BugKu-杂项(Misc)的部分Writeup（持续更新，直到刷完）

原创

Starry`Quan 已于 2022-03-18 14:26:19 修改 3712 收藏 10

分类专栏： BugKu 文章标签： 信息安全 经验分享 安全

于 2020-06-17 19:51:06 首次发布

MISC题的部分writeup（持续更新）

~~断更了，太监了，对不起，大佬们~~
第一次写博客，有不好的地方，麻烦大佬指正，我用了些时间把BugKu里的Misc的解题思路和套路整理了一下，发出来分享，基本上是最完全的，各位大佬看完帮忙转发一下呗

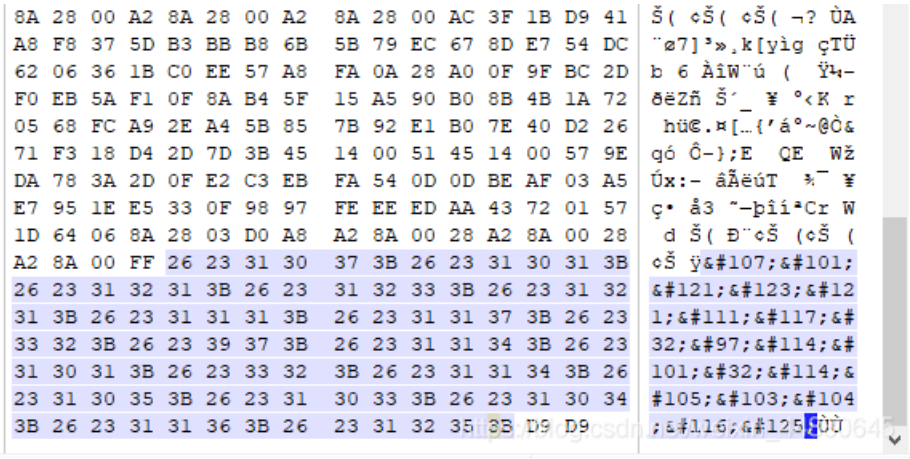
1.签到题

给了一张二维码，扫描关注就得到了flag

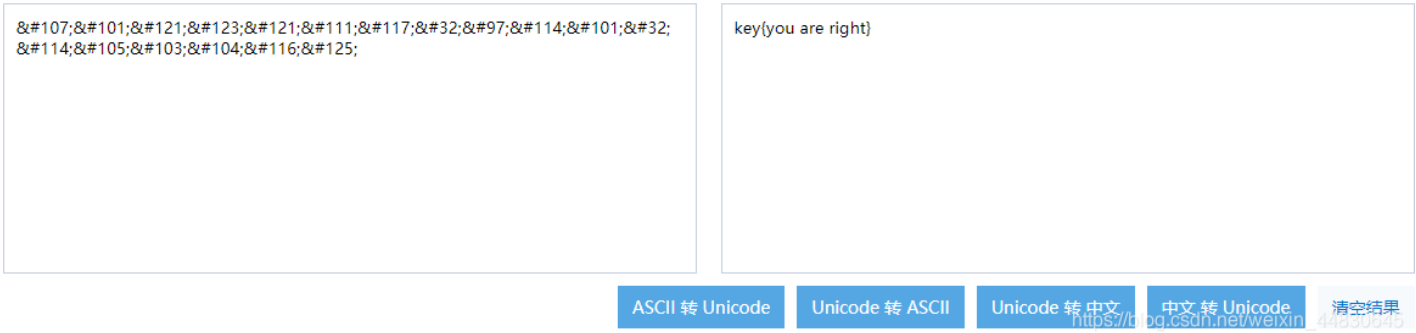
2.这是一张单纯的图片

题目给了一张图片，下载下来，然后用Winhex打开，发现末尾处有一段Unicode编码

```
&#107;&#101;&#121;&#123;&#121;&#111;&#117;&#32;&#97;&#114;&#101;&#32;&#114;&#105;&#103;&#104;&#116;&#125;
```



然后打开一个在线解码的网站解码就可以得到flag



> 这里放出解码的网址：https://tool.chinaz.com/tools/unicode.aspx
> 附件下载地址

3.隐写

把题目给的rar下载下来，解压后得到一张图片，先用了Winhex打开，发现没什么提示，然后又放到kali里面用binwalk检测，没有隐藏文件，最后想到了修改图片的高度

> 修改图片的高度和宽度，binwalk分析一下发现什么也没有藏，那就可能是图片被改“短”了（一般改成和宽度一样，把高去用八进制转换成十六进制，然后去winhex里面找到相同的十六进制，替换成想要的高的十六进制，改宽的原理相同）

先右键查看一下文件的高和宽

图像

分辨率	500 x 420
宽度	500 像素
高度	420 像素
位深度	32

发现宽是500，然后拿去网站转化为16进制

2进制 8进制 10进制 16进制 32进制 36进制 58进制 62进制 | 更多: 10

500 转换

进制	结果	解释
2	111110100	
8	764	
10	500	
16	1f4	
26	tg	小写字母
32	FM	不包含 ILOU 字符
36	dw	数字 + 小写字母
52	jG	大写字母 + 小写字母
58	9C	不包含 0OlI 字符
62	84	数字 + 小写字母 + 大写字母

16进制为01f4，到winhex里面去搜索01f4

```
89 50 4E 47 0D 0A 1A 0A  00 00 00 0D 49 48 44 52 | ‰PNG        IHDR
00 00 01 F4 00 00 01 A4  08 06 00 00 00 CB D6 DF |    ô    ¤     ËÖß
8A 00 00 00 09 70 48 59  73 00 00 12 74 00 00 12 | Š    pHYs   t
74 01 DE 66 1F 78 00 00  0A 4D 69 43 43 50 50 68 | t Þf x   MiCCPPh
6F 74 6F 73 68 6F 70 20  49 43 43 20 70 72 6F 66 | otoshop ICC prof
```

我们把高01A4改成和宽一样的01F4，然后保存退出，重新打开图片，就可以看到flag了

> 附件下载地址

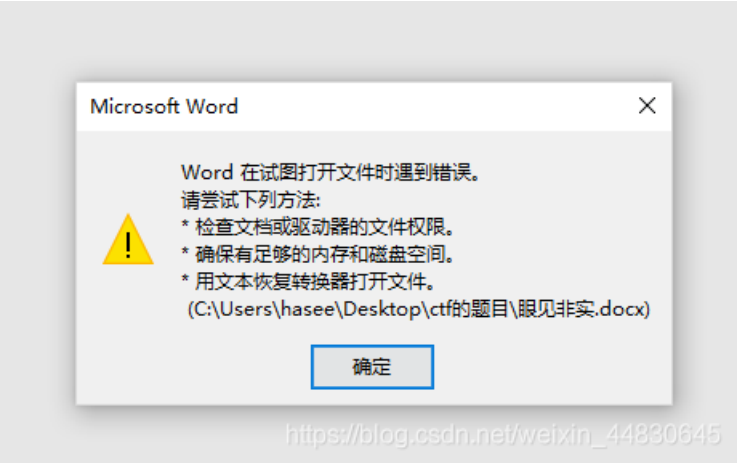
4.telnet

下载zip，解压出来一个数据包，用wireshark打开
筛选telnet协议，然后直接右键追踪TCP流，就可以直接看到flag了

> 附件下载地址

5.眼见非实（ISCCCTF）

下载文件，名字是zip，不管他，直接丢到winhex里面看
发现还真的是zip文件，错怪他了，修改后缀为zip，打开之后发现里面有个docx文件，解压出来打开



打开报错，联想题目，眼见非实，丢到winhex里面去看

```
50 4B 03 04 0A 00 00 00  00 00 E2 20 0F 49 00 00 | PK         â  I
00 00 00 00 00 00 00 00  00 00 09 00 16 00 D1 DB |                 ÑÛ
BC FB B7 C7 CA B5 2F 75  70 12 00 01 19 91 A4 C1 | ¼û·ÇÊµ/up     ‘¤Á
E7 9C BC E8 A7 81 E9 9D  9E E5 AE 9E 2F 50 4B 03 | çœ¼è§ é žå®ž/PK
04 0A 00 00 00 00 00 C1  20 0F 49 00 00 00 00 00 |        Á  I
```

发现也是一个zip文件，修改后缀为zip打开，能解压出来一个文件夹，flag在\眼见非实\word\document.xml文件下，问我怎么知道的？一个一个试出来的

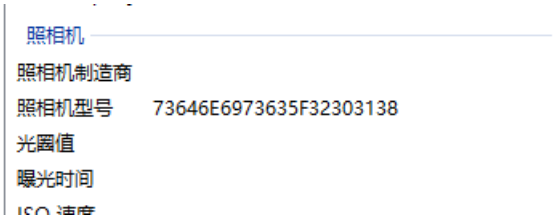
> 附件下载地址

6.啊哒

下载zip文件，能解压出来一个表情包，不多说，直接丢到winhex里面查看

```
7D 52 4D 3E A1 8B 5D 22  00 00 00 16 00 00 00 08  }RM>¡‹]"
00 00 00 66 6C 61 67 2E  74 78 74 3F B1 6E 80 97     flag.txt?±n€—
FE BF 44 3B 35 6A 56 E1  E5 75 A1 C3 C1 A4 2D 93  þ¿D;5jVáåu¡ÃÁ¤-"
5A FA C4 B9 49 B0 AC A9  D5 00 AD 07 73 50 4B 01  ZúÄ¹I°¬©Õ - sPK
02 3F 03 14 03 01 00 00  00 9D 7D 52 4D 3E A1 8B   ?         }RM>¡‹
5D 22 00 00 00 16 00 00  00 08 00 24 00 00 00 00  ]"          $
00 00 00 20 80 A4 81 00  00 00 00 66 6C 61 67 2E      €¤     flag.
74 78 74 0A 00 20 00 00  00 00 00 01 00 18 00 00  txt
F9 C3 77 B6 66 D4 01 00  26 F5 78 B6 66 D4 01 00  ùÃw¶fÔ  &õx¶fÔ
F9 C3 77 B6 66 D4 01 50  4B 05 06 00 00 00 00 01  ùÃw¶fÔ PK
00 01 00 5A 00 00 00 48  00 00 00 00 00           Z   H
```


发现末尾有flag.txt字样，猜测是个压缩包，改后缀后打开压缩包，发现有flag被加密了，一开始猜测是伪加密，但测试后，发现不是，那就只能去找密码了，找了半天没找到，最后抱着试一试的态度，点开了图片的详细属性，发现密码就在里面…



不难看出是一个16进制，直接16进制转文本，得出密码

输入密码解压，得到flag

> 在线解密网站：在线16进制转文本
> 附件下载地址

7.又一张图片，还单纯吗

下载下来，丢到winhex里面无果之后，在linux里面使用命令binwalk查找隐藏文件，发现很多隐藏文件

> 可以用binwalk -e 图片路径 或者 formost 图片路径 的方法把里面的东西分离出来
> 不知道是不是我操作问题，我用binwalk分离不出来，所以就用了fomost

```
root@kali:/mnt/hgfs/ctf的题目# binwalk 2.jpg

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, EXIF standard
12            0xC             TIFF image data, big-endian, offset of first image
 directory: 8
158792        0x26C48         JPEG image data, JFIF standard 1.02
158822        0x26C66         TIFF image data, big-endian, offset of first image
 directory: 8
159124        0x26D94         JPEG image data, JFIF standard 1.02
162196        0x27994         JPEG image data, JFIF standard 1.02
168370        0x291B2         Copyright string: "Copyright (c) 1998 Hewlett-Pack
ard Company"
```


分离出来之后，会在当前目录下生成一个文件夹，在里面能看到flag

> 附件下载地址

8.猜

下载图片，是个半人脸的照片，丢到winhex和linux里面一顿操作，什么都没有发现o(╥﹏╥)o，然后重新看了看题目，感觉是靠这半张脸去猜明星，直接用百度搜图或者谷歌搜图

发现是刘亦菲，按照题目要求格式提交即可

> 附件下载地址

9.宽带信息泄露

下载下来发现是个bin文件，丢到winhex里面看了一番，无果后百度了一下bin文件用什么软件可以打开，结果发现

> .bin 是个万能的后缀，就是说啊，许多软件的作者如果写数据文件的时候（跟我一样）起名困难，很可能不知道该把自己的软件存出来的文件叫做什么格式，于是啊，既然是二进制存储的，不如就叫 .bin 吧。
> 所以，当你看到一个 .bin 格式的文件时，一方面可以先在心里默默咒骂一下软件作者，另外一方面，可以视图通过 magic number 猜测这是个什么文件。unix 下的 file 程序就是专门通过文件头部信息跟 magic number 来猜测文件格式的。猜到之后，就可以试试通过具体的格式是否能够找到对应的软件了。

我有什么好说的呢…
联想题目，应该是路由器的配置文件，用routerpassview软件打开，题目要求提交用户名，所以搜索字段为username

把username所对应的val提交即可

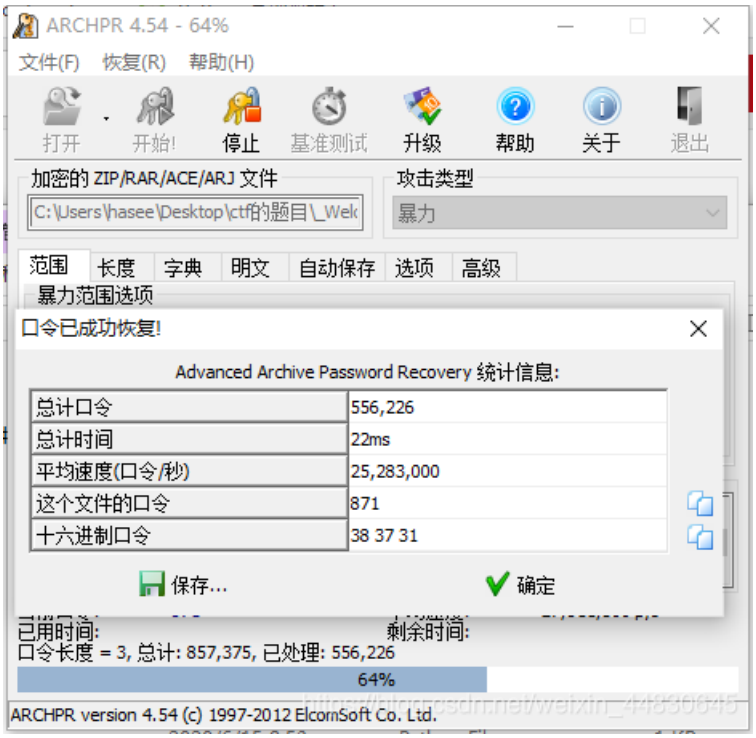
> routerpassview这个软件，火绒会报毒，如果要保险起见，可以去虚拟机运行ヾ(๑°▽°๑)ﾉ
> 附件下载地址

10.隐写2

下载图片下来，发现这个图片有点猖狂呀！不管他，改盘他还得盘他
用binwalk可以看到有隐藏文件，可以用 binwalk -e 图片路径 分离出来

```
root@kali:/mnt/hgfs/ctf的题目# binwalk -e Welcome_.jpg

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, JFIF standard 1.01
30            0x1E            TIFF image data, big-endian, offset of first image
 directory: 8
52516         0xCD24          Zip archive data, at least v1.0 to extract, compre
ssed size: 6732, uncompressed size: 6732, name: flag.rar
59264         0xE780          End of Zip archive, footer length: 22
147852        0x2418C         End of Zip archive, footer length: 22

root@kali:/mnt/hgfs/ctf的题目#
```


可以分理处一个flag.rar，和提示.jpg，说解压密码为3位数，直接用ARCHPR破解



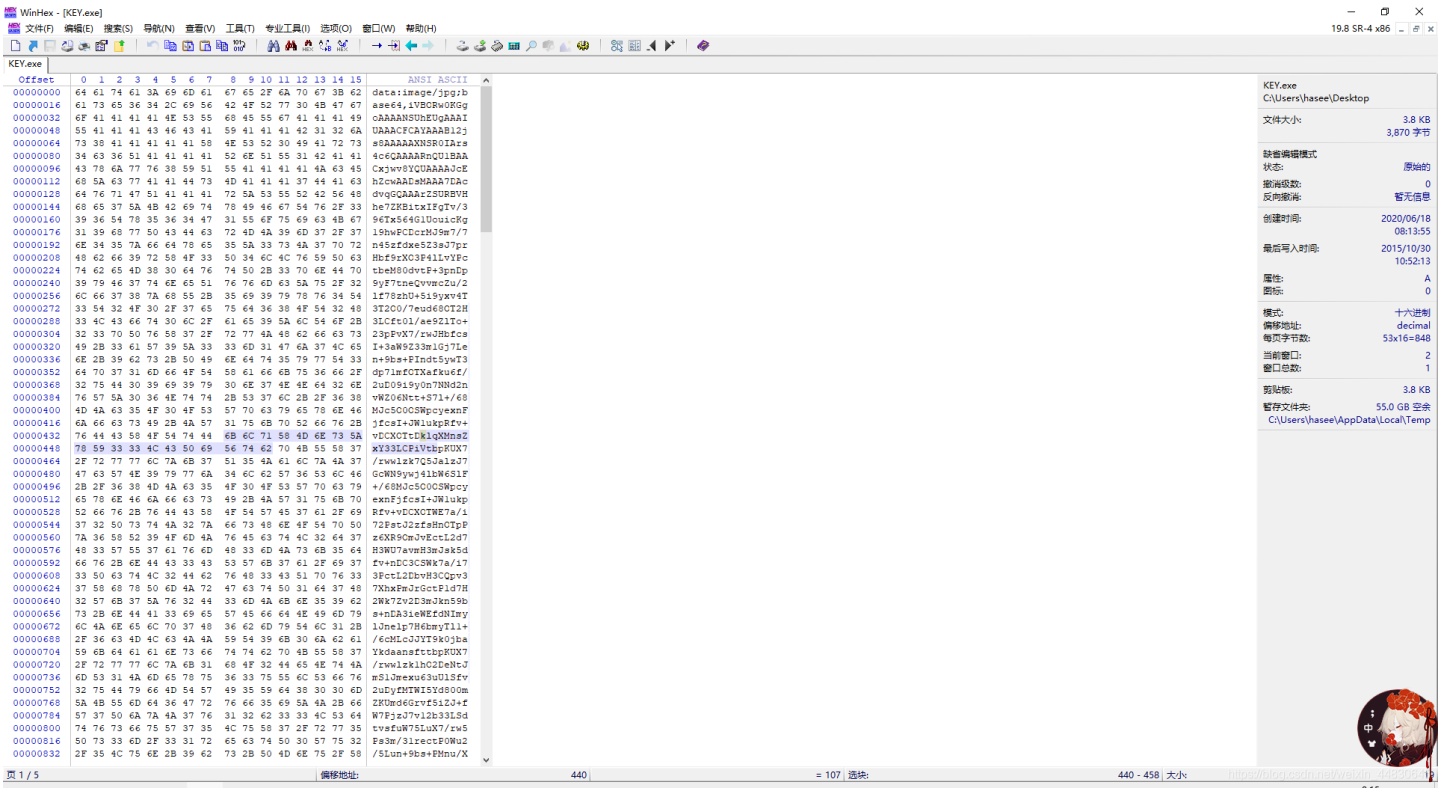
成功爆破出密码，密码为：871
解压出3.jpg，用winhex打开，在末尾发现flag，flag被base64加密了，解密即为正确flag

> 这里给出解密网站：在线base64加密解密
> 附件下载地址

11.多种方法解决

下载得到一个exe文件，发现运行不了

所以就用Winhex打开，发现格式就是base64转图片的格式



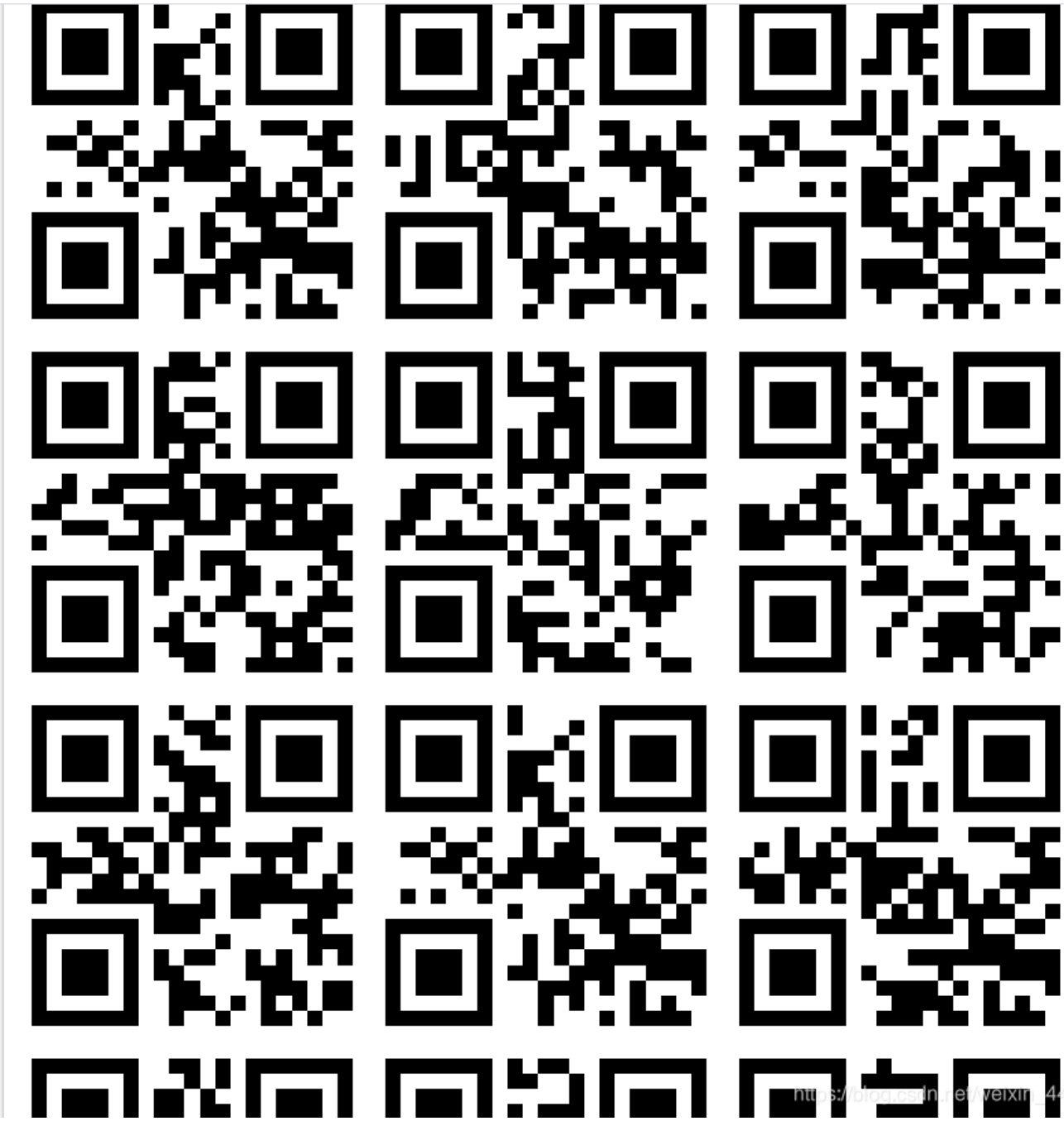
所以就放到网站里面去转成图片

得到了一个二维码，扫描即可

> 这里放出转换的网址：base64在线转图片
> 附件下载地址

12.闪的好快

给了一个gif图，可以用网站把他们分离开来

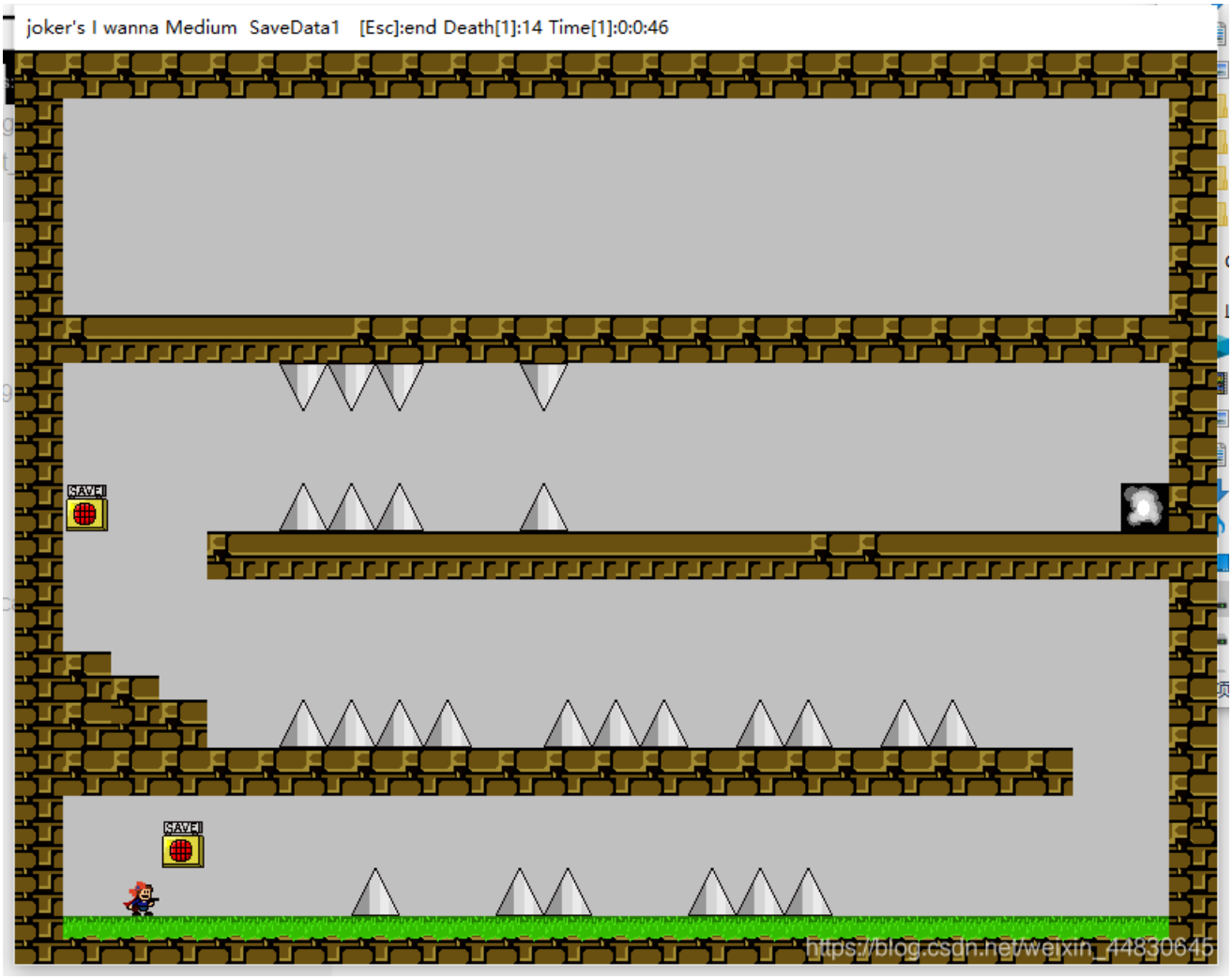


菜鸡的我只能一个一个扫了，扫完可以得到flag

> 这题有点麻烦，就把flag给出来SYC{F1aSh_so_f4sT}
> 这里给出图片分离的网站gif分离
> 附件下载

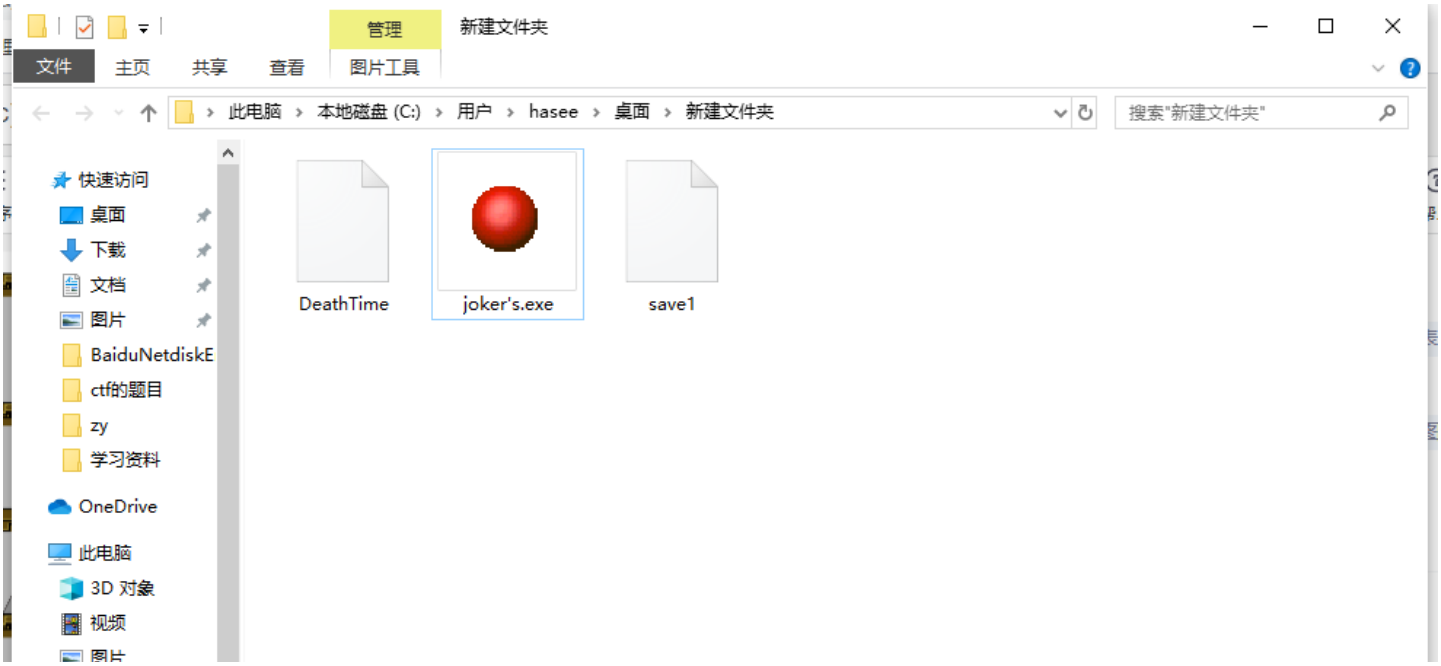
13.come_game

打开zip解压出来一个exe文件，双击打开，发现是一个自己以前玩过的游戏

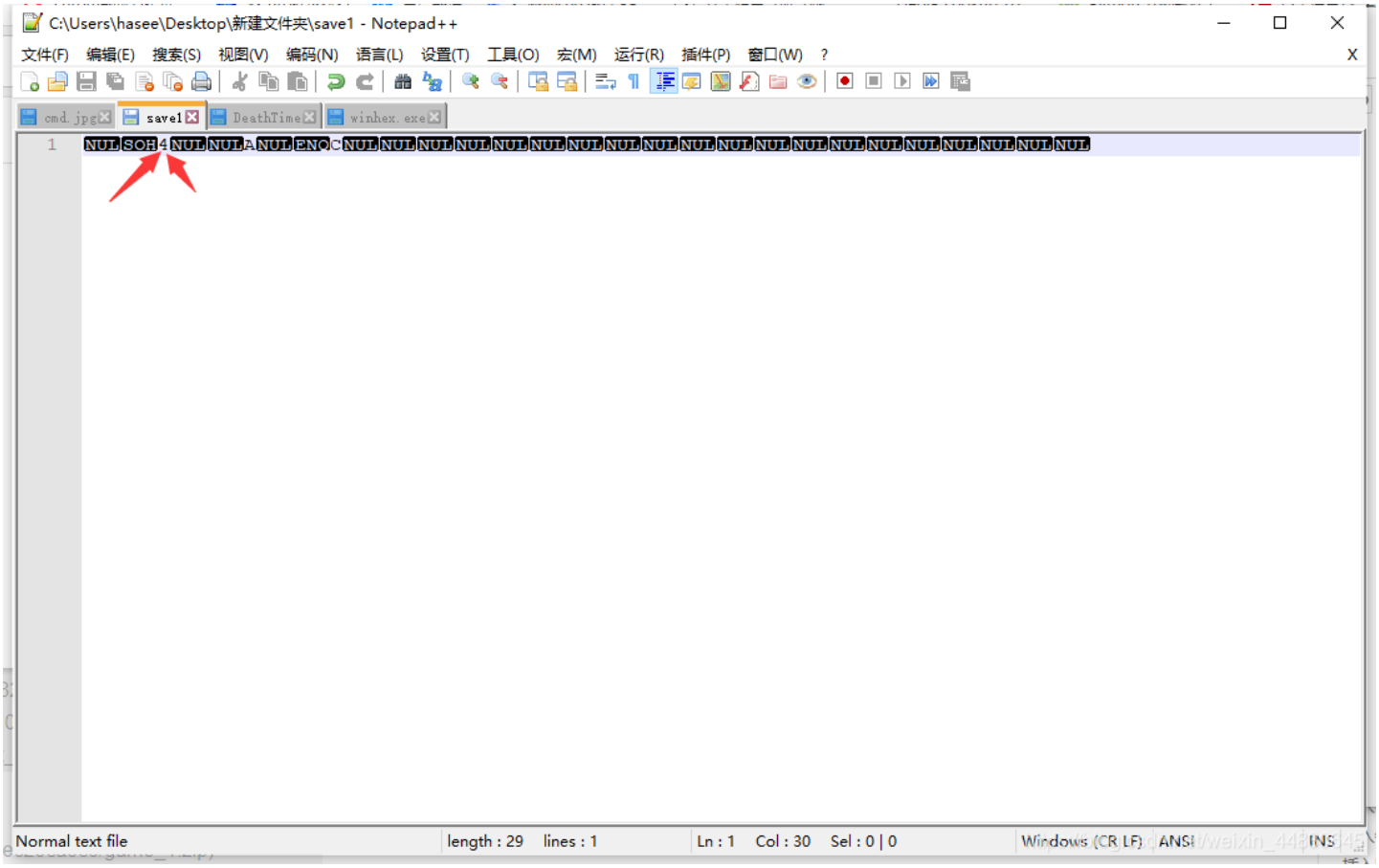


> 以前就把我玩吐了，这次又尝试了上百条命，没过关...

玩了几条命后，发现没用，就退出游戏了
回到游戏文件夹，发现游戏创建了两个文件



有个叫save的，猜测里面是关卡数



把4改成5，重新进游戏，发现还真的是关卡数，得到flag

附件下载地址

14.白哥的鸽子

下载下来得到一个名叫jpg的文件，先不管他，直接用winhex打开，拉到最后，发现了一串奇怪的字符串

```
6A 13 76 93 78 8D D4 BB  9F 12 F3 B4 E5 5C 8C F4
C3 35 0F A6 72 10 97 10  63 88 4E 21 5D 43 0C 91
5C 08 66 08 A5 04 24 44  48 CA C1 11 FA 8A B1 E0
DE 52 03 94 80 6C 8E D9  3D 49 A0 78 C8 2D E1 8B
EE 3D FF 00 B8 F7 A2 76  C7 41 30 56 02 CB 22 75
13 B4 67 FF 00 88 8D 7E  A2 BA 88 EB C6 3B FF 00
C4 F5 FF 00 10 AF 8C 11  82 6A 1F 92 12 05 A2 03
44 0B 50 10 24 C1 B9 65  F3 1C 22 0D CE C1 1D C8
8D 47 DA 3E D1 EE CF 94  1A E6 2A C5 8E 3C F8 00
EA 03 A8 35 12 39 F0 8E  6C A2 9E 1D 66 E2 BB 87
74 F7 4B 65 B0 58 2F 01  3A 92 BF 1E 73 2A C7 49
E6 03 A7 9D 14 11 1D 79  D0 9D 28 0E A5 1D 40 20
78 DC 59 69 DA 8F 64 6E  E6 7B A3 57 31 EE 8D DC
CB 62 45 62 89 EE 5B DC  B6 73 01 E3 FF D9 66 67
32 69 76 79 6F 7D 6C 7B  32 73 33 5F 6F 40 61 77
5F 5F 72 63 6C 40
```

```
j v‰x Ô»Ÿ ó´å\ŒôÃ5 ¦r – c^N!]C '\ f ¥ $DHÊÁ úŠ±àÞR ”€lŽÜ=I xÈ-á‹î=ÿ ¸÷¢vÇA0V Ë"u ´gÿ ^ ~¢°^ëÆ;ÿÄðÿ ¯Œ ,j ’ ¢D P $Á²eó " ÎÁ È GÚ>ÑîÏ” æ*ÅŽ<øê ¨5 9ðŽl¢ž fâ»‡t÷Ke°X/ :’¿ s*ÇIæ § yÐ ( ¥ @xÜYiÚ dnæ{£Wlî ÜËbEb‰î[Ü‰s ãÿÙfg2ivyo}l{2s3_o@aw__rcl@
```


看这样子，猜测是栅栏密码，就去解密网站解密

栅栏密码加密解密

fg2ivyo}l{2s3_o@aw__rcl@

每组字数 3 加密 解密

flag{w22_is_v3ry_cool}@@

提交发现是错误的，去掉两个@@之后提交

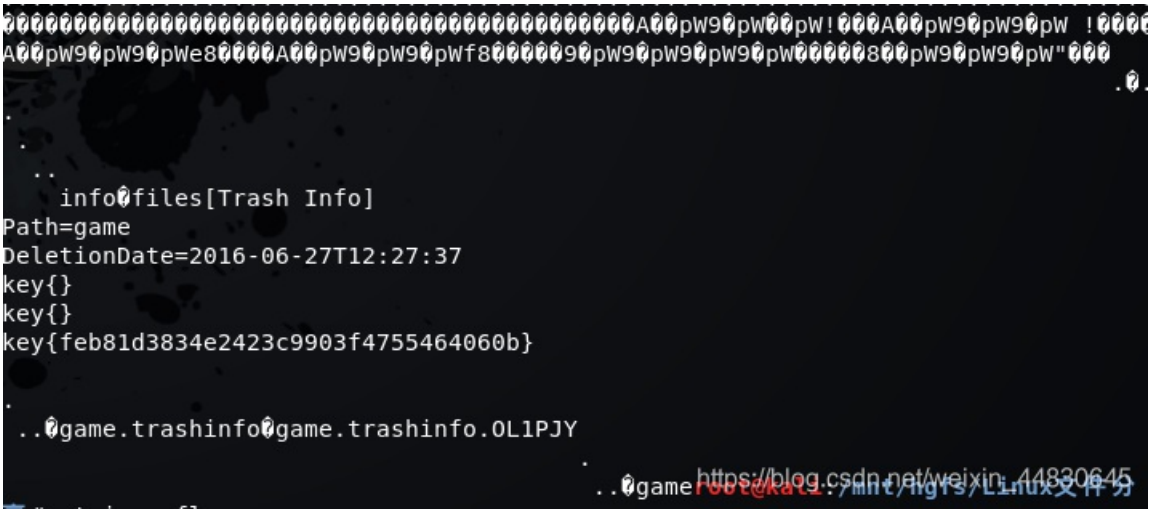
附件下载

15.linux

解压得到一个名叫flag的文件
这道题有多重做法，这里说两种

第一种做法

用cat命令，直接输入cat flag会冒出一大串的字符，拉倒最底下，能发现flag



第二种做法

用strings命令，输入strings flag，能查找flag这个文件里面的所有字符串，同样的，在最后可以发现flag

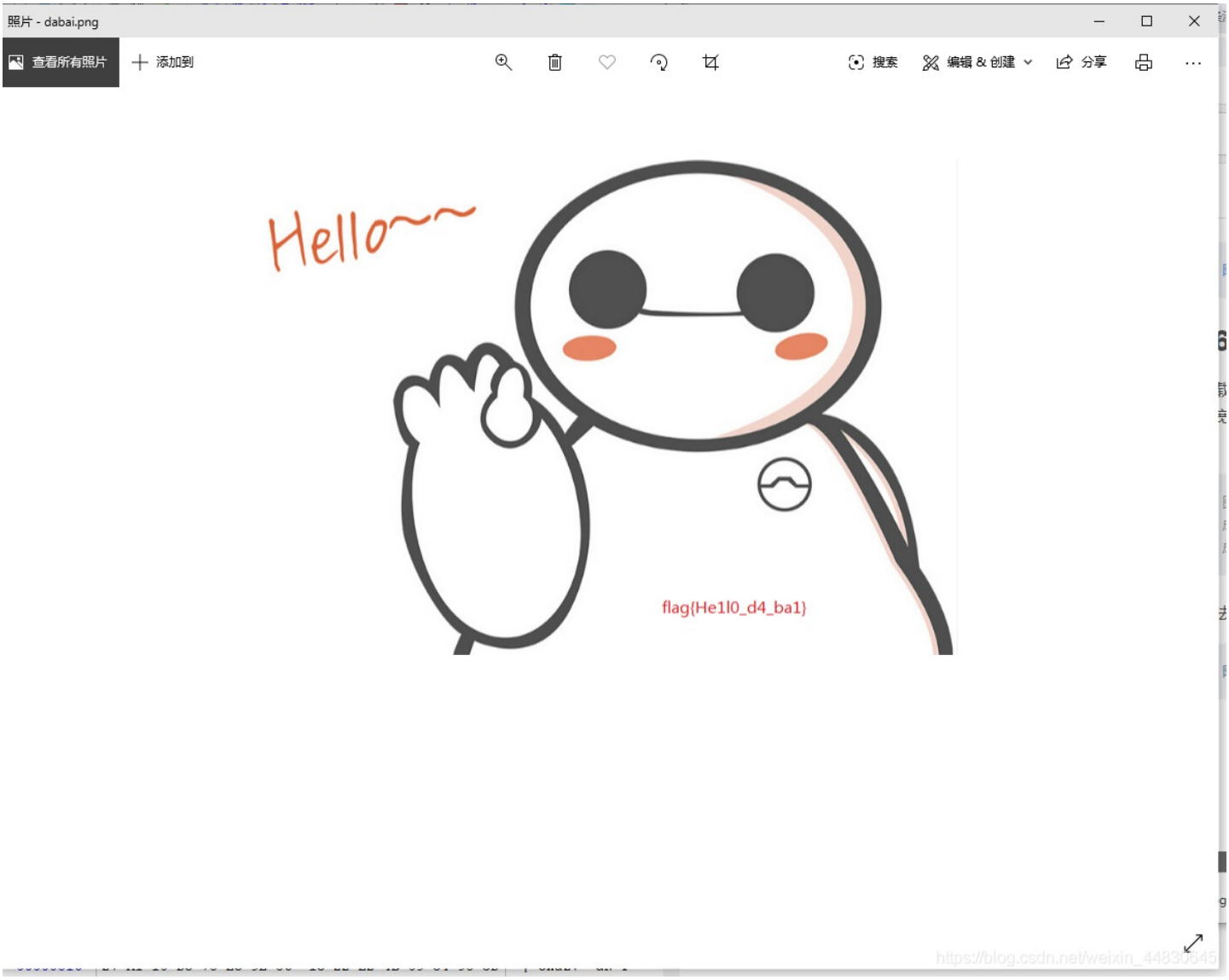
> 附件下载

16.隐写3

下载下来得到一张叫dabai的png图片，右键看了属性，发现宽跟高很不匹配，猜测是被改“短”了，所以用winhex打开

> 图片被改“短”了（一般改成和宽度一样，把高去用进制转换成十六进制，然后去winhex里面找到相同的十六进制，替换成想要的高的十六进制，改宽的原理相同）

做法大致和第3题相同，想知道做法的，跳回去看看就行了，做出来的结果如下图

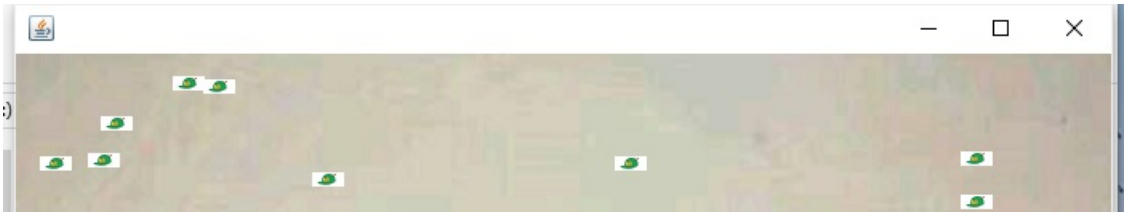


> 附件下载

17.做个游戏（08067CTF）

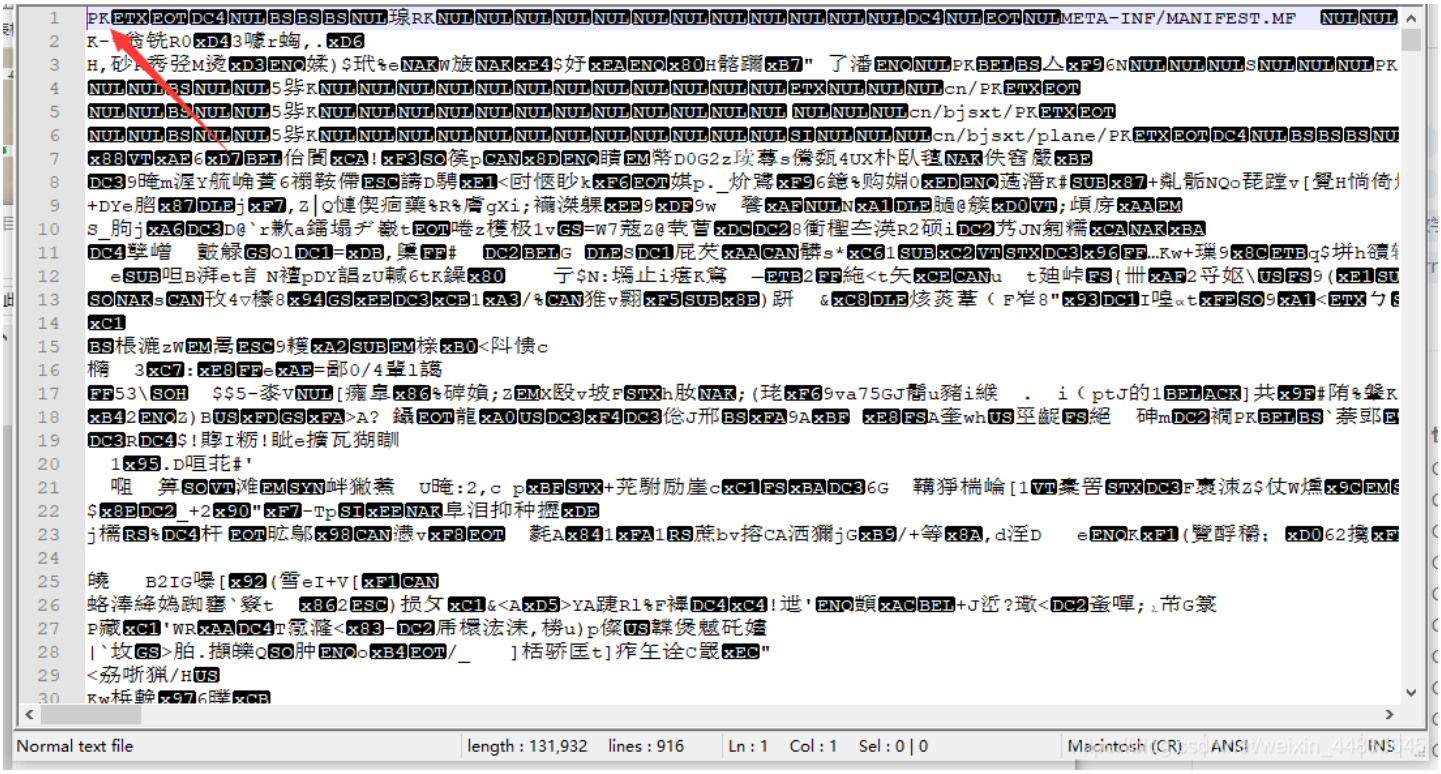
下载下来发现是个jar文件，玩了一把

> 注意打开这个文件需要有java的环境



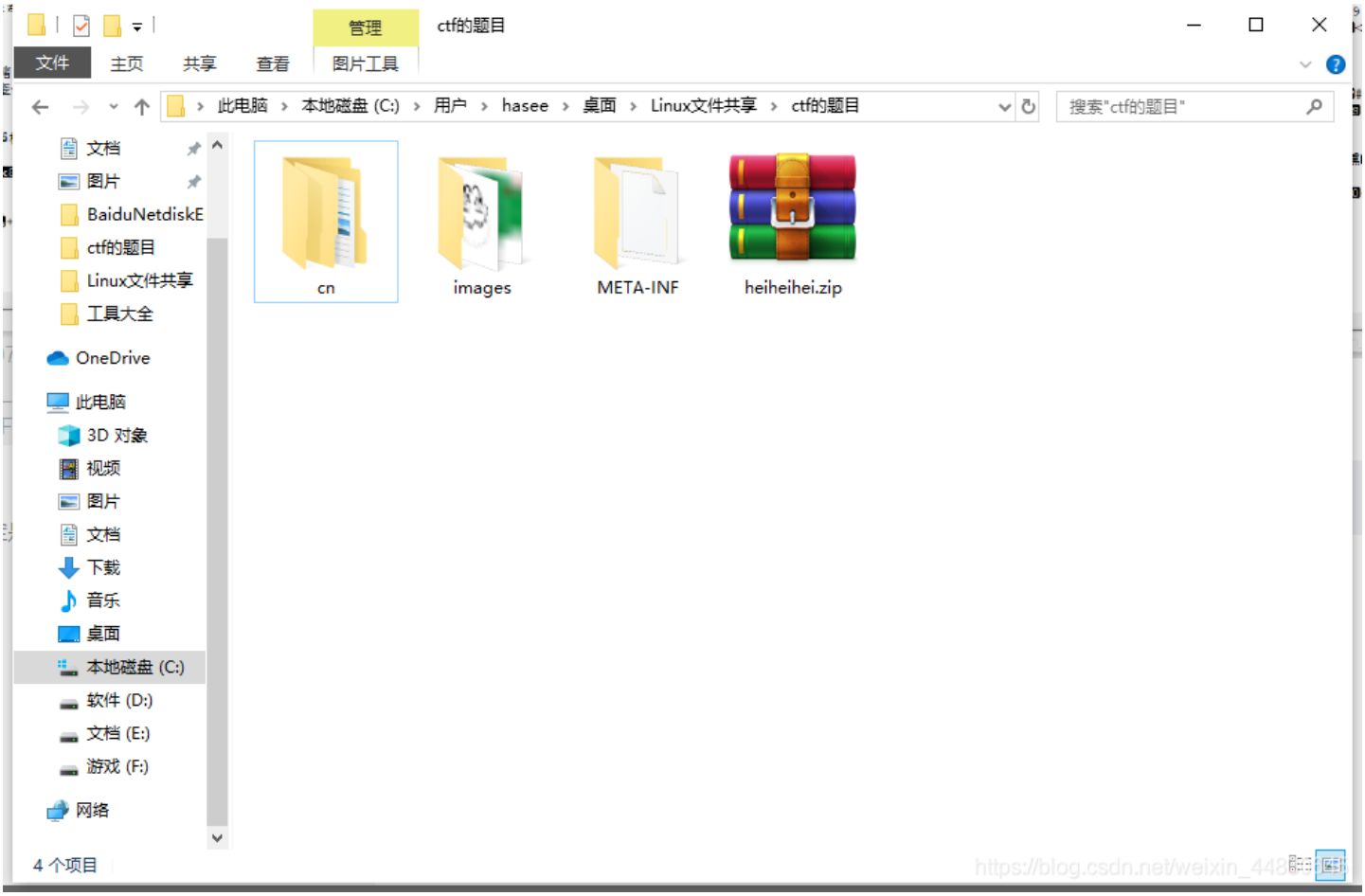
发现他居然嘲讽我，不可饶恕，必须得盘他

右键用Notepad++打开

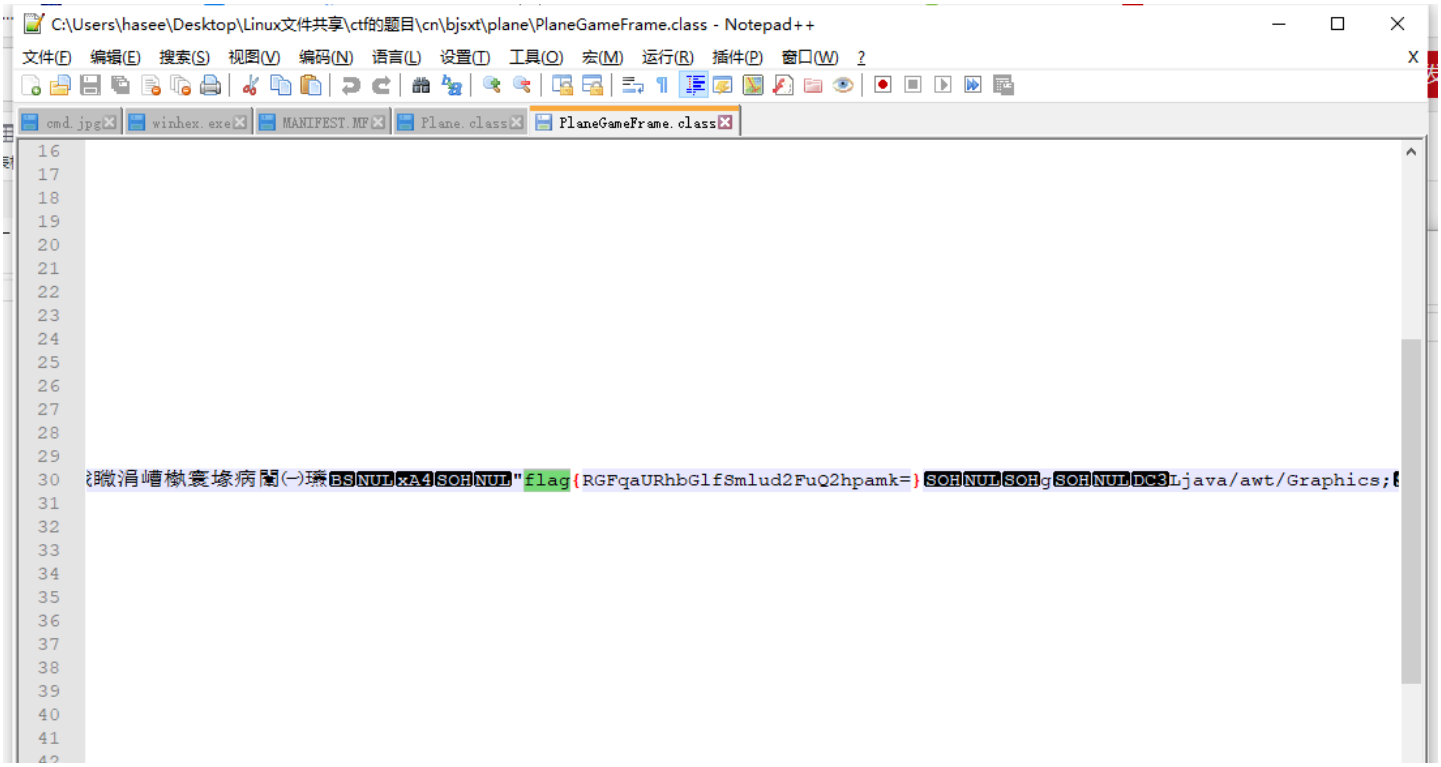


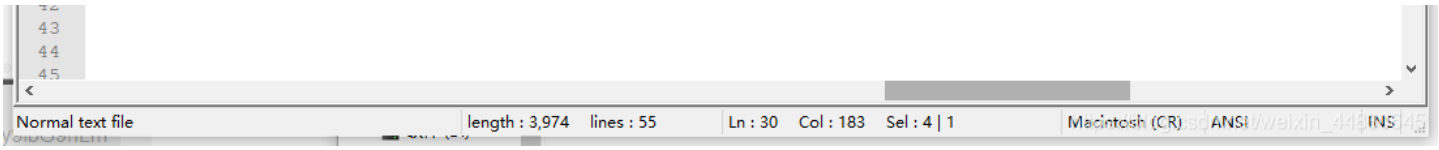
第一眼就看到了PK头，直接修改后缀为zip，解压出来三个文件夹

> 因为zip算法的创始人的名字缩写是PK，所以看到PK就基本可以断定是zip文件了



flag在cn\bjsxt\plane\PlaneGameFrame.class文件下，因为PlaneGameFram翻译过来是游戏框架的意思，所以用Notepad++打开搜索字符串flag就可以得到flag



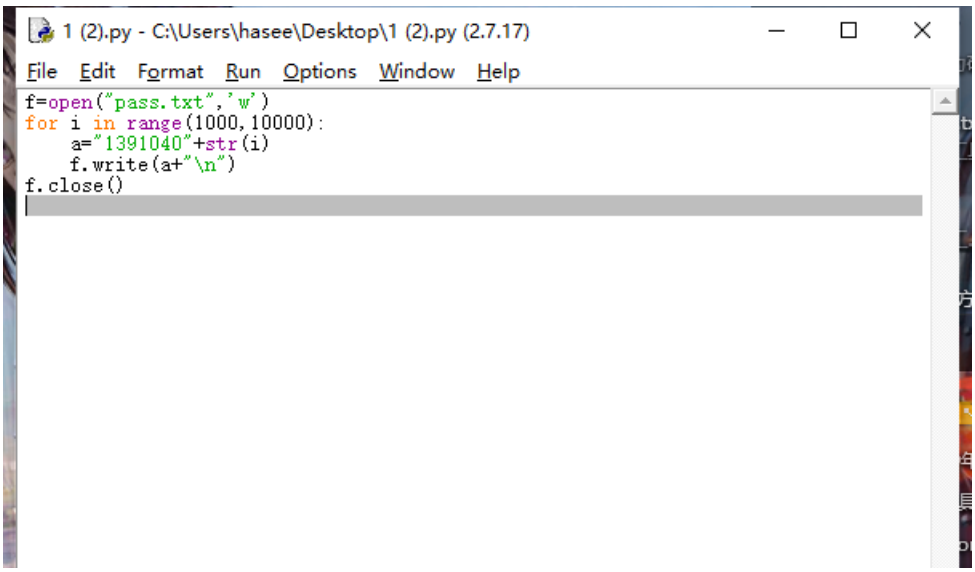


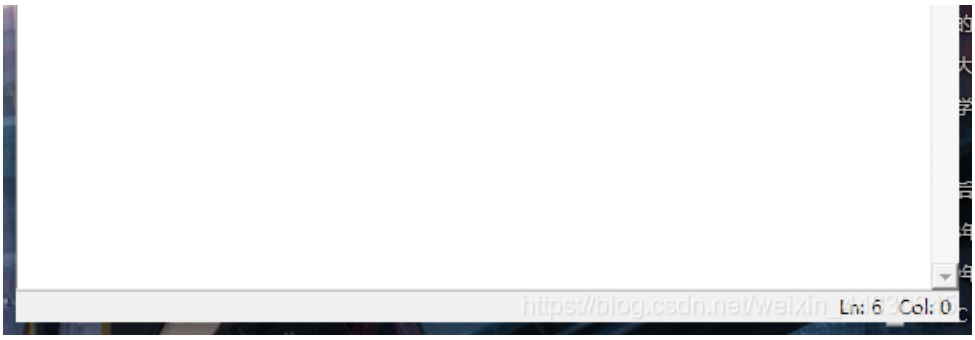
这里要注意的是，flag是base64位加密过的，要提交解密后的flag才能得分

> base64解密网站

18.想蹭网先解开密码

下载下来得到了一个数据包，再看了看题目的提示
知道了是破解wifi密码，先用软件或者python生成一个11位数的字典





在kali里面用aircrack-ng -w pass.txt wifi.cap这条命令就可以开始爆破了

破出来密码为13910407686，按题目要求格式提交即为flag

> 附件下载

19.Linux2

把题目下载下载，是个文件，放到kali里面用binwalk扫描一下，发现很多隐藏的文件

```
root@kali:/mnt/hgfs/Linux文件分享/ctf的题目# binwalk brave

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             Linux EXT filesystem, rev 1.0, ext3 filesystem dat
a, UUID=cf6d7bff-c377-403f-84ae-956ce3c9e3c9
4127744       0x3EFC00        Executable script, shebang: "/usr/bin/env bash"
4127746       0x3EFC02        Unix path: /usr/bin/env bash
8388608       0x800000        Linux EXT filesystem, rev 1.0, ext3 filesystem dat
a, UUID=cf6d7bff-c377-403f-84ae-956ce3c9e3c9
9298944       0x8DE400        Linux EXT filesystem, rev 1.0, ext3 filesystem dat
a, UUID=cf6d7bff-c377-403f-84ae-956ce3c9e3c9
9308160       0x8E0800        Linux EXT filesystem, rev 1.0, ext3 filesystem dat
a, UUID=cf6d7bff-c377-403f-84ae-956ce3c9e3c9
9324544       0x8E4800        Linux EXT filesystem, rev 1.0, ext3 filesystem dat
a, UUID=cf6d7bff-c377-403f-84ae-956ce3c9e3c9
9345024       0x8E9800        Linux EXT filesystem, rev 1.0, ext3 filesystem dat
a, UUID=cf6d7bff-c377-403f-84ae-956ce3c9e3c9
13712384      0xD13C00        JPEG image data, JFIF standard 1.01

root@kali:/mnt/hgfs/Linux文件分享/ctf的题目#
```


> 用bindwalk -e brave把隐藏的东西分离出来，或者用formost分离出来

进入分离出来的文件夹，找一下，就可以找到flag，flag在_brave.extracted\ext-root\o8\huas.txt文件下

做法二

根据题目的提示，可以直接用grep命令查找字符串"KEY"

```
root@kali:/mnt/hgfs/Linux文件分享/ctf的题目# grep 'KEY' -a brave
KEY{24f3627a86fc740a7f36ee2c7a1c124a}
U3210#"! Utp@ad@@@KEY{}@ N@@F@  @@@2V@@S@@@2V
```

> Linux系统中grep命令是一种强大的文本搜索工具，它能使用正则表达式搜索文本，并把匹 配的行打印出来。grep全称是Global Regular Expression Print，表示全局正则表达式版本，它的使用权限是所有用户

提交即可

附件下载，提取码：s7jy

20.账号被盗了

这道题，我去的时候，网页已经404了…

想知道做法的话，可以去看别人的writeup

21.细心的大象

下载下来是一张图片，结合题目，细心，猜测是右键看图片的详细信息

果不其然在备注那里有一串疑似base64编码的东西，解密之后是一串疑似密码的字符串

Base64编码转换

Base64编码转换

TVNEUzQ1NkFTRDEyM3p6

清空 加密 解密 ☐解密结果以16进制显示

MSDS456ASD123zz

复制

然后去kali里面用binwalk查看有没有隐藏文件，发现有一个rar的文件然后把他们分离出来

```
root@kali:/mnt/hgfs/Linux文件分享/ctf的题目# binwalk 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
6391983       0x6188AF        RAR archive data, version 4.x, first volume type:
MAIN_HEAD

root@kali:/mnt/hgfs/Linux文件分享/ctf的题目# 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

6391983       0x6188AF        RAR archive data, version 4.x, first volume type:
MAIN_HEAD

root@kali:/mnt/hgfs/Linux文件分享/ctf的题目#
```


打开发现是一个加密了的压缩包，那密码肯定就是我们刚刚解密出来的字符串了，解压出来之后，是一张什么提示都没有的图片，猜测是图片被改短了

我们把01A4改成和宽一下的01F4就可以看到flag了

BUGKU{a1e5aSA}

附件下载

22.爆照(08067CTF)

把图片下载下来，发现是一个动漫图片（手动滑稽）缘什么空的，挺不错的，推荐看看（doge）
不多说了，直接开始做题，我看着题目，一开始还以为是ms08067呢，想了想发现不可能，把他放到kali里面用binwalk查看一下有没有隐藏的文件，发现有东西，把它们分离出来

```
root@kali:/mnt/hgfs/Linux文件分享/ctf的题目# binwalk -e 8.jpg

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, JFIF standard 1.01
40499         0x9E33          Zip archive data, encrypted at least v2.0 to extra
ct, compressed size: 8362, uncompressed size: 92278, name: 8
48892         0xBEFC          Zip archive data, at least v2.0 to extract, compre
ssed size: 14906, uncompressed size: 15739, name: 88
63830         0xF956          Zip archive data, at least v2.0 to extract, compre
ssed size: 11129, uncompressed size: 18479, name: 888
74992         0x124F0         Zip archive data, at least v2.0 to extract, compre
ssed size: 10371, uncompressed size: 11782, name: 8888
85397         0x14D95         Zip archive data, at least v2.0 to extract, compre
ssed size: 6945, uncompressed size: 92278, name: 88888
92377         0x168D9         Zip archive data, at least v2.0 to extract, compre
ssed size: 6824, uncompressed size: 92278, name: 888888
99237         0x183A5         Zip archive data, at least v2.0 to extract, compre
ssed size: 7076, uncompressed size: 92278, name: 8888888
106350        0x19F6E         Zip archive data, at least v2.0 to extract, compre
ssed size: 8219, uncompressed size: 92278, name: 88888888
168452        0x29204         End of Zip archive, footer length: 22
```

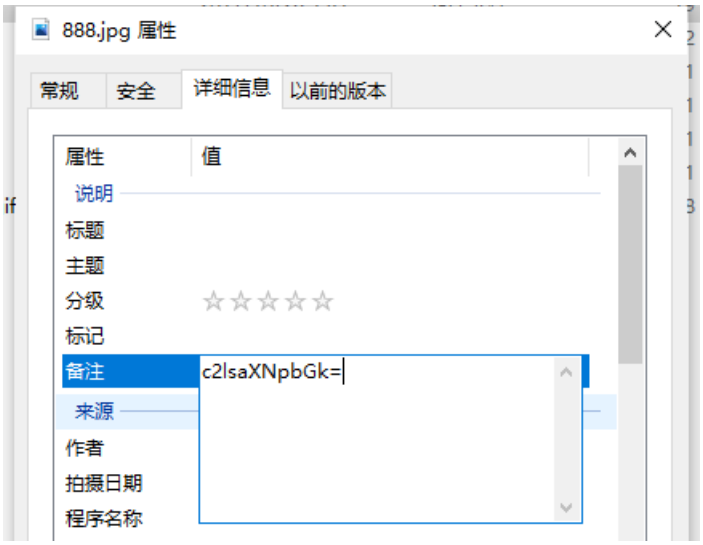

能分离出来一个zip压缩包，打开发现就一些文件和一个gif动图，叫我慢慢排序

解压出来，用file命令查看了一下，都是jpg文件

```
root@kali:/mnt/hgfs/Linux文件分享/ctf的题目/_8.jpg.extracted# file 88
88: JPEG image data, JFIF standard 1.01, resolution (DPI), density 96x96, segmen
t length 16, Exif Standard: [TIFF image data, big-endian, direntries=1, software
=www.meitu.com], baseline, precision 8, 303x300, components 3
root@kali:/mnt/hgfs/Linux文件分享/ctf的题目/_8.jpg.extracted# file 888
888: JPEG image data, JFIF standard 1.01, resolution (DPI), density 96x96, segme
nt length 16, Exif Standard: [TIFF image data, big-endian, direntries=3], baseli
ne, precision 8, 303x299, components 3
root@kali:/mnt/hgfs/Linux文件分享/ctf的题目/_8.jpg.extracted# file 8888
8888: JPEG image data, JFIF standard 1.01, resolution (DPI), density 96x96, segm
ent length 16, Exif Standard: [TIFF image data, big-endian, direntries=1, softwa
re=www.meitu.com], baseline, precision 8, 293x303, components 3
root@kali:/mnt/hgfs/Linux文件分享/ctf的题目/_8.jpg.extracted#
```


那就一个一个改后缀成jpg咯，然后一张一张图片查看，发现88.jpg包含一个二维码

扫描能得到bilibili，提交发现是错的，猜测是不完全，看了全部图片的详细信息后，发现888图片有备注



拿去base64解码，能得到silisili
然后用binwalk一个一个找，会发现8888的图片有藏一个压缩包

```
root@kali:/mnt/hgfs/Linux文件分享/ctf的题目/_8.jpg.extracted# binwalk 8888.jpg

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, JFIF standard 1.01
30            0x1E            TIFF image data, big-endian, offset of first image
 directory: 8
10976         0x2AE0          Zip archive data, at least v2.0 to extract, compre
ssed size: 644, uncompressed size: 1202, name: 1509126368.png
11760         0x2DF0          End of Zip archive, footer length: 22

root@kali:/mnt/hgfs/Linux文件分享/ctf的题目/_8.jpg.extracted#
```

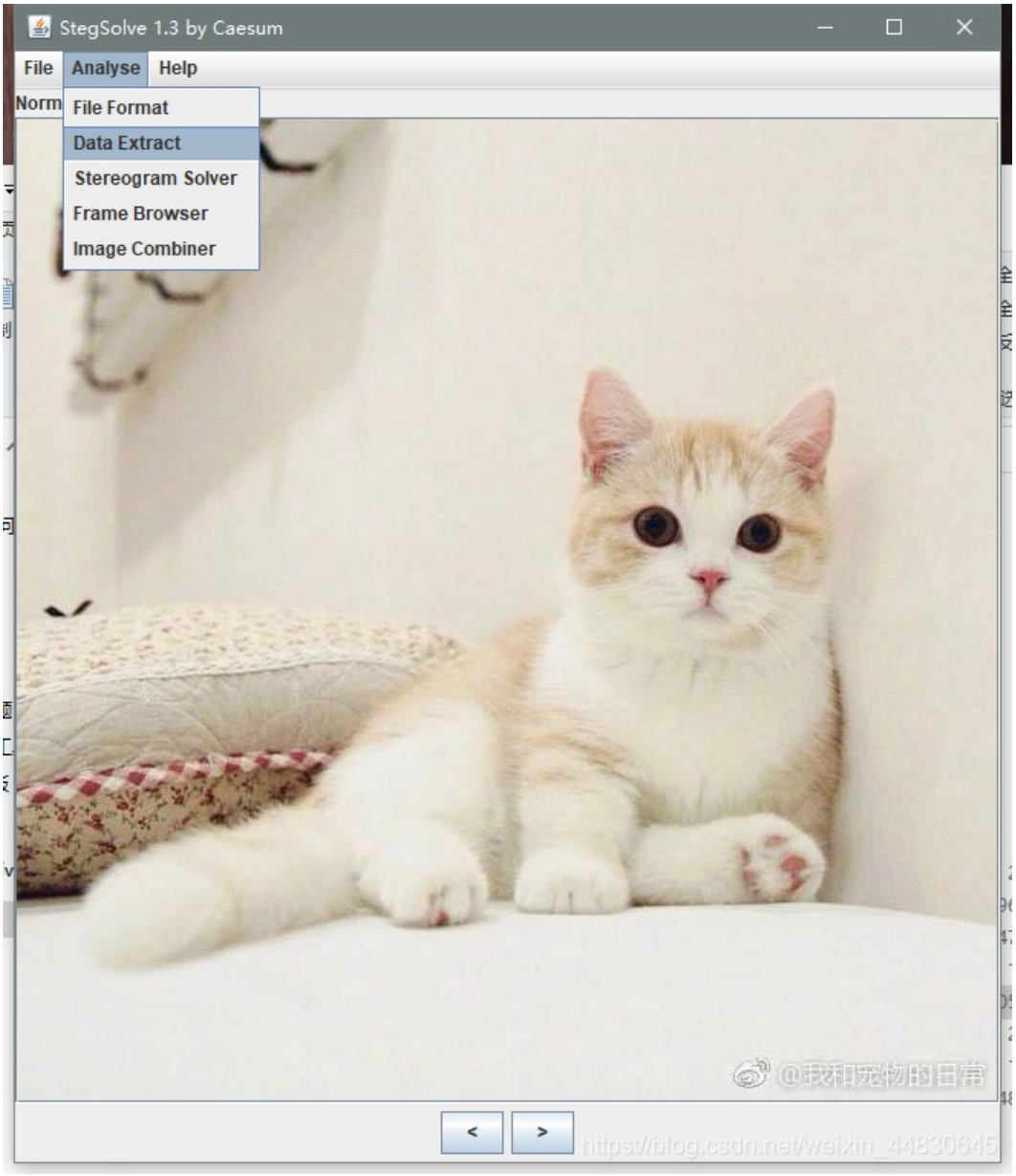

把他们分离出来，能分离出来一个二维码，扫描能得到panama，其他的图片一切正常， 猜测这三个就是flag的组成，然后结合之前的提示，叫我慢慢排序，多试几次就可以得到flag为flag{bilibili_silisili_panama}，提交即可

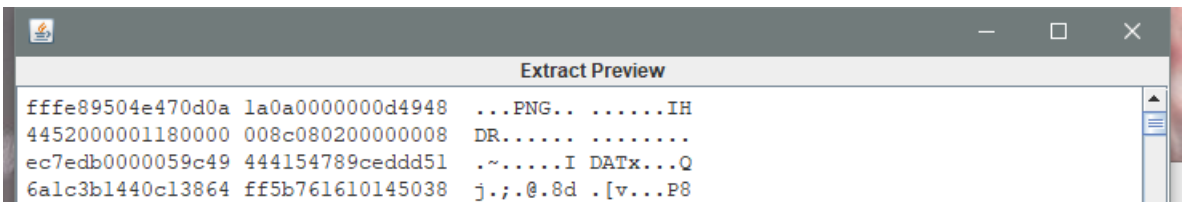
> 附件下载

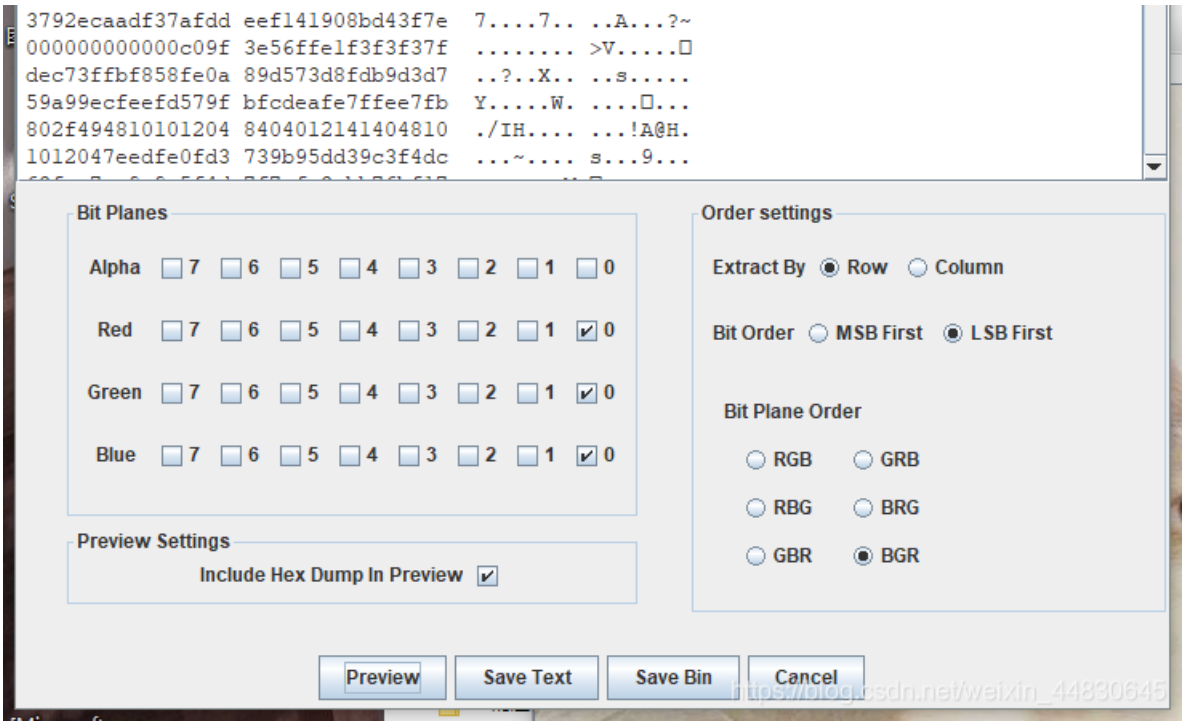
23.猫片（安恒）

下载下来是一个名字叫png的图片，根据以往的经验，改png后缀直接打开

用binwalk分析之后无果，遂用stegsolve打开，并且根据题目给的提示LSB BGR NTFS，设置图片属性







把Red Green Blue三项设为0，不显示他们三种颜色，Alpha不用设置，因为它代表了透明度，如果设置为0了，那么图片就看不到了

能看到是一个png的文件，按Save Bin把它保存为png文件，然后把FFFE删去，保留PNG头

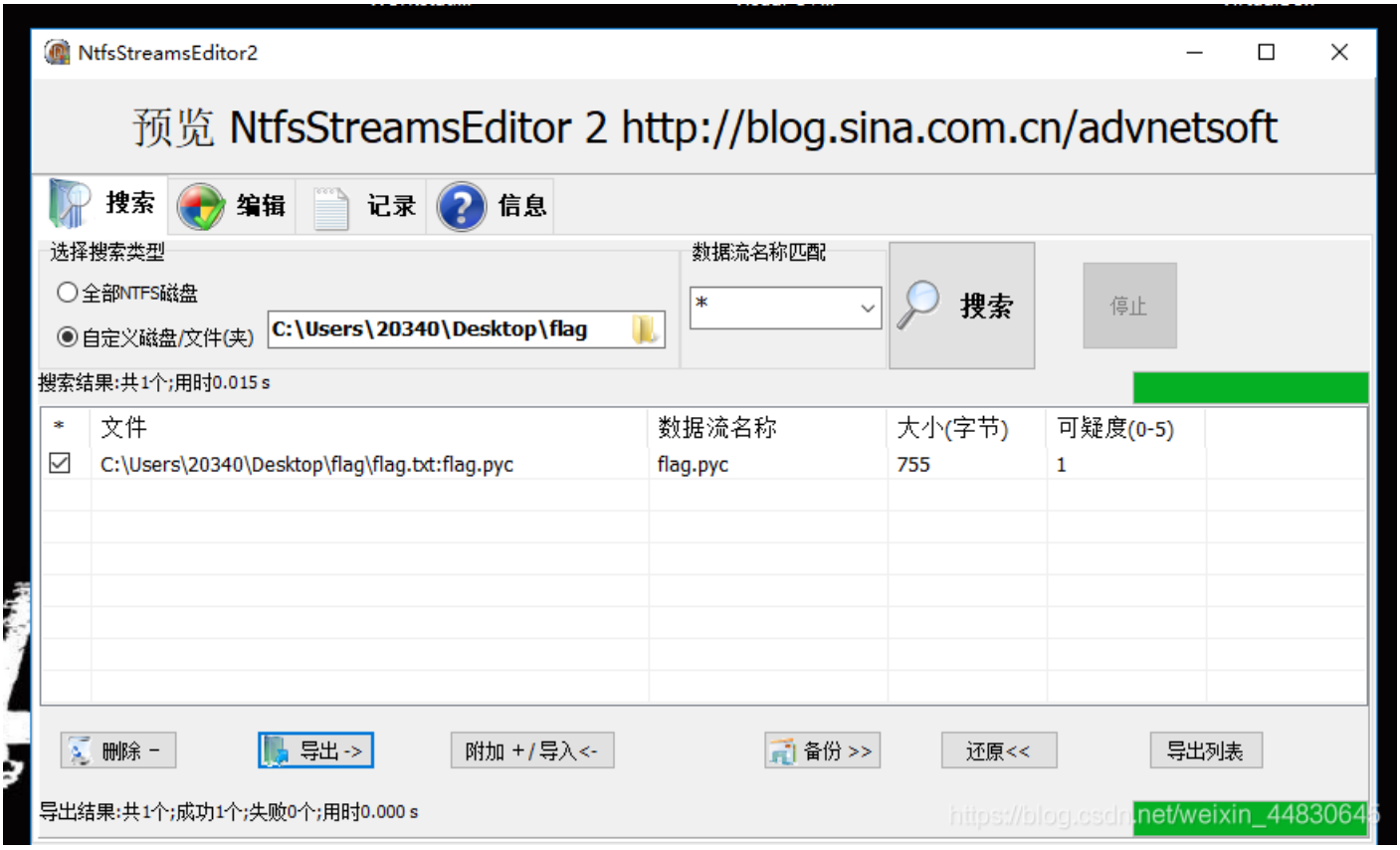
打开图片发现是半截二维码

然后修改图片的高度，关于方法我前面第三题有讲，这里就不再赘述

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	ANSI ASCII
89	50	4E	47	0D	0A	1A	0A	00	00	00	0D	49	48	44	52	%PNG IHDR
00	00	01	18	00	00	01	18	08	02	00	00	00	08	EC	7E	ì~
DB	00	00	05	9C	49	44	41	54	78	9C	ED	DD	51	6A	1C	Û œIDATxœíÝQj
3B	14	40	C1	38	64	FF	5B	76	16	10	14	50	38	37	92	; @Á8dÿ[v P87'

修改好之后扫描可以得到一个百度网盘下载地址

下载下来之后，发现怎么样都找不到真的flag，去翻了翻大佬的博客，发现要用NtfsStreamsEditor2这个工具打开才可以看到文件，但是我怎么样都看不到，在此只能搬运这位大佬的博客了



去pyc反编译网站把这个反编译一下可以得到

```
# uncompyle6 version 3.4.0
# Python bytecode 2.7 (62211)
# Decompiled from: Python 3.7.3 (default, Mar 27 2019, 17:13:21) [MSC v.1915 64 bit (AMD64)]
# Embedded file name: flag.py
# Compiled at: 2017-12-05 23:42:15
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',
# okay decompiling C!_Users_20340_Desktop_flag_flag.txt!flag.pyc
```

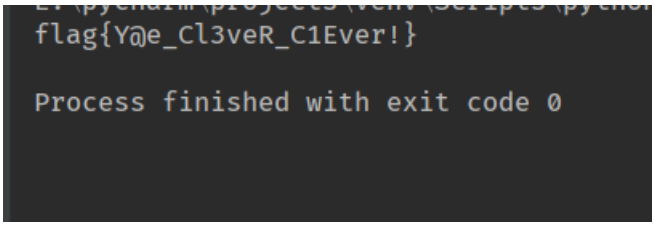

写一个解密脚本解决

注意两点：

> encode（）只是演示一下ciphertext的由来，也就是演示了如果flag='*******' 加密的过程，让你明白加密过程 然后现在给你一个真的flag的ciohertext 求解flag，解密的脚本一定是按照加密脚本 反过来写 也就是按加密脚本从下往上走。一层层解密。

这里给出解密的脚本

```
import base64
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']
def decode():
flag = ''
ciphertext.reverse()
for i in range(len(ciphertext)):
if i % 2 == 0:
s = int(ciphertext[i]) - 10
else:
s = int(ciphertext[i]) + 10
s = chr(i^s)
flag = flag + s
print(flag)
decode()
```

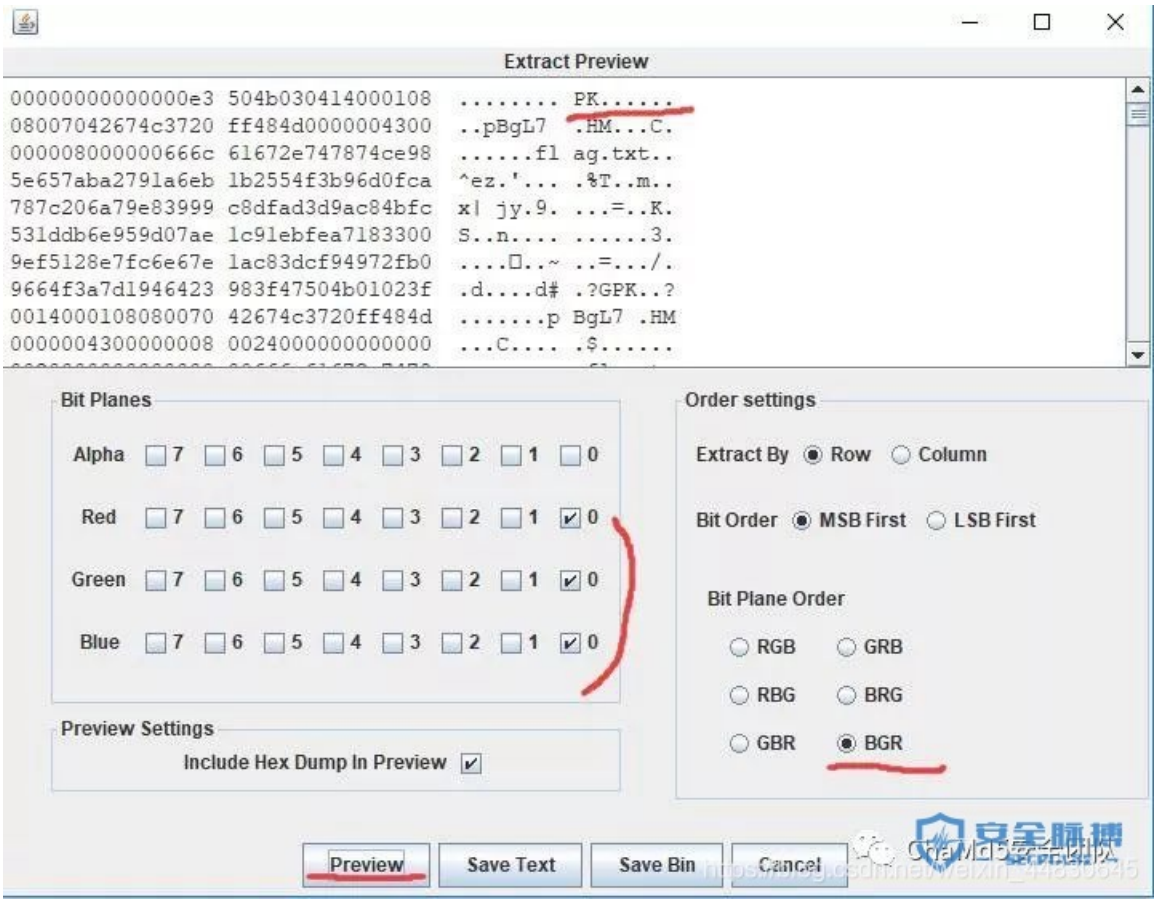


最后得出 flag{Y@e_Cl3veR_C1Ever!}

24.多彩

这道题，我想了想，空着也不好，就搬运一下业内大佬ChaMD5的wp吧
首先使用隐写神器Stegsolve

在中间地带发现了YSL（杨树林，b（￣▽￣）d）这个口红品牌的字样。再继续深入，Analyse→Data Extract



Save Bin保存为一个zip包

这里用winrar打开会报错，得用7z等压缩工具打开才可以。

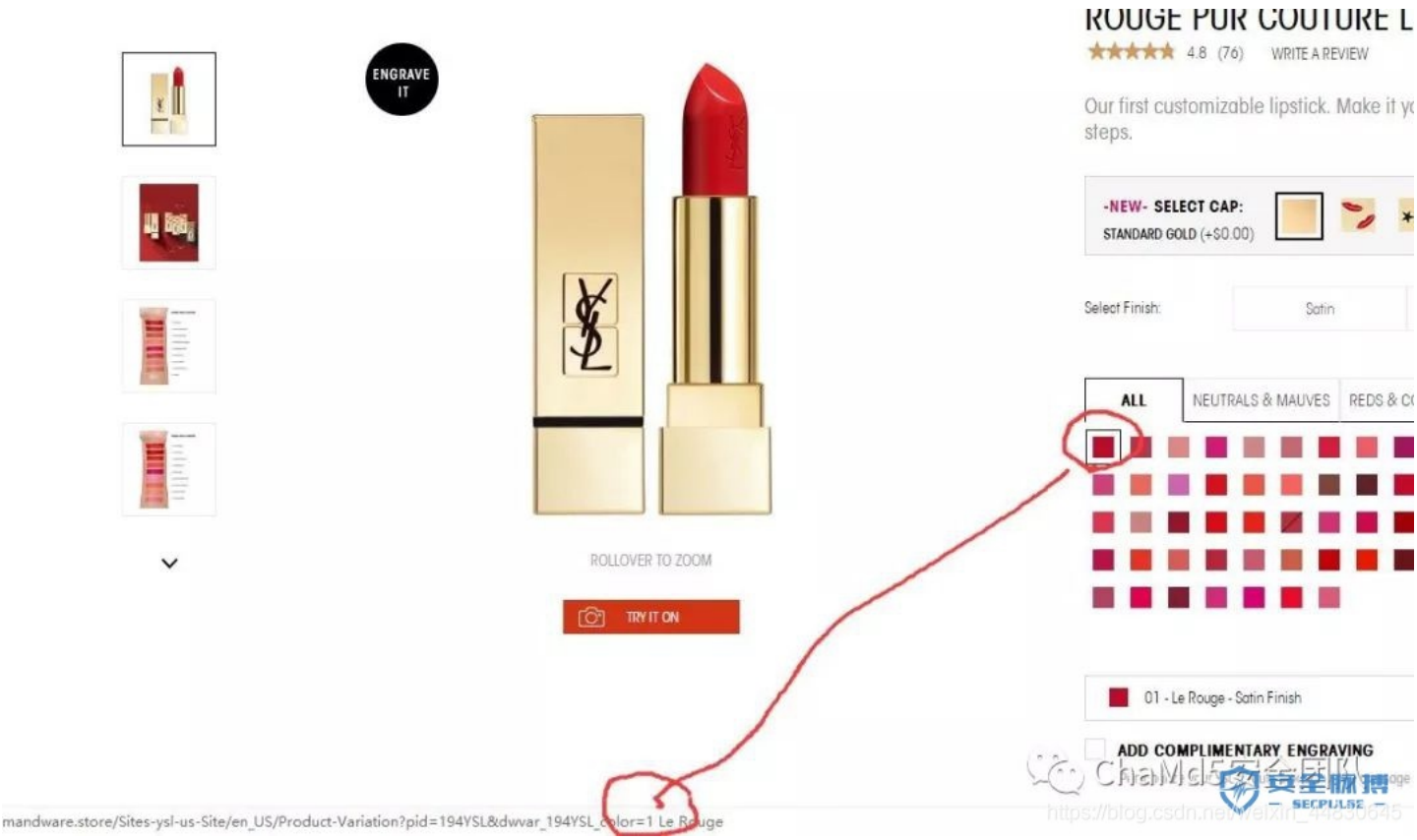
尝试了下伪加密，无果。于是整个过程就剩下一个密码。一般来说图片隐写的话，要么是二进制里藏了东西，要么就是图形藏了东西。这里二进制里藏了zip包，剩下的密码就只能从图形里入手。图形里是21个颜色格，我分别取色

> BC0B28D04179D47A6FC2696FEB8262CF1A77C0083EBC0B28BC0B28D132746A1319BC0B28BC0B28D4121DD75B59DD8885CE0A4AD4121D7E453AD75B59DD8885

这里折腾了好久，发现是要找颜色所对应的YSL口红的色号(ㅖ￣ω￣)

搜到一个网址：

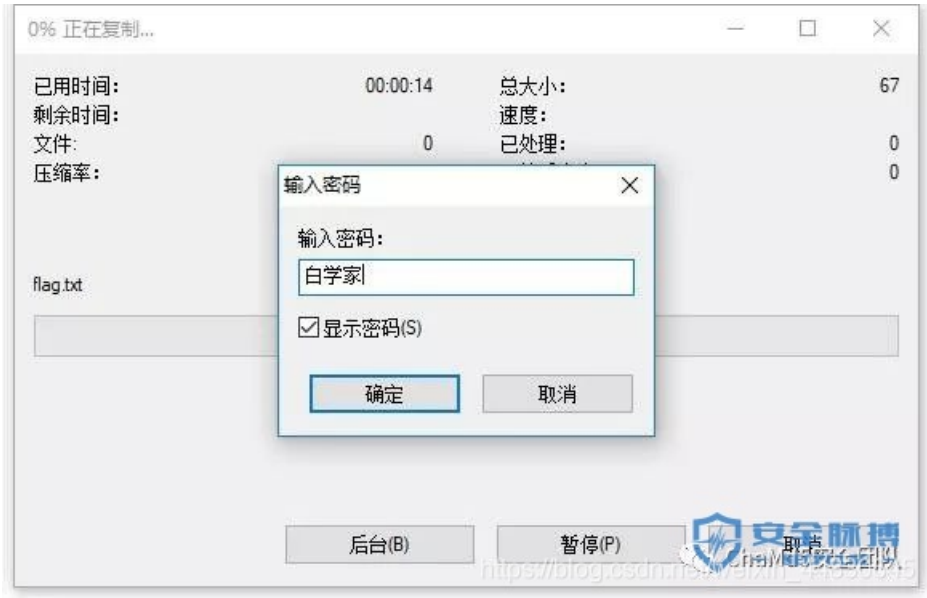
https://www.yslbeautyus.com/on/demandware.store/Sites-ysl-us-Site/en_US/Product-Variation?pid=194YSL



这里颜色值可以对应上色号，于是写脚本收集颜色值对应的色号，并把色号转换为二进制，再组合，再bin2text

```
# -*- coding:utf8 -*-
__author__='pcat@chamd5.org'
import requestsimport re
import libnum

def foo():
url=r'https://www.yslbeautyus.com/on/demandware.store/Sites-ysl-us-Site/en_US/Product-Variation?pid=194YSL'
cont=requests.get(url).content
# print cont
pattern=r'YSL_color=(.*?)%20[sS]*?background-color: #(.*?)"'
rst=re.findall(pattern,cont)
dYSL={}
for num,color in rst:
dYSL[color]=int(num.lstrip('0'))
lst=['BC0B28','D04179','D47A6F','C2696F','EB8262', 'CF1A77','C0083E','BC0B28','BC0B28','D13274', '6A1319','BC0B2
8','BC0B28','D4121D','D75B59', 'DD8885','CE0A4A','D4121D','7E453A','D75B59', 'DD8885']
flag=''.join('{:b}'.format(dYSL[i]) for i in lst)
print libnum.b2s(flag)
pass

if __name__ == '__main__':
foo()
print 'ok'
```

打印出来是“白学家”，用7z进行解压缩



解压后打开flag.txt即可。

flag{White_Album_is_Really_worth_watching_on_White_Valentine’s_Day}

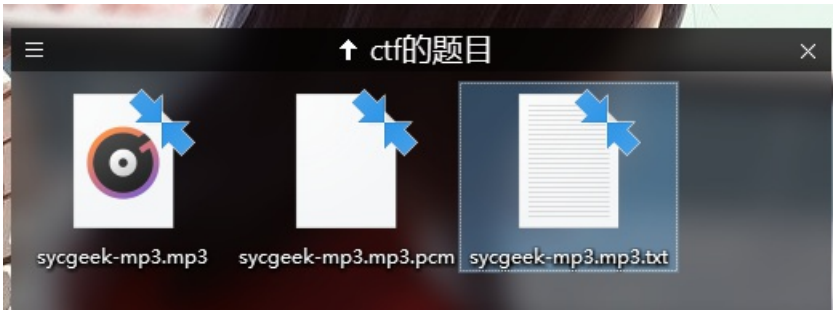
~~出题人老白学家了，白雪家给爷死！不对我好像也是白学家，那没事了。~~

25.旋转跳跃

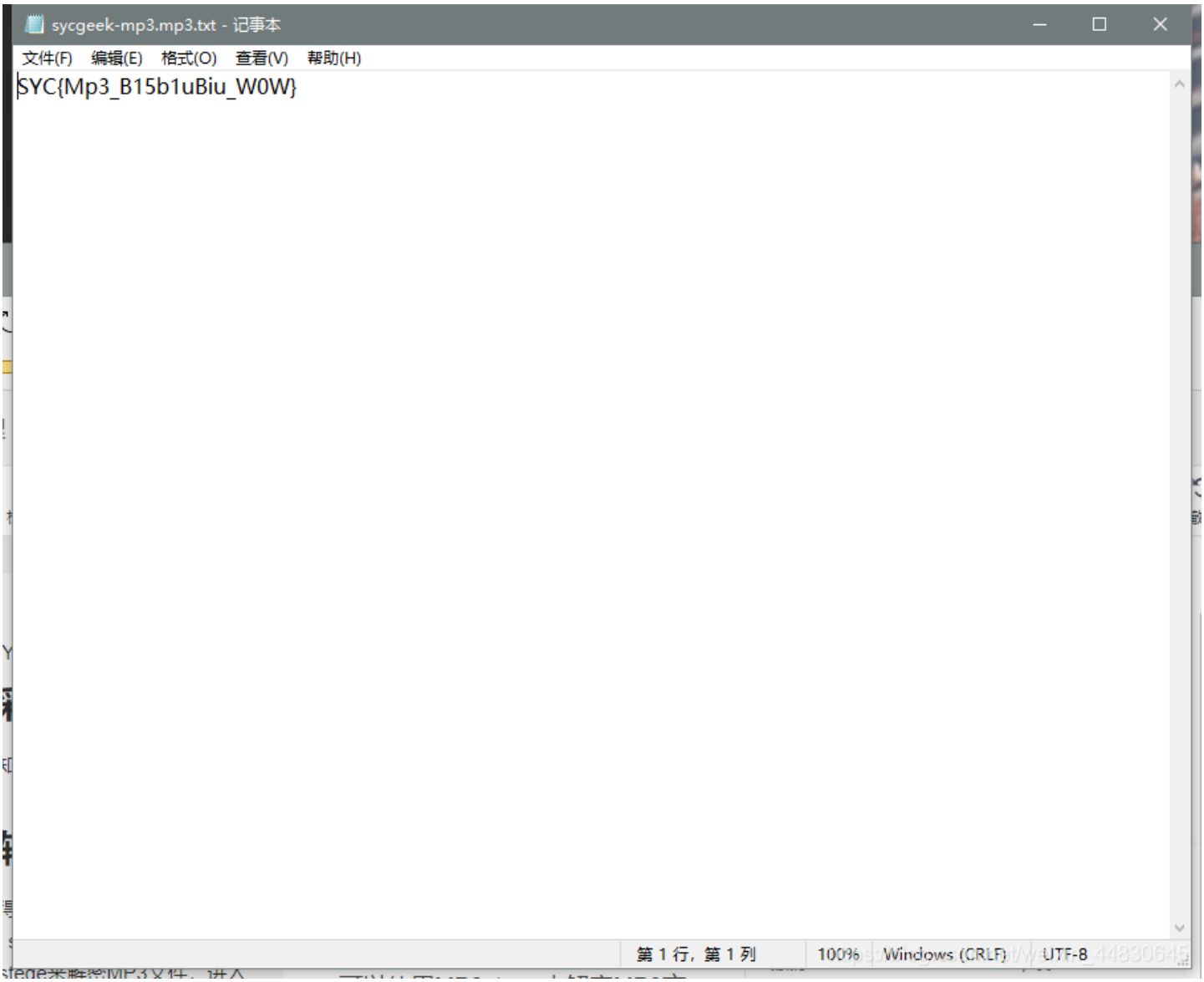
下载题目文件得到一个mp3文件，然后有个提示说key：syclovergeek
可以使用MP3stege来解密MP3文件，进入到MP3stege文件目录下，运行以下命令即可

```
Decode.exe -X -P syclovergeek sycgeek-mp3.mp3
```

运行结束后会在文件夹下创建



打开即可得到FLAG



26.普通的二维码

下载解压出来是一张二维码图片，扫描可得到

已扫描到以下内容

哈哈!就不告诉你flag就在这里!

扫描所得内容并非微信提供，请谨慎使用

如需使用，可通过复制操作获取内容

老规矩，走一套流程，右键属性==>winhex==>然后binwalk

右键看完没有任何信息，然后用winhex打开

```
00002848  FF FF FF FF FF FF FF FF  FF FF FF FF F0 00 FF FF  ÿÿÿÿÿÿÿÿÿÿÿÿð ÿÿ
00002864  FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF  ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
00002880  F0 00 FF FF FF FF FF FF  FF FF FF FF FF FF FF FF  ð ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
00002896  FF FF FF FF F0 00 FF FF  FF FF FF FF FF FF FF FF  ÿÿÿÿð ÿÿÿÿÿÿÿÿÿÿ
00002912  FF FF FF FF FF FF FF FF  F0 00 FF FF FF FF FF FF  ÿÿÿÿÿÿÿÿð ÿÿÿÿÿÿ
00002928  FF FF FF FF FF FF FF FF  FF FF FF FF F0 00 FF FF  ÿÿÿÿÿÿÿÿÿÿÿÿð ÿÿ
00002944  FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF  ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
00002960  F0 00 FF FF FF FF FF FF  FF FF FF FF FF FF FF FF  ð ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
00002976  FF FF FF FF F0 00 FF FF  FF FF FF FF FF FF FF FF  ÿÿÿÿð ÿÿÿÿÿÿÿÿÿÿ
00002992  FF FF FF FF FF FF FF FF  F0 00 FF FF FF FF FF FF  ÿÿÿÿÿÿÿÿð ÿÿÿÿÿÿ
00003008  FF FF FF FF FF FF FF FF  FF FF FF FF F0 00 31 34  ÿÿÿÿÿÿÿÿÿÿÿÿð 14
00003024  36 31 35 34 31 34 31 31  34 37 31 37 33 31 31 30  6154141147173110
00003040  31 34 31 31 36 36 31 34  35 31 33 37 31 37 31 30  1411661451371710
00003056  36 30 31 32 35 31 33 37  31 32 30 31 37 31 31 33  60125137120171 13
00003072  37 31 36 33 31 34 33 31  36 32 31 35 31 31 36 30  7163143162151160
00003088  31 36 34 31 33 37 31 31  37 31 36 34 31 34 33 31  1641371171641431
00003104  33 37 31 32 34 31 35 37  31 33 37 31 32 34 31 34  3712415713712414
00003120  35 31 35 36 31 33 37 31  30 31 31 36 33 31 34 33  5156137101163143
00003136  31 35 31 31 35 31 30 34  31 31 37 35 40 78 6A 73  151151041175@xjs
00003152  65 63 6B 21                                       eck!
```


最后那里有一串奇怪的字符串，仔细查看后发现最高只有7，并没有到8，猜测是八进制，去网站八进制转成字符串，发现三个一组的时候可以转出f，猜测是三个一组转换

把它们放到一个文件里面，但注意要把@xjseck!删去，然后可以写一个脚本来转换（注意本脚本为Python3.x版本才可运行）

```
f = open('flag.txt')
temp = []
while True:
    k = f.read(3)
    if k:
        temp.append(k)
    else:
        break

f.close()
for i in temp:
    num = '0o' + i
    num = int(num, base=0)
    num = chr(num)
    print(num, end='')
```

转换出来的结果如下图所示

```
-------
flag{Have_yOU_Py_script_Otc_To_Ten_Ascii!}
>>>
```

27.乌云邀请码

下载解压得到

您好：

这是来自于WooYun的一封邀请邮件，非常高兴你通过WooYun发布有价值的漏洞，很荣幸的邀请阁下为WooYun白帽子中的一员，你可以通过如下的链接来注册

http://www.wooyun.org/user.php?action=register&code=b6d75821211e338dd56623c8825456ab&invite_email=504038236@qq.com&invite_type=0

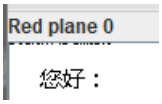
WooYun会给你发送一封确认邮件，可以点击其中的链接完成注册，希望你继续支持WooYun

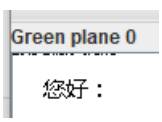
漏洞处理流程：http://www.wooyun.org/help#bug
白帽注意事项：http://www.wooyun.org/help#whitehat

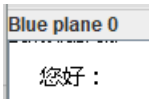
本邮件由WooYun自动发送，请勿回复
WooYun是一个自由平等的漏洞和安全信息报告平台
其他关于WooYun的更多详细信息请访问http://www.wooyun.org/about.php

谢谢!

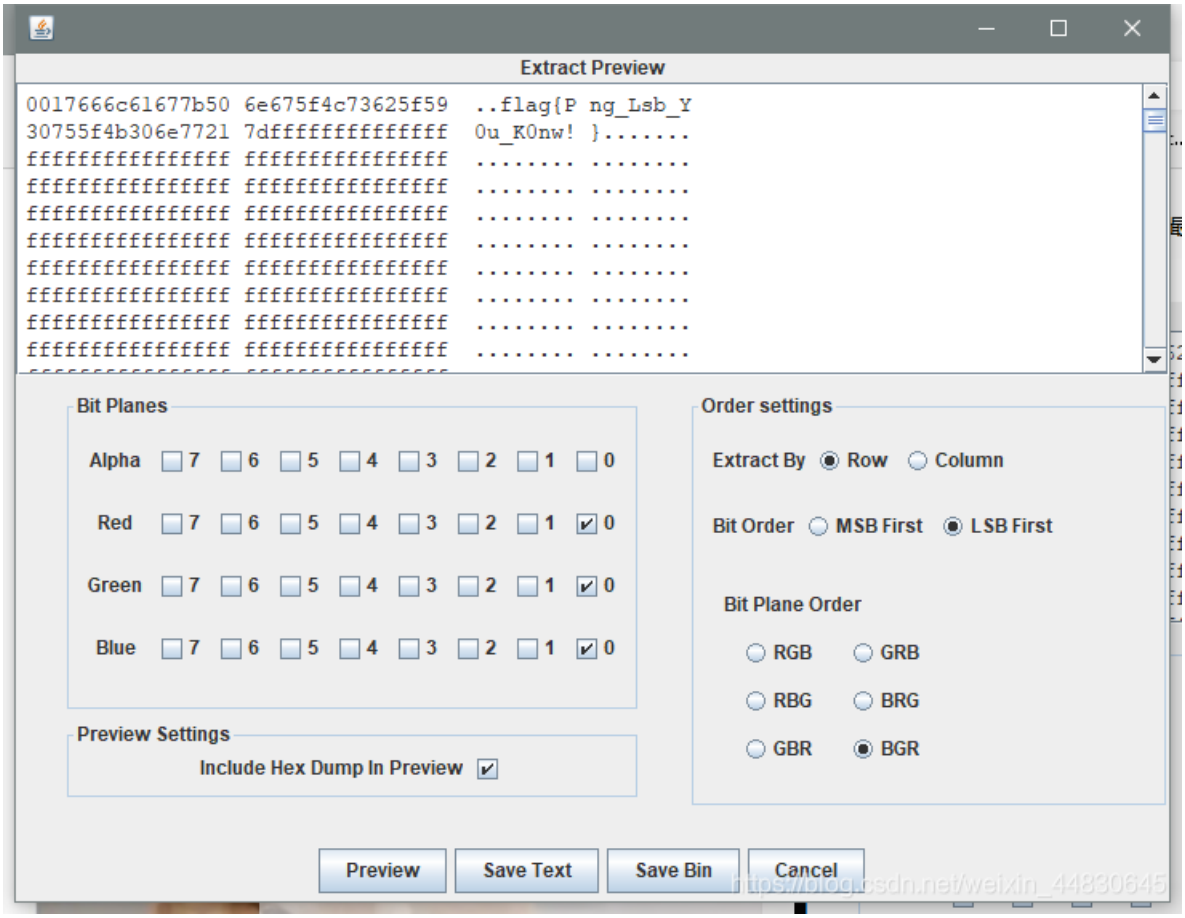
使用winhex和binwalk无果后，用stegsolve打开





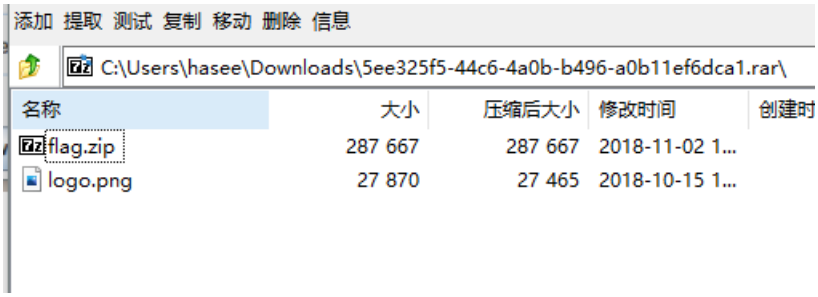


很明显的低位隐写有东西，明显的lsb隐写，设置之后可以得到flag

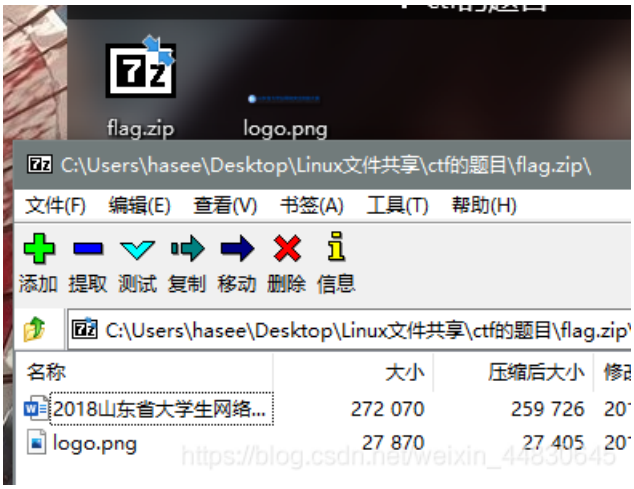


28.神秘的文件

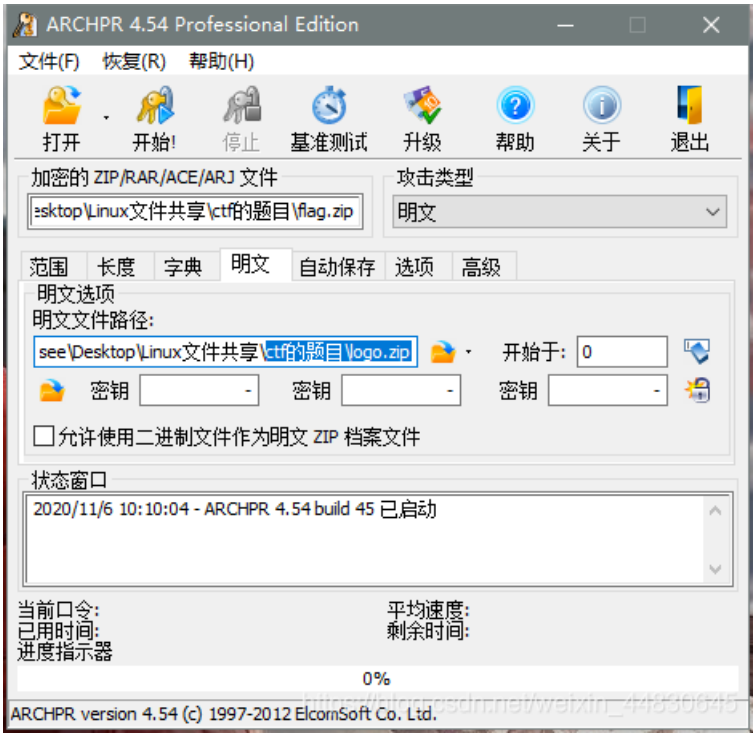
下载得到一个压缩包，解压出来是为两个文件，一个压缩包，一个图片



打开flag.zip发现，里面文件有加密，并且也有一张名为logo.png的图片



很明显的明文攻击了，我们使用ARCHPR明文攻击破解



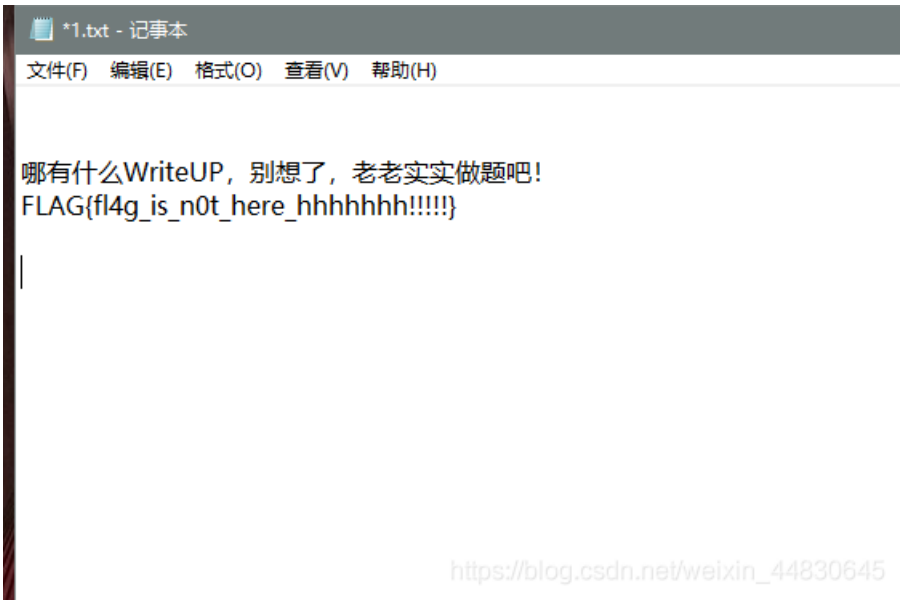
我尝试了好久的时间，怎么都爆破不出来，去百度翻了翻，可能是我版本的问题，所以我就用别人爆出来的密码解压

> 密码为：q1w2e3r4

解压出来有一个docx的文档，打开得到一张漫威图片

解压出来有一个docx的文档，打开得到一张滑稽图片

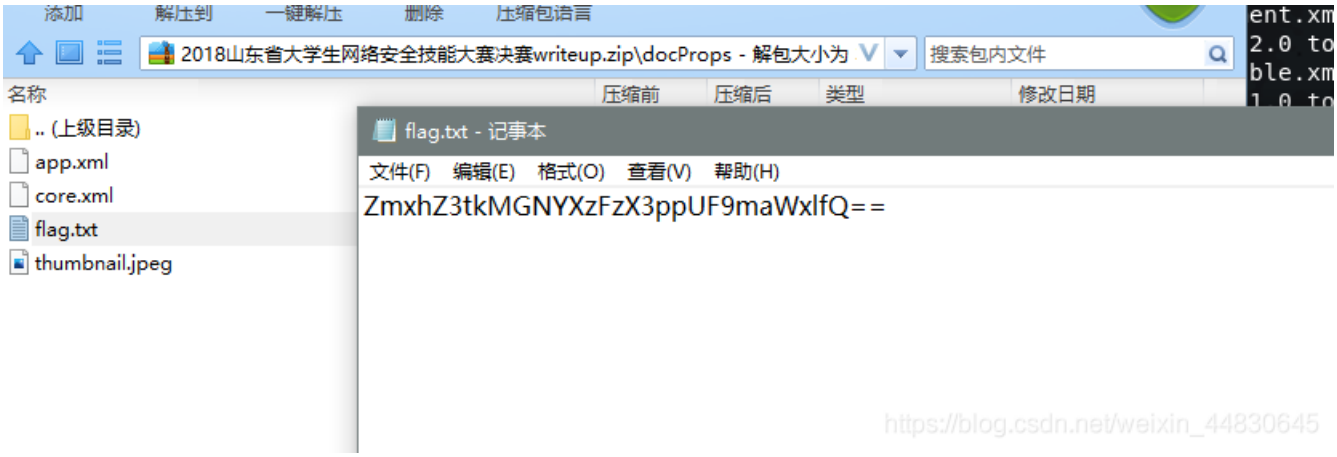
ctrl+a全选后复制出来得到了一个假的flag



然后用binwalk能看到有很多的文件

```
root@kali:/mnt/hgfs/Linux文件分享/ctf的题目# binwalk 2018山东省大学生网络安全技
能大赛决赛writeup.docx

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             Zip archive data, at least v2.0 to extract, compr
ssed size: 362, uncompressed size: 695, name: docProps/app.xml
672           0x2A0           Zip archive data, at least v2.0 to extract, compr
ssed size: 387, uncompressed size: 773, name: docProps/core.xml
1370          0x55A           Zip archive data, at least v1.0 to extract, compr
ssed size: 36452, uncompressed size: 36452, name: docProps/thumbnail.jpeg
37875         0x93F3          Zip archive data, at least v2.0 to extract, compr
ssed size: 1285, uncompressed size: 4056, name: word/document.xml
39207         0x9927          Zip archive data, at least v2.0 to extract, compr
ssed size: 476, uncompressed size: 1529, name: word/fontTable.xml
39731         0x9B33          Zip archive data, at least v1.0 to extract, compr
ssed size: 222845, uncompressed size: 222845, name: word/media/image1.png
262627        0x401E3         Zip archive data, at least v2.0 to extract, compr
ssed size: 1117, uncompressed size: 2847, name: word/settings.xml
263791        0x4066F         Zip archive data, at least v2.0 to extract, compr
ssed size: 2920, uncompressed size: 29509, name: word/styles.xml
266756        0x41204         Zip archive data, at least v2.0 to extract, compr
ssed size: 1512, uncompressed size: 6803, name: word/theme/theme1.xml
268319        0x4181F         Zip archive data, at least v2.0 to extract, compr
```


把它改成zip后缀后直接打开，在docProps目录下有一个flag.txt打开里面是一串base64加密



拿去解密可以得到flag

29.论剑

下载可得到一张图片


```
31 31 30 31 31 30 31 20  30 31 31 31 31 30 30 31  1101101 01111001
20 30 31 31 30 31 31 31  30 20 30 31 31 30 30 30   01101110 011000
30 31 20 30 31 31 30 31  31 30 31 20 30 31 31 30  01 01101101 0110
30 31 30 31 20 30 31 31  30 31 30 30 31 20 30 31  0101 01101001 01
31 31 30 30 31 31 20 30  31 31 30 31 30 31 31 20  110011 01101011
30 31 31 30 30 31 30 31  20 30 31 31 31 31 30 30  01100101 0111100
31 20 30 30 31 30 30 30  30 31 20 30 30 31 30 30  1 00100001 00100
30 30 31 20 30 30 31 30  30 30 30 31 20 30 31 31  001 00100001 011
30 31 30 30 30 20 30 31  31 30 31 30 30 30 20 30  01000 01101000 0
31 31 30 31 30 30 30 38  7B BC AF 27 1C 00 04 CB  11010008{¼¯'   Ë
B2 17 DF A0 1E 00 00 00  00 00 00 6A 00 00 00 00  ² ß        j
00 00 00 6D C5 15 1F D8  C3 E1 A9 0E D6 5B F2 33     mÅ  ØÃá© Ö[ò3
```

用winhex打开，看到中间有一串二进制
把它8个一组转换为16进制，然后在转换成ascii码，可以得到mynameiskey!!!hhh，肯定有用的

然后我注意到图片的宽高很不平等，猜测是宽高被修改过，遂可以修复一下

> 不知道怎么修复的小伙伴，可以看看第三题，我那里有讲过

得到一张图片，但是有一些地方被挡住了，我们应该还有一些线索没有找到，回到winhex，发现紧跟在二进制后面有一串我们比较熟悉的十六进制

```
31 31 30 31 31 30 31 20  30 31 31 31 31 30 30 31  1101101 01111001
20 30 31 31 30 31 31 31  30 20 30 31 31 30 30 30   01101110 011000
30 31 20 30 31 31 30 31  31 30 31 20 30 31 31 30  01 01101101 0110
30 31 30 31 20 30 31 31  30 31 30 30 31 20 30 31  0101 01101001 01
31 31 30 30 31 31 20 30  31 31 30 31 30 31 31 20  110011 01101011
30 31 31 30 30 31 30 31  20 30 31 31 31 31 30 30  01100101 0111100
31 20 30 30 31 30 30 30  30 31 20 30 30 31 30 30  1 00100001 00100
30 30 31 20 30 30 31 30  30 30 30 31 20 30 31 31  001 00100001 011
30 31 30 30 30 20 30 31  31 30 31 30 30 30 20 30  01000 01101000 0
31 31 30 31 30 30 30 38  7B BC AF 27 1C 00 04 CB  11010008{¼¯'   Ë
B2 17 DF A0 1E 00 00 00  00 00 00 6A 00 00 00 00  ² ß        j
00 00 00 6D C5 15 1F D8  C3 E1 A9 0E D6 5B F2 33     mÅ  ØÃá© Ö[ò3
```

翻了翻记录文件头的小笔记，发现这个头跟7z的文件头很像，所以就修复了一下，修复之后用binwalk看了一下，不出所料，出现了7-zip文件

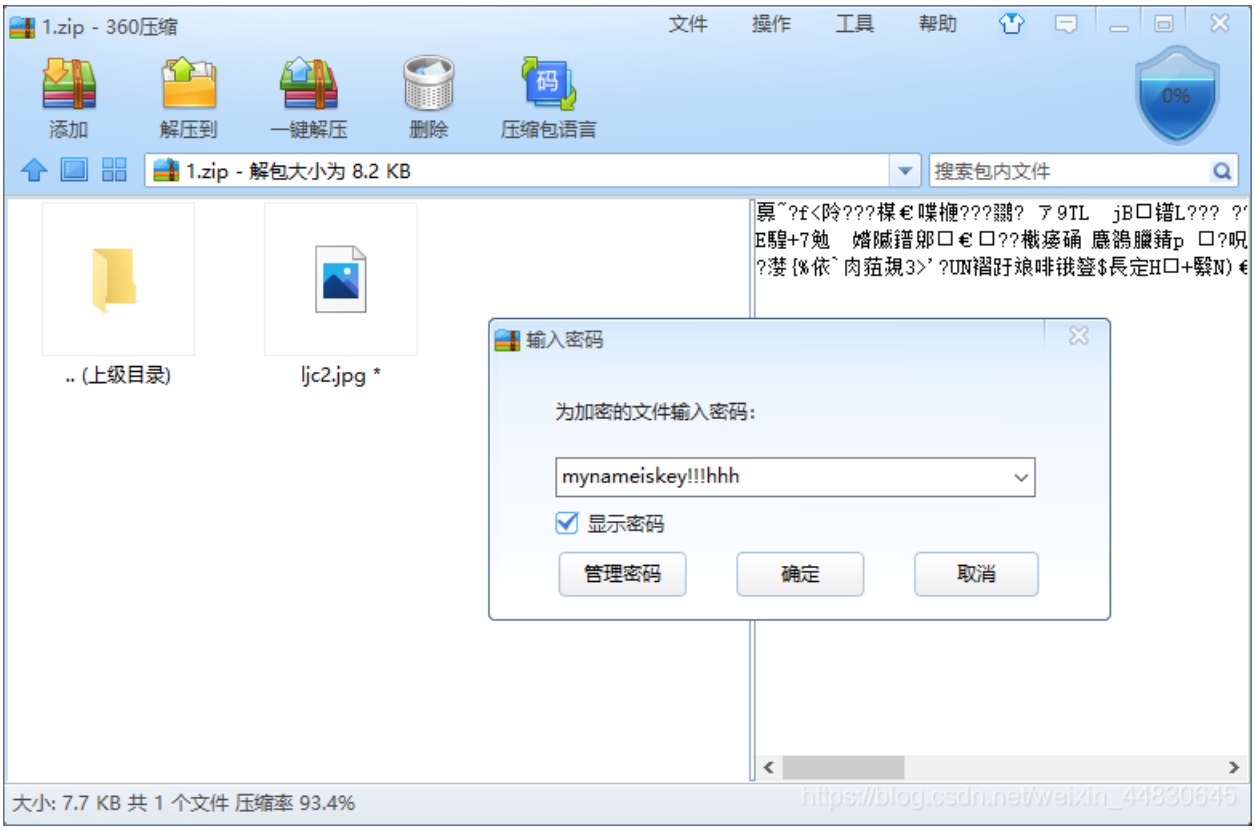
```
root@kali:/mnt/hgfs/Linux文件分享/ctf的题目# binwalk lunjian.jpg
```

```
DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, JFIF standard 1.01
9591          0x2577          7-zip archive data, version 0.4
17569         0x44A1          JPEG image data, JFIF standard 1.01

root@kali:/mnt/hgfs/Linux文件分享/ctf的题目#
```

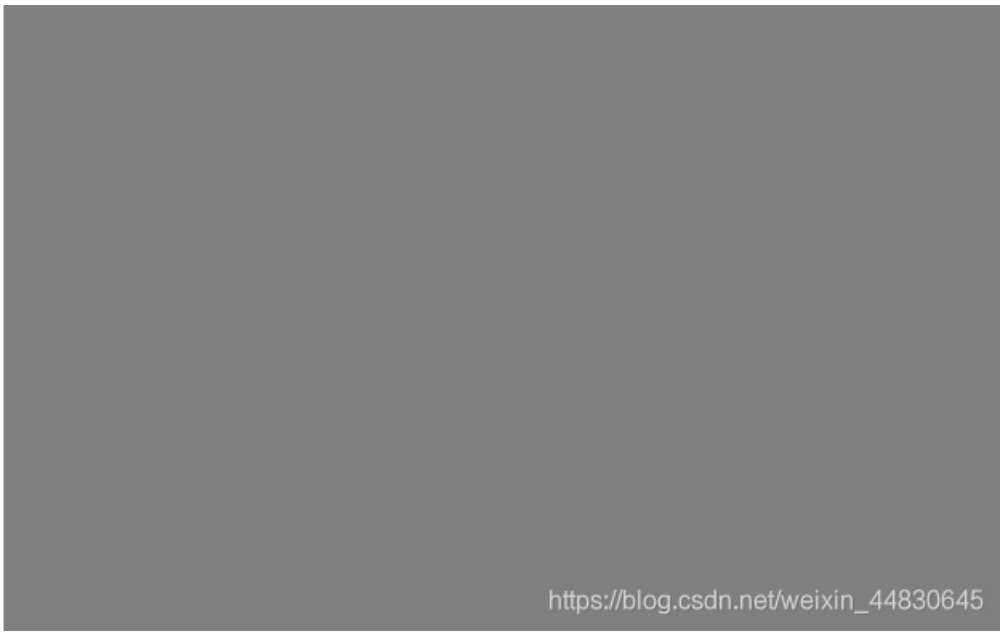
我们用dd分离出来

```
root@kali:/mnt/hgfs/Linux文件分享/ctf的题目# ls
lunjian.jpg
root@kali:/mnt/hgfs/Linux文件分享/ctf的题目# dd if=lunjian.jpg of=1.zip skip=959
1 bs=1
记录了12399+0 的读入
记录了12399+0 的写出
12399 bytes (12 kB, 12 KiB) copied, 1.62815 s, 7.6 kB/s
root@kali:/mnt/hgfs/Linux文件分享/ctf的题目#
```

得到一个压缩包，但是被加密了，解密的密码就是我们之前解出来的那个字符串



解压出来可以得到一张和题目一样的图片，同样修改一下它的宽高，可以得到如下图片

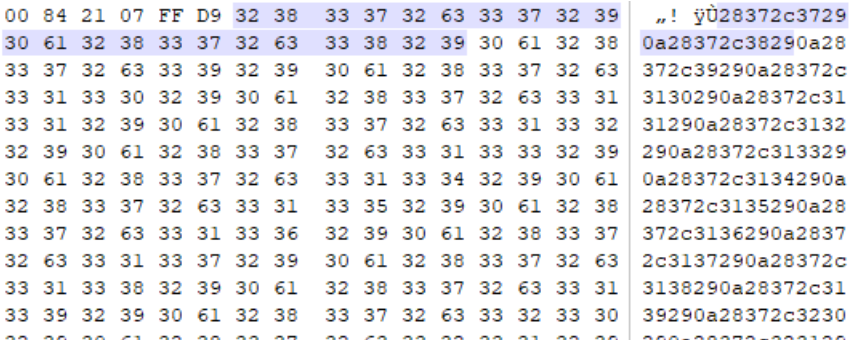


然后结合之前的那张图片可以得到Not flag{666c61677B6D795F6E616D655F482121487D}
中间的那串字符是一串bash16加密，解密即可得到flag

> Base16编码使用16个ASCII可打印字符（数字0-9和字母A-F）对任意字节数据进行编码。
> Base16先获取输入字符串每个字节的二进制值（不足8比特在高位补0），然后将其串联进来
> 再按照4比特一组进行切分，将每组二进制数分别转换成十进制
> 然后在下面找到对应的编码串接起来就是Base16编码。
> 编码 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
> 字符 0 1 2 3 4 5 6 7 8 9 A B C D E F

30.图穷匕见

下载下来得到一张图片，用winhex打开，搜索文件尾ffd9可以看到跟在文件尾后面有很多的字符串



```
32 39 30 61 32 38 33 37  32 63 33 32 33 31 32 39  290a28372c323129
30 61 32 38 33 37 32 63  33 32 33 32 32 39 30 61  0a28372c3232290a
32 38 33 37 32 63 33 32  33 33 32 39 30 61 32 38  28372c3233290a28
33 37 32 63 33 32 33 34  32 39 30 61 32 38 33 37  372c3234290a2837
32 63 33 32 33 35 32 39  30 61 32 38 33 37 32 63  2c3235290a28372c
33 32 33 36 32 39 30 61  32 38 33 37 32 63 33 32  3236290a28372c32
33 37 32 39 30 61 32 38  33 37 32 63 33 32 33 38  37290a28372c3238
32 39 30 61 32 38 33 37  32 63 33 32 33 39 32 39  290a28372c323929
30 61 32 38 33 37 32 63  33 33 33 30 32 39 30 61  0a28372c3330290a
32 38 33 37 32 63 33 33  33 31 32 39 30 61 32 38  28372c3331290a28
33 37 32 63 33 33 33 32  32 39 30 61 32 38 33 37  372c3332290a2837
32 63 33 33 33 33 32 39  30 61 32 38 33 37 32 63  2c3333290a28372c
33 33 33 34 32 39 30 61  32 38 33 37 32 63 33 33  3334290a28372c33
33 35 32 39 30 61 32 38  33 37 32 63 33 33 33 36  35290a28372c3336
32 39 30 61 32 38 33 37  32 63 33 33 33 37 32 39  290a28372c333729
30 61 32 38 33 37 32 63  33 33 33 38 32 39 30 61  0a28372c3338290a
32 38 33 37 32 63 33 33  33 39 32 39 30 61 32 38  28372c3339290a28
33 37 32 63 33 34 33 30  32 39 30 61 32 38 33 37  372c3430290a2837
32 63 33 34 33 31 32 39  30 61 32 38 33 37 32 63  2c3431290a28372c
33 34 33 32 32 39 30 61  32 38 33 37 32 63 33 34  3432290a28372c34
33 33 32 39 30 61 32 38  33 37 32 63 33 34 33 34  33290a28372c3434
32 39 30 61 32 38 33 37  32 63 33 34 33 35 32 39  290a28372c343529
30 61 32 38 33 37 32 63  33 34 33 36 32 39 30 61  0a28372c3436290a
32 38 33 37 32 63 33 34  33 37 32 39 30 61 32 38  28372c3437290a28
33 37 32 63 33 34 33 38  32 39 30 61 32 38 33 37  372c3438290a2837
32 63 33 34 33 39 32 39  30 61 32 38 33 37 32 63  2c343920 90a28372c
33 35 33 30 32 39 30 61  32 38 33 37 32 63 33 35  3530290a28372c35
33 31 32 39 30 61 32 38  33 37 32 63 33 35 33 32  31290a28372c3532
32 39 30 61 32 38 33 37  32 63 33 35 33 33 32 39  290a28372c353329
30 61 32 38 33 37 32 63  33 35 33 34 32 39 30 61  0a28372c3534290a
32 38 33 37 32 63 33 35  33 35 32 39 30 61 32 38  28372c3535290a28
33 37 32 63 33 35 33 36  32 39 30 61 32 38 33 37  372c3536290a2837
32 63 33 37 33 38 32 39  30 61 32 38 33 37 32 63  2c3738290a28372c
33 37 33 39 32 39 30 61  32 38 33 37 32 63 33 38  3739290a28372c38
33 30 32 39 30 61 32 38  33 37 32 63 33 38 33 31  30290a28372c3831
```

把他们保存下载，然后用16进制转成ascii码，就有思路了

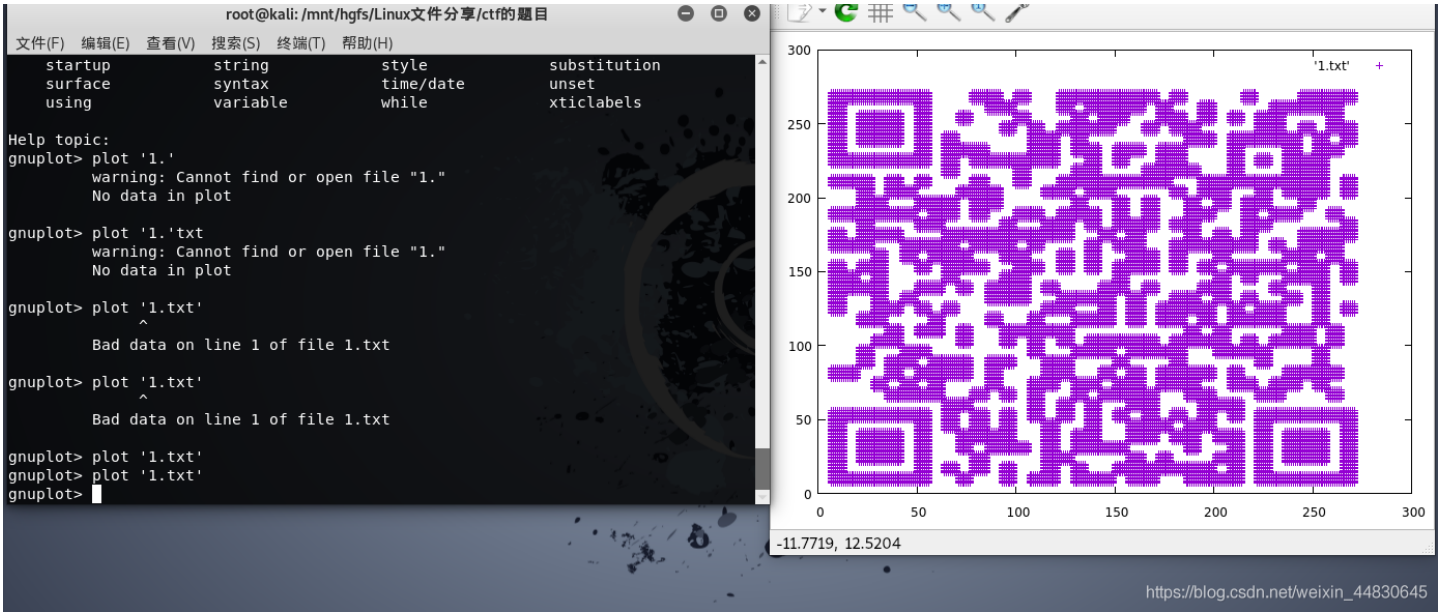
这一看就知道是坐标，以下有两种做法

第一种、可以用python写个脚本把它画出来，这里我就直接借用一下大佬的脚本了

```
import  matplotlib.pyplot  as plt
i=0
fig=plt.figure()
with open("text.txt") as f:
 for data in f.readlines():
  data=data.strip()
  data=eval(data)
  plt.scatter(data[0],data[1],c="255",marker=".")
  i=i+1
  print("\r\n[+] Has dealed",i,"lines")
plt.show()
```

最后可以得到一张二维码但是特别慢

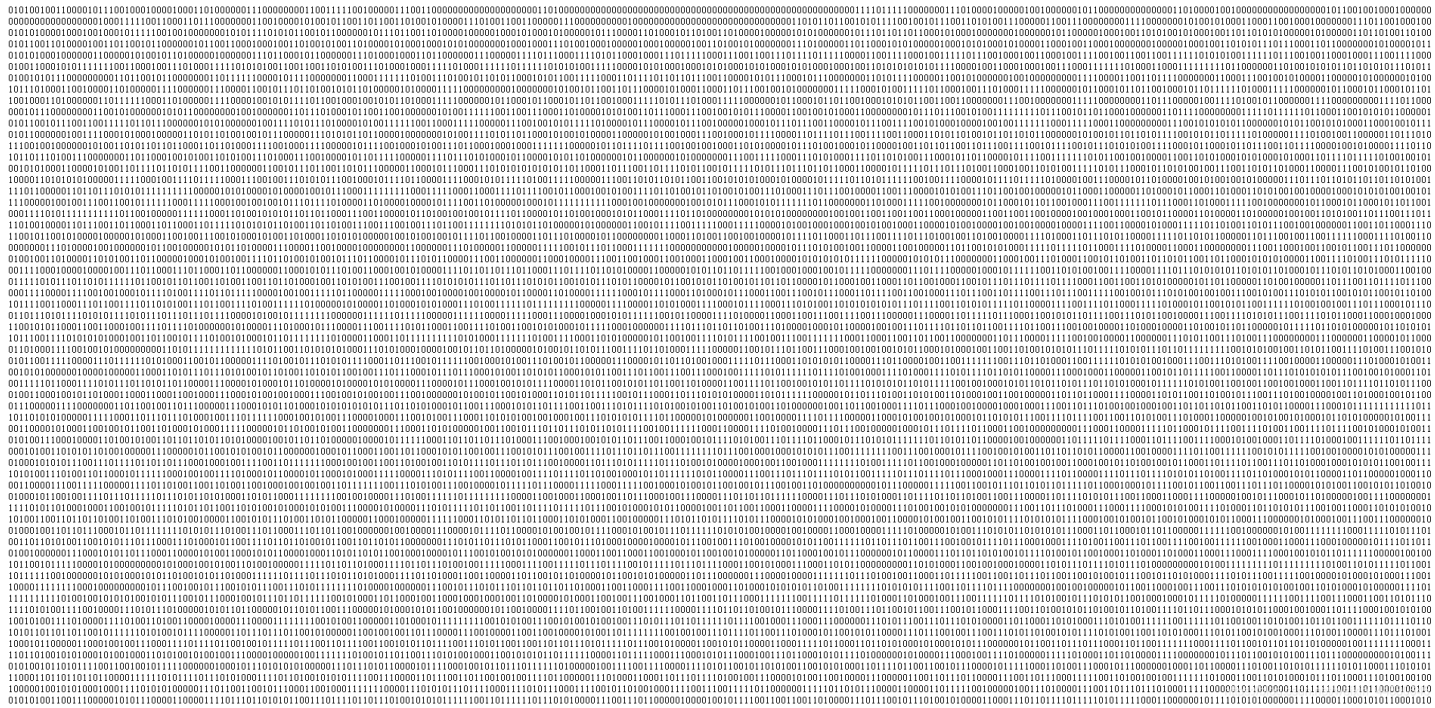
第二种、在linux里面用gnuplot工具



这里需要注意的是，gnuplot一定要把括号换成空，把逗号换成空格才可以识别画图
扫描即可得到flag

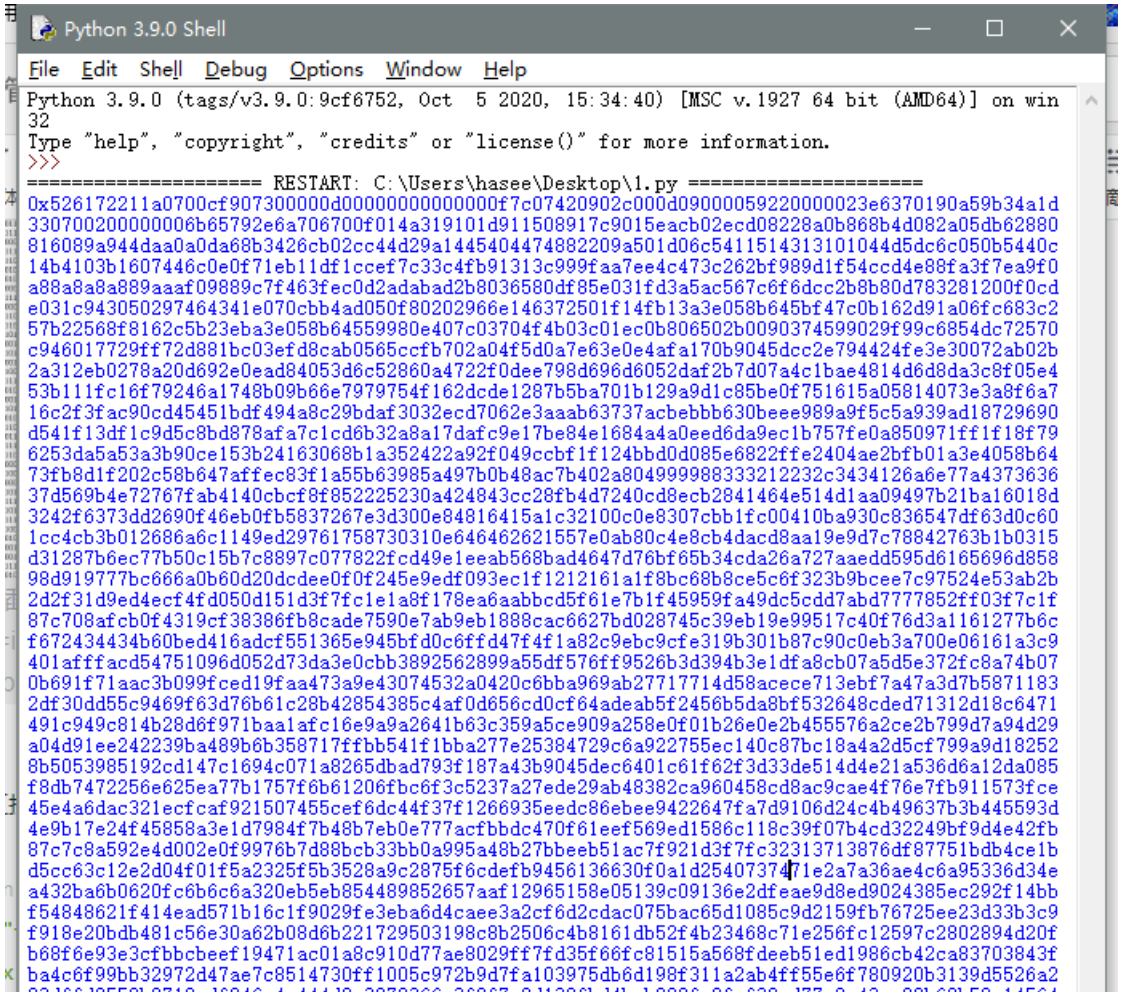
31.convert

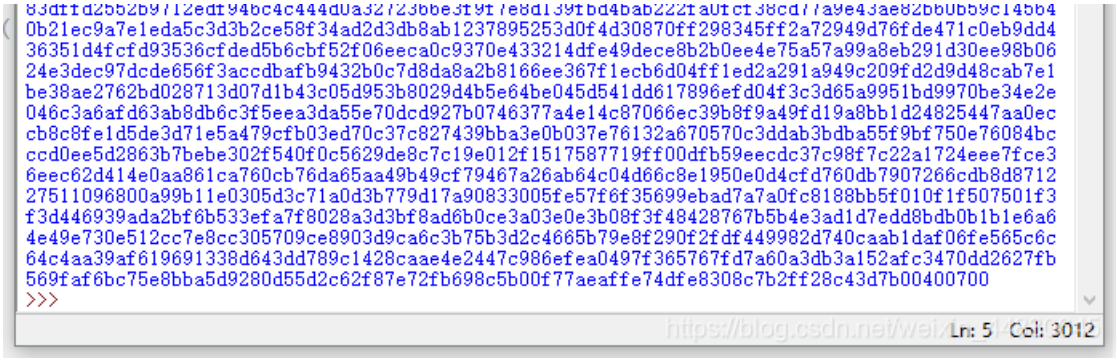
下载文件得到一长串的二进制字符串



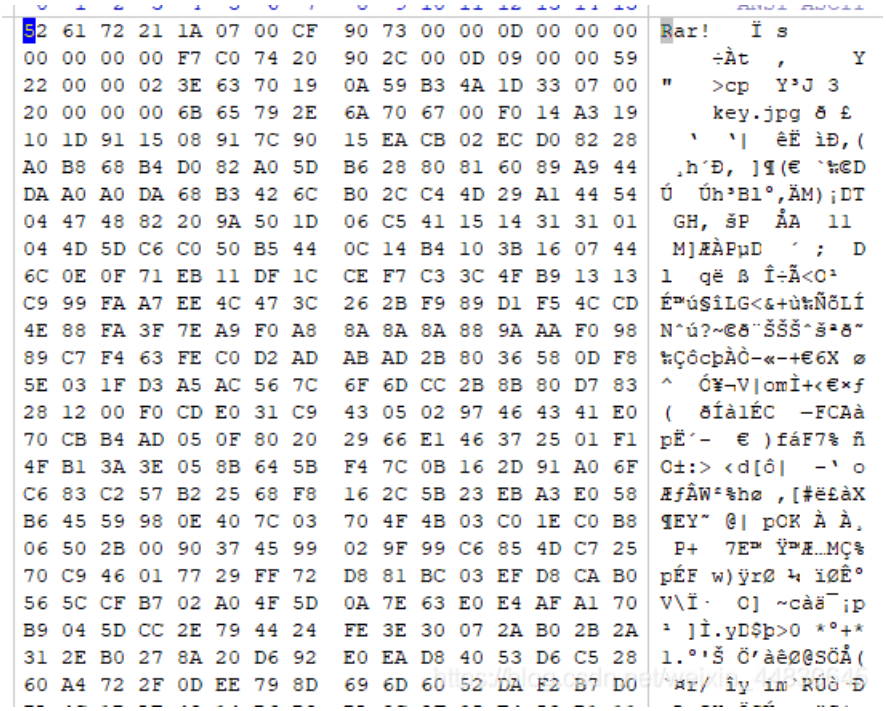
把它们直接转换为16进制，这里我就写了个python来转换为16进制

```
f1=open("flag.txt","r")
print(hex(int(str(f1.read()),2)))
f1.close()
```

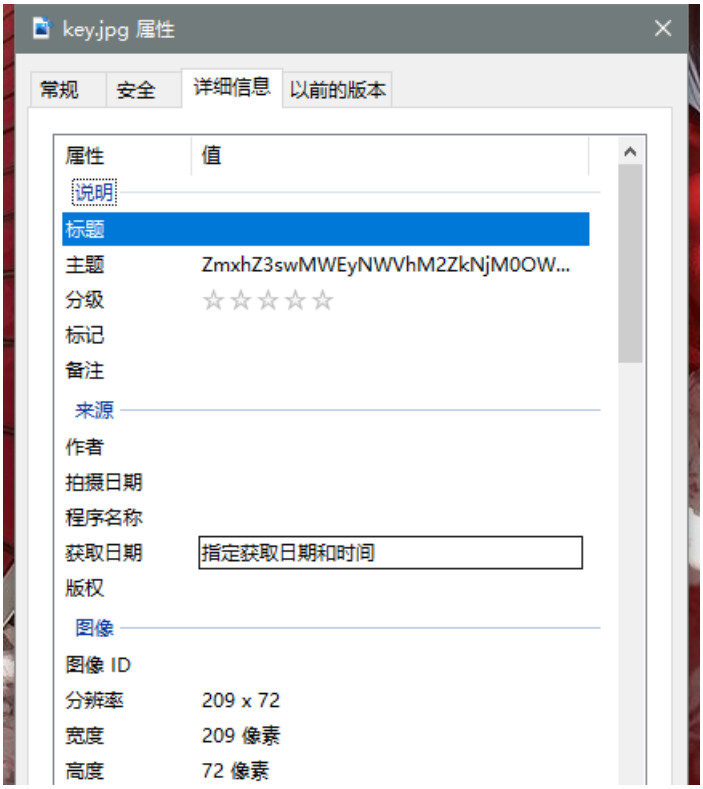


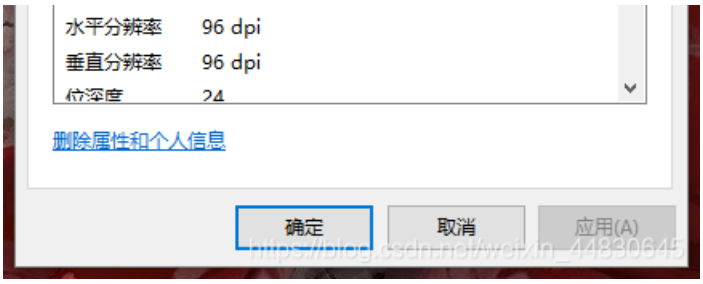


把他们复制到winhex里面



我们可以发现，他是一个rar文件，把后缀改为rar然后可以解压出来一张图片，图片的详细信息里面有一串base64加密

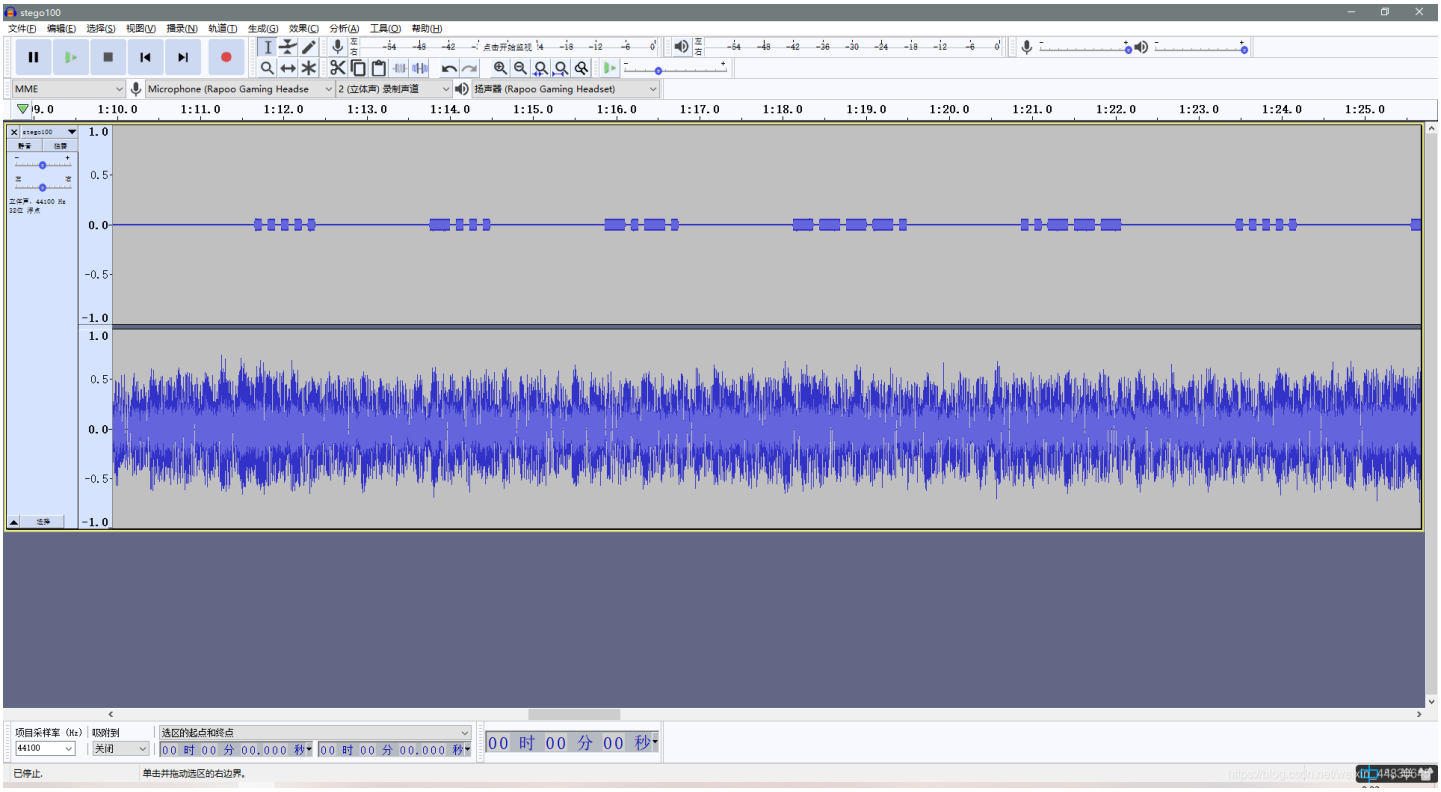




解密即可得到flag

32.听首音乐

下载可以得到一个音频文件，题目没有任何提示，我们把它放进Audacity分析一下



上面那段音频，有短有长有空格，很容易让人联想到摩斯密码，把它们按照顺序一个一个打出来可以得到如下密码：

```
... -... -.-. ----. ...— ... -... ...- ----. -.-. -... ----- .---- —... —... ...-. ... ...— . -... .---- --... -... --... ----- ----. ...— ----. .---- ----. .---- -.-.
```

然后去一个莫斯密码解密的网站解密即可得到flag

摩斯密码

摩斯密码在线加密解密工具

```
..... -... -.-. ----. ..--- ..... -.... ....- ----. -.-. -... ----- .---- ---.. ---.. ..-. ..... ..--- . -.... .---- --... -.. --... ----- ---
-. ..--- ----. .---- ----. .---- -.-.
```

Space: Short: . Long: -

加密 解密

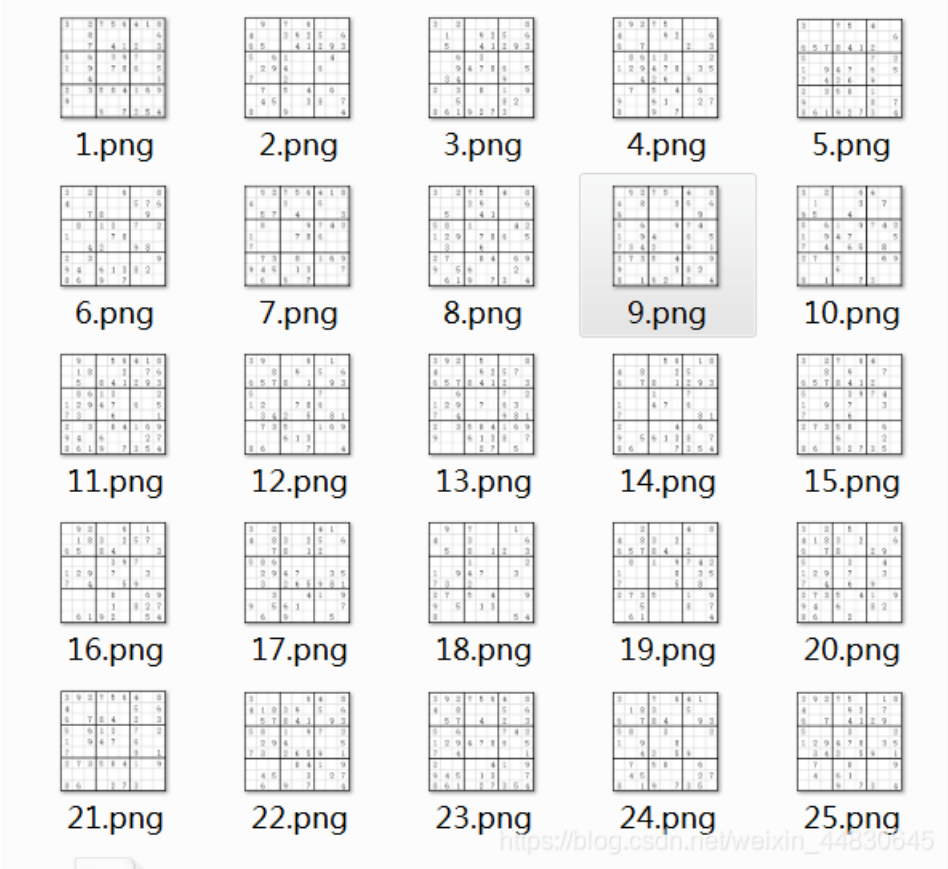
```
5BC925649CB0188F52E617D70929191C
```

动力来自 CTF论坛. 没有美工-。-.

33.好多数组

下载下来得到一堆的数独，自己解出来也没啥用，也没啥思路，这里解直接借用一位大佬的wrtieup了，因为他写的比较详细，链接如下：乔悟空

zip没有后缀，winhex打开，发现确实是zip，修改后缀，解压得到一堆图片，内容是数独



非常壮观，出题人也是费劲了，这个就是二维码分成了25部分，把有数字的涂黑，拼到一起就能组成一个二维码，思路是看来的，但是怎么才能快速完成这一系列的工作呢，怎么能快速获取黑点呢，全点一遍有点呆吧……

> 每一个有九行，如果能选某一行的像素点遍历，估计就可以实现了，go，
> （n years later……□）

他终于来了，全网独家脚本，哈哈哈哈哈哈，可能没人和我这么闲吧哈哈哈哈
我们知道，每个数独有九行，我们要得到的是有哪几行填充了数字，我的思路就是扫描每一格中间的像素，如果是黑色，那就是填充了数字，白色反之
此时文件结构是这样的：

```python
# 此脚本用于从数独图片获取密码，有值为1，无值为0
import os
import cv2
import numpy as np
from PIL import Image

black = (0,0,0) # 黑色RGB
white = (255,255,255) # 白色RGB

# 获取给定路径图片的结果数组
def getBin(picPath):
    image = cv2.imread(picPath)
    height = image.shape[0]  # 图片宽度
    wide = image.shape[1]  # 图片高度
    formWide = (wide-6)/9  # 每一格宽度
    res = [] # 结果数组
    tem = '' # 暂存数组
    for i in range(9):
        for j in range(wide-6):
            px = image[int(i*(formWide-1)+(formWide/2)), j+3]
            if (px == white).all() and (j+3)%formWide > (formWide*3/4) and (j+3)/formWide > len(tem):
                tem += '0'
            if (px == black).all() and (j+3) % formWide > (formWide/4) and (j+3) % formWide < (3*formWide/4) and
 (j+3)/formWide > len(tem):
                tem += '1'
        res.append(tem)
        tem = ''
    return res

# 获取整合25张图片，获取最终结果
def getRes():
    tem = [] # 临时存储返回值
    res = [] # 存储结果数组
    for i in range(5):
        for j in range(5):
            picPath = 'zip/' + str(i*5+j+1) + '.png' # 构造文件名
            tem = getBin(picPath)
            if len(res)==0:
                res = tem
            else:
                if len(res)==(i+1)*9:
                    for x in range(len(tem)):
                        res[(i*9)+x] += tem[x]
                else:
                    for x in range(len(tem)):
                        res.append(tem[x])
    return res

resList = getRes()
resImg = Image.new('RGB',(45,45))
# 绘制图片
for x in range(45):
    for y in range(45):
        if resList[x][y] == '0':
            resImg.putpixel((x,y),white)
        else:
            resImg.putpixel((x,y),black)
resImg.save('res.png')
```

脚本获取的图片还是有一些偏差，因为题目给的图片不规范（不管我的事□），我微调脚本之后，得到下图，QR Search竟然扫不出来，但是腾讯QQ可以，我只能说一句马老板□□哈哈哈哈哈，扫码结果：
Vm0xd1NtUXlWa1pPVldoVFlUSlNjRlJVVGtOamJGWnlWMjFHVlUxV1ZqTldNakZlWVcxS1lxTnNhRmhoTVZweVdWUkdXbVZHWkhOWGJGcHBWa1paZWxaclpEUmhNVXBYVW14V2FHVnFRVGs9
base64解码七次后结果：flag{y0ud1any1s1}

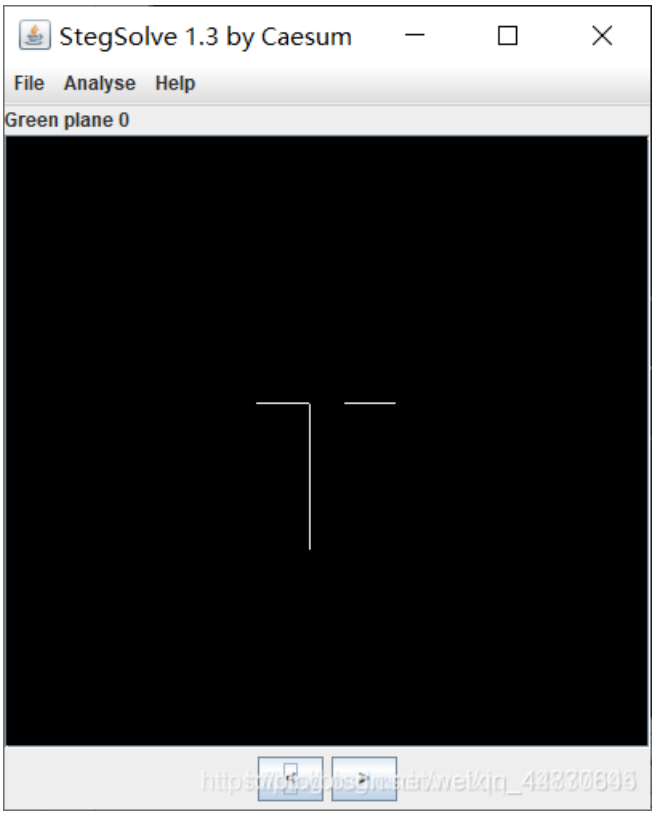
34.PEN_AND_APPLE

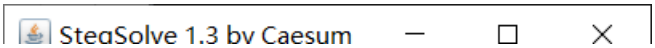
下载得到一个mp4文件，然后这是一个NTFS的隐写，但我做不出来hhhh,这道题先留着，等我之后会了我再来补上，对不起啦(>人<；)

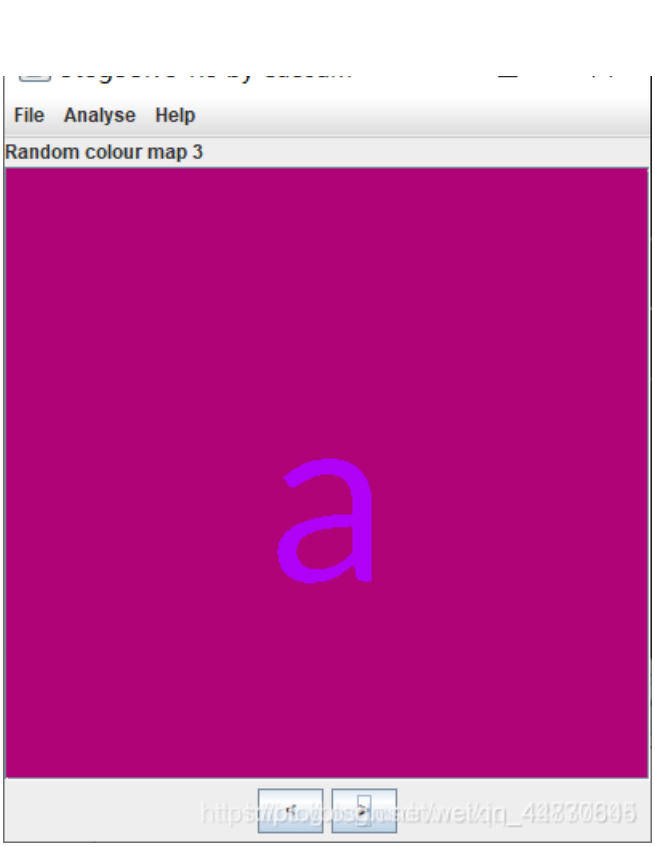
35.color

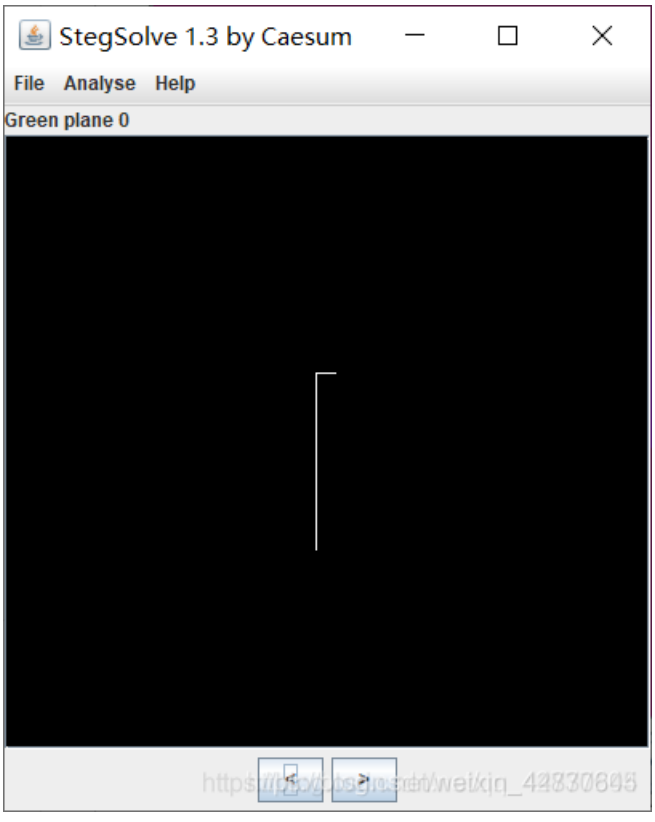
下载下来得到一组图片，用StegSolve依次打开查看，可以发现一些英文字母

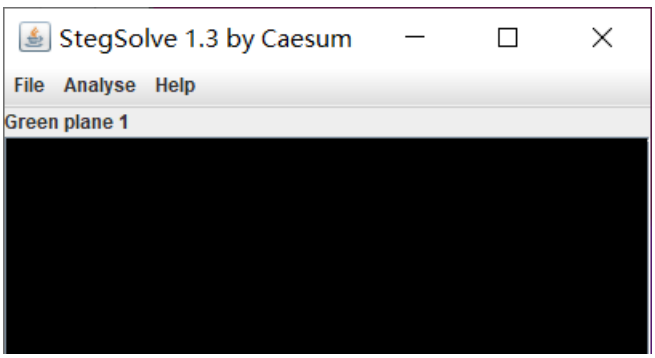

StegSolve 1.3 by Caesum
File Analyse Help
Red plane 0


StegSolve 1.3 by Caesum
File Analyse Help
Green plane 0

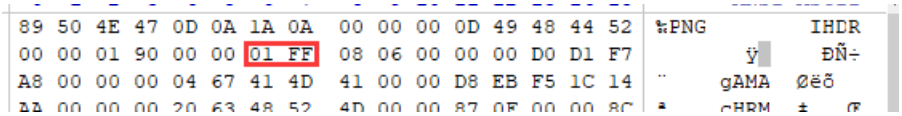

StegSolve 1.3 by Caesum


File Analyse Help
Random colour map 3
a


StegSolve 1.3 by Caesum
File Analyse Help
Green plane 0

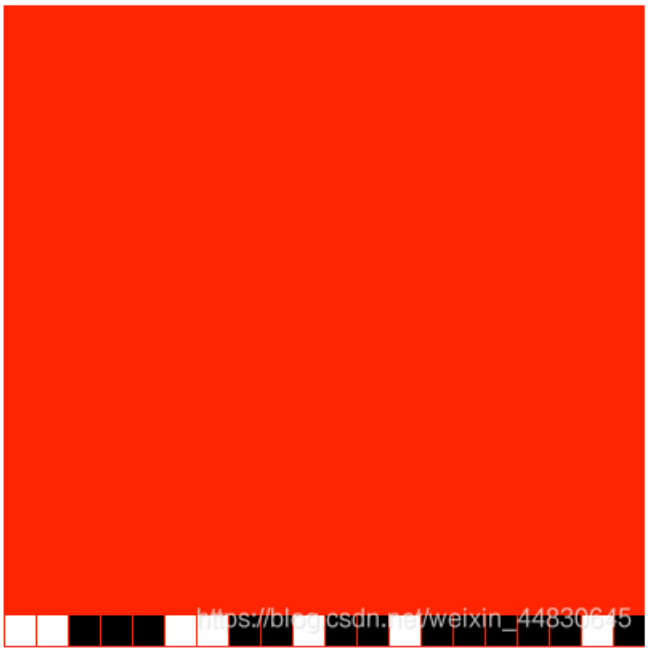

StegSolve 1.3 by Caesum
File Analyse Help
Green plane 1

连起来可以组成一句话：Make me tall，意思就是让他变高，那我们可以把他们放到winhex里面去修改他们的高度



```
89 50 4E 47 0D 0A 1A 0A  00 00 00 0D 49 48 44 52  %PNG        IHDR
00 00 01 90 00 00 01 FF  08 06 00 00 00 D0 D1 F7        ÿ    ĐÑ÷
A8 00 00 00 04 67 41 4D  41 00 00 D8 EB F5 1C 14  ¨    gAMA  Øëõ
```

之后可以看到有黑白块在图片的底部



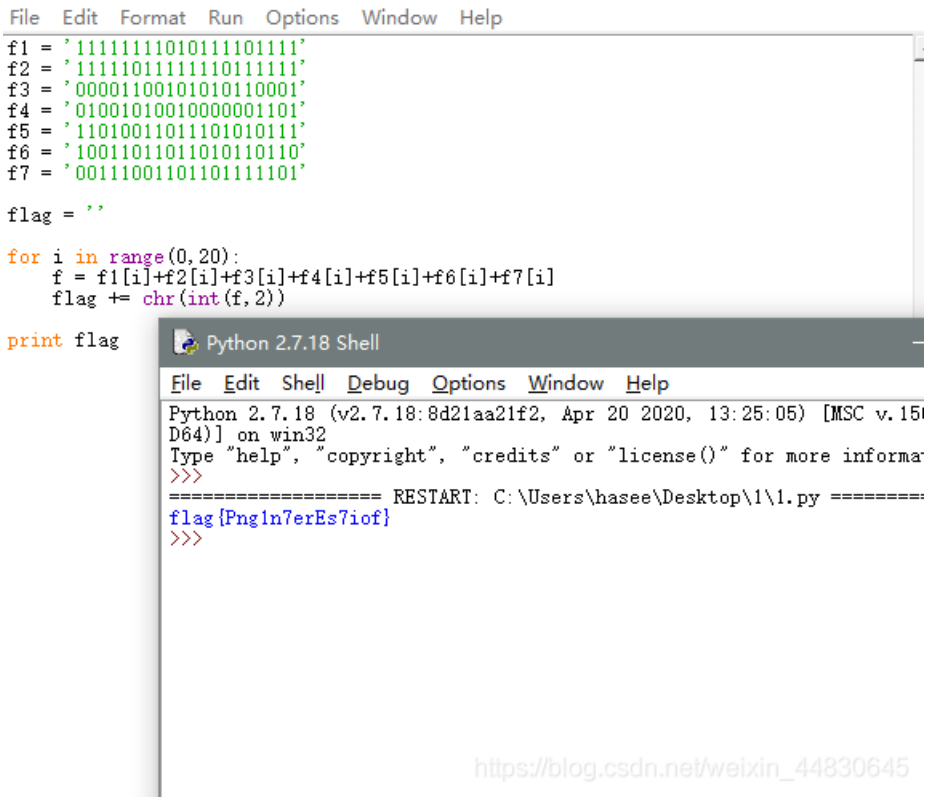
把他们以颜色转换为二进制，白色转为0，黑色转为1，我们可以拿到七串字符串

```
'11111111010111101111'
'11111011111110111111'
'00001100101010110001'
'01001010010000001101'
'11010011011101010111'
'10011011011010110110'
'00111001101101111101'
```

发现横向解密解密不出来，所以就尝试了一下纵向解密，发现当纵向排列的时候，1100110能解出来为f
那我们可以写一个脚本来解密

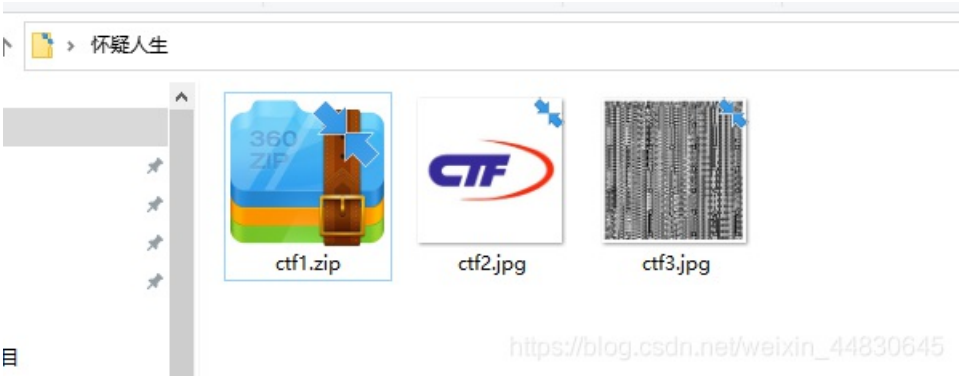
```
f1 = '11111110101111101111'
f2 = '11111011111110111111'
f3 = '00001100101010110001'
f4 = '01001010010000001101'
f5 = '11010011011101010111'
f6 = '10011011011010110110'
f7 = '00111001101101111101'

flag = ''

for i in range(0,20):
    f = f1[i]+f2[i]+f3[i]+f4[i]+f5[i]+f6[i]+f7[i]
    flag += chr(int(f,2))

print flag
```

运行即可得到flag



36.怀疑人生

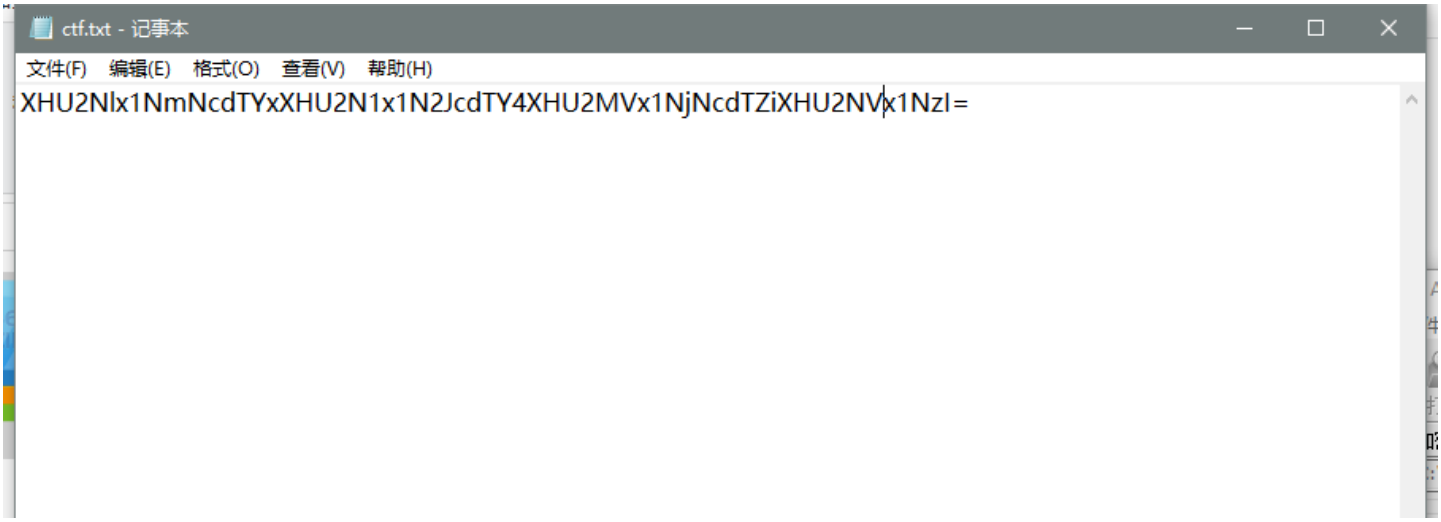
下载下来是一个名为zip的文件，把它改后缀为zip打开，能解压出来三个文件

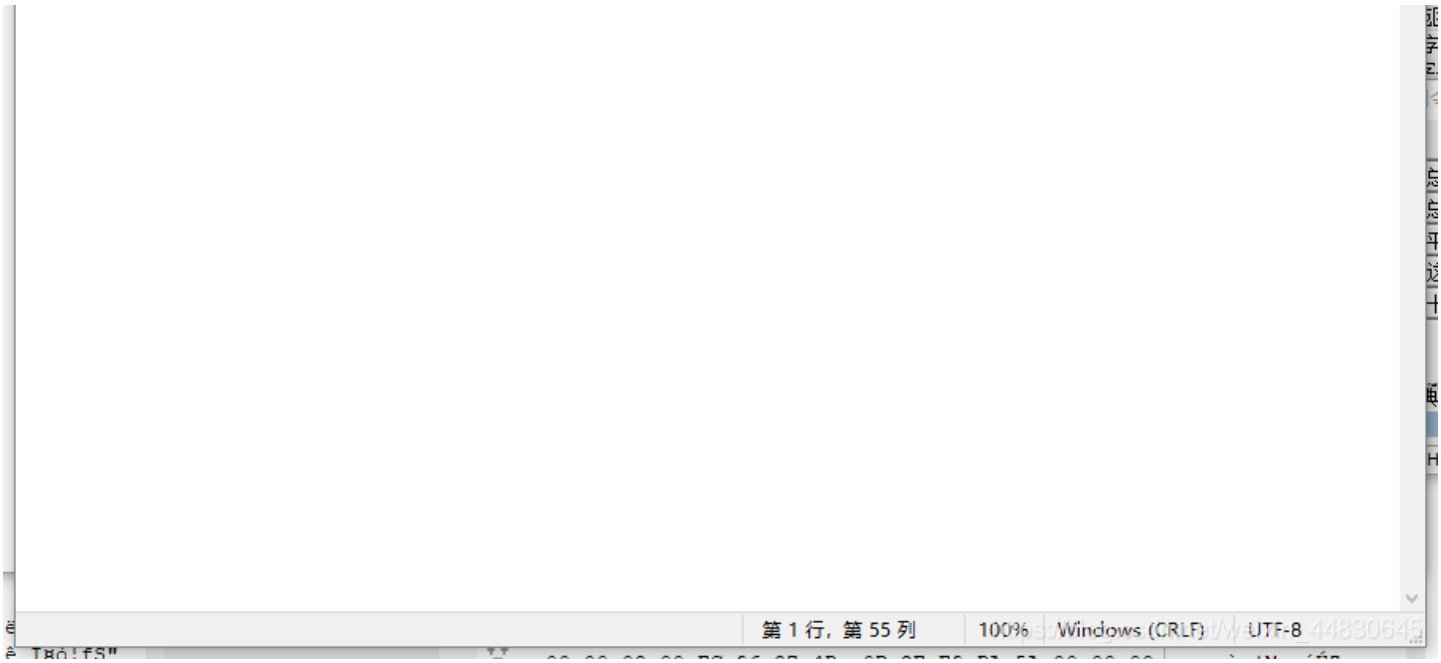


线索一、cfg1.zip

发现zip文件有加密，尝试了一下，发现不是伪加密，看了看题目，并没有给什么提示，所以就直接用字典暴力破解了，我用的字典是archpr自带的字典

得到解压密码后，解压可以得到一串base64的字符串





把它拿去解密后，可以得到一串16进制

Base64编码转换

XHU2N1x1NmNcdTYxXHU2N1x1N2JcdTY4XHU2MVx1NjNcdTZiXHU2NVx1NzI=

清空 加密 解密 解密结果以16进制显示

\u66\u6c\u61\u67\u7b\u68\u61\u63\u6b\u65\u72

复制

再拿去解密，可以得到一个

16进制到文本字符串

加密或解密字符串长度不可以超过10M

1 \u66\u6c\u61\u67\u7b\u68\u61\u63\u6b\u65\u72

16进制转字符 字符转16进制 测试用例 清空结果 复制结果

1 flag{hacker

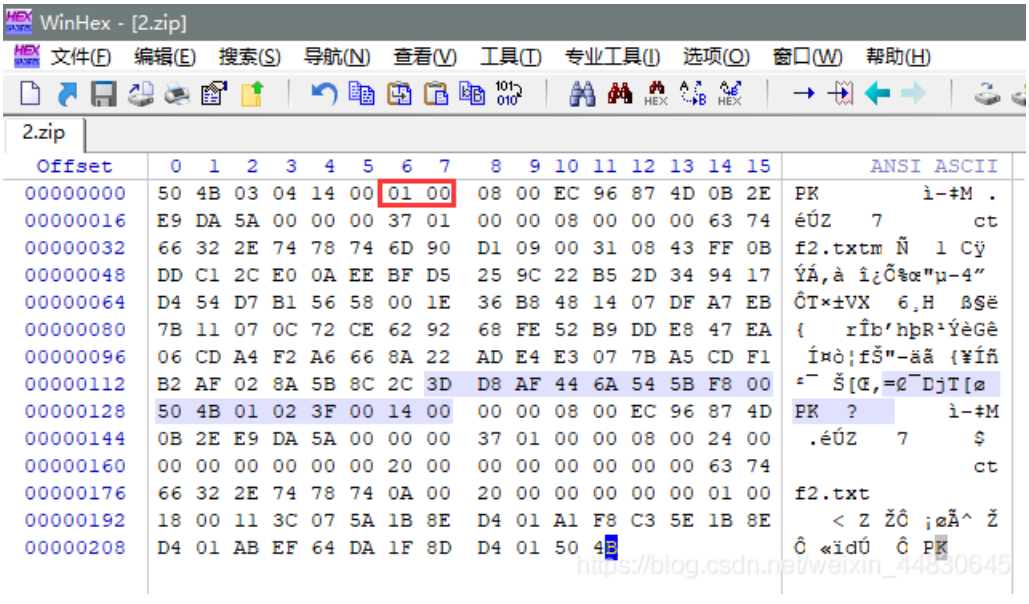
线索二、ctf2.jpg

用winhex打开，可以发现最后面有一个zip的文件

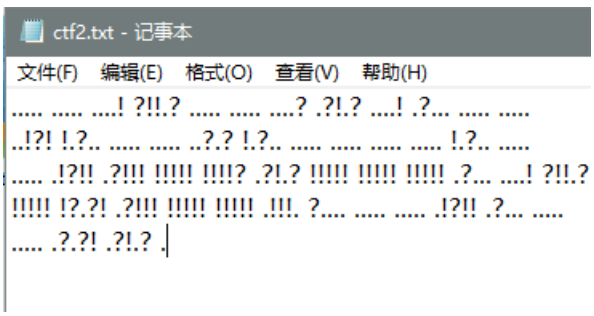
```
00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00 00 00 01 FF D9 00 00  50 4B 03 04 14 00 01 00
08 00 EC 96 87 4D 0B 2E  E9 DA 5A 00 00 00 37 01
00 00 08 00 00 00 63 74  66 32 2E 74 78 74 6D 90
D1 09 00 31 08 43 FF 0B  DD C1 2C E0 0A EE BF D5
25 9C 22 B5 2D 34 94 17  D4 54 D7 B1 56 58 00 1E
36 B8 48 14 07 DF A7 EB  7B 11 07 0C 72 CE 62 92
68 FE 52 B9 DD E8 47 EA  06 CD A4 F2 A6 66 8A 22
AD E4 E3 07 7B A5 CD F1  B2 AF 02 8A 5B 8C 2C 3D
D8 AF 44 6A 54 5B F8 00  50 4B 01 02 3F 00 14 00
00 00 08 00 EC 96 87 4D  0B 2E E9 DA 5A 00 00 00
37 01 00 00 08 00 24 00  00 00 00 00 00 00 20 00
00 00 00 00 00 00 63 74  66 32 2E 74 78 74 0A 00
20 00 00 00 00 00 01 00  18 00 11 3C 07 5A 1B 8E
D4 01 A1 F8 C3 5E 1B 8E  D4 01 AB EF 64 DA 1F 8D
D4 01 50 4B 05 06 00 00  00 00 01 00 01 00 5A 00
00 00 80 00 00 00 00 00
```

```
    ÿÙ  PK
  ì-‡M .éÚZ   7
      ctf2.txtm
Ñ  1 Cÿ ÝÁ,à î¿Õ
%œ"µ-4” ÔT×‡VX
6¸H  ß§ë{   rÎb’
hÞR¹ÝèGê Í¤ò¦fŠ"
-ää {¥Íñ²¯ Š[Œ,=
Ø¯DjT[ø PK  ?
    ì-‡M .éÚZ
7     $
      ctf2.txt
           < Z Ž
Ô ¡øÃ^ ŽÔ «ïdÚ
Ô PK          Z
  €
```

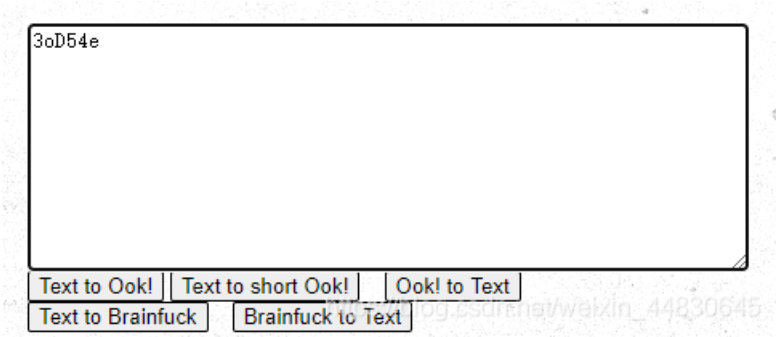

把图片后缀改为zip，打开发现需要密码，输入上一个密码发现不行，爆破软件说不是一个加密过的zip，猜测是伪加密，丢进winhex查看，的确是伪加密，把红框里面的01改为00即可



解压出来是一串我看不懂的东西



百度查了一下，发现是ook加密，解密可得到第二段的flag（这里的字符串是ook省略没写，可以看出来是ook加密）



线索三、ctf3.jpg

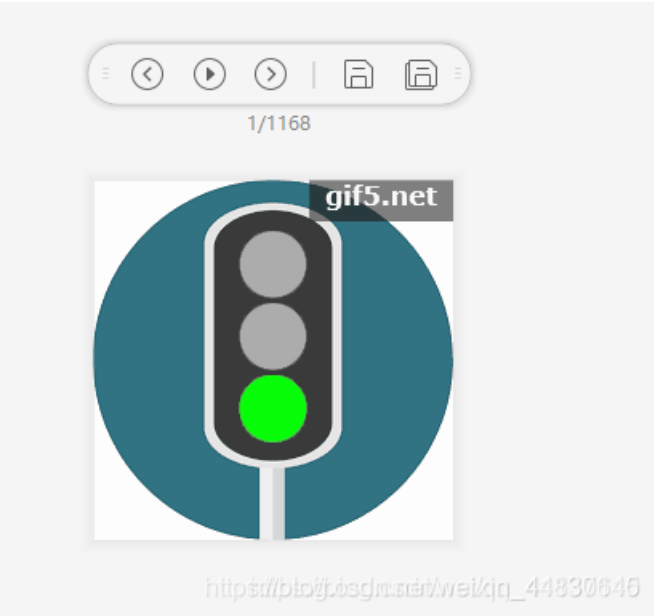
是一张图片，看起来像一张二维码，但我又觉得不是，看起来太奇怪了，想着怎么修复这张二维码，但是一点办法，一点头绪都没有，就抱着试一试的心态，拿出QQ，扫描二维码，神器的就来了

wc，它居然扫出来了，不得不说，QQ还是nb的

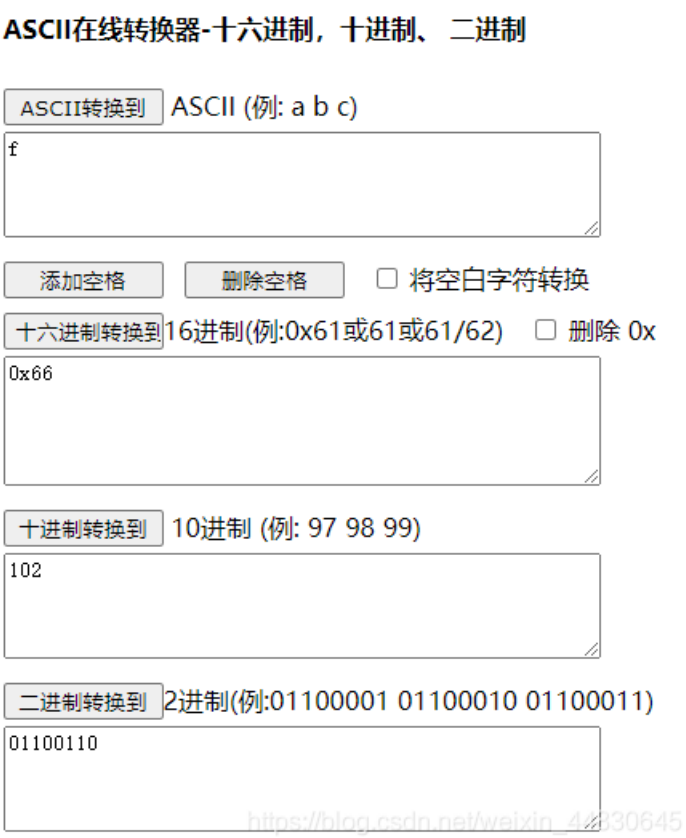
最后把他们三段拼接起来就是flag了
flag{hackermisc12580}

37.红绿灯

下载下来得到一个GIF的图片，然后用分离工具把他们全部分离出来，能分离出来1168张图片



观察一下，发现黄灯闪的是最少的，每一个黄灯之前都有8个灯在亮，所以可以联想到二进制
我们可以推测红色和绿色对应二进制0和1，黄色作为分隔，那么久只有两种结果
01100110或10011001，这两种都拿去转换为ASCII码，发现01100110对应为f



就可以确定绿灯对应0，红灯对应1，把他们全部转换一遍，可以手动转换，也可以写一个python的脚本
这里我就直接搬运这位大佬的脚本了

```
# -*- coding: cp936 -*-
from PIL import Image

savepath='D:\\gif\\'
im=Image.open('D:\\Traffic_Light.gif')
try:
#tell是帧数，而seek是取当前帧数的图片。
  im.save(savepath+'light{0}.png'.format(im.tell()))
  while True:
    im.seek(im.tell()+1)
    im.save(savepath+'light{0}.png'.format(im.tell()))
except:
  pass

flag=""
for i in range(1168):
  image=Image.open(savepath+'light'+str(i)+'.png')
  #print image.getpixel((115,55))#输出颜色值
  #print image.getpixel((115,145))
  if image.getpixel((115,55))==251:
    flag+=str(1)
  elif image.getpixel((115,145))==186:
    flag+=str(0)
flag= hex(int(flag,2))[2:-1].decode('hex')#二进制转字符串
print flag
```

然后即可得到flag

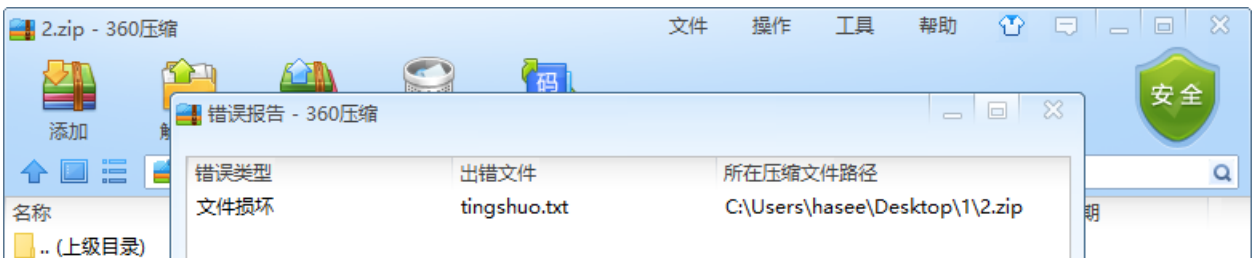
```
flag{Pl34s3_p4y_4tt3nt10n_tO_tr4ff1c_s4f3ty_wh3n_yOu_4r3_0uts1d3}
```

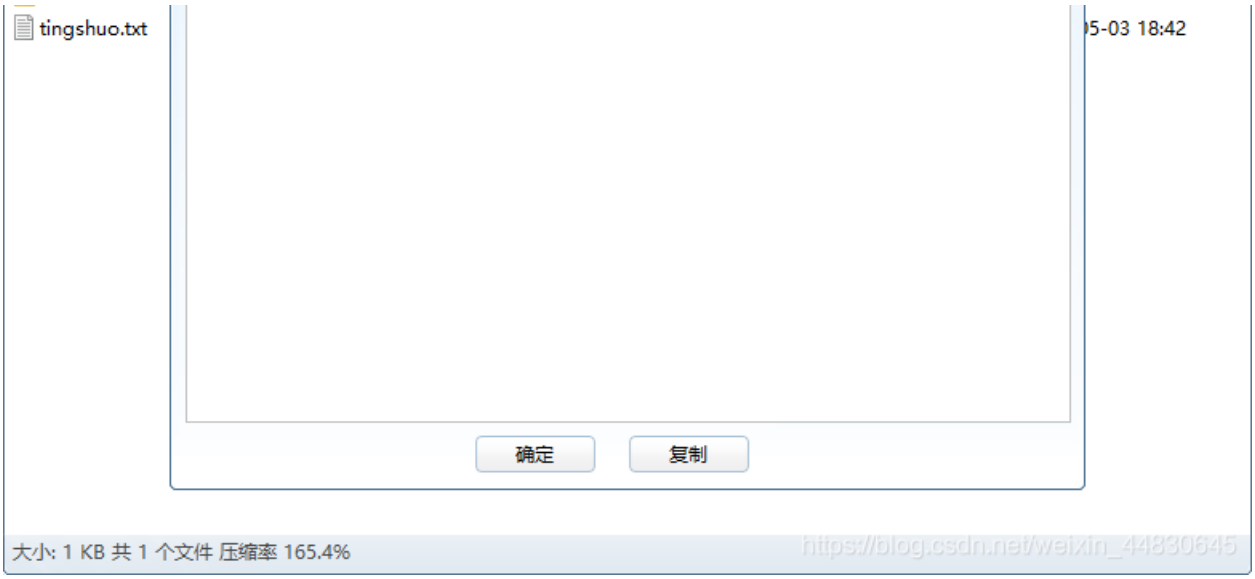
38.不简单的压缩包

下载下来得到一个加密过的zip文件，里面有一个加密过的swf文件，然而并不知道密码，放进winhex里面去看看

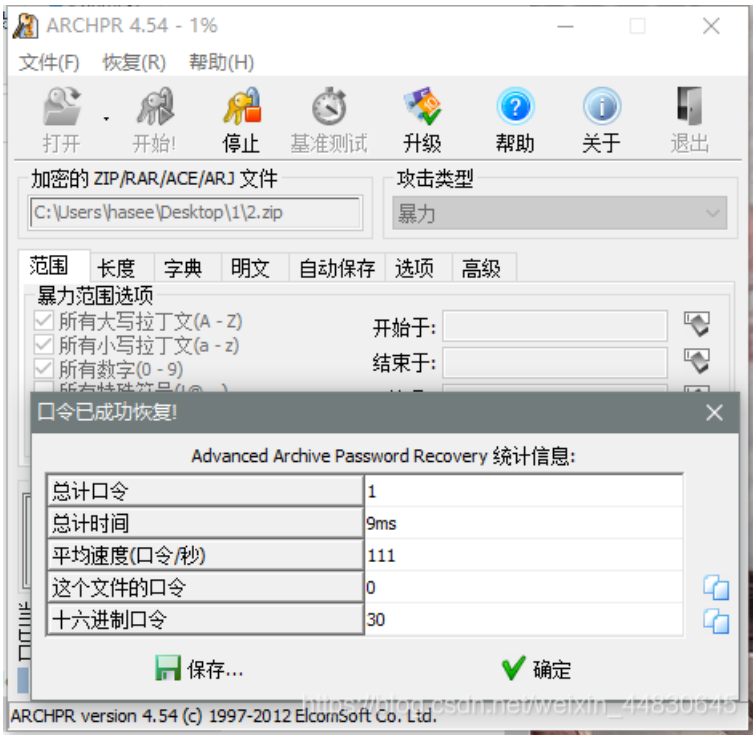
```
00 F9 24 3B 3F 9E 01 D5  01 3B DF 0A E8 9E 01 D5   ù$;?ž Õ ;ß èž Õ
01 3B DF 0A E8 9E 01 D5  01 50 4B 05 06 00 00 00   ;ß èž Õ PK
00 01 00 01 00 5A 00 00  00 5F 3B 06 00 00 00 50       Z    _;     P
4B 03 04 14 00 09 00 08  00 4A 95 A3 4E 65 1C 30   K         J•£Ne 0
5E 2B 00 00 00 1A 00 00  00 0C 00 00 00 74 69 6E   ^+            tin
67 73 68 75 6F 2E 74 78  74 AA C5 51 DA 25 C6 68   gshuo.txt ªÅQÚ%Æh
42 FE B6 9C 58 63 2B 80  C5 12 7C DC F1 6A 88 2F   Bþ¶œXc+€Å |Üñjˆ/
A1 A5 60 32 B7 87 C7 DE  70 11 B1 66 FC DE EC 0F   ¡¥`2·‡ÇÞp ±füÞì
11 80 FF C3 50 4B 07 08  65 1C 30 5E 2B 00 00 00    €ÿÃPK  e 0^+
1A 00 00 00 50 4B 01 02  1F 00 14 00 09 00 08 00       PK
4A 95 A3 4E 65 1C 30 5E  2B 00 00 00 1A 00 00 00   J•£Ne 0^+
0C 00 24 00 00 00 00 00  00 00 20 00 00 00 00 00     $
00 00 74 69 6E 67 73 68  75 6F 2E 74 78 74 0A 00     tingshuo.txt
20 00 00 00 00 00 01 00  18 00 93 F1 52 E2 9C 01            "ñRâœ
D5 01 7B 86 DF DC 9C 01  D5 01 7B 86 DF DC 9C 01   Õ {†ßÜœ Õ {†ßÜœ
D5 01 50 4B 05 06 00 00  00 00 01 00 01 00 5E 00   Õ PK           ^
00 00 65 00 00 00 00 00                            e
```


可以发现最后面有个压缩包，把它手动分离出来，发现分离出来的还是有加密，我以为是伪加密，试了一下，好家伙，直接报错了

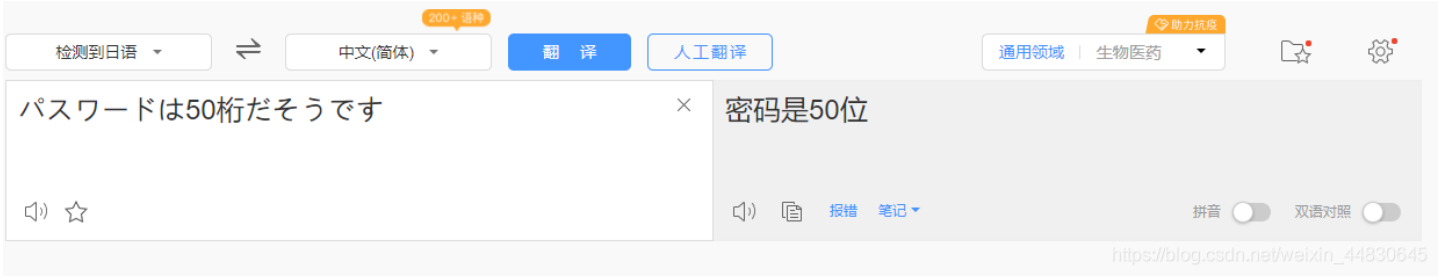




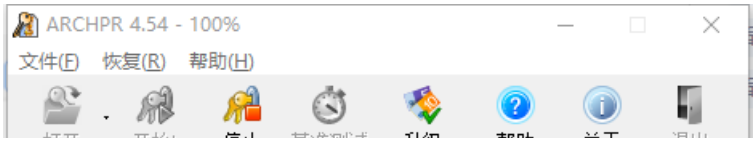
然后去看了看题干，发现并没有什么提示，那就直接爆破呗，爆破出来显示密码是0



用这个密码解压可以得到一个文档，里面写着一串日文，拿去翻译可以得到提示

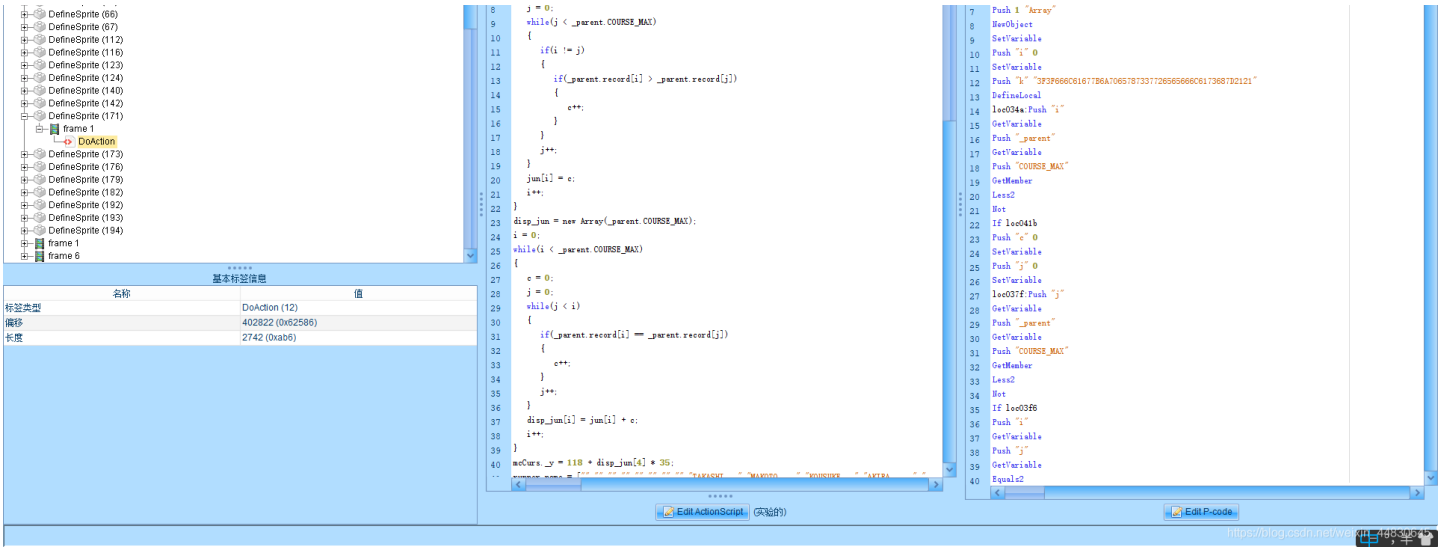


emmmm，这是要爆破死我=-=，没办法咯，那就直接爆破，一下子就爆破出来了，密码是50个a



用这个密码解压可以得到一个小游戏，可能是我比较菜，玩不通关，然后就去百度怎么破解swf的文件，让我找到了一个工具

把游戏放进去，可以看到游戏的源码，然后找了一下，在脚本里面能看到一串奇怪的字符串

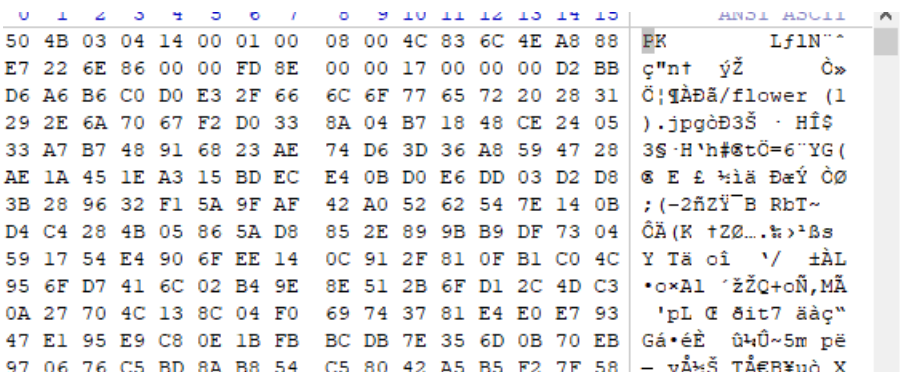


把它转为字符串就可以得到flag了

提交的时候记得把?!符号去掉哦

39.一枝独秀

下载下来得到一个压缩包，然后解压出来是一个jpg文件，但是并打不开，然后用winhex分析，发现它是一个zip文件

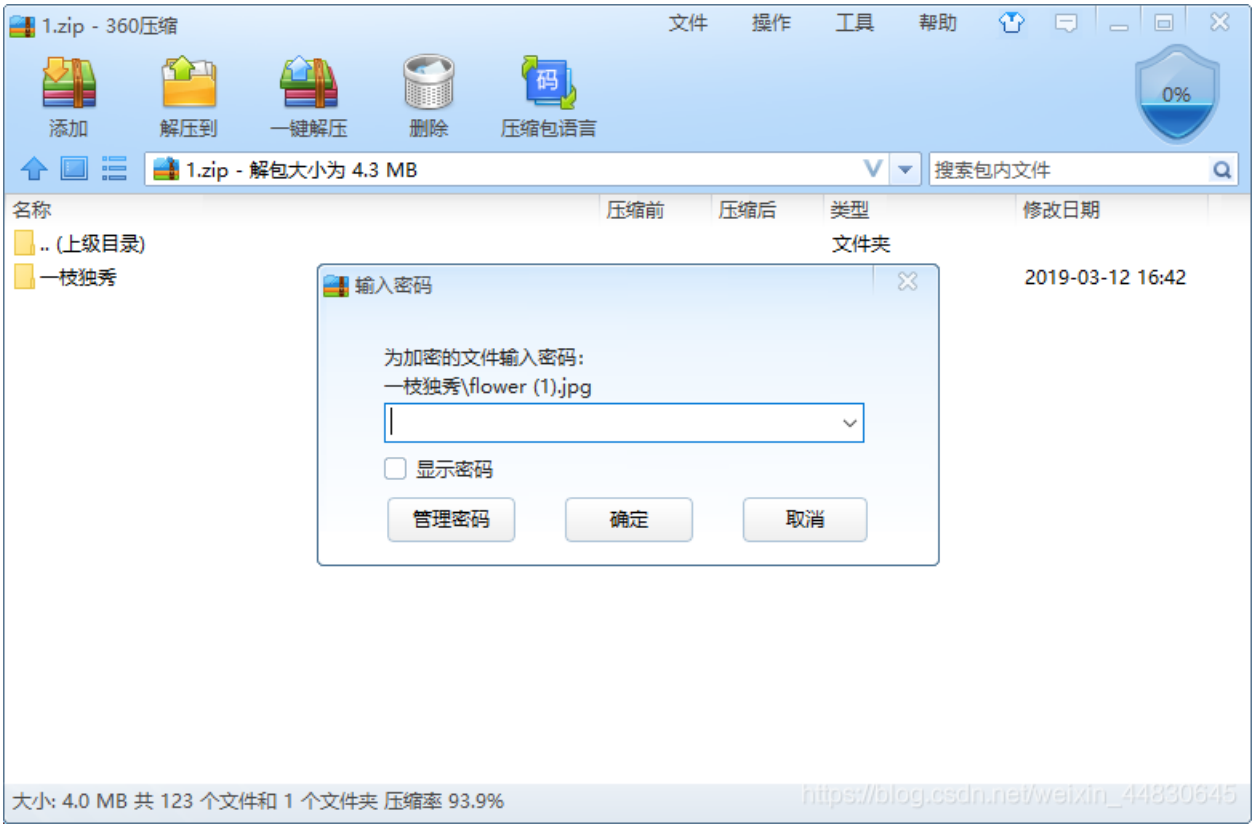


```
7C 63 CD D2 C3 C7 A9 32  65 AF 59 81 4C D6 20 EF  |cÍÒÃÇ©2e¯Y LÖ ï
27 21 F5 03 D2 81 3C 70  51 3F E4 4E 6B A3 E6 93  '!õ Ò <pQ?äNk£æ"
0D 2B C9 C1 03 67 05 C7  ED E7 C0 50 93 F1 54 AE   +ÉÁ g ÇíçÀP"ñT®
3B E9 5B A4 4F F3 48 A6  AA 12 60 F5 08 5D 51 14  ;é[¤OóH¦ª `õ ]Q
6B D6 63 74 E4 13 1E 65  FA 36 F8 BB C1 64 E0 3E  kÖctä  eú6ø»Ádà>
3E 87 EF CD AD EC 81 C8  B3 0E F8 EC C2 6E 7B DA  >‡ïÍ-ì È³ øìÂn{Ú
59 74 17 52 D2 92 E8 43  4B A4 B2 ED E8 8A 93 8D  Yt RÒ'èCK¤²íèŠ"
C6 82 8B 61 A5 23 77 71  14 E1 4B 4E 72 93 5E A4  Æ‚‹a¥#wq áKNr"^¤
```

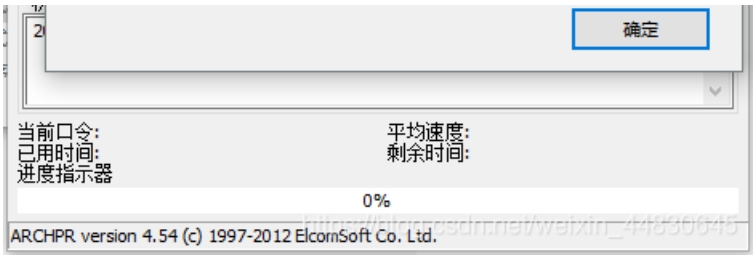

把它后缀改为zip就可以打开了**需要注意的是，文件上面有一段没用的文件头需要删除**

```
50 4B 03 04 14 00 00 00  00 00 4C 85 6C 4E 00 00  PK        L…lN
12 34 56 78 00 00 00 00  00 00 09 00 00 00 D2 BB   4Vx           Ò»
D6 A6 B6 C0 D0 E3 2F 50  4B 03 04 14 00 01 00 08  Ö¦¶ÀÐã/PK
00 4C 83 6C 4E A8 88 E7  22 6E 86 00 00 FD 8E 00   LƒlN¨ˆç"n†  ýŽ
00 17 00 00 00 D2 BB D6  A6 B6 C0 D0 E3 2F 66 6C       Ò»Ö¦¶ÀÐã/fl
6F 77 65 72 20 28 31 29  2E 6A 70 67 F2 D0 33 8A  ower (1).jpgòÐ3Š
```

删除之后打开，说需要密码，然后就直接爆破密码



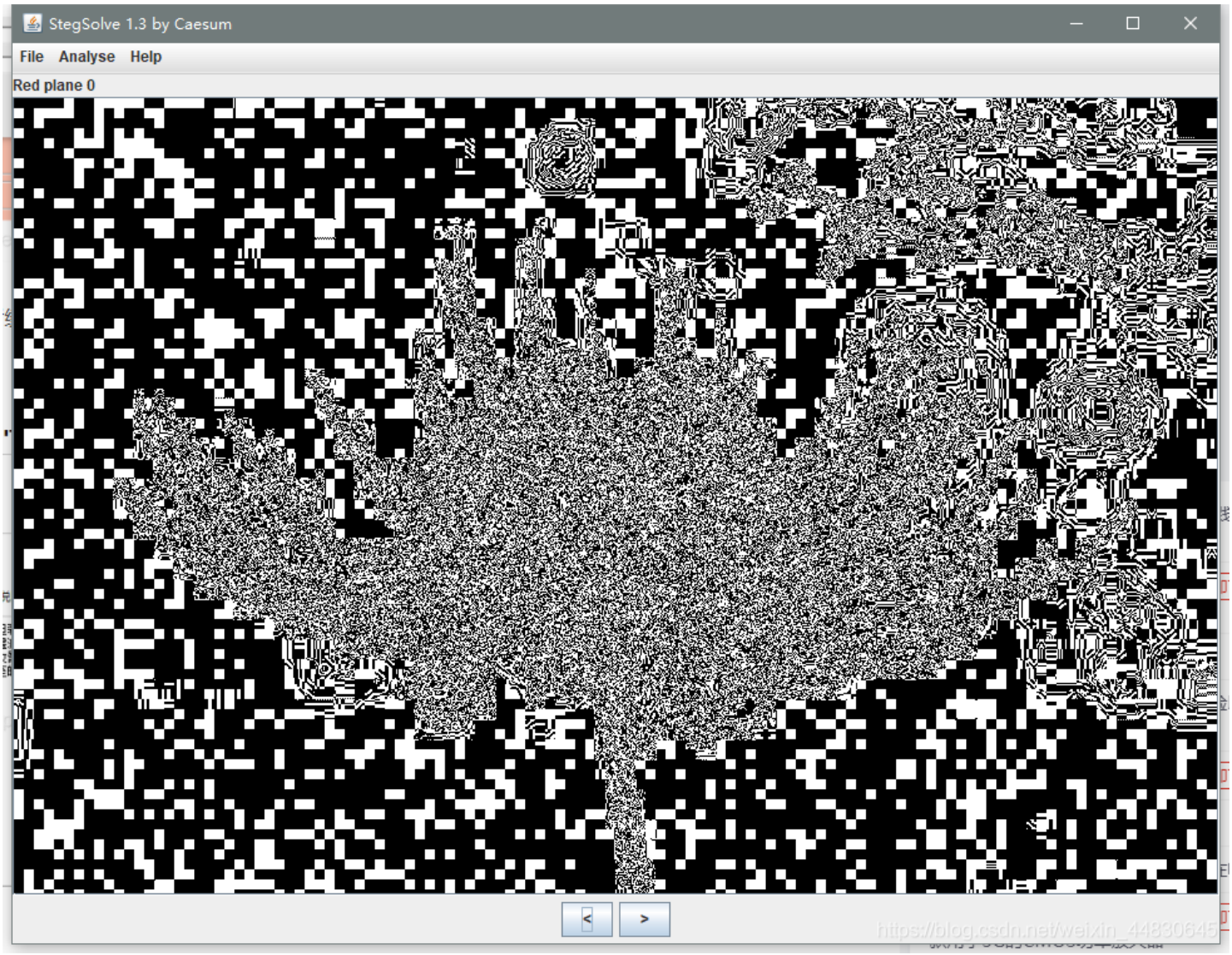
爆破出来密码是12345678，不知道为什么我的爆破不了



然后解压出来可以看到很多的花的图片，观察了一下，发现就这张图片的大小和别的图片不一样

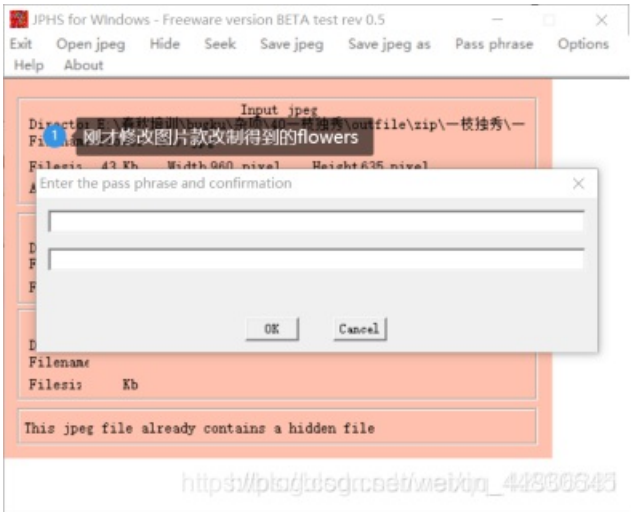
flower (78).jpg	2019/3/12 16:26	JPG 文件	36 KB
flower (79).jpg	2019/3/12 16:26	JPG 文件	36 KB
flower (80).jpg	2019/3/12 16:26	JPG 文件	36 KB
flower (81).jpg	2019/3/12 16:26	JPG 文件	43 KB
flower (82).jpg	2019/3/12 16:26	JPG 文件	36 KB
flower (83).jpg	2019/3/12 16:26	JPG 文件	36 KB
flower (84).jpg	2019/3/12 16:26	JPG 文件	36 KB

发现了一个很像二维码，但又不是二维码的东西

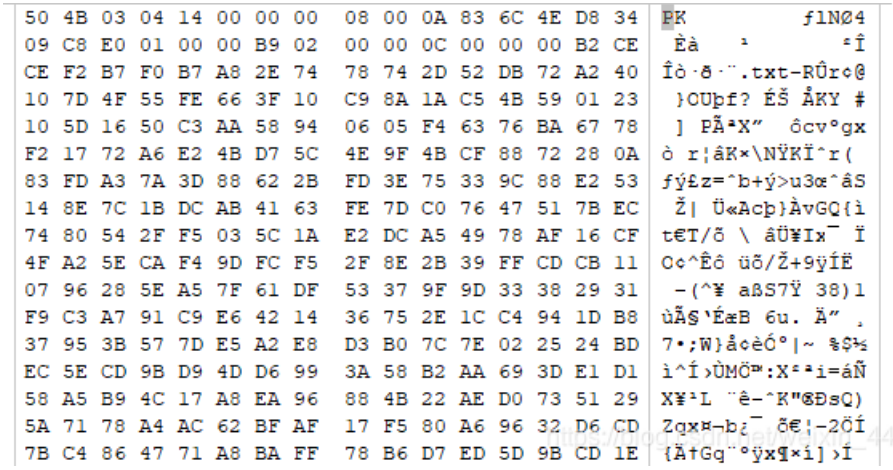


做到这里我就不会了，就去翻了翻大佬们的博客，发现需要JPHS这个工具
打开工具-open JPEG-打开刚才标记的81号

Seek–发现需要密码，于是继续观察81号



右键查看属性得到了主题flowers，可能是密码，填上试试，发现对了，把文件保存下载然后用winhex打开，能看出来这是一个zip文件



改zip后缀解压出来能得到一个文档，里面有些文字

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

从一朵花中看到一个世界，那从一段佛文中能看到什么呢?
你站在4楼的栏杆上眺望远方，如果参悟不出来就打算跳下去

佛曰：阿罰豆缽娑提諳竟諳迦亦侄栗侄大梵尼朋梵彌哆耨除怛奢般是諳爍悉哆爍冥參特參怯涅皤吉缽阿藝耶諳勝侄竟離諳諸弋缽曰。梵究吶耨盧他姪明漫究吶得哆藐集能冥盡滅知俱朋怯室神奢羅姪豆罰帝遠蘇明梵苦奢密侄曰缽者特哆呼勝蘇不冥死等那阿冥悉奢薩豆涅缽波罰。罰摩侄故罰夢缽恐皤寫諳闍舍哆得波苦奢即罰恐冥道一哆究梵呼冥闍哆上罰南訶諳寫冥依皤者哆諦故死哆夷菩侄曰吶逝至皤佛諳耶

下面那个加密很常见了，是与佛论禅加密，解密一下能得到

H-hDs1OOZL3lhIZZbeRSbbbVRZNm32W2X33mGm3Txt999RdV9hxO

听佛说宇宙的真谛 参悟佛所言的真意 普度众生

春来花自青，秋至叶飘零

佛曰：阿罰豆缽娑提諳竟諳迦亦侄栗侄大梵尼朋梵彌哆耨除怛奢般是諳爍悉哆爍冥參特參怯涅皤吉缽阿藝耶諳勝侄竟離諳諸尼缽曰。梵究吶耨盧他姪明漫究吶得哆藐集能冥盡滅知俱朋怯室神奢羅姪豆罰帝遠蘇明梵苦奢密侄曰缽者特哆呼勝蘇不冥死等那阿冥悉奢薩豆涅缽波罰。罰摩侄故罰夢缽恐皤寫諳闍舍哆得波苦奢即罰恐冥道一哆究梵呼冥闍哆上罰南訶諳寫冥依皤者哆諦故死哆夷菩侄曰吶逝至皤佛諳耶

然后再根据题目的提示，拿去栅栏密码解密

H-hDs1OOZL31hIZZbeRSbbbVRZNm32W2X33mGm3Txt999RdV9hxO

每组字数 4 加密 解密

HINT-ZmxhZ3tDb29seW91R290SXROb3dZb3VLbm93VGhlRmxhZ30

能得到一个HINT，这个明显是一个base64的加密，再拿去解密就能得到flag了

ZmxhZ3tDb29seW91R290SXROb3dZb3VLbm93VGhlRmxhZ30

清空 加密 解密 ☐解密结果以16进制显示

flag{CoolyouGotItNowYouKnowTheFlag}

40.小猪佩奇

还在研究