bugku CTF杂项wp(2)

原创

giunwr 于 2019-07-24 20:55:46 发布 433 收藏 1

分类专栏： misc

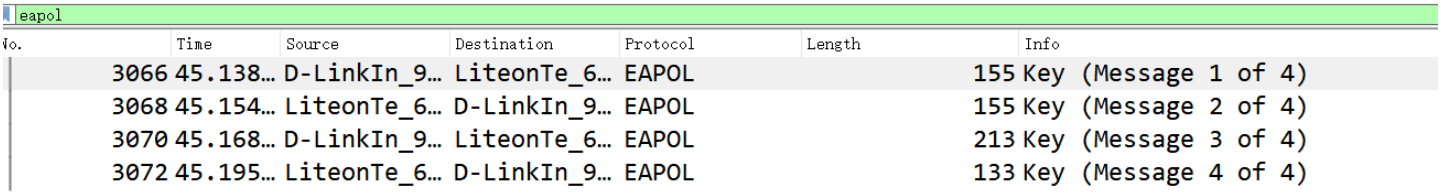
本文链接：https://blog.csdn.net/qq_44105778/article/details/95341326

十六、想蹭网络先破解wifi密码

题目给了我们一个数据包，且提示我们flag为wifi密码，

第一步：

打开文件，WIFI连接认证的重点在WPA的四次握手包，即eapol协议的包，过滤一下：
刚好四个包



eapol

No.	Time	Source	Destination	Protocol	Length	Info
3066	45.138…	D-LinkIn_9…	LiteonTe_6…	EAPOL	155	Key (Message 1 of 4)
3068	45.154…	LiteonTe_6…	D-LinkIn_9…	EAPOL	155	Key (Message 2 of 4)
3070	45.168…	D-LinkIn_9…	LiteonTe_6…	EAPOL	213	Key (Message 3 of 4)
3072	45.195…	LiteonTe_6…	D-LinkIn_9…	EAPOL	133	Key (Message 4 of 4)

第二步

既然是密码，还给了我们数据包，那么我们开始爆破，首先需要一个字典

因为手机号为11位，给了我们7位，故我们使用一下python代码生成字典
代码如下：

```
import string
s = string.digits
f = open('1.txt', 'w')
for i in s:
    for j in s:
        for k in s:
            for o in s:
                f.write("1391040"+i+j+k+o+'\n')
```

第三步

使用aircrack-ng进行爆破。

首先进行安装aircrack-ng，我在kali下进行操作。无法锁定的原因是我之前使用终端时，可能强制退出，还有命令在执行，那么我们就强制解除,命令如下

```
sudo rm /var/cache/apt/archives/lock
sudo rm /var/lib/dpkg/lock
```

```
root@kali:~/Desktop# apt-get install aircrack-ng
E: 无法获得锁 /var/lib/dpkg/lock - open (11: 资源暂时不可用)
E: 无法锁定管理目录(/var/lib/dpkg/)，是否有其他进程正占用它？
root@kali:~/Desktop# cd
root@kali:~# ^C
root@kali:~# sudo rm /var/cache/apt/archives/lock
root@kali:~# sudo rm /var/lib/dpkg/lock
```

然后进行安装**aircrack-ng**

命令如下

```
apt-get install aircrack-ng
```

安装好用，进行爆破，得出密码

```
aircrack-ng -a2 wifi.cap -w password.txt
```

```
root@kali:~# aircrack-ng -a2 wifi.cap -w 1.txt
Opening wifi.cap please wait...
Read 4257 packets.

   #  BSSID              ESSID                     Encryption

   1  3C:E5:A6:20:91:60  CATR                      WPA (0 handshake)
   2  3C:E5:A6:20:91:61  CATR-GUEST                WPA (0 handshake)
   3  BC:F6:85:9E:4E:A3  D-Link_DIR-600A           WPA (1 handshake, with PMKID)

Index number of target network ?

3

Opening wifi.cap please wait...
Read 4257 packets.

1 potential targets



                          Aircrack-ng 1.5.2

      [00:00:04] 9920/9999 keys tested (2438.93 k/s)

      Time left: 0 seconds                                      99.21%

                        KEY FOUND! [ 13910407686 ]


      Master Key     : 31 44 9B 6D 05 DA 32 31 E1 11 72 14 81 9C 3C 66
                       2D 06 E1 6B D6 B4 FC 57 76 4F 63 A9 91 3F A5 9A

      Transient Key  : B9 A4 61 84 0F 16 26 7C 49 EC B6 5C DC 60 61 4A
                       87 2E 21 DB C1 F4 FB 66 AC 72 85 80 8F 5D 8D EE
                       E8 C6 B0 34 14 78 6A F8 F2 E0 BA 90 0E E5 AD 8B
                       5C 0C 6A 4A A2 44 8A 9A FC AE 03 8F EE D8 9C 53

      EAPOL HMAC     : 47 B2 C1 CE 91 63 0D 67 7E 4F 44 5C DA 8B D3 88
```


aircrack-ng使用

```
aircrack-ng -w 字典文件 目标

破解KEY,漫长的过程.漫不漫长取决于两个方面:一是网管的聪明程度(能否设置出复杂的密码),二是电脑的速度.
usage:aircrack-ng[options]<.cap/.ivsfile(s)>

Commonoptions:
-a<amode>:暴破(1/WEP,2/WPA-PSK)

-e<essid>:选择essid为目标

-b<bssid>:选择ap的mac为目标,就是破解识别的关键字

-q:使用安静模式,无数出模式

-C<macs>:将所有的AP合并为一个虚拟的

-help:显示这个帮助
```

十七、linux2

根据题目提示，flag为key{}，那么使用winhex打开文件，并搜索key发现flag.

十八、账号被盗了

点击getf lag，发现没有管理员权限，那么我们就想到cookie,使用抓包工具将cookie的值，改成ture，即可得到一个网站然而发现该网站我怎么也进去，打扰了。

十九、细心的大象

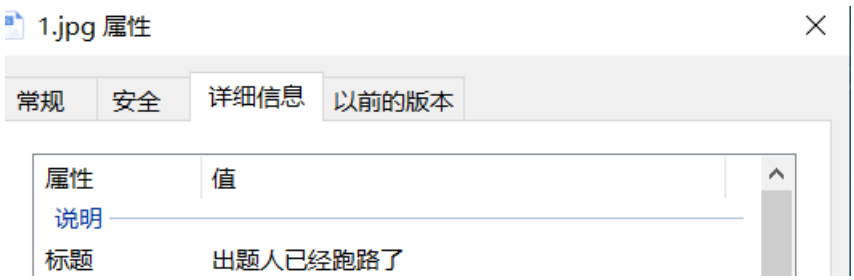
1、下载图片，使用winhex打开没发现什么有用的信息

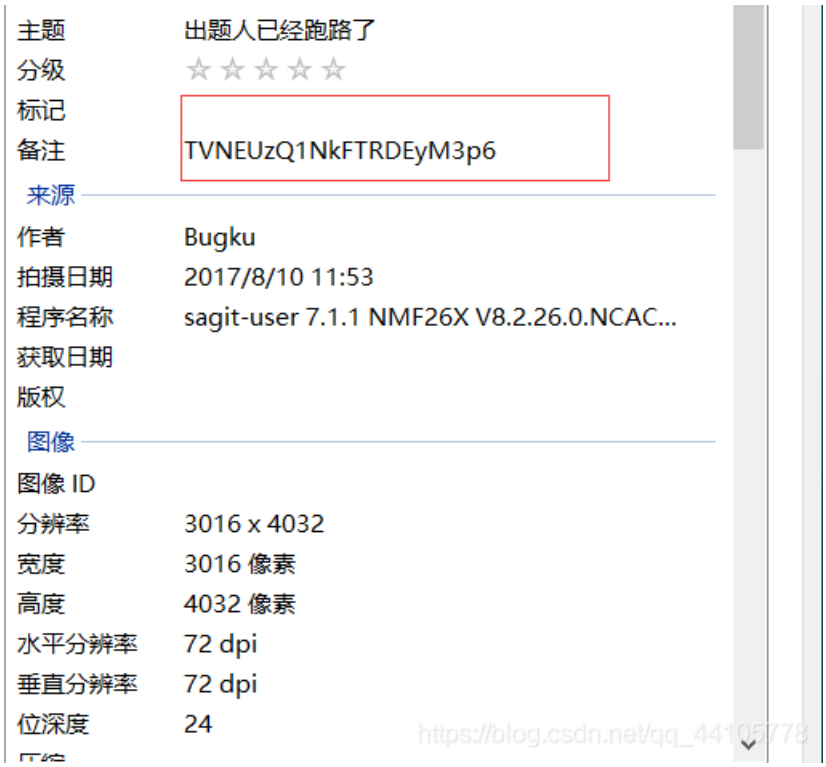
2、用binwalk查看图片是否包含其他文件，成功分离出一个压缩文件

```
C:\Program Files\python36\Scripts>python binwalk -e 1.jpg

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, EXIF standard
12            0xC             TIFF image data, big-endian, offset of first image directory: 8
5005118       0x4C5F3E        PARity archive data

WARNING: Extractor.execute failed to run external extractor 'unrar e '%e'': [WinError 2] 系统找不到指定的文件。, 'u
e '%e'' might not be installed correctly

WARNING: Extractor.execute failed to run external extractor 'unrar -x '%e'': [WinError 2] 系统找不到指定的文件。, '
-x '%e'' might not be installed correctly
5391983       0x6188AF        RAR archive data, version 4.x, first volume type: MAIN_HEAD
```


3、发现该压缩文件里有个图片是加密的，于是我们寻找密码

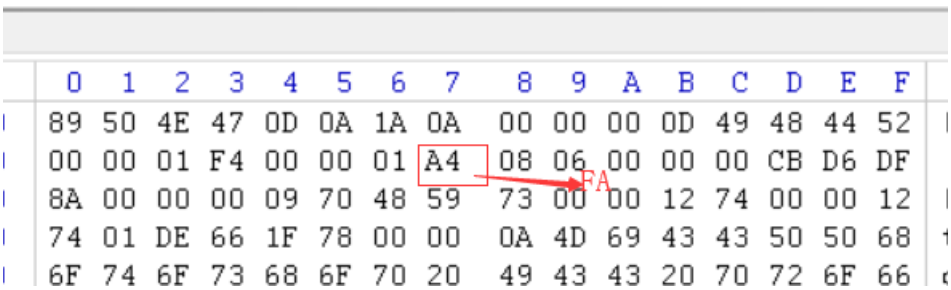
4、在用winhex看一下大象的图片，还是没发现重要信息，我们打开文件的属性，去看看文件的详细信息里面有什么信息，发现一串类似于base64加密的东西。





5、把该串信息当作密码输入，发现失败了，base64解密后，再输入，可以得到以下图片

使用winhex打开修改图片的高为500，就可以在图片里看到flag了

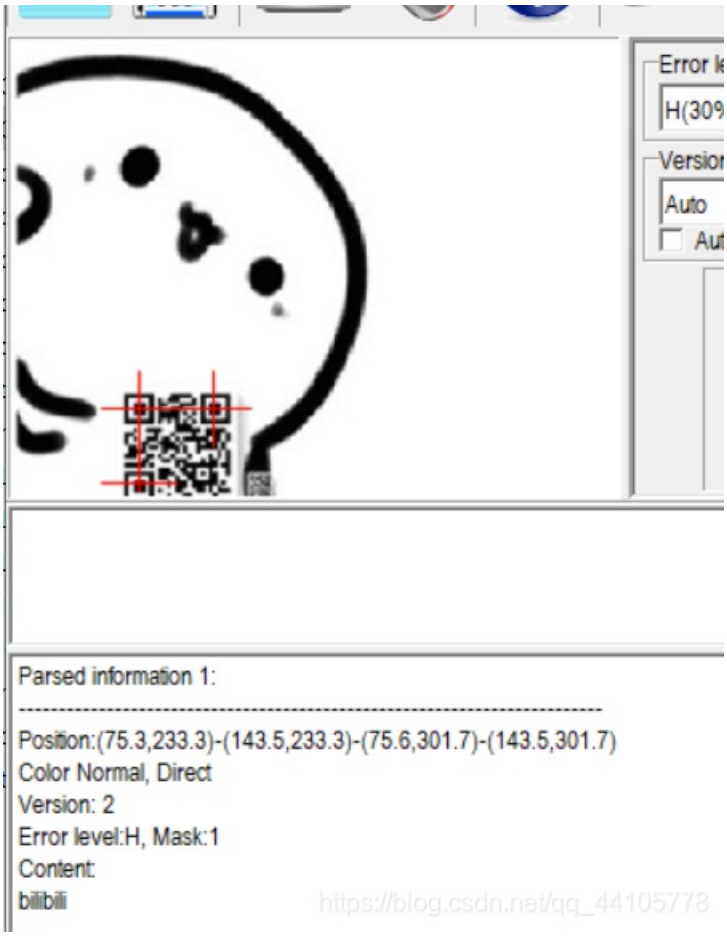


二十、爆照

1、首先对文件进行分析，发现1个压缩包，压缩包里有一个动态图和8个文件。更具文件的大小可以发现88、888、8888比较特殊。flag应该在这3张图片里。

使用winhex打开可以依次打开这3个文件可以发现这3个文件都是jpg文件。修改后缀名为jpg

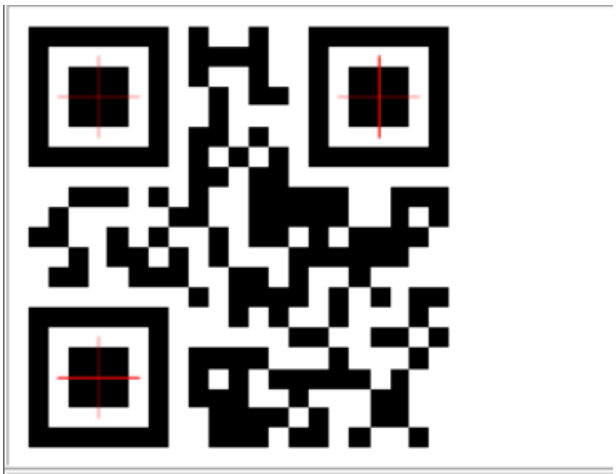
可以发现88文件有个二维码，扫描得bilibili



888文件是个图片，查看文件详细信息可以发现一段base64加密的数据，解密得silisili

分辨率	303 x 299
宽度	303 像素
高度	299 像素
水平分辨率	96 dpi
垂直分辨率	96 dpi
位深度	24

8888文件修改后缀名得到一张图片，详细信息没有有用信息，使用binwalk分析下，发现了一个压缩包。压缩包里有一个二维码图片，扫描得panama



Parsed information 1:

Position:(12.2,12.1)-(267.1,12.1)-(12.2,266.9)-(267.1,266.9)
Color Normal, Direct
Version: 1
Error level:H, Mask:2
Content:
panama

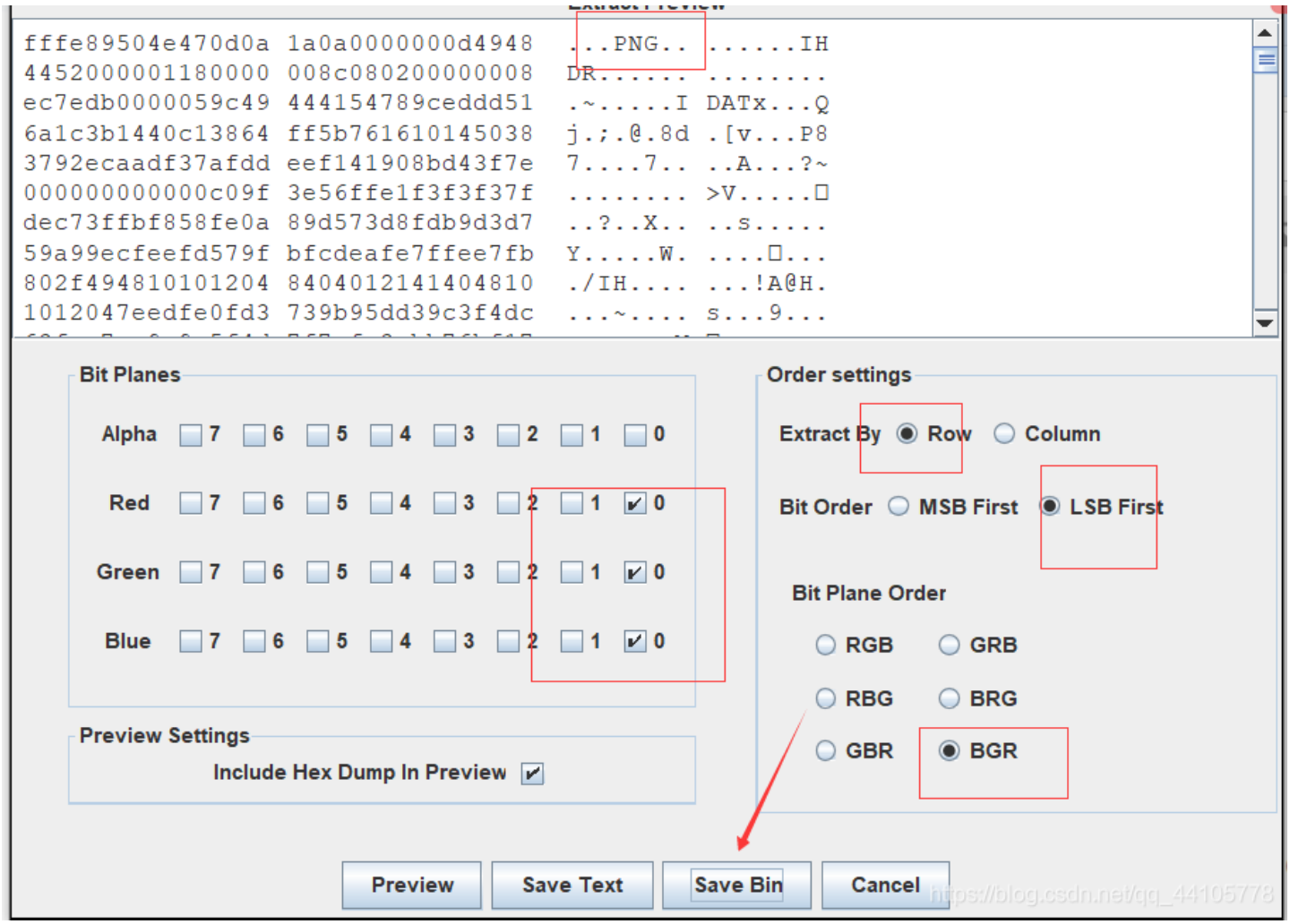
故该题flag为{bilibili_silisili_panama}

二十一、猫片（安恒）

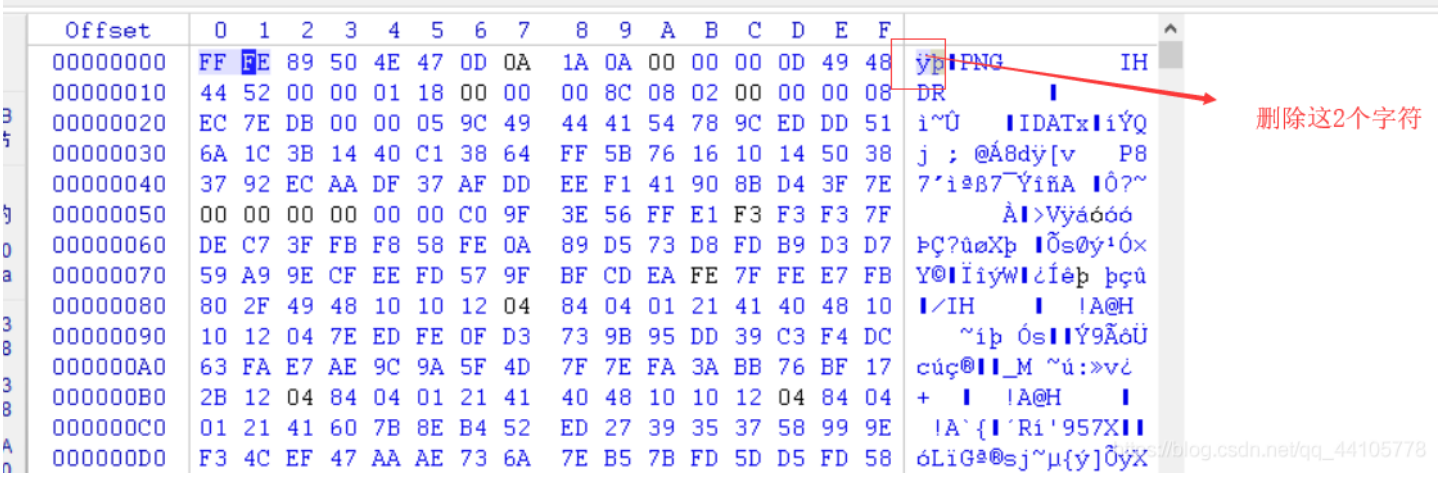
根据题目提示，这样应该是LBS隐写

1、下载附件修改后缀名为png

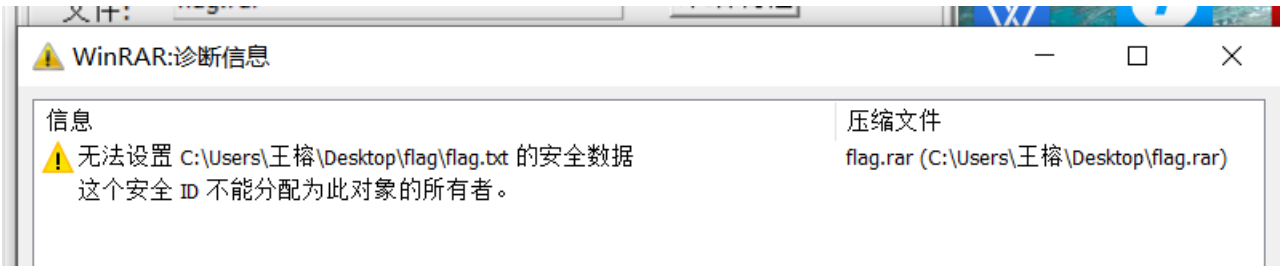
2、使用stegsolve打开，使用其DATA extract功能进行分析，根据提示为LSB,BGR隐写，经过下图操作，保存为png文件。



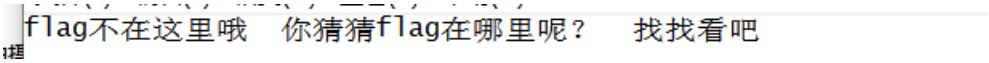
3、发现图片打不开，使用winhex打开查看，进行下图操作，然后保存，发现是半张二维码



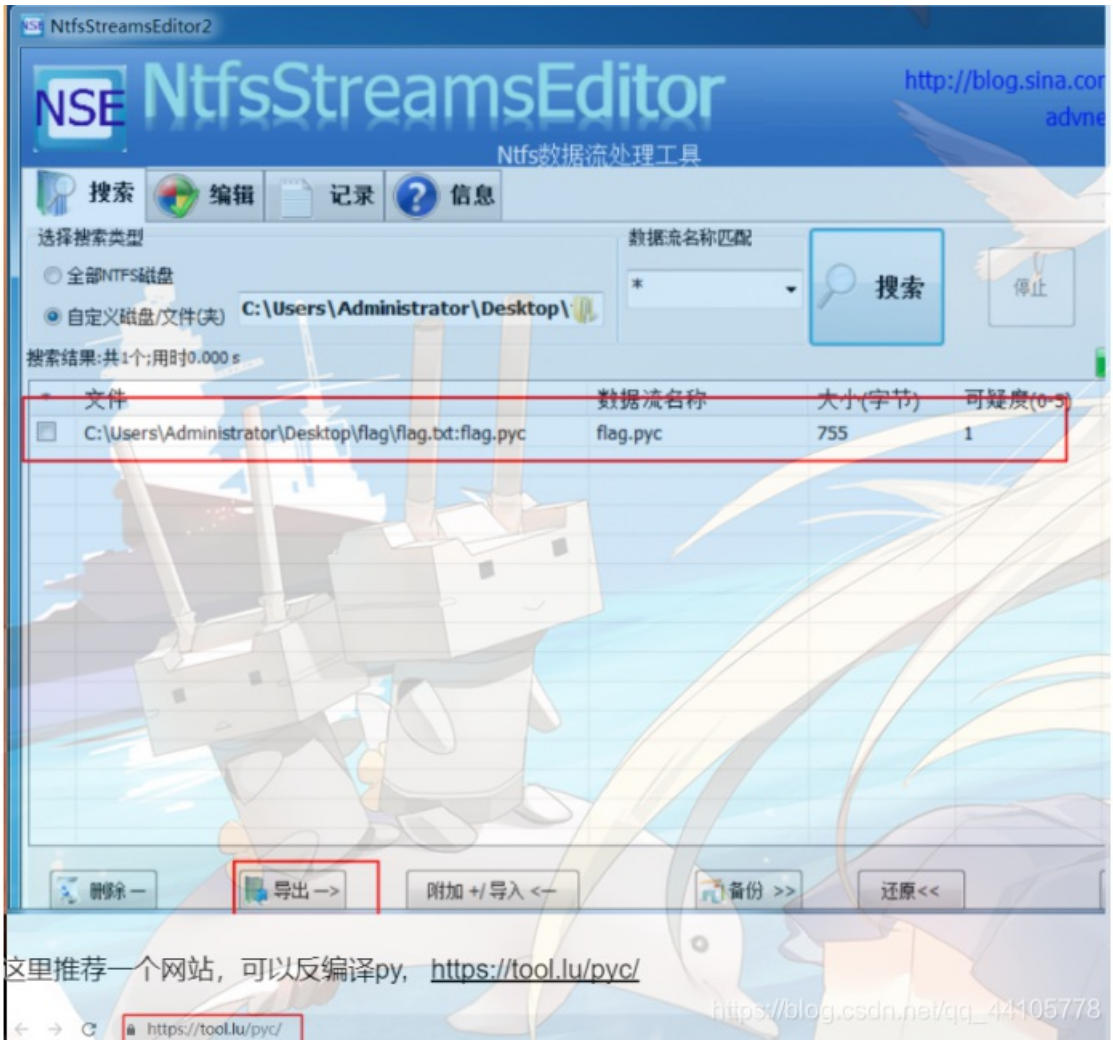
4、进行图片高度的修改，可以得到一张完整的二维码，扫描下载flag.rar压缩包



打开文件，有上面的错误，flag文件里的内容，有点气，费尽脑子，发现无可奈何，于是开始查各种资料。



发现是ntfs文件数据流隐写，就说题目的提示为什么ntfs没用到。
使用ntfsstreamsEditor工具查找数据流，然后导出（注意这边一个坑：flag.rar这个压缩文件一定要用winrar来解压才能找得到数据流）



发现导出的数据流文件是python反编译文件。直接扔到在线我就进行python反编译 ， https://tool.lu/pyc/
得到以下结果

```
import base64

def encode():
    flag = '**************'
    ciphertext = []
    for i in range(len(flag)):
        s = chr(i ^ ord(flag[i]))
        if i % 2 == 0:
            s = ord(s) + 10
        else:
            s = ord(s) - 10
        ciphertext.append(str(s))

    return ciphertext[::-1]

ciphertext = [
    '96',
    '65',
    '93',
    '123',
    '91',
    '97',
    '22',
    '93',
    '70',
    '102',
    '94',
    '132',
    '46',
    '112',
    '64',
    '97',
    '88',
    '80',
    '82',
    '137',
    '90',
    '109',
    '99',
    '112']
```

进行解密脚本的编写

```
def decode():
    ciphertext = [
        '96',
        '65',
        '93',
        '123',
        '91',
        '97',
        '22',
        '93',
        '70',
        '102',
        '94',
        '132',
        '46',
        '112',
        '64',
        '97',
        '88',
        '80',
        '82',
        '137',
        '90',
        '109',
        '99',
        '112']
    ciphertext.reverse()             # 加密中使用ciphertext[::-1]进行取反，故使用reverse取反，
    flag = ''
    for i in range(len(ciphertext)):     # 加密中使用的是flag的长度，而ciphertext与flag长度一样
        if i % 2 == 0:
            s = int(ciphertext[i]) - 10    # 加密中s的值其实就是int(ciphertext[i])
        else:
            s = int(ciphertext[i]) + 10
        s = chr(i ^ s)                     # 加密中为i和flag异或，那么i与s异或即可的到flag
        flag += s
    return flag


if __name__ == '__main__':
    flag = decode()
    print(flag)
```