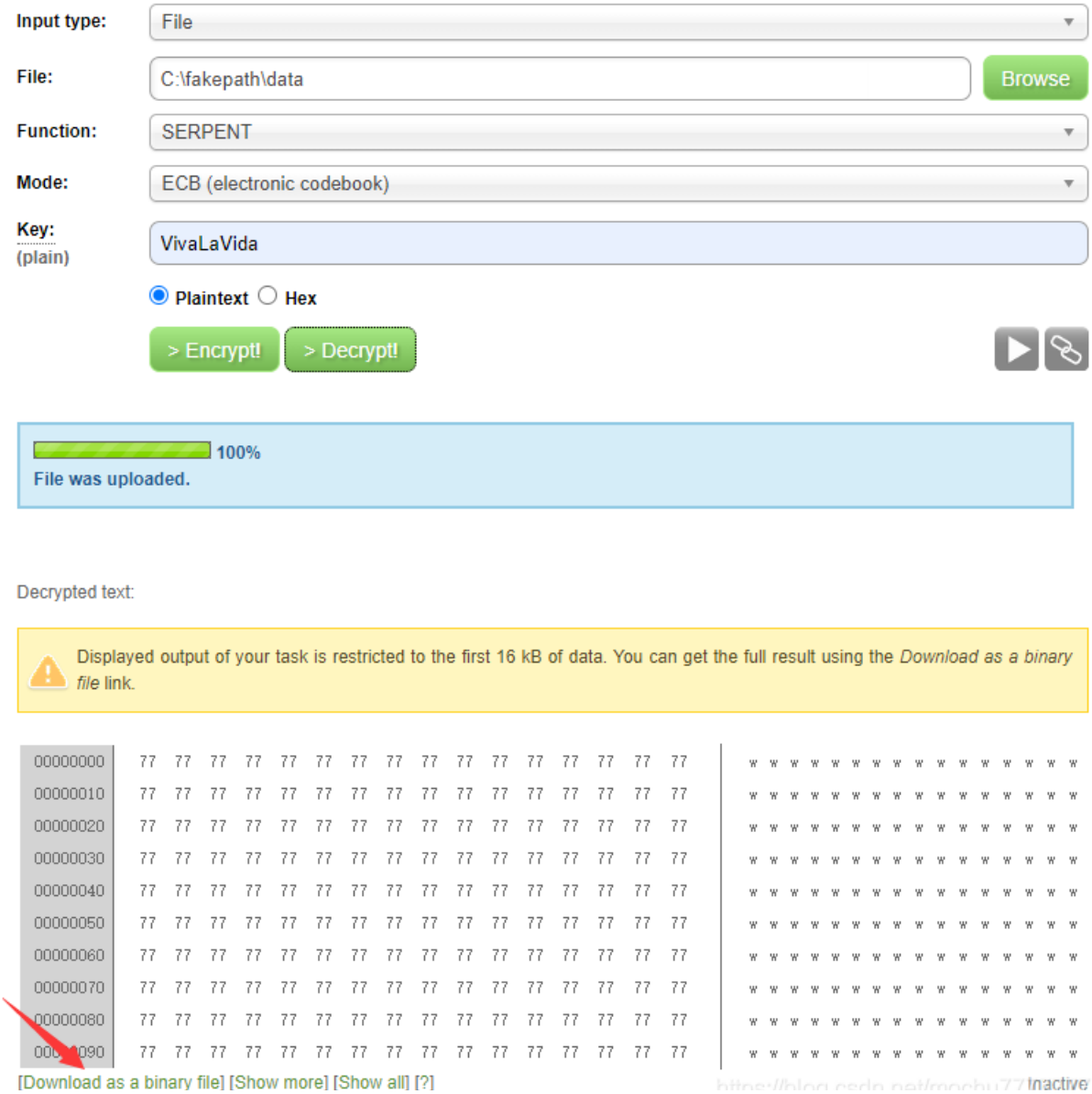
`Serpent`加密，尝试寻找`key`

`Steghide`发现有隐写`key.txt`，无密码

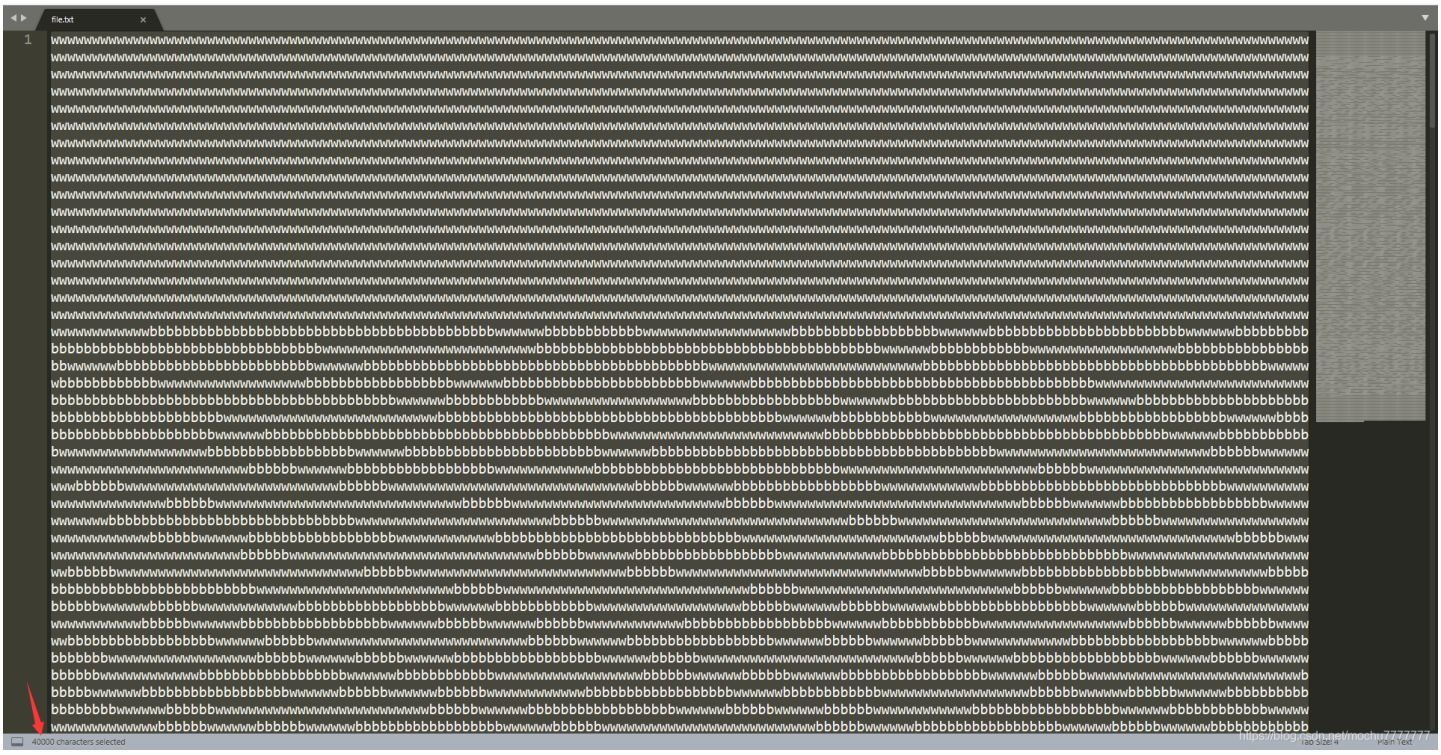

```
key: VivaLaVida
```

知道了是`Serpent`加密，也知道了`key`，那么`data`应该就是加密后的数据，直接解密

Serpent-Online-Encrypt：http://serpent.online-domain-tools.com/



下载解密后的数据，发现内容如下：



然后从以下几点看出这是个二维码的二进制数据：

- 只有`w`和`b`两种字符，二进制
- `w`对应`white`，`b`对应`black`
- 总字符`200 x 200 = 40000`正方形，可能是张二维码
- 做过二进制转二维码题目的同学从内容的首尾都是`w`字符，就应该看得出这是个二维码数据

使用Python将这些转换为二维码

```
import PIL
from PIL import Image

width=height=200
img = Image.new("RGB",(width,height))
i = 0
char = "这里填数据"
for w in range(width):
 for h in range(height):
  if (char[i]=='w'):
   img.putpixel([w,h],(255,255,255))
  else:
   img.putpixel([w,h],(0, 0, 0))
  i = i + 1
img.save('flag.png')
```

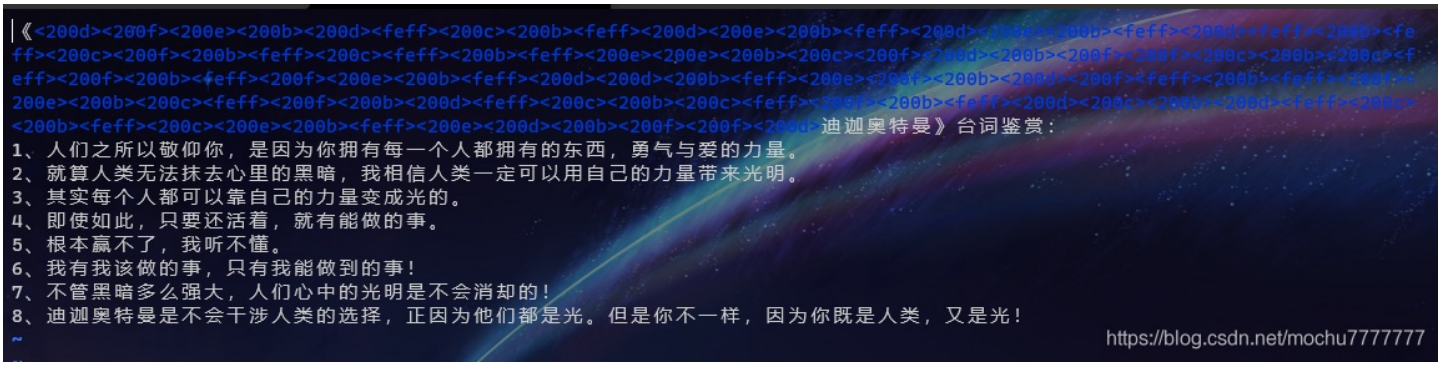
扫描即可得到flag


```
flag{67bd09fc-e252-4c21-858f-2a7d698d555f}
```

Tiga

> 难度：中下
> 考察知识点：零宽度字符隐写、CRC爆破、zip密码爆破、zip密码明文攻击、字节流数据还原、Base16/32/64/85混淆解码

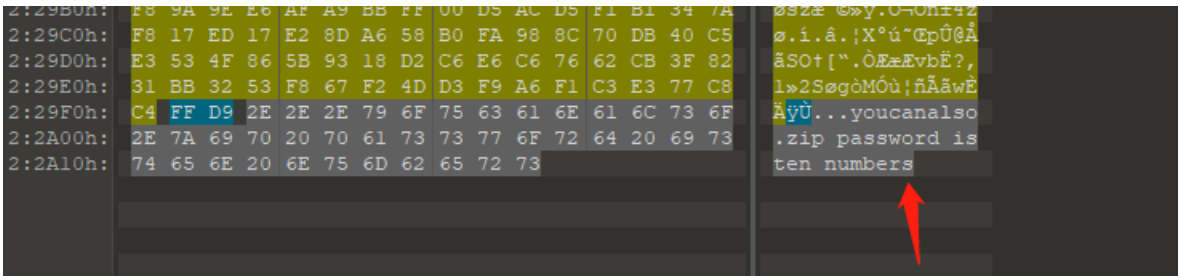
vim查看`tiga.txt`发现存在零宽度字符



零宽度字符隐写在线站：https://yuanfux.github.io/zero-width-web/

得到`misc.zip`密码：`GiveTiGaGuang!`

`迪迦.jpg`文件尾附加了`youcanalso.zip`的密码信息



`file.zip`的密码很明显需要通过爆破这些密码片段文件的CRC得到内容

爆破脚本如下：

```python
import binascii
import string

def crack_crc():
    print('-------------Start Crack CRC-------------')
    crc_list = [0x14433530, 0xaf251007, 0xd554e7b6, 0xebb3156, 0xbb474d49, 0x2cb8a39b, 0x75fe76f0]
    comment = ''
    chars = string.printable
    for crc_value in crc_list:
        for char1 in chars:
            for char2 in chars:
                for char3 in chars:
                    res_char = char1 + char2 + char3
                    char_crc = binascii.crc32(res_char.encode())
                    calc_crc = char_crc & 0xffffffff
                    if calc_crc == crc_value:
                        print('[+] {}: {}'.format(hex(crc_value),res_char))
                        comment += res_char
    print('-----------CRC Crack Completed-----------')
    print('Result: {}'.format(comment))

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

```python
import binascii
import string

def crack_crc():
    print('-------------Start Crack CRC-------------')
    crc_list = [0x14433530, 0xaf251007, 0xd554e7b6, 0xebb3156, 0xbb474d49, 0x2cb8a39b, 0x75fe76f0]
    comment = ''
    chars = string.printable
    for crc_value in crc_list:
        for char1 in chars:
            for char2 in chars:
                for char3 in chars:
                    res_char = char1 + char2 + char3
                    char_crc = binascii.crc32(res_char.encode())
                    calc_crc = char_crc & 0xffffffff
                    if calc_crc == crc_value:
                        print('[+] {}: {}'.format(hex(crc_value),res_char))
                        comment += res_char
    print('-----------CRC Crack Completed-----------')
    print('Result: {}'.format(comment))

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

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

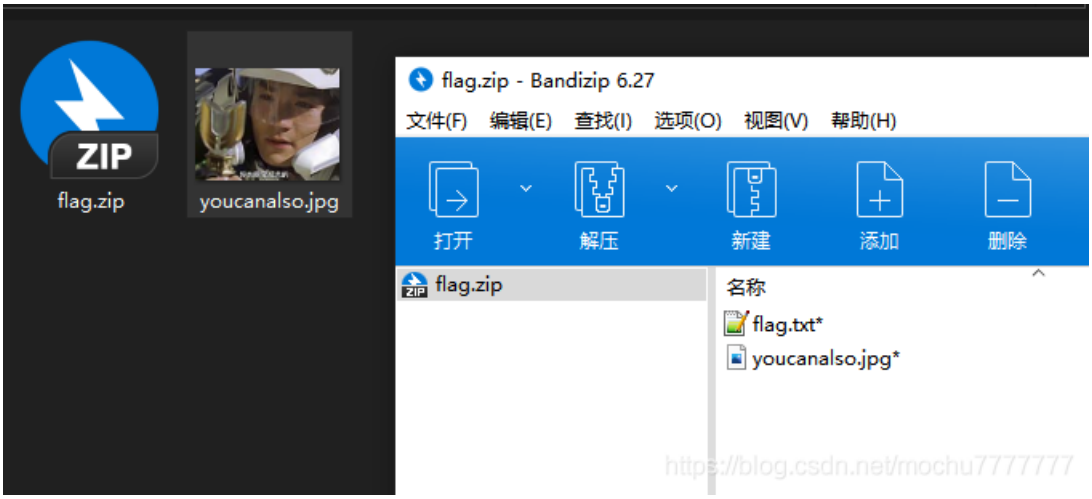
```
[Running] D:/Python/Python3/python.exe "c:\Users\Administrator\Desktop\新建文件夹\tiga\misc\code.py"
-------------Start Crack CRC-------------
[+] 0x14433530: T&h
[+] 0xaf251007: g%W
[+] 0xd554e7b6: L0^
[+] 0xebb3156: rm@
[+] 0xbb474d49: c!V
[+] 0x2cb8a39b: K$x
[+] 0x75fe76f0: Et~
-----------CRC Crack Completed-----------
Result: T&hg%WL0^rm@c!VK$xEt~

[Done] exited with code=0 in 2.703 seconds
```


得到`file.zip`密码：`T&hg%WL0^rm@c!VK$xEt~`

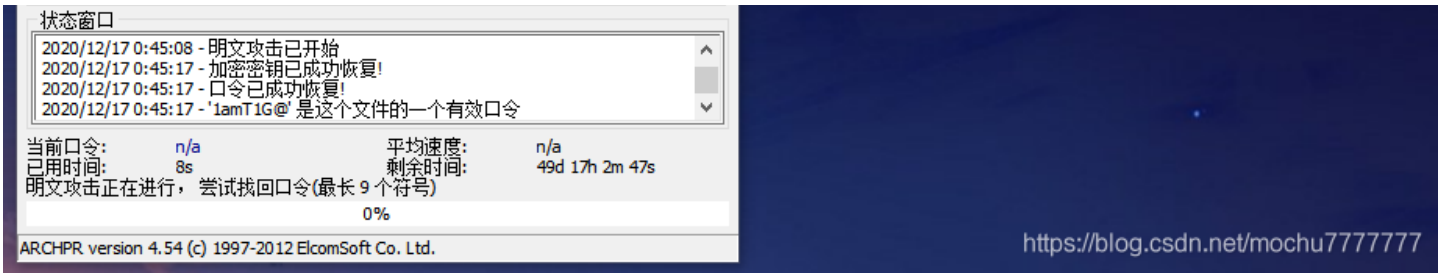
解压得到`youcanalso.zip`根据之前的得到的信息，使用ARCHPR掩码爆破十位数字

得到密码：`2001701725`

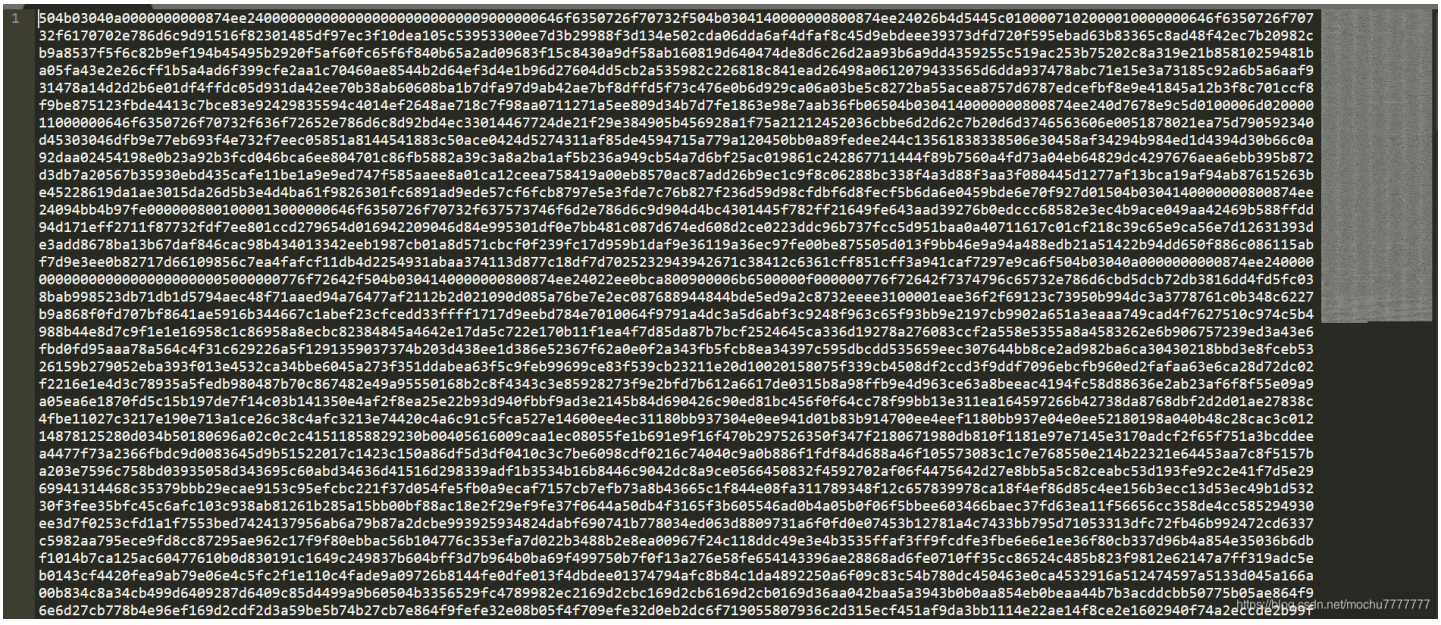
解压得到`flag.zip`和`youcanalso.zip`



这里很明显可以进行明文攻击



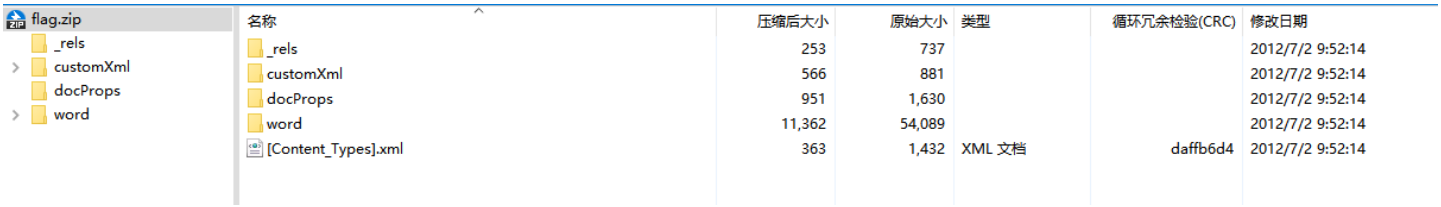
得到`flag.zip`密码: `1amT1G@`得到`flag.txt`，打开发现通过观察头和尾的内容是`zip`文件的的字节流数据



将字节流写成`zip`文件，脚本如下：

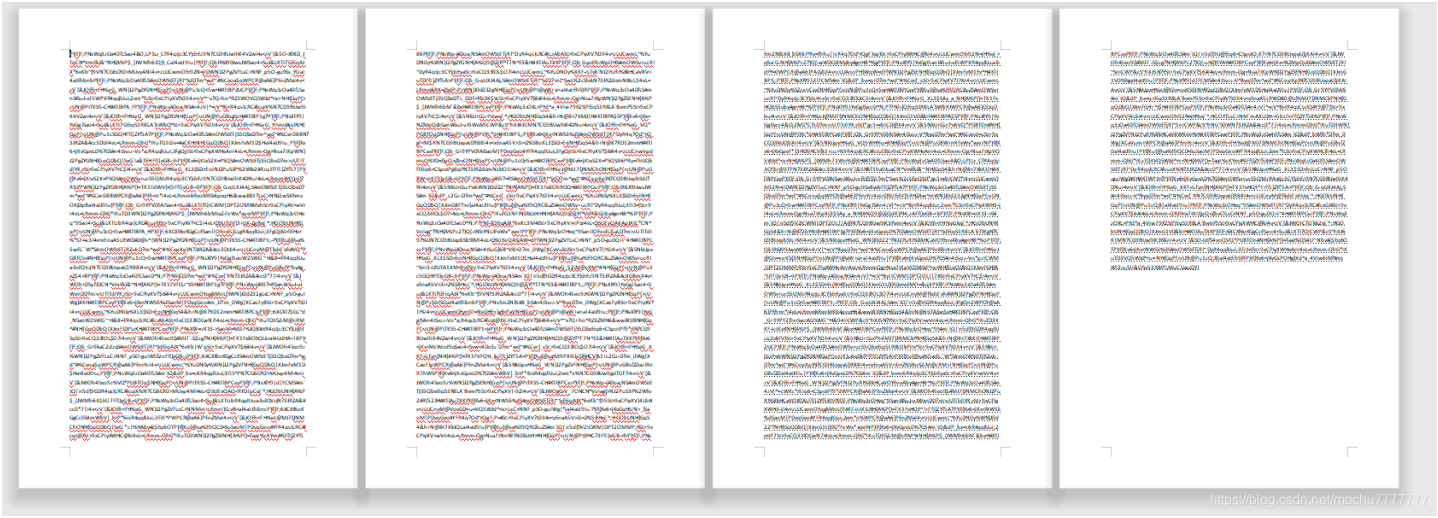
```
import struct

a = open("flag.txt","r")#十六进制数据文件
lines = a.read()
res = [lines[i:i+2] for i in range(0,len(lines),2)]

with open("data.zip","wb") as f:
 for i in res:
  s = struct.pack('B',int(i,16))
  f.write(s)
```

得到`data.zip`发现是`doc`文件的内容



修改后缀为得到`data.docx`，打开发现是`base85`数据

注意：后两页是隐藏文字，需要开启隐藏文字查看选项才能发现



将这个`base85`数据解了一下发现是：base16/32/64/85的随机套娃编码

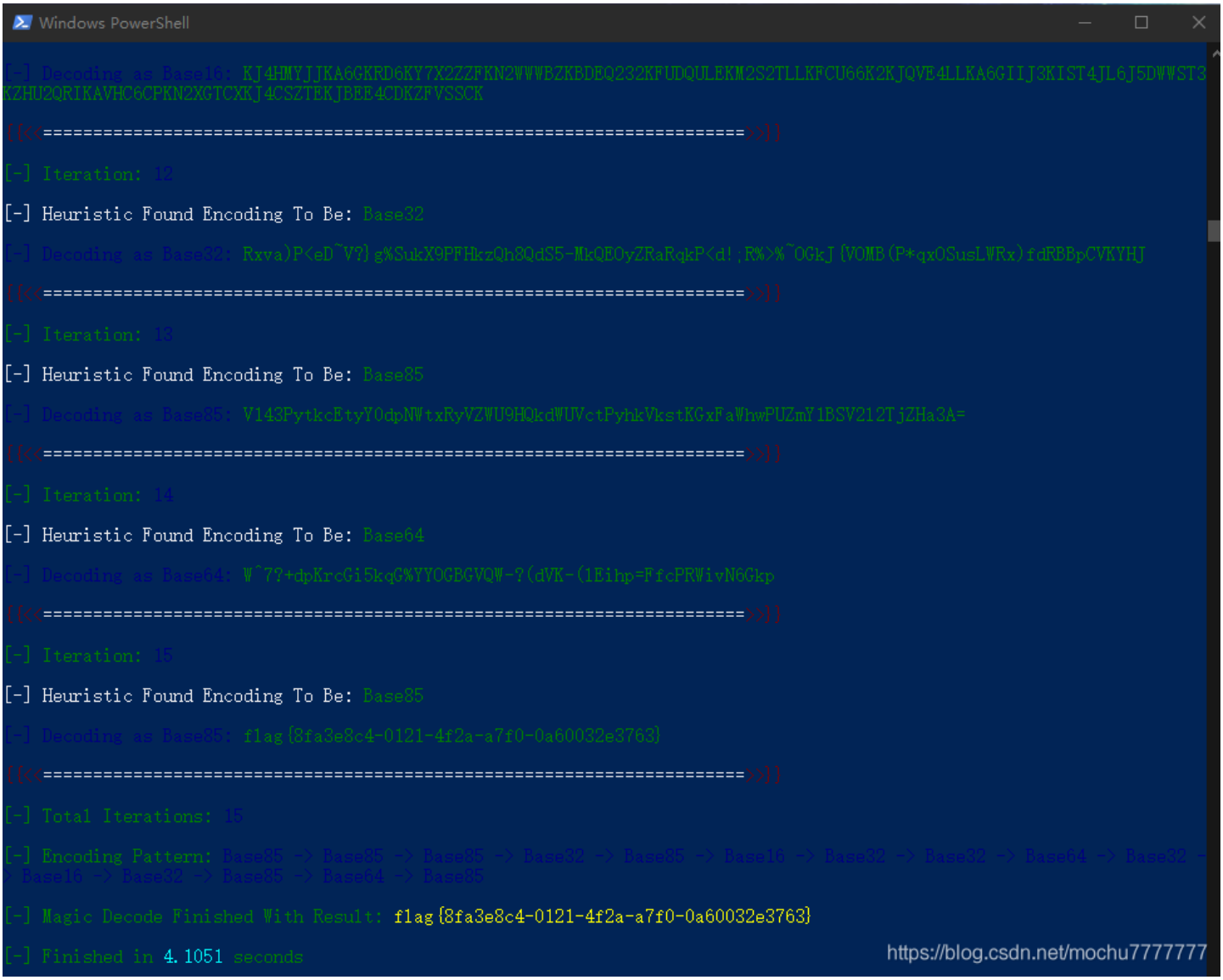
这里可以通过写个识别脚本通过识别base16/32/64/85进行逐步解码，也可以手工，因为这里设置的编码次数不多，就15次

> PS：本来是想设置的编码次数多一点的，不过我比较懒，不想写识别脚本2333

不过推荐使用`basecrack`这个工具直接解码最快：https://github.com/mufeedvh/basecrack

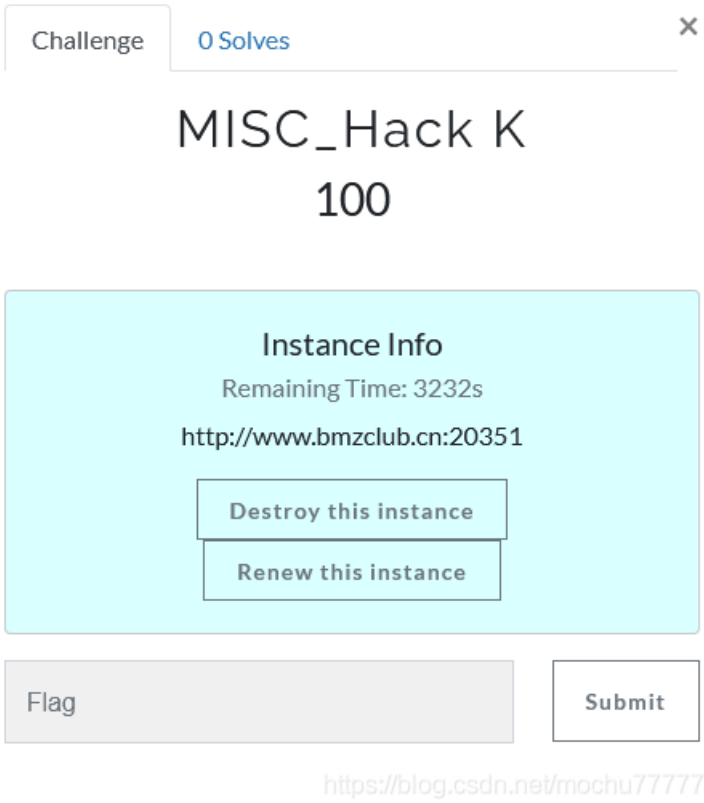
```
python .\basecrack.py --magic
```

然后输入编码数据，得到flag



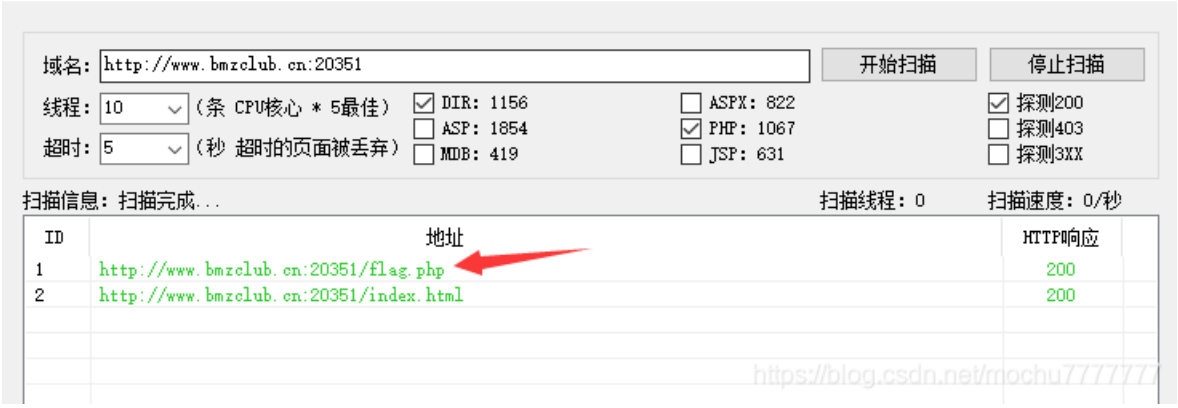

```
flag{8fa3e8c4-0121-4f2a-a7f0-0a60032e3763}
```

Hack K



这题其实我也和出题的师傅商量过

> flag最后的存放文件名`flag.php`太过平常(应该改复杂点)，用平常做web的思路，目录扫描一扫就出



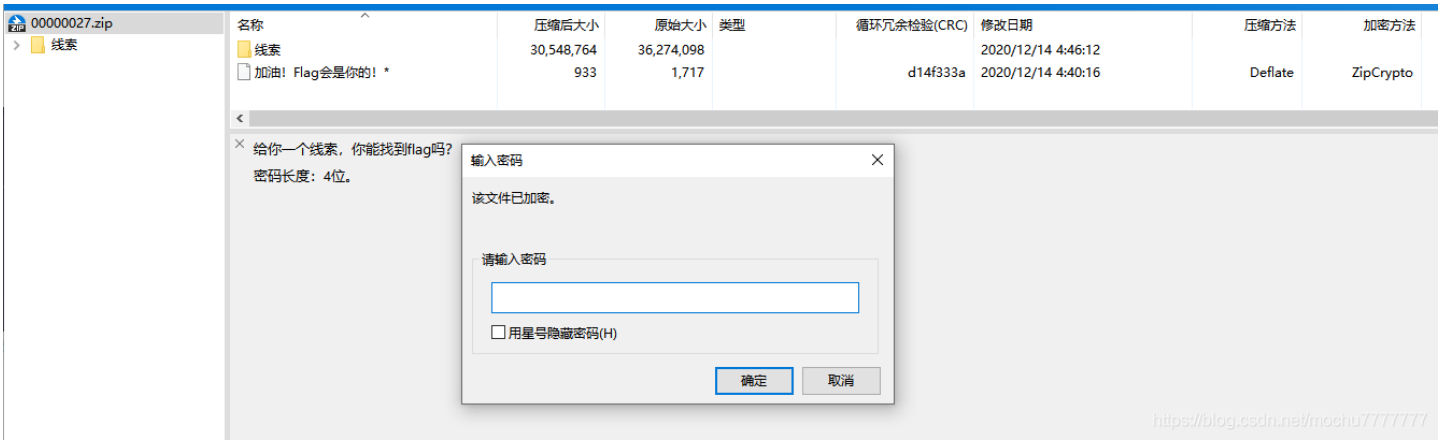
访问即可得到flag

首先观察了这个网站，唯一可以想到出题方向的就是这几张图片了，下载下来发现其中一张图片容量非常大

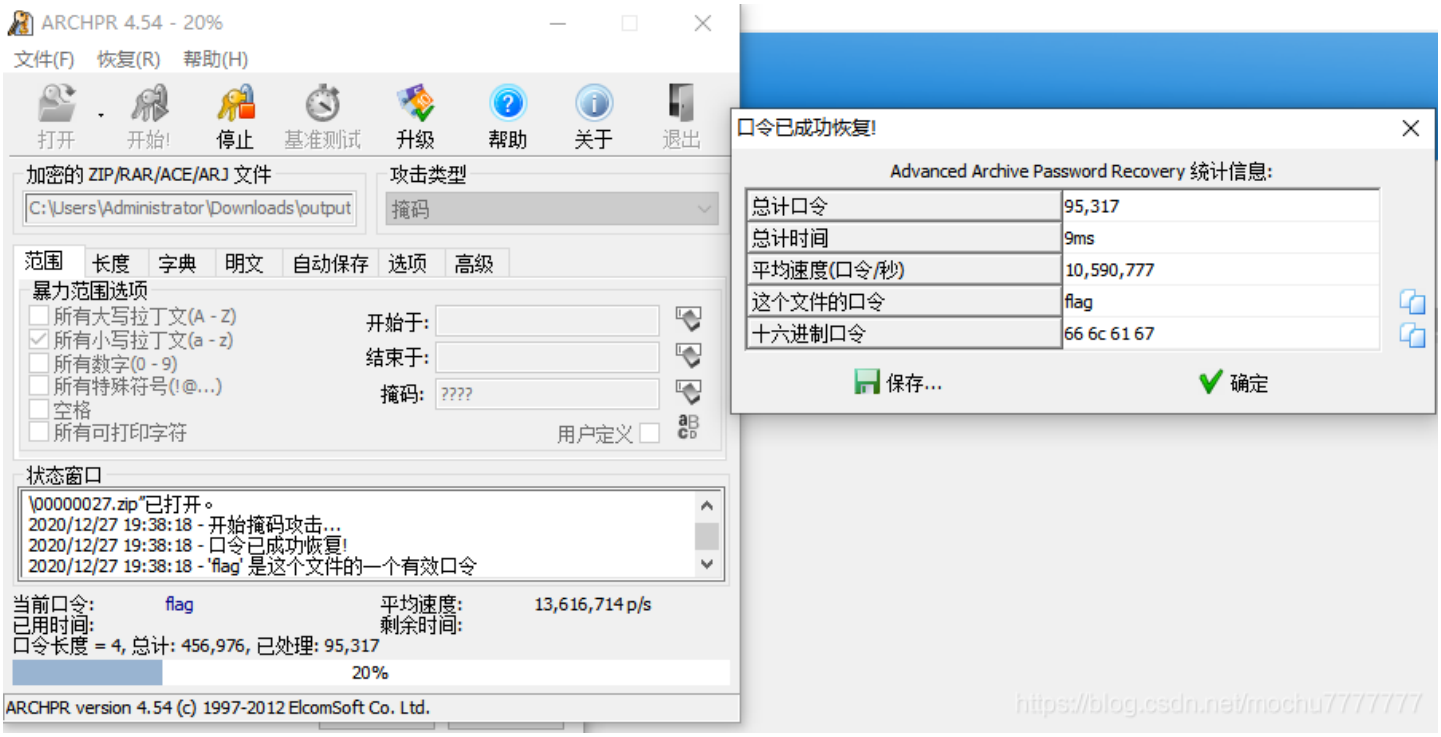
`binwalk`分析图片有附加别的数据

```
root@mochu7-pc:/mnt/c/Users/Administrator/Downloads# binwalk social.png

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             PNG image, 74 x 74, 8-bit/color RGBA, non-interlaced
1594          0x63A           Zlib compressed data, default compression
14823291      0xE22F7B        COBALT boot rom data (Flat boot rom or file system)

root@mochu7-pc:/mnt/c/Users/Administrator/Downloads#
```

`foremost`直接分离，得到一个压缩包，有密码

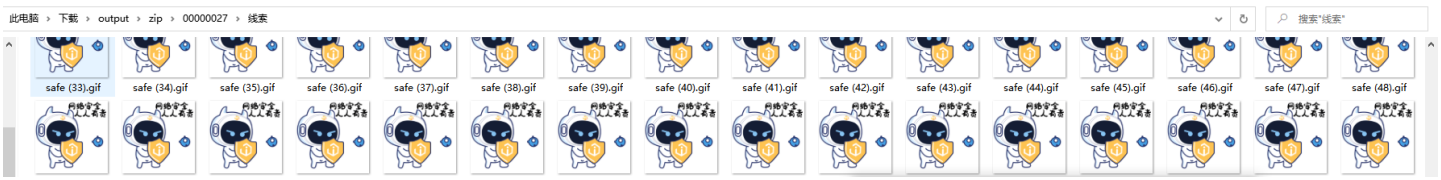


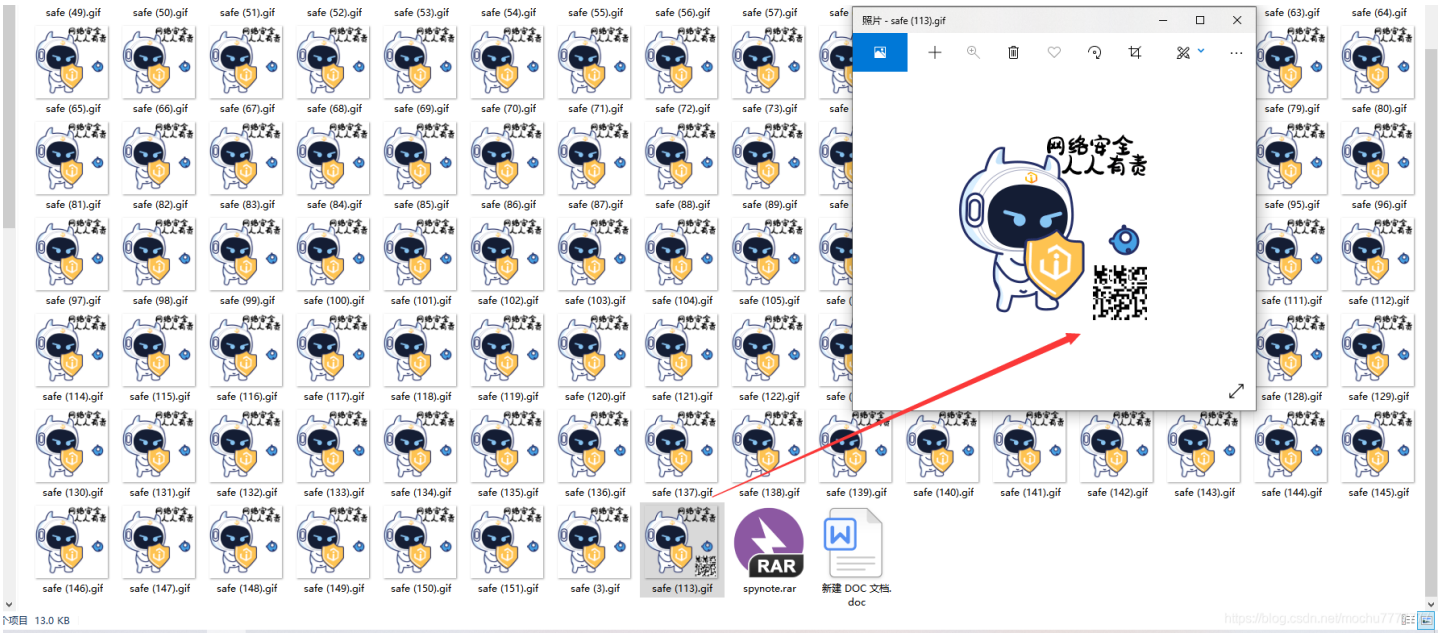
根据压缩包注释提示使用`ARCHPR`爆破，得到密码`flag`



得到一个好像是二维码定位符的png图片

以及一大堆gif图片和文件，其中很容易发现`safe (113).gif`中有张好像没有定位符的二维码





二维码取出来

使用 PS 修补一下

访问`flag.php`得到flag


```
flag{Zme12a9rqsk123S14RH}
```