CTF攻防世界刷题51-

原创

五五六六0524 已于 2022-03-17 21:15:11 修改 3188 收藏
分类专栏： CTF积累及刷题 文章标签： 安全
于 2022-03-02 21:14:42 首次发布

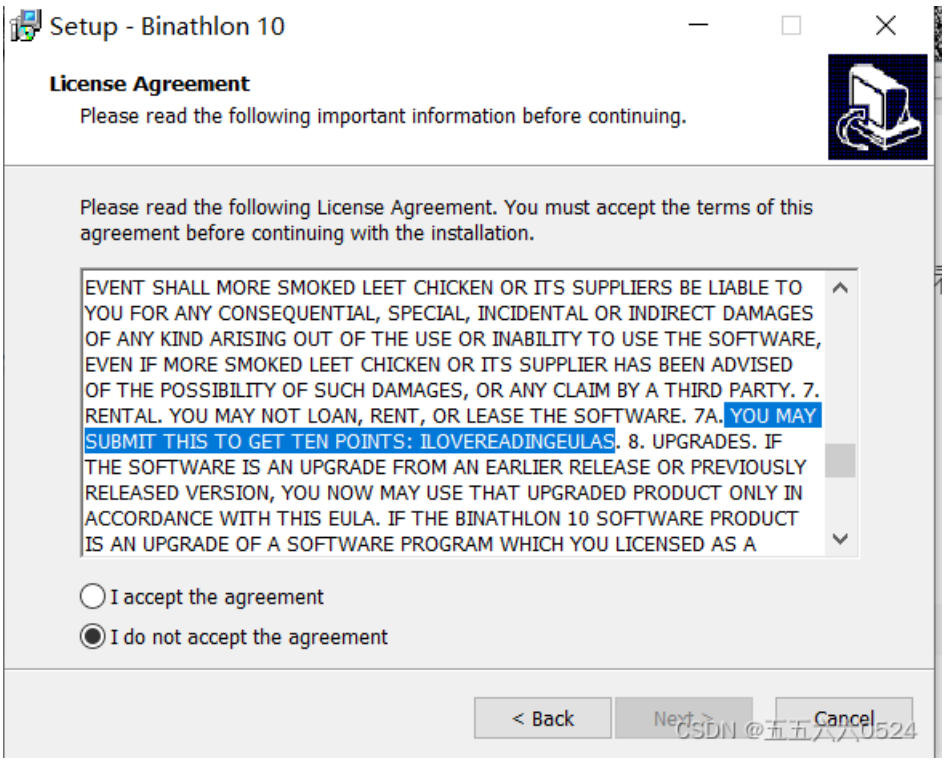
目录

51、Just-No-One

这一题实在一言难尽，它居然在协议里，我拖进kail里看了半天，也没发现什么，最后看的wp

攻防世界 Misc高手进阶区 4分题 Just-No-One_闵行小鱼塘-CSDN博客

在这，翻译一下就是“你可以提交这个来得到10分: 违反规定”，真离谱



52、labour

拖进kail里看一看

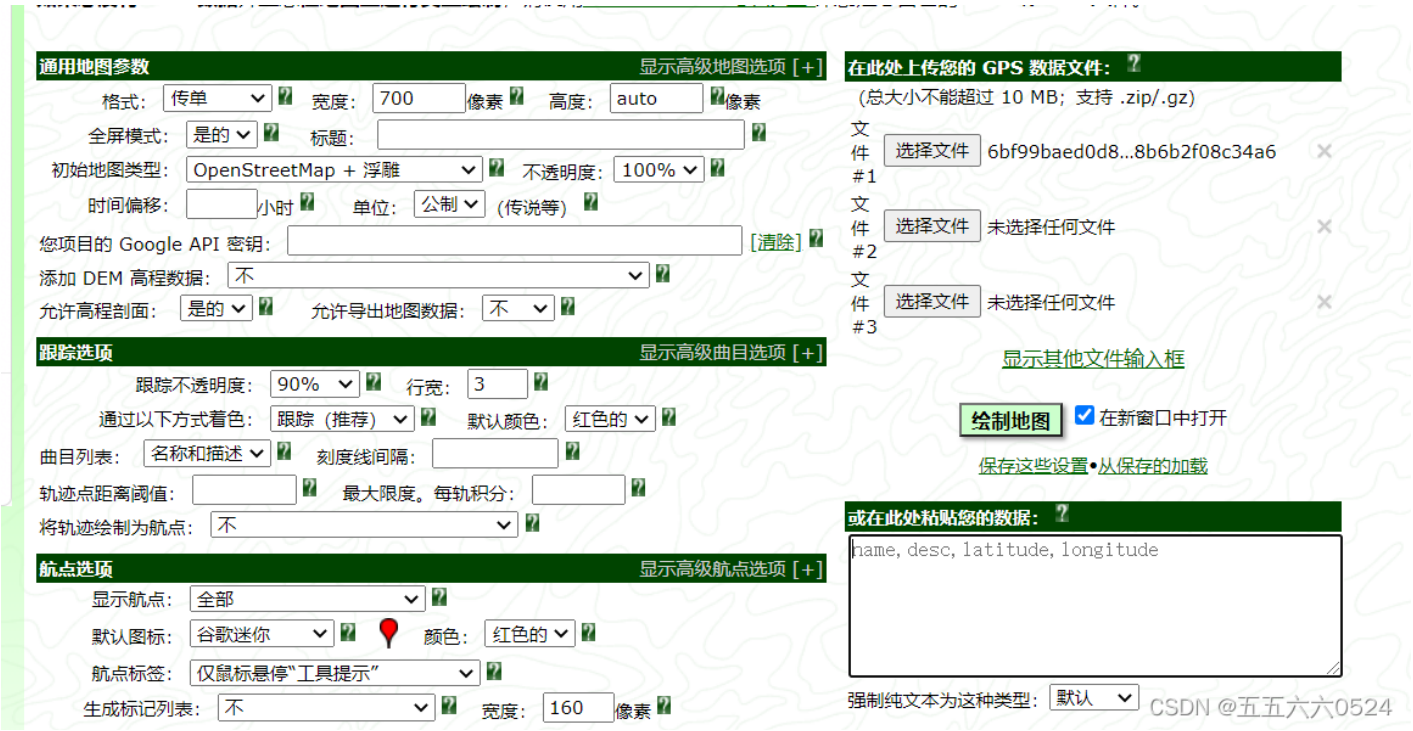
```
└─$ strings 1                                                                    1 ×
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<gpx version="1.1" creator="BITSCTF" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.topografi
x.com/GPX/1/1" xsi:schemaLocation="http://www.topografix.com/GPX/1/1 http://www.topografix.com/GPX/1/1/gpx.xsd">
<!--Use appropriate brackets and underscores to separate words if you succeed-->
<wpt lat="23.71697" lon="89.45508">
<ele>12.1</ele>
<name>WP01-A</name>
</wpt>
<wpt lat="22.82885" lon="80.79786">
<name>WP02-B</name>
</wpt>
<wpt lat="39.88276" lon="58.81642">
<name>WP03-C</name>
</wpt>
<wpt lat="15.43674" lon="27.65039">
<name>WP04-D</name>
</wpt>
<wpt lat="12.69179" lon="17.50781">
<ele>288.7</ele>
<name>WP05-E</name>
</wpt>
<wpt lat="14.91081" lon="100.47656">
<ele>13.1</ele>
<name>WP06-F</name>
</wpt>
<wpt lat="45.9267" lon="2.21484">
<ele>557.9</ele>
<name>WP07-G</name>
</wpt>
```


Use appropriate brackets and underscores to separate words if you succeed-->翻译就是“如果成功，使用适当的方括号和下划线分隔单词 -- >”

下面是经纬度，把对应的地点查出来，用这个网站

GPS Visualizer: Draw a map from a GPS data fileGPS Visualizer can read GPS data files (tracklogs & waypoints), street addresses, or simple coordinates, and plot them on Leaflet maps or Google Maps.https://www.gpsvisualizer.com/map_input

把文件传进去，然后点绘制地图



然后选择OSM(TF landscape)

把各个地点标出来

> WP01-A Bangladesh
> WP02-B India
> WP03-C Uzbekistan
> WP04-D Sudan
> WP05-E Chad
> WP06-F Thailand
> WP07-G France
> WP08-H Malaysia
> WP09-I Afghanistan
> WP10-J Pakistan
> WP11-K Turkey
> WP12-M Romania
> WP13-M Egypt
> WP16-P China
> WP17-Q Kazakhstan

首字母连起来就是，还要记得用下划线分割单词，BITSCTF{MAP_THE_HACK}

53、hong

binwalk一下发现有东西

```
└─$ binwalk hong.mp3

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
55334         0×D826          Zlib compressed data, default compression
56734         0×DD9E          Zlib compressed data, default compression
82483         0×14233         JPEG image data, JFIF standard 1.01
82513         0×14251         TIFF image data, big-endian, offset of first image directory: 8
112115        0×1B5F3         Zlib compressed data, default compression
112861        0×1B8DD         Zlib compressed data, default compression
136938        0×216EA         Zlib compressed data, default compression
138170        0×21BBA         JPEG image data, JFIF standard 1.01
138200        0×21BD8         TIFF image data, big-endian, offset of first image directory: 8
192612        0×2F064         Zlib compressed data, default compression
194332        0×2F71C         Zlib compressed data, default compression
195518        0×2FBBE         Zlib compressed data, default compression
274685        0×430FD         Zlib compressed data, default compression
275977        0×43609         Zlib compressed data, default compression
276905        0×439A9         Zlib compressed data, default compression
277642        0×43C8A         Zlib compressed data, default compression
285855        0×45C9F         Zlib compressed data, default compression
294828        0×47FAC         Zlib compressed data, default compression
303158        0×4A036         Zlib compressed data, default compression
311823        0×4C20F         Zlib compressed data, default compression
320796        0×4E51C         Zlib compressed data, default compression
333060        0×51504         Zlib compressed data, default compression
335054        0×51CCE         Zlib compressed data, default compression
498178        0×79A02         Zlib compressed data, default compression
501817        0×7A839         Zlib compressed data, default compression
```


提取出来两张图，二维码扫出来一大堆数字，另一张图直接出BCTF{cute&fat_cats_does_not_like_drinking}

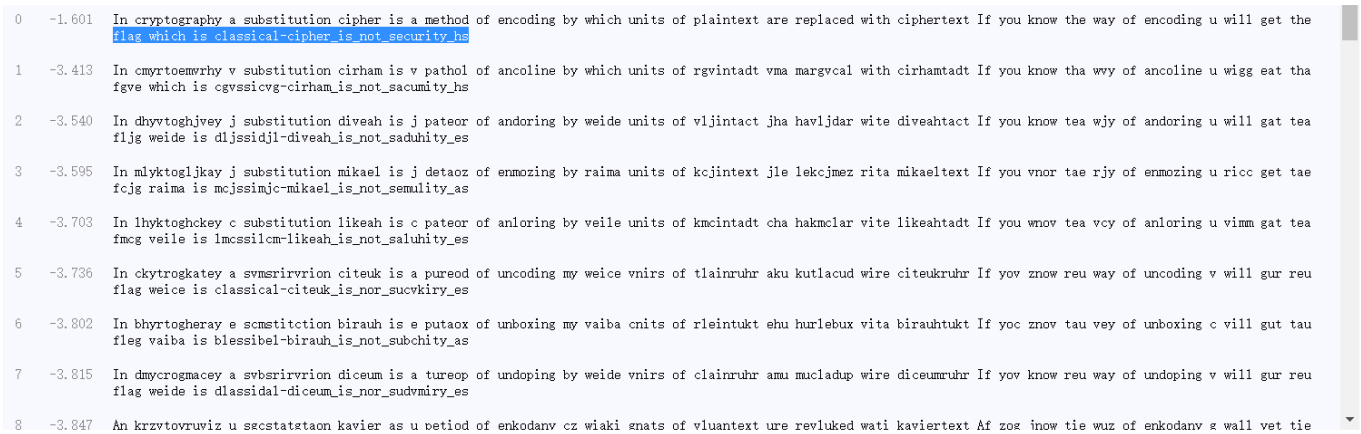
54、4-2

题目得到这些

> Eg qnlyjtcnzydl z umaujejmjetg qeydsn eu z bsjdtx tw sgqtxegc al kdeqd mgeju tw yrzegjsoj zns nsyrzqsx kejd qeydsnjsoj
> Ew ltm fgtk jds kzl tw sgqtxegc m kerr csj jds wrzc kdeqd eu qrzuueqzr-qeydsn_eu_gtj_usqmnejl_du

我还以为要翻译，没啥结果，binwalk也没有，找了好几个wp，这玩意不是凯撒密码，是要词频分析（叫古典密码自动化爆破，我不太懂），flag{classical-cipher_is_not_security_hs}

quipqiup - cryptoquip and cryptogram solver

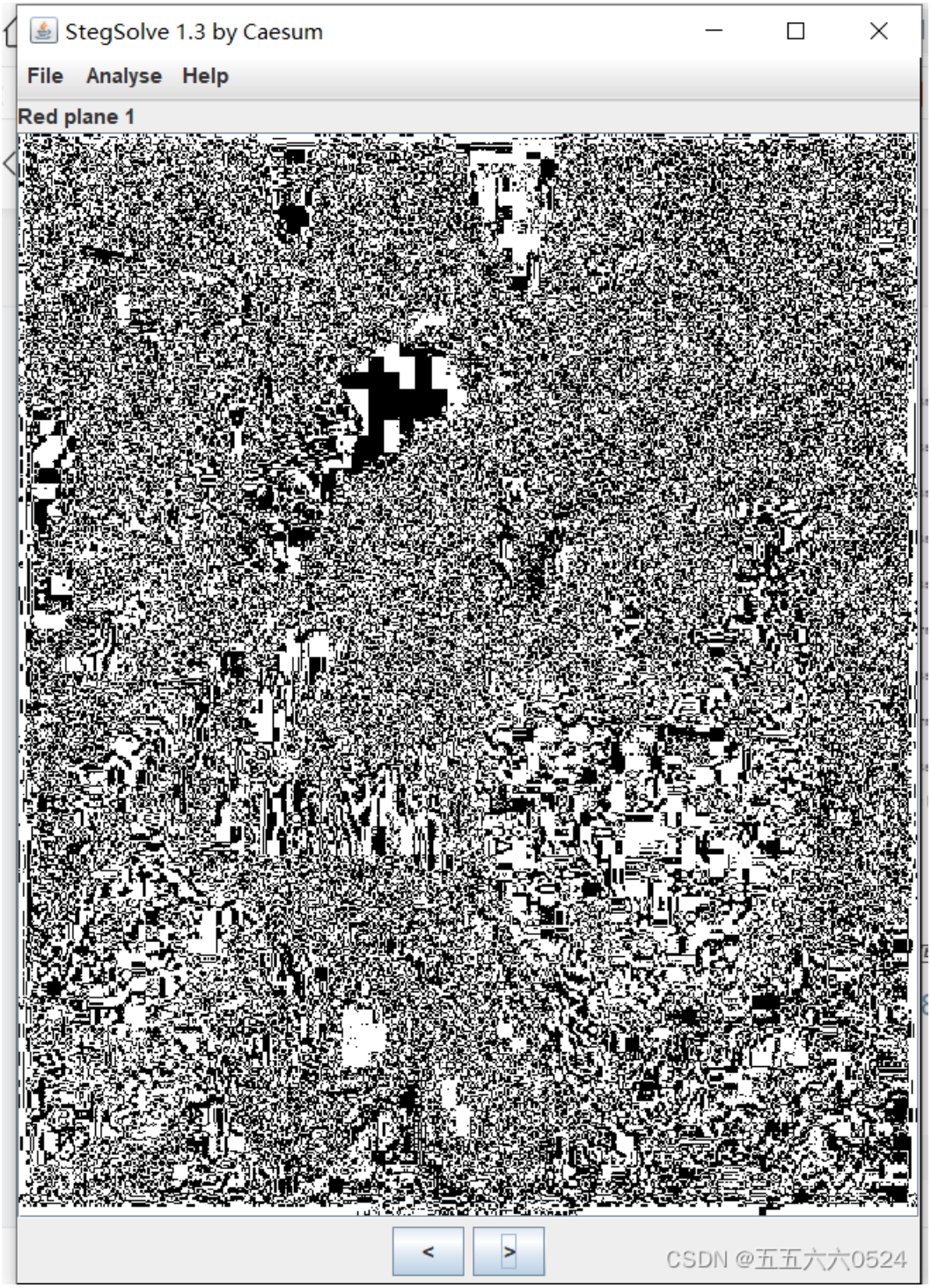


0	-1.601	In cryptography a substitution cipher is a method of encoding by which units of plaintext are replaced with ciphertext If you know the way of encoding u will get the flag which is classical-cipher_is_not_security_hs
1	-3.413	In cmyrtoemvrhy v substitution cirham is v pathol of ancoline by which units of rgvintadt vma margvcal with cirhamtadt If you know tha wvy of ancoline u wigg eat tha fgve which is cgvssicvg-cirham_is_not_sacumity_hs
2	-3.540	In dhyvtoghjvey j substitution diveah is j pateor of andoring by weide units of vljintact jha havljdar wite diveahtact If you know tea wjy of andoring u will gat tea fljg weide is dljssidjl-diveah_is_not_saduhity_es
3	-3.595	In mlyktogljkay j substitution mikael is j detaoz of enmozing by raima units of kcjintext jle lekcjmez rita mikaeltext If you vnor tae rjy of enmozing u ricc get tae fcjg raima is mcjssimjc-mikael_is_not_semulity_as
4	-3.703	In lhyktoghckey c substitution likeah is c pateor of anloring by veile units of kmcintadt cha hakmclar vite likeahtadt If you wnov tea vcy of anloring u vimm gat tea fmcg veile is lmcssilcm-likeah_is_not_saluhity_es
5	-3.736	In ckytrogkatey a svmsrirvrion citeuk is a pureod of uncoding my weice vnirs of tlainruhr aku kutlacud wire citeukruhr If yov znow reu way of uncoding v will gur reu flag weice is classical-citeuk_is_nor_sucvkiry_es
6	-3.802	In bhyrtogheray e scmstitction birauh is e putaox of unboxing my vaiba cnits of rleintukt ehu hurlebux vita birauhtukt If yoc znov tau vey of unboxing c vill gut tau fleg vaiba is blessibel-birauh_is_not_subchity_as
7	-3.815	In dmycrogmacey a svbsrirvrion diceum is a tureop of undoping by weide vnirs of clainruhr amu mucladup wire diceumruhr If yov know reu way of undoping v will gur reu flag weide is dlassidal-diceum_is_nor_sudvmiry_es
8	-3.847	An krzytovruviz u sgcstatgtaon kavier as u petiod of enkodanv cz wiaki gnats of vluantext ure revluked wati kaviertext Af zog inow tie wuz of enkodanv g wall vet tie

Thanks for using quipqiup.com! The code and website are (C) 2014-2020 by Edwin Olson, ebolson@umich.edu. Quotes were compiled by James F Thompson.

55、low

binwalk、stegslove均无果，找wp说是低位隐写，既然是LSB隐写，不懂为什么stegslove不行，最后找到了xctf攻防世界 MISC高手进阶区 low_l8947943的博客-CSDN博客



确实有点像下面有一张二维码，被盖住了

```
# lsb隐写
import PIL.Image as Image
img = Image.open('low.bmp')
img_tmp = img.copy()
pix = img_tmp.load()
width,height = img_tmp.size
for w in range(width):
   for h in range(height):
      if pix[w,h]&1 == 0:
         pix[w,h] = 0
      else:
         pix[w,h] = 255
img_tmp.show()
```

安装不了PIL，直接安装pillow也行，pip install pillow

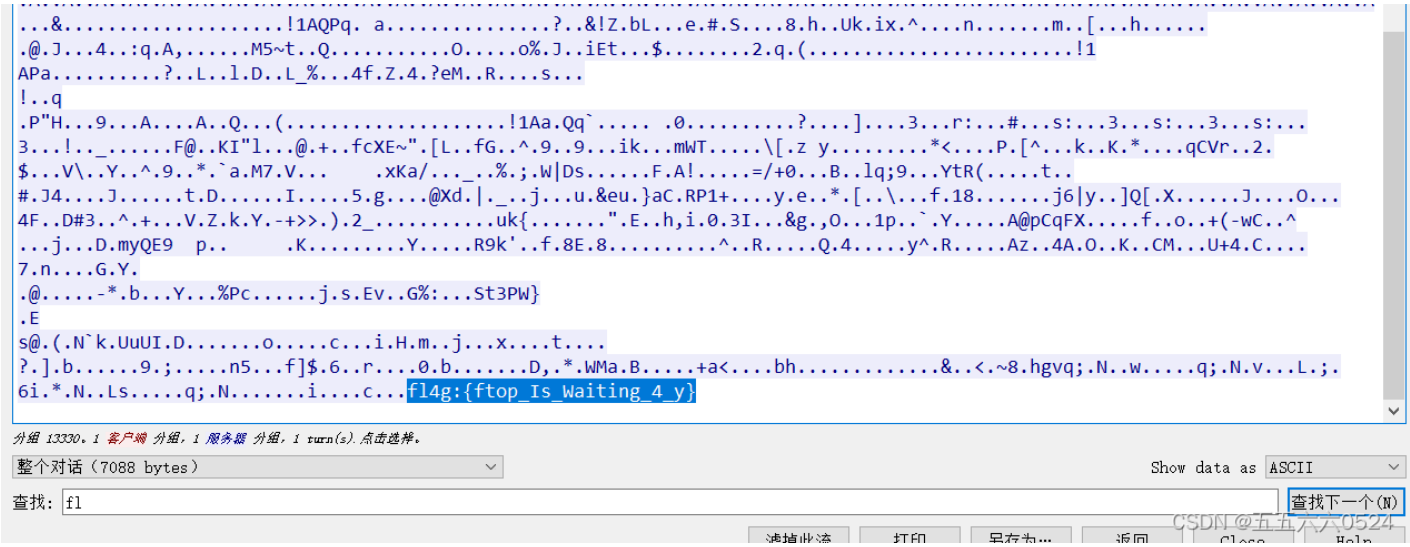
> Pillow是PIL的一个派生分支，但如今已经发展成为比PIL本身更具活力的图像处理库。pillow可以说已经取代了PIL，将其封装成python的库（pip即可安装），且支持python2和python3，目前最新版本是3.0.0。

在这里我被卡了，总显示“Import "PIL" could not be resolved from sourcePylancereportMissingModuleSource”，下载好的库在vs code上用不了，最后发现版本不对，vs上是3.9，库下载到3.7上了

扫一下，直接出flag{139711e8e9ed545e}

56、心仪的公司

这个搜flag搜不到，只能搜fl，用strings命令搜也行，在wireshark里搜也行，fl4g:{ftop_Is_Waiting_4_y}



57、misc1

> d4e8e1f4a0f7e1f3a0e6e1f3f4a1a0d4e8e5a0e6ece1e7a0e9f3baa0c4c4c3d4c6fbb9e1e6b3e3b9e4b3b7b7e2b6b1e4b2

根据wp【愚公系列】2021年11月 攻防世界-进阶题-MISC-050(misc1)_时光隧道-CSDN博客，每两个分组十六进制，转成十进制后-128(偏移量为128)，再转成ascii码得到flag，DDCTF{9af3c9d377b61d269b11337f330c935f}

```
import re
s = 'd4e8e1f4a0f7e1f3a0e6e1f3f4a1a0d4e8e5a0e6ece1e7a0e9f3baa0c4c4c3d4c6fbb9e1e6b3e3b9e4b3b7b7e2b6b1e4b2b6b9
num = re.findall('\w{2}' ,s)
flag = ''
for i in num:
   ch = chr(int(i,16)-128)
   flag += ch
print(flag)
```

关于偏移量这个玩意是什么，以及怎么搞这事，得研究研究

58、Miscellaneous-300

得到一个压缩包，需要密码，winhex看了下，不是伪加密，爆破发现密码和文件名一模一样，解压得到下一个加密文件，密码依旧和文件名相同，连续好多个，俄罗斯套娃一样，找了个脚本

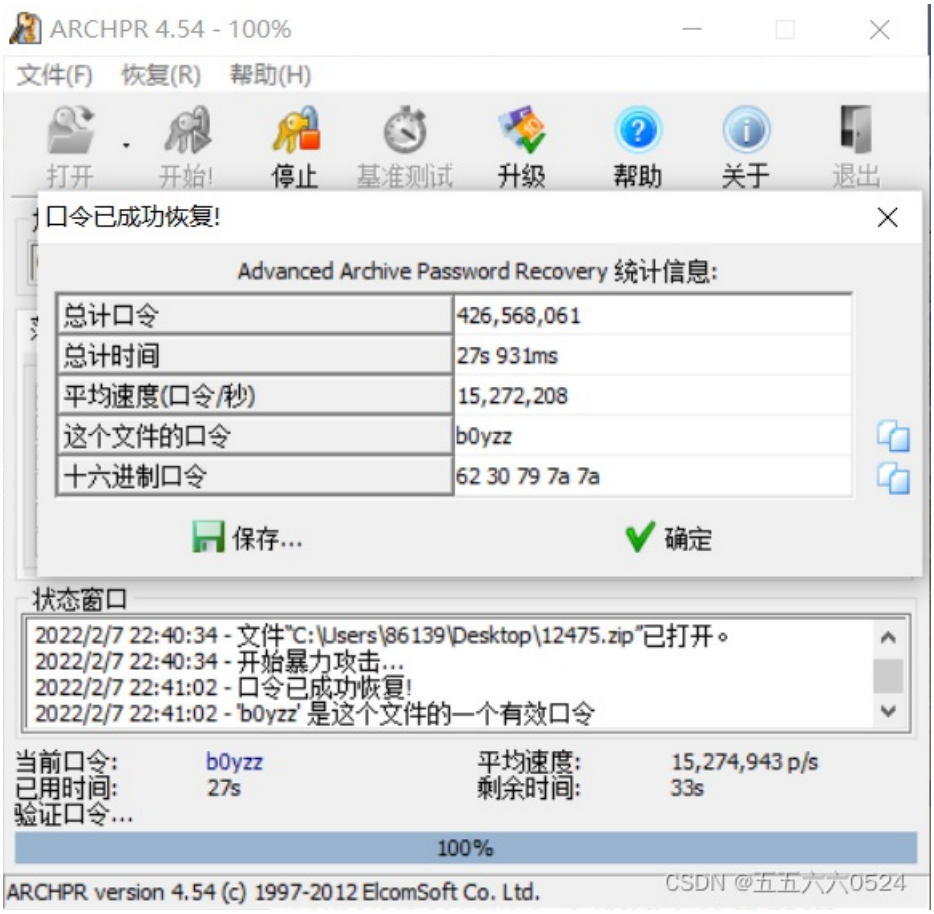
攻防世界-MISC-进阶-Miscellaneous-300_岁月仓行的博客-CSDN博客

```
import zipfile
import re
zipname = "C:\\Users\\86139\\Desktop\\tmp\\"+"47096.zip"
while True:
    if zipname != "C:\\Users\\86139\\Desktop\\tmp\\73168.zip":
        ts1 = zipfile.ZipFile(zipname)
        res = re.search('[0-9]*',ts1.namelist()[0])
        print(res.group())
        passwd = res.group()
        ts1.extractall("C:\\Users\\86139\\Desktop\\tmp\\",pwd=passwd.encode('ascii'))
        zipname = "C:\\Users\\86139\\Desktop\\tmp\\"+ts1.namelist()[0]
    else:
        print("find")
```

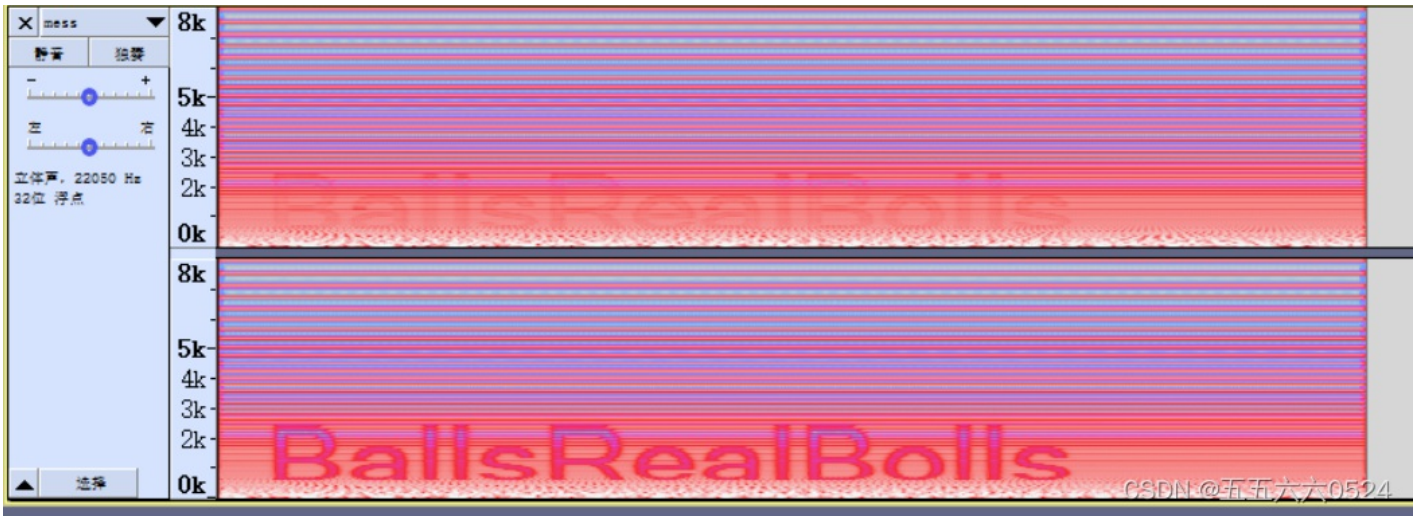
实在是套了好多层，跑了好久

```
77004
80289
12475

Traceback (most recent call last):
  File "c:/Users/86139/Desktop/crc3图片宽高破解/压缩包套娃.py", line 10, in <module>
    ts1.extractall("C:\\Users\\86139\\Desktop\\tmp\\",pwd=passwd.encode('ascii'))
  File "C:\Users\86139\AppData\Local\Programs\Python\Python37\lib\zipfile.py", line
1619, in extractall
    self._extract_member(zipinfo, path, pwd)
  File "C:\Users\86139\AppData\Local\Programs\Python\Python37\lib\zipfile.py", line
1672, in _extract_member
    with self.open(member, pwd=pwd) as source, \
  File "C:\Users\86139\AppData\Local\Programs\Python\Python37\lib\zipfile.py", line
1524, in open
    "required for extraction" % name)
RuntimeError: File <ZipInfo filename='mess.wav' compress_type=deflate external_attr=
0x20 file_size=440880 compress_size=418560> is encrypted, password required for extr
action
```


运行到12475.zip时报错了，爆破得到密码是b0yzz



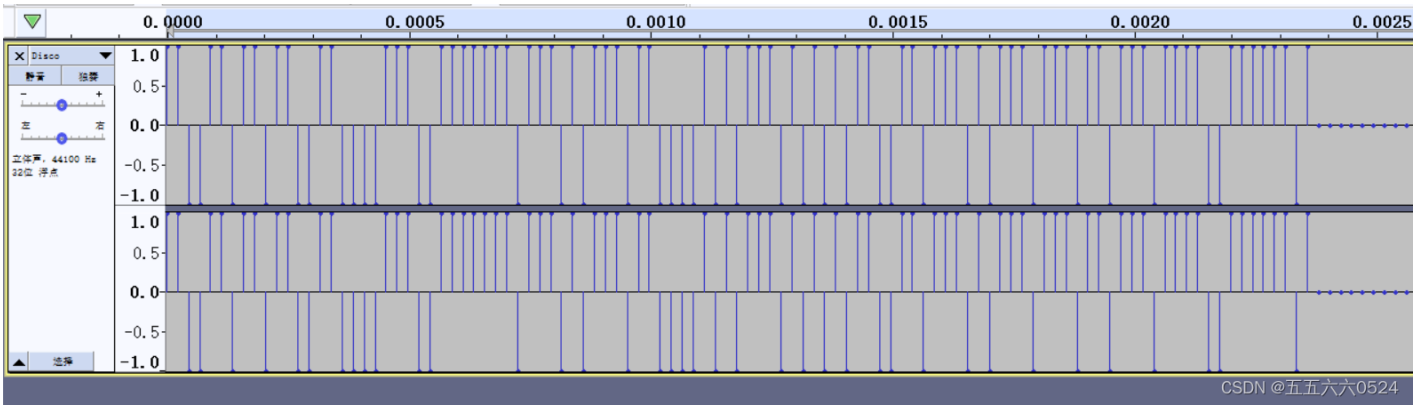
得到mess.wav，仍进AUdacity里，调成频谱图，发现flag，BallsRealBolls



59、很普通的Disco

攻防世界 Misc高手进阶区 3分题 很普通的Disco_闵行小鱼塘-CSDN博客

用Audacity打开，直接放大，把进度条拖到最前面，按上为1，下为0，能得到一堆二进制




```
11001101101100110000111001111111011101011101100001010111010101011001101110101110111011011101111001111110
```

总共105个，按7个一位，f对应的二进制0110 0110，l对应0110 1100，刚好每7个一组，每组前加0，转换成字符串就是flag，flag{W0W*funny}

```
a='11001101101100110000111001111111011101011101100001010111010101011001101110101110111011011101111001111110
flag=''
for i in range(0,len(a),7):
    flag+=chr(int('0'+a[i:i+7],2))
    print(flag)
```

```
f
fl
fla
flag
flag{
flag{W
flag{W0
flag{W0W
flag{W0W*
flag{W0W*f
flag{W0W*fu
flag{W0W*fun
flag{W0W*funn
flag{W0W*funny
flag{W0W*funny}
```


60、肥宅快乐题

攻防世界-Misc-肥宅快乐题(.swf文件查看帧)_Sea_Sand息禅-CSDN博客_攻防世界肥宅快乐题

swf文件，第一次接触potplayer，气死了，potplayer还是打不开swf文件，不过拖进浏览器里倒是能打开，但是我不想打游戏通关啊，根据题目提示（注意与NPC的对话哦），找到那一帧，然后base64翻译

61、warmup

解压得到一个压缩包和一张图，把图片用winrar压缩，加压的图片和原来的压缩包里的图片CRC32值一样，明文攻击

乌鱼子，我是真的破不开放弃，然后就是得到两张图片盲水印，flag{bWm_Are_W0nderfu1}

62、签到题

SSCTF线上选举美男大赛开始了，泰迪拿着他的密码去解密了，提交花括号内内容（Z2dRQGdRMWZxaDBvaHRqcHRfc3d7Z2ZoZ3MjfQ==）

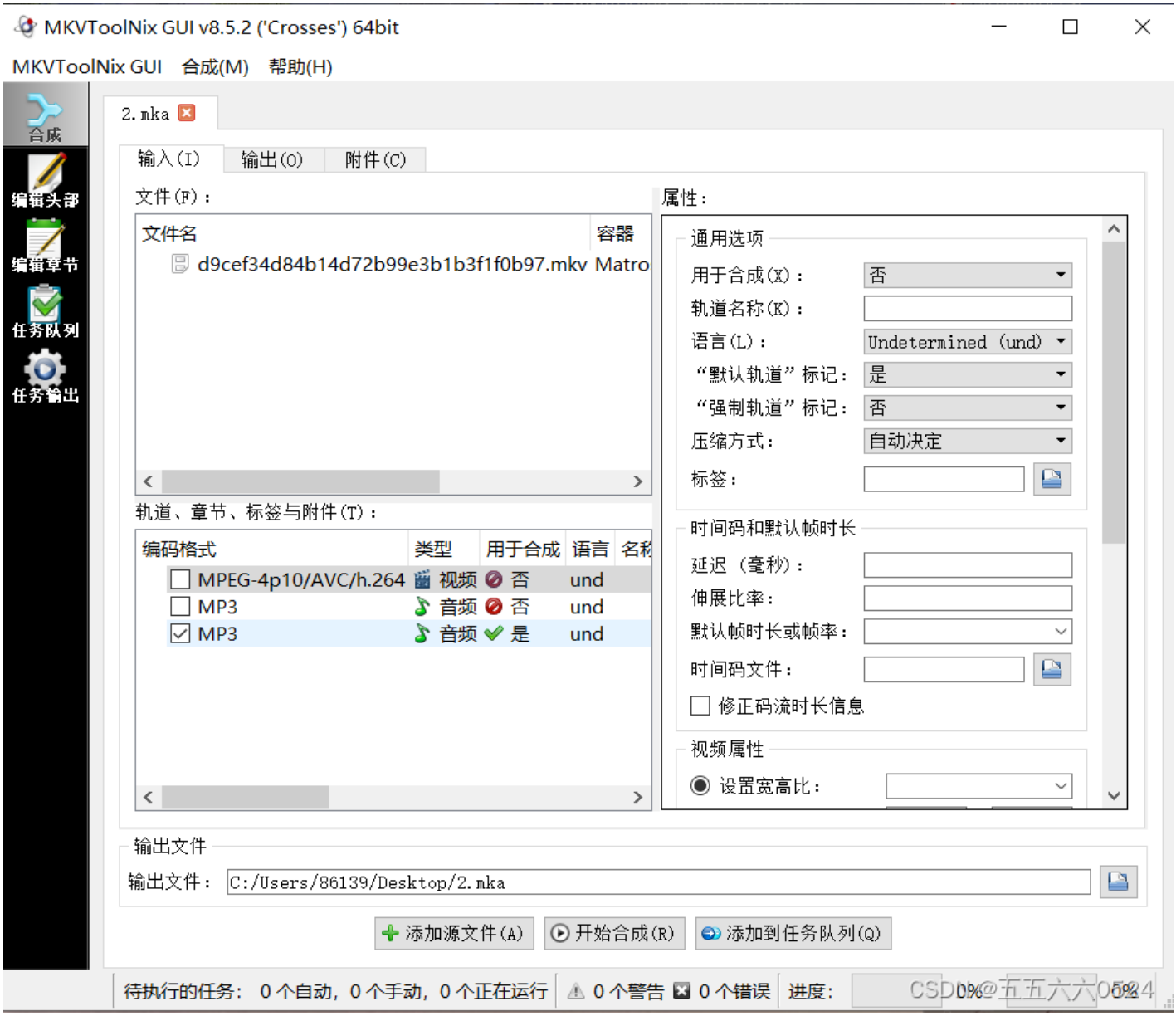
base64解密：ggQ@gQ1fqh0ohtjpt_sw{gfhgs#}

栅栏解密（位移7）：ggqht{ggQht_gsQ10jsf#@fopwh}

凯撒密码（位移14）：ssctf{ssCtf_seC10ver#@rabit}

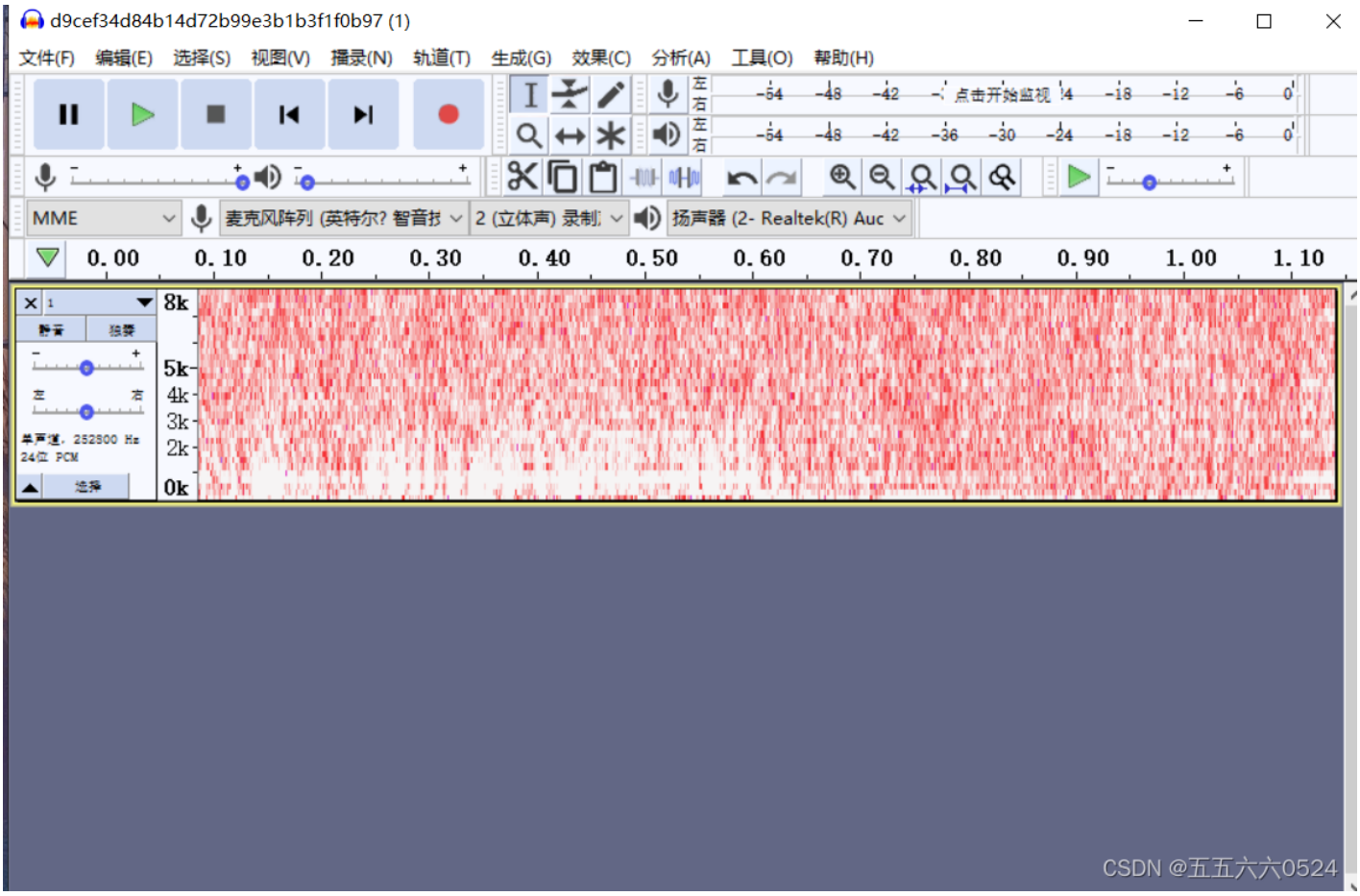
63、funny_video

播放视频发现视频有两个音轨，并且两个音轨声音相似，先将音频提取出来MKVToolnixPortable



然后用audicity打开，wp上说调到频谱图上直接出，我的最接近flag的就长这样，失败

> flag{fun_v1d30_mu51c}

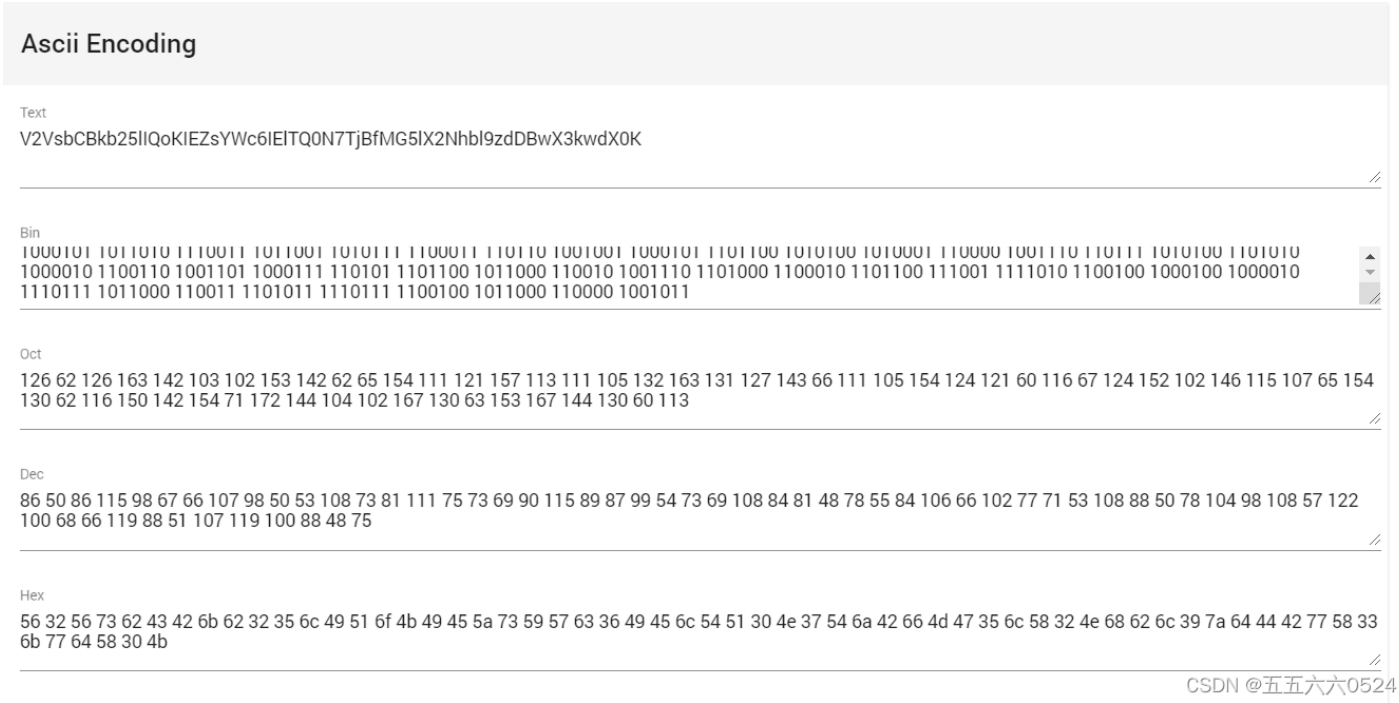


64、隐藏的信息

得到一堆数字，原来以为是九键解密，后来发现是八进制数字，用CaptfEncoder(网络安全工具套件)转换，还挺好用的

> 0126 062 0126 0163 0142 0103 0102 0153 0142 062 065 0154 0111 0121 0157 0113 0111 0105 0132 0163 0131 0127 0143 066 0111 0105 0154 0124 0121 060 0116 067 0124 0152 0102 0146 0115 0107 065 0154 0130 062 0116 0150 0142 0154 071 0172 0144 0104 0102 0167 0130 063 0153 0167 0144 0130 060 0113

八进制转ASCII，发现长得像base64



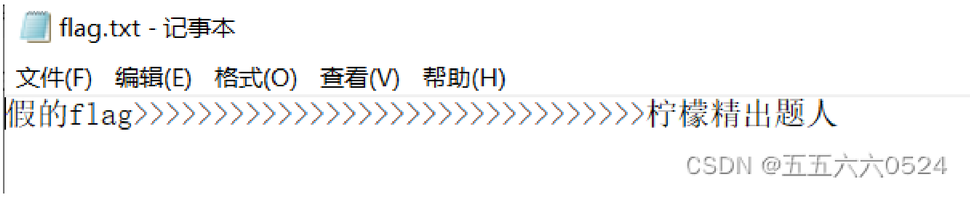
base64解密，出ISCC{N0_0ne_can_st0p_y0u}

65、miscmisc

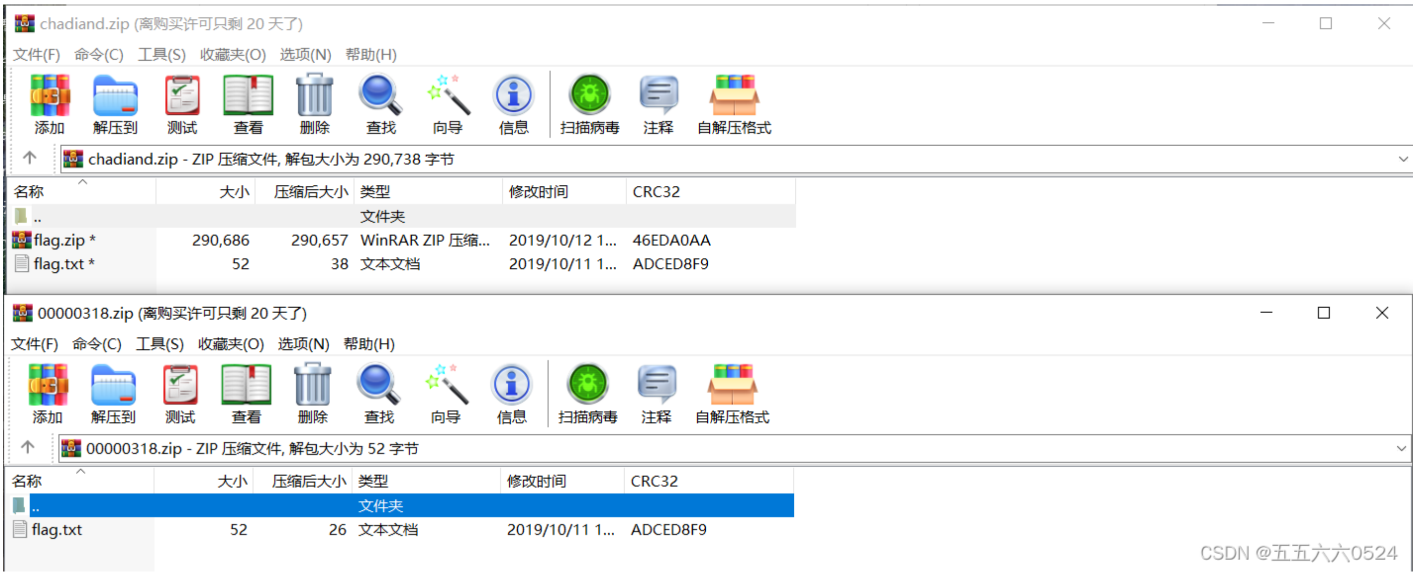
参考2019湖湘杯 misc3 之miscmisc_胖虎很忙-CSDN博客_miscmisc

得到一张表情包，foremost提取出来两个压缩包，解压之后里面都是一个压缩包和一张表情包，压缩包一个叫chadian.zip，一个叫chadiand.zip，不知道有什么联系，表情包都叫chayidian.jpg

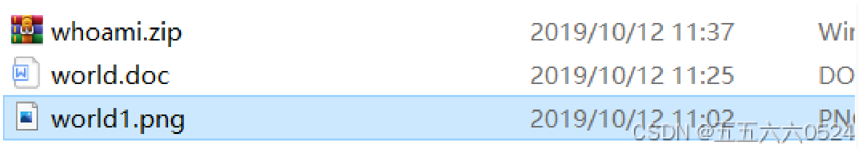
解压出来的表情包，foremost提取出来一个压缩包，发现里面是一个flag.txt



两个压缩包里都有flag.txt，发现两个的CRC32相同，应该是明文攻击，ARCHPR跑不出来，用AZPR试试，总是跑不出来（解压密码是z$^58a4w）

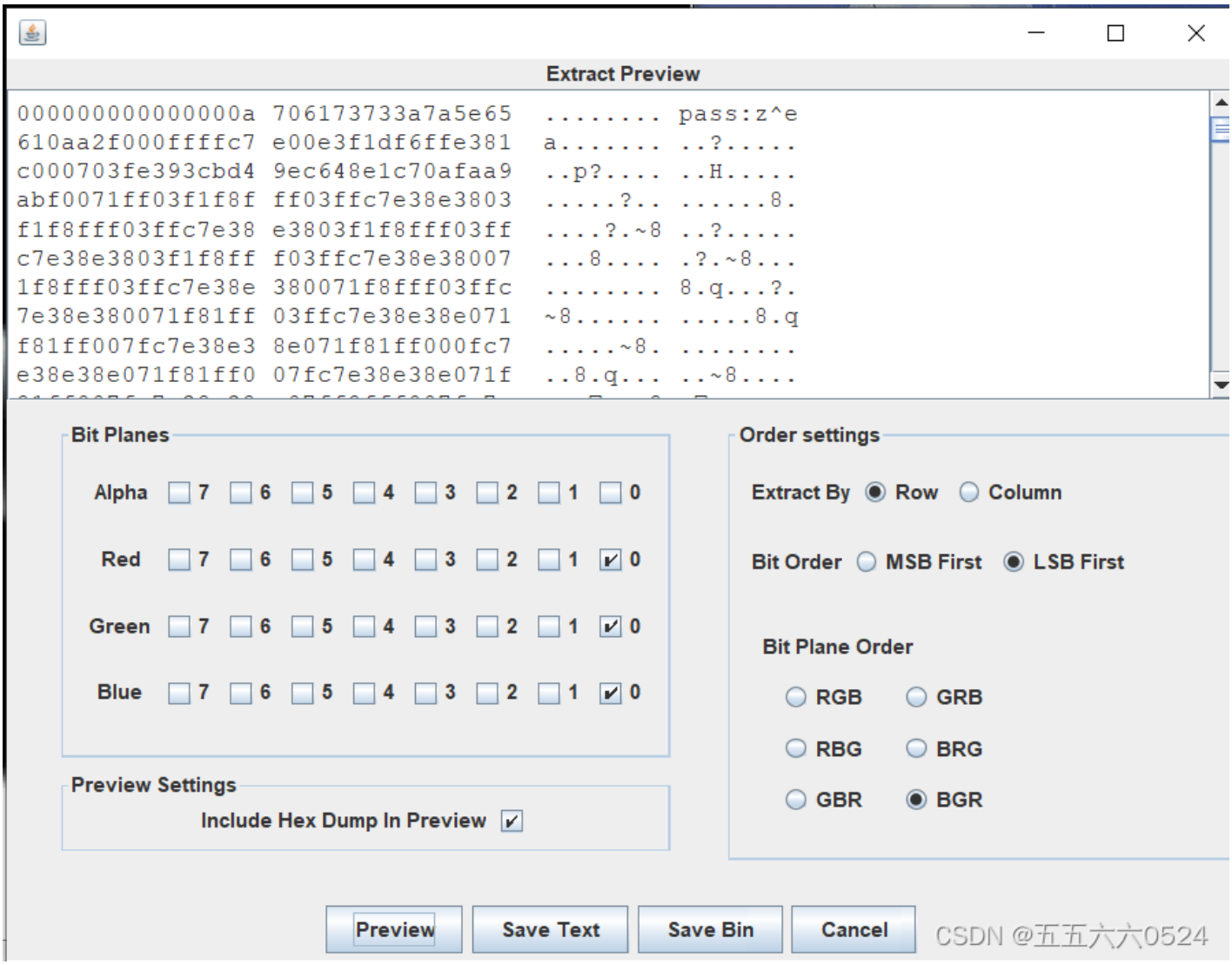


解压后得到一张图，一个文档，一个加密压缩包，猜想应该是图和文档分别得到半截密码解开zip得到flag



图片长这样

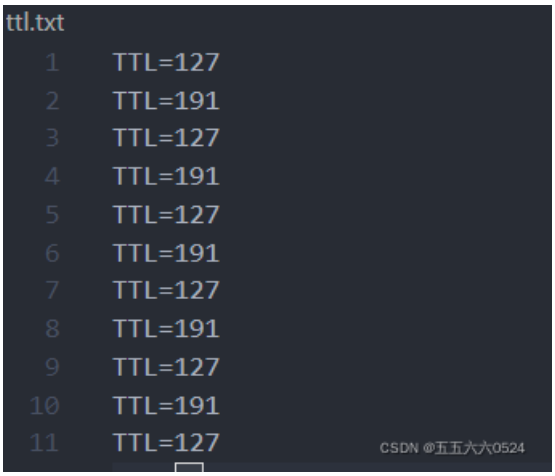
扔进stegslove，lsb低位隐写，得到pass:z^ea，前半截密码



文档里面是这样，后半截密码是每一行末尾的数字或字母，这谁想得到，4zaa3azf8

解压密码z^ea4zaa3azf8，flag{12sad7eaf46a84fe9q4fasf48e6q4f6as4f864q9e48f9q4fa6sf6f48}，这一题好狗

66、奇怪的TTL字段



127是 01111111、191是 10111111 、63是 00111111、255是 11111111，那应该就是前两位在变化

攻防世界 Misc高手进阶区 3分题 奇怪的TTL字段_闵行小鱼塘-CSDN博客_攻防世界奇怪的ttl字段

```
fp = open('ttl.txt','r')
a = fp.readlines()
p = []
for i in a:
    p.append(int(i[4:]))
s = ''
for i in p:
    if i == 63:
        a = '00'
    elif i == 127:
        a = '01'
    elif i == 191:
        a = '10'
    elif i == 255:
        a = '11'
    s += a
# print(s)

import binascii
flag = ''
for i in range(0,len(s),8):
    flag += chr(int(s[i:i+8],2))
flag = binascii.unhexlify(flag)
wp = open('res.jpg','wb')
wp.write(flag)
wp.close()
```

得到一张图片，foremost提取出来6张，用wps画图拼了老半天

扫描是key:AutomaticKey cipher:fftu{2028mb39927wn1f96o6e12z03j58002p}，自动密钥密码Practical Cryptography

Plaintext

flagabdfdeabee

keyword = AutomaticKey

v Encrypt v | ^ Decrypt ^

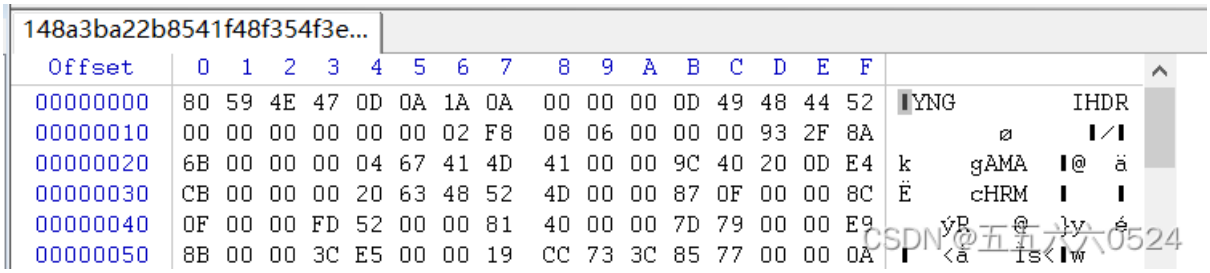
Ciphertext

fftu{2028mb39927wn1f96o6e12z03j58002p}

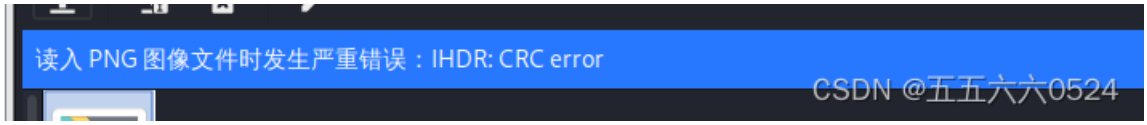
得到flagabdfdeabee，但是这个结果不对，还需要对应的把数字填充进去，flag{2028ab39927df1d96e6a12b03e58002e}

67、2-1

图片无法打开，扔进winhex发现文件头错了，改为：89 50 4E 47



改完之后还是无法打开，扔进kali里发现是CRC错误，CRC碰撞不对，找了个脚本



攻防世界 Misc高手进阶区 3分题 2-1_闵行小鱼塘-CSDN博客

```
import os
import binascii
import struct

misc = open("2-1.png","rb").read()

for i in range(1024):
    data = misc[12:16] + struct.pack('>i',i)+ misc[20:29]
    crc32 = binascii.crc32(data) & 0xffffffff
    if crc32 == 0x932f8a6b:
        print (i)
```

得到宽度应该是709， 十六进制就是02 C5

再改，出wdflag{Png_C2c_u_kn0W}

68、halo

这一题确实挺奇葩的，下面是下载下来的原txt，攻防世界里的wp跟攻防世界 Misc高手进阶区 5分题 halo_闵行小鱼塘-CSDN博客_攻防世界halo这位的wp上的原编码都不一样.....xctf攻防世界 MISC高手进阶区 halo_l8947943的博客-CSDN博客这位指出来了

> aWdxNDs0NDFSOzFpa1I1MWliT09w

过程就是原码base64解密后异或运算，flag{jdr78672Q82jhQ62jaLL3}异或运算加密/解密 - 一个工具箱 - 好用的在线工具都在这里！

69、互相伤害!!!

下载下来是一个没有后缀的文件，扔进kali里自动识别显示是流量包，foremost提取出来很多图片但是很糊，还是用的wireshark导出的（http），下面这张图用手机扫不出来，最后用的 草料二维码解码器这个网站，成功了

得到U2FsdGVkX1+VpmdLwwhbyNU80MDlK+8t61sewce2qCVztitDMKpQ4fUl5nsAZOl7bE9uL8lW/KLfbs33aC1XXw==，AES解密AES加密-AES解密-在线AES加密解密工具，密钥应该是CTF，解得668b13e0b0fc0944daf4c223b9831e49

根据题目提示，这张图大概率有点问题

扔进winhex发现里面有压缩包，压缩包里有张图片

```
00003030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00003040  00 00 FF D9 00 50 4B 03  04 14 00 01 00 08 00 5B    ÿÙ PK         [
00003050  62 7C 4A DA 75 D2 97 79  A9 01 00 BD EB 04 00 24  b|JÚuÒ—y©  ½ë  $
00003060  00 00 00 61 31 64 36 33  62 62 33 65 64 31 66 39     a1d63bb3ed1f9
00003070  64 66 38 39 62 37 32 33  37 35 66 31 65 64 37 39  df89b72375f1ed79
00003080  65 35 64 2E 6A 70 67 BF  09 DB A0 39 6B C9 51 9D  e5d.jpg¿ Û 9kÉQ
00003090  1A 36 8F B5 85 CF 7B 5D  47 47 9D B7 B3 99 07 D0   6 µ…Ï{]GG ·³™ Ð
000030A0  43 44 B3 42 02 D7 4D 27  0E 3D 54 4F FF 14 8F F4  CD³B ×M' =TOÿ  ô
000030B0  10 58 47 B5 13 B4 29 53  5B DF A5 DF 60 2E BB E7   XGµ ´)S[ß¥ß`.»ç
000030C0  8E 96 AD 47 36 DA 45 9F  A7 34 C5 F9 EC 44 DB 28  Ž–-G6ÚE Š§4ÅùìDÛ(
000030D0  A5 69 1E 7A DD 63 58 44  F1 9E EA 23 6B AA 3B 93  ¥i zÝcXDñžê#kª;“
```


foremost提取出来

大的二维码扫出来是一句话

里面的小的二维码扫出flag，flag{97d1-0867-2dc1-8926-144c-bc8a-4d4a-3758}，一定要注意题目，题目里面说的是flag里面的内容！！！

70、Keyes_secret

下载下来一堆乱七八糟的字母

```
RFVGYHNWSXCDEWSXCVWSXCVTGBNMJUY,WSXZAQWDVFRQWERTYTRFVBTGBNMJUYXSWEFTYHNNBVCXSWERF
WERTYWSXCDEWSXCFETGBNMJUTRFVBGRDXCVBTYUIOJMWSXTGBNMJUYZAQWDVFRGRDXCVBWSXCVQWERTYW
WSXCFEQWERTY(WSX.WSXCDE.,QWERTYYHNMKJTGBNMJUCVGREDQWERTYYHNMKJTGBNMJUYTGBNMJUZAQW
VBWSXCFEXSWEFTYHNWSXZAQWDVFRWSXIUYHNBVTYUIOJMMNBVCDRTGHUGRDXCVBTYUIOJMWSXTGBNMJUY
```

还以为是字频统计，但是根本没有规律

```
[('R', 285), ('T', 282), ('W', 279), ('V', 273), ('Y', 231), ('E', 221), ('B', 207), ('C', 20
), ('U', 164), ('S', 155), ('F', 150), ('M', 133), ('Q', 128), ('H', 99), ('J', 94), ('I', 77
', 2), (')', 2), ('-', 1), ('{', 1), ('}', 1), ('a', 0), ('b', 0), ('c', 0), ('d', 0), ('e',
 0), ('k', 0), ('l', 0), ('m', 0), ('n', 0), ('o', 0), ('p', 0), ('q', 0), ('r', 0), ('s', 0
), ('y', 0), ('z', 0), ('L', 0), ('P', 0), ('1', 0), ('2', 0), ('3', 0), ('4', 0), ('5', 0),
 ('!', 0), ('@', 0), ('#', 0), ('$', 0), ('%', 0), ('^', 0), ('&', 0), ('*', 0), ('_', 0), (
(']', 0)]
RTWVYEBCXDGNUSFMQHJIAOZK()-{}abcdefghijklmnopqrstuvwxyzLP1234567890!@#$%^&*_+-[]
```


自动换行后，很容易发现两个大括号之间的存在，前面的几个构成了IS，是键盘密码跑不了，往前推几个就是FLAG，FLAG{ISCC-KEYBOARD-CIPHER}

keyes.txt - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

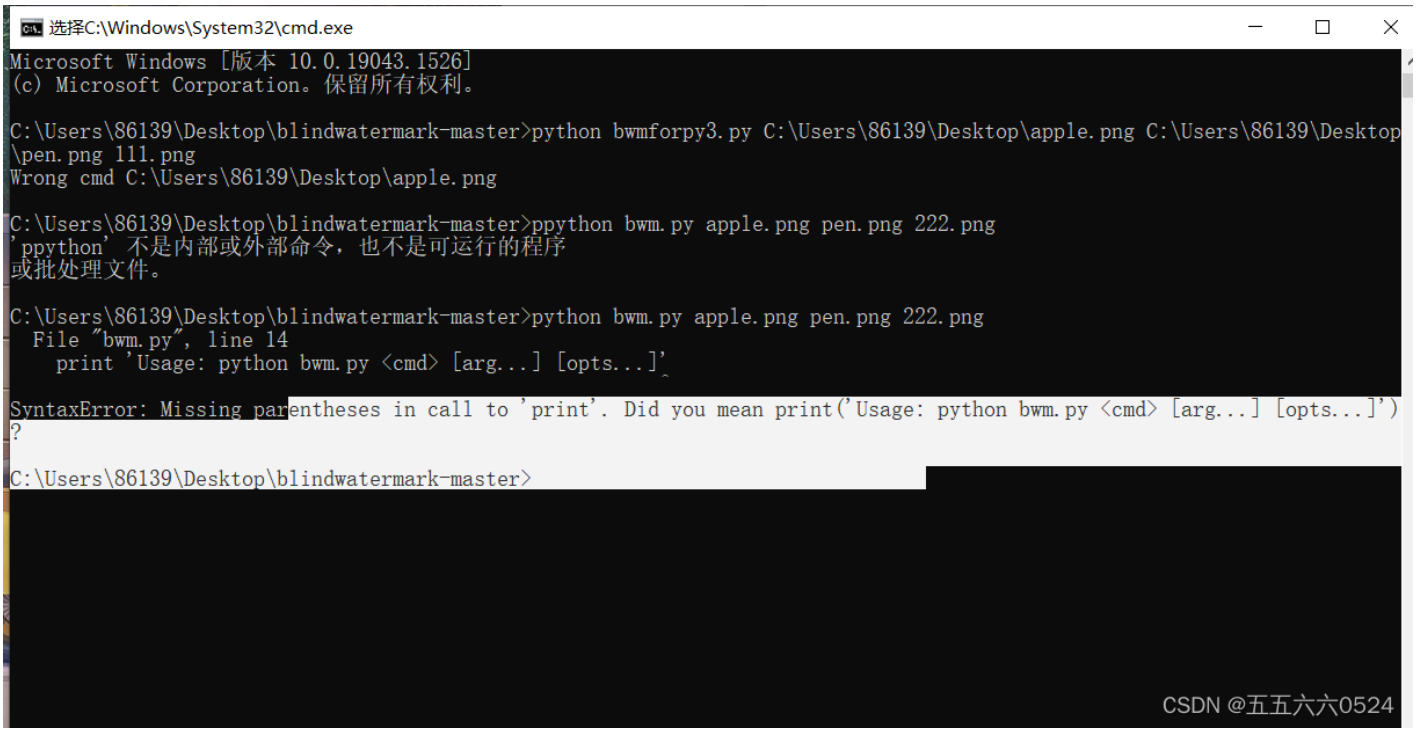
RFVGYHNWSXCDEWSXCVWSXCVTGBNMJUY,WSXZAQWDVFRQWERTYTRFVBTGBNMJUYXSWEFTYHNNBVCXSWERFTGBNMJUTYUIOJMWSXCDEMNBVCDRTGHUQWERTYIUYHNBVWSXCDETRFVBTGBNMJUMNBVCDRT
VCXSWERFTYUIOJMTGBNMJUMNBVCDRTGHUWSXCDEQWERTYTYUIOJMRFVGYHNWSXCDEQWERTYTRFVGWSXCVGRDXCVBCVGREDQWERTY
(TRFVBTYUIOJMTRFVG),QWERTYGRDXCVBQWERTYTYUIOJMEFVTNBVCXSWERFWSXCDEQWERTYTGBNMJUYTRFVGQWERTYTRFVBMNBVCDRTGHUEFVTNBVCXSWERFTYUIOJMTGBNMJUYIUYHNBVNBVCXSWE
IUYHNBVQWERTYGRDXCVBQWERTYTRFVBTGBNMJUYXSWEFTYHNNBVCXSWERFTGBNMJUTYUIOJMWSXCDEMNBVCDRTGHUQWERTYIUYHNBVWSXCDETRFVBTGBNMJUMNBVCDRTGHUWSXTYUIOJMEFVTQWERTY
TYUIOJMWSXTYUIOJMWSXTGBNMJUYZAQWDVFR.QWERTYTRFVBTYUIOJMTRFVGQWERTYTRFVBTGBNMJUYZAQWDVFRTYUIOJMWSXCDEIUYHNBVTYUIOJMIUYHNBVQWERTYGRDXCVBMNBVCDRTGHUWSXCDE
VWSXCVEFVTQWERTYWSXCFEWSXCDEIUYHNBVWSXCVGREDZAQWDVFRWSXCDEWSXCFEQWERTYTYUIOJMTGBNMJUYQWERTYIUYHNBVWSXCDEMNBVCDRTGHUEFVGYWSXCDEQWERTYGRDXCVBIUYHNBVQWERT
MJUTRFVBGRDXCVBTYUIOJMWSXTGBNMJUYZAQWDVFRGRDXCVBWSXCVQWERTYWSXCDERGNYGCWSXCDEMNBVCDRTGHUTRFVBWSXIUYHNBVWSXCDEQWERTYTYUIOJMTGBNMJUYQWERTYCVGREDWSXEFVGYW
UTYUIOJMWSXTRFVBWSXNBVCXSWERFGRDXCVBZAQWDVFRTYUIOJMIUYHNBVQWERTYWSXCDERGNYGCNBVCXSWERFWSXCDEMNBVCDRTGHUWSXWSXCDEZAQWDVFRTRFVBWSXCDEQWERTYWSXZAQWDVFRQWE
UWSXZAQWDVFRCVGREDQWERTYGRDXCVBQWERTYXSWEFTYHNGRDXCVBTRFVBRFVGYHNWSXZAQWDVFRWSXCDE,QWERTYGRDXCVBIUYHNBVQWERTYEFVGYWDCFTWSXCDEWSXCVWSXCVQWERTYGRDXCVBIUY
GBNMJUTRFVBTYUIOJMWSXZAQWDVFRCVGREDQWERTYGRDXCVBZAQWDVFRWSXCFEQWERTYMNBVCDRTGHUWSXCDEGRDXCVBTRFVBTYUIOJMWSXZAQWDVFRCVGREDQWERTYTYUIOJMTGBNMJUYQWERTYTYU
MNBVCDRTGHUTYUIOJMQWERTYTGBNMJUYTRFVGQWERTYGRDXCVBTYUIOJMTYUIOJMGRDXCVBTRFVBQAZSCEIUYHNBVQWERTYTRFVGTGBNMJUYTGBNMJUZAQWDVFRWSXCFEQWERTYWSXZAQWDVFRQWERT
WSXCDEGRDXCVBWSXCVQWERTYEFVGYWDCFTTGBNMJUYMNBVCDRTGHUWSXCVWSXCFEQWERTY
(WSX.WSXCDE.,QWERTYYHNMKJTGBNMJUCVGREDQWERTYYHNMKJTGBNMJUYTGBNMJUZAQWDVFRTYUIOJMEFVTQWERTYNBVCXSWERFMNBVCDRTGHUTGBNMJUYCVGREDMNBVCDRTGHUGRDXCVBXSWEFTYH
ERFMNBVCDRTGHUTGBNMJUYTRFVGWSXCDEIUYHNBVIUYHNBVWSXTGBNMJUYZAQWDVFRGRDXCVBWSXCVQWERTYIUYHNBVWSXCDETYUIOJMTYUIOJMWSXZAQWDVFRCVGREDIUYHNBV).QWERTYRFVGYHNV
VCDRTGHUWSXCDEQWERTYEFVTTGBNMJUYTGBNMJUMNBVCDRTGHUQWERTYTRFVGWSXCVGRDXCVBCVGRED
{WSXIUYHNBVTRFVBTRFVBQWERTYQAZSCEWSXCDEEFVTYHNMKJTGBNMJUYGRDXCVBMNBVCDRTGHUWSXCFEQWERTYTRFVBWSXNBVCXSWERFRFVGYHNWSXCDEMNBVCDRTGHU}
QWERTYMNBVCDRTGHUWSXCDEEFVGYWSXCDEMNBVCDRTGHUIUYHNBVWSXCDE-
WSXCDEZAQWDVFRCVGREDWSXZAQWDVFRWSXCDEWSXCDEMNBVCDRTGHUWSXZAQWDVFRCVGRED,QWERTYZAQWDVFRWSXCDETYUIOJMEFVGYWDCFTTGBNMJUYMNBVCDRTGHUQAZSCEQWERTYIUYHNBVZAQW
RTYNBVCXSWERFMNBVCDRTGHUTGBNMJUYTYUIOJMTGBNMJUYTRFVBTGBNMJUYWSXCVQWERTYGRDXCVBZAQWDVFRGRDXCVBWSXCVEFVTIUYHNBVWSXIUYHNBV,QWERTYIUYHNBVEFVTIUYHNBVTYUIOJM
FTYHNWSXZAQWDVFRWSXIUYHNBVTYUIOJMMNBVCDRTGHUGRDXCVBTYUIOJMWSXTGBNMJUYZAQWDVFR,QWERTYNBVCXSWERFMNBVCDRTGHUTGBNMJUYCVGREDMNBVCDRTGHUGRDXCVBXSWEFTYHNXSWEF
WDVFRWSXCFEQWERTYTRFVBMNBVCDRTGHUEFVTNBVCXSWERFTYUIOJMGRDXCVBZAQWDVFRGRDXCVBWSXCVEFVTIUYHNBVWSXIUYHNBVQWERTYGRDXCVBMNBVCDRTGHUWSXCDEQWERTYGRDXCVBWSXCVV
YHNBVQWERTYEFVGYWDCFTRFVGYHNWSXTRFVBRFVGYHNQWERTYRFVGYHNGRDXCVBEFVGYWSXCDEQWERTYYHNMKJWSXCDEWSXCDEZAQWDVFRQWERTYMNBVCDRTGHUWSXCDEQAZXCDEWVTGBNMJUWSXMNE
ERTYNBVCXSWERFMNBVCDRTGHUWSXTGBNMJUYMNBVCDRTGHUQWERTYTRFVBTYUIOJMTRFVGQWERTYTRFVBTGBNMJUYZAQWDVFRTYUIOJMWSXCDEIUYHNBVTYUIOJMIUYHNBVQWERTYGRDXCVBTYUIOJM
MJUYZAQWDVFR.

71、saleae

.ligicdata是逻辑分析仪数据文件，用logic软件打开，不会用，logicdata后缀打不开，遂放弃，flag{12071397-19d1-48e6-be8c-784b89a95e07}

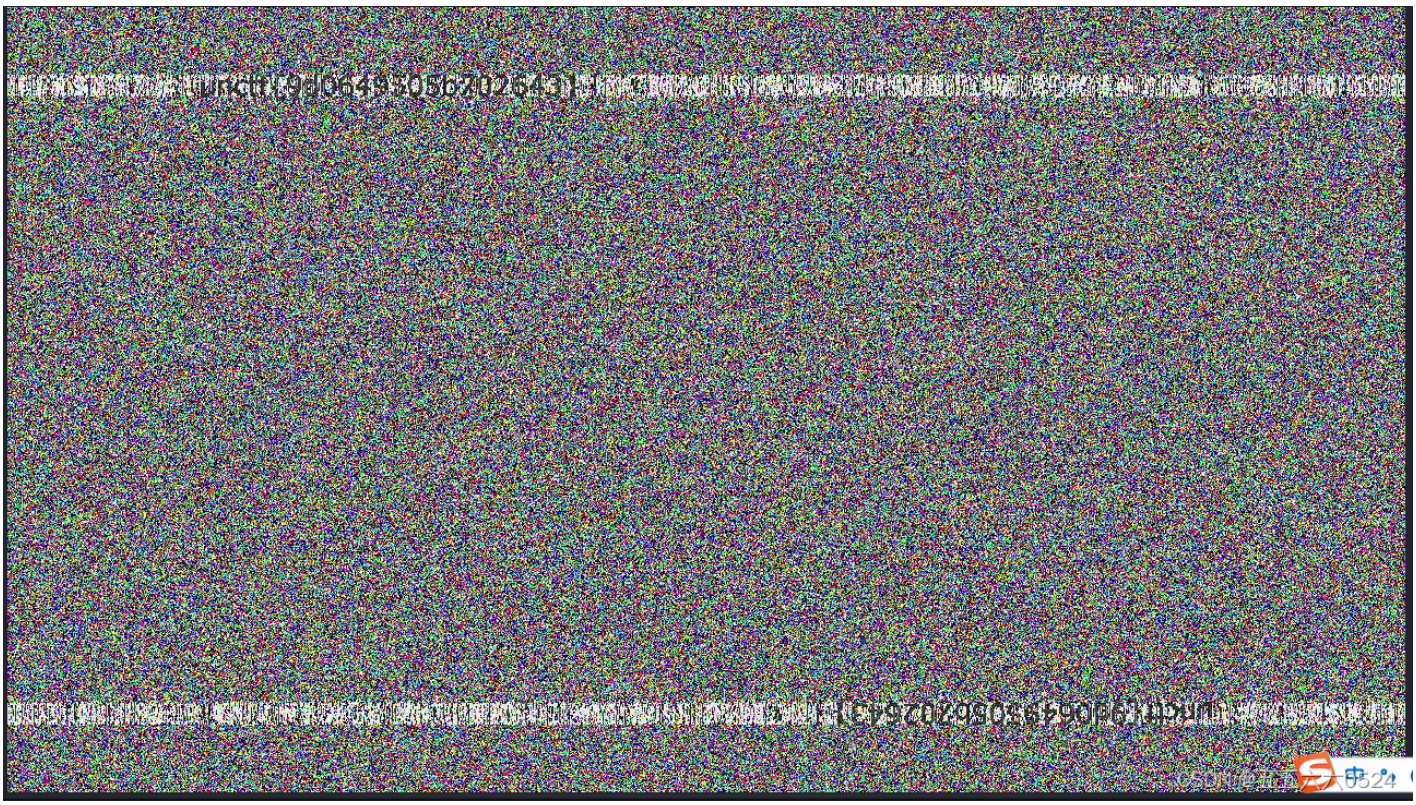
72、信号不好先挂了

apple.png扔进kali里发现里面有东西，提取出来一张一样的图，名字叫pen.png，接下来的操作是盲水印解，奈何我实在没学会



卡死在这了，不知道出了什么错

3.9号更，windows不成功，嘿嘿kali成功了，unctf{9d0649505b702643}



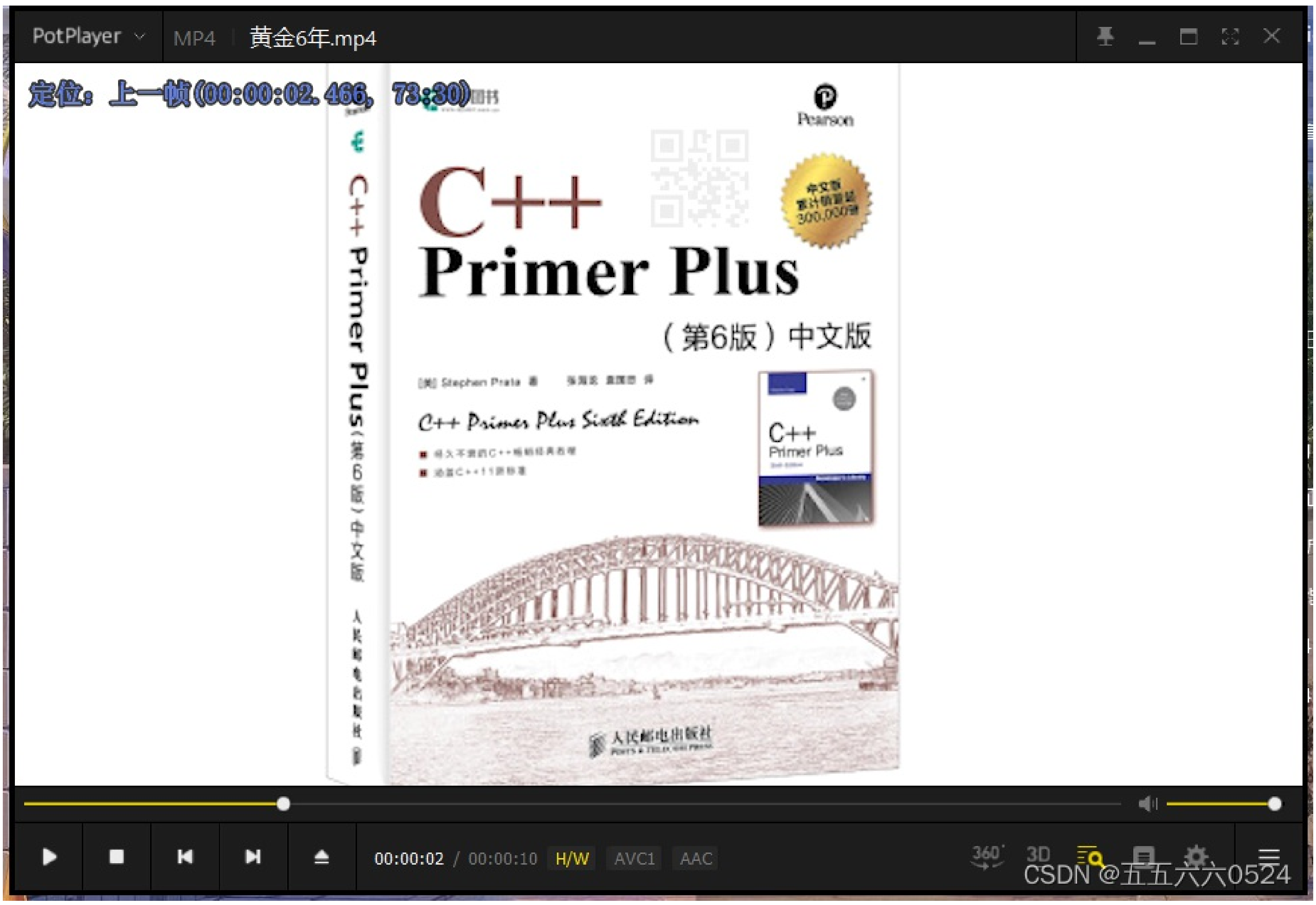
73、黄金六年

mp4文件用winhex打开，可以发现末尾有base64编码，转一下发现，是rar压缩包，里面还有flag.txt，十六进制里面有52617221（CaptfEncoder是真的很好用，就是转十六进制的内容好像不太对，得到的rar显示文件格式损坏）

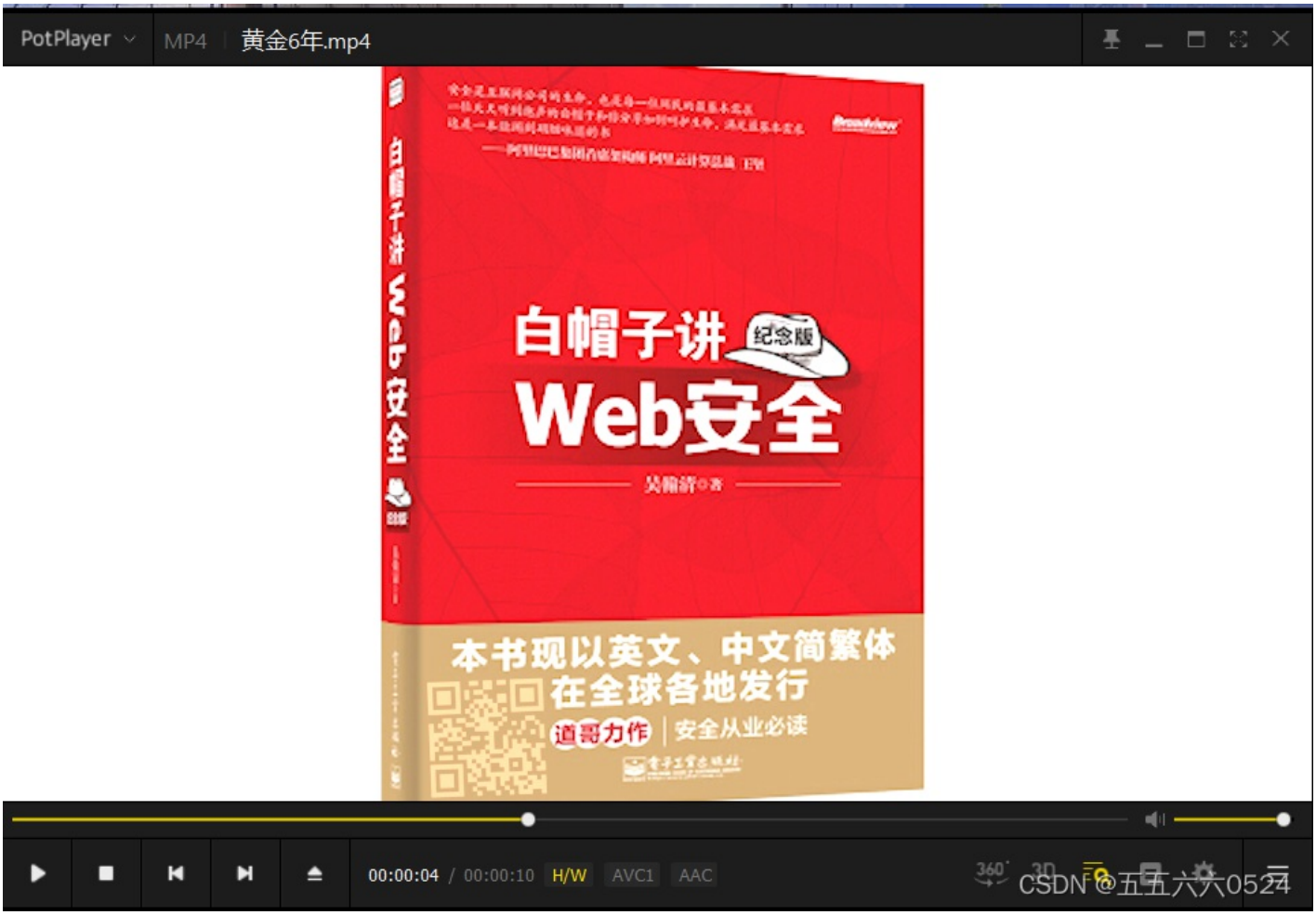
[RoarCTF2019]黄金6年 - 春告鳥 - 博客园根据这位的学会了base64转十六进制，得到的压缩包需要解密

```
import base64
a="UmFyIRoHAQAzkrXlCgEFBgAFAQGAgADh7ek5VQIDPLAABKEAIEvsUpGAAwAIZmxhZy50eHQwAQADDx43HyOdLMGWfCE9WEsBZprAJQoB
b=base64.b64decode(a)
c=open('1.rar','wb')
c.write(b)
c.close()
```

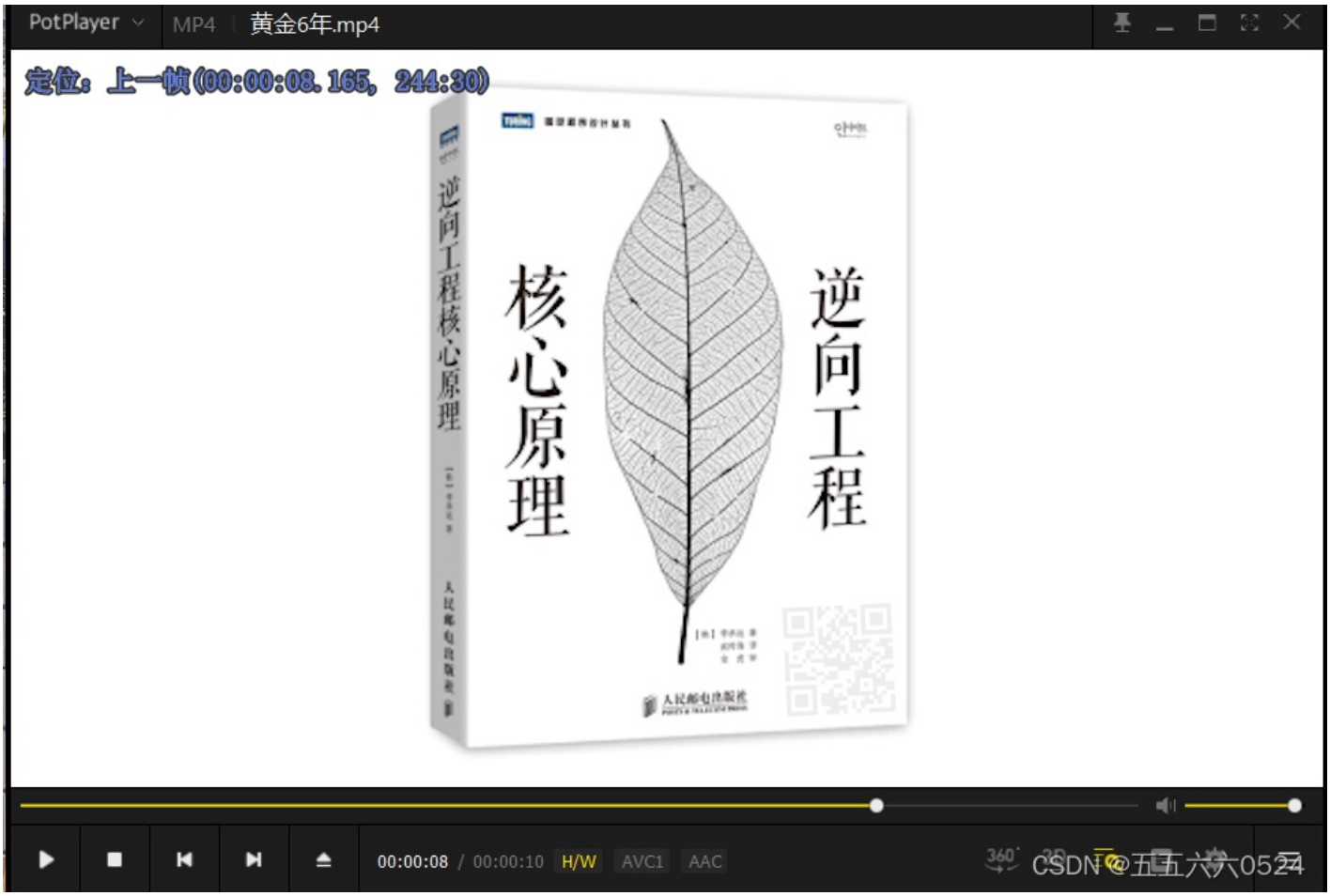
下面的这些二维码是真真不好找，麻烦死了，能猜到密码在视频里面，但是真不好找啊



key1:i



key2:want



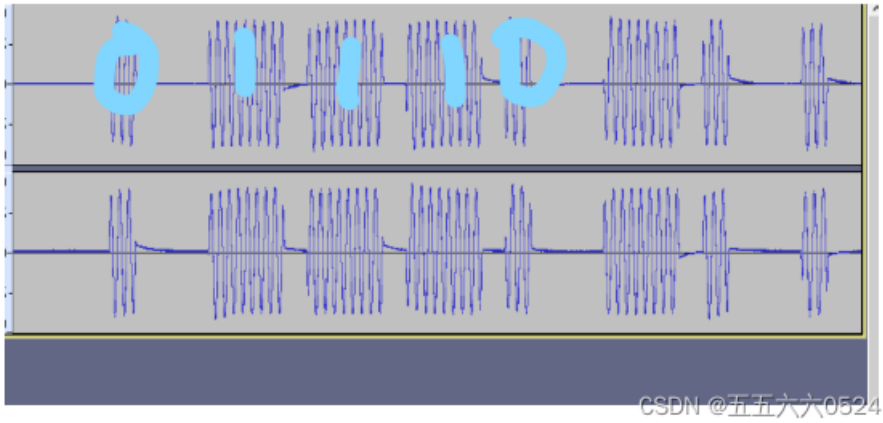
key3:play

key4:ctf

密码iwantplayctf，解得flagroarctf{CTF-from-RuMen-to-RuYuan}

74、打开电动车

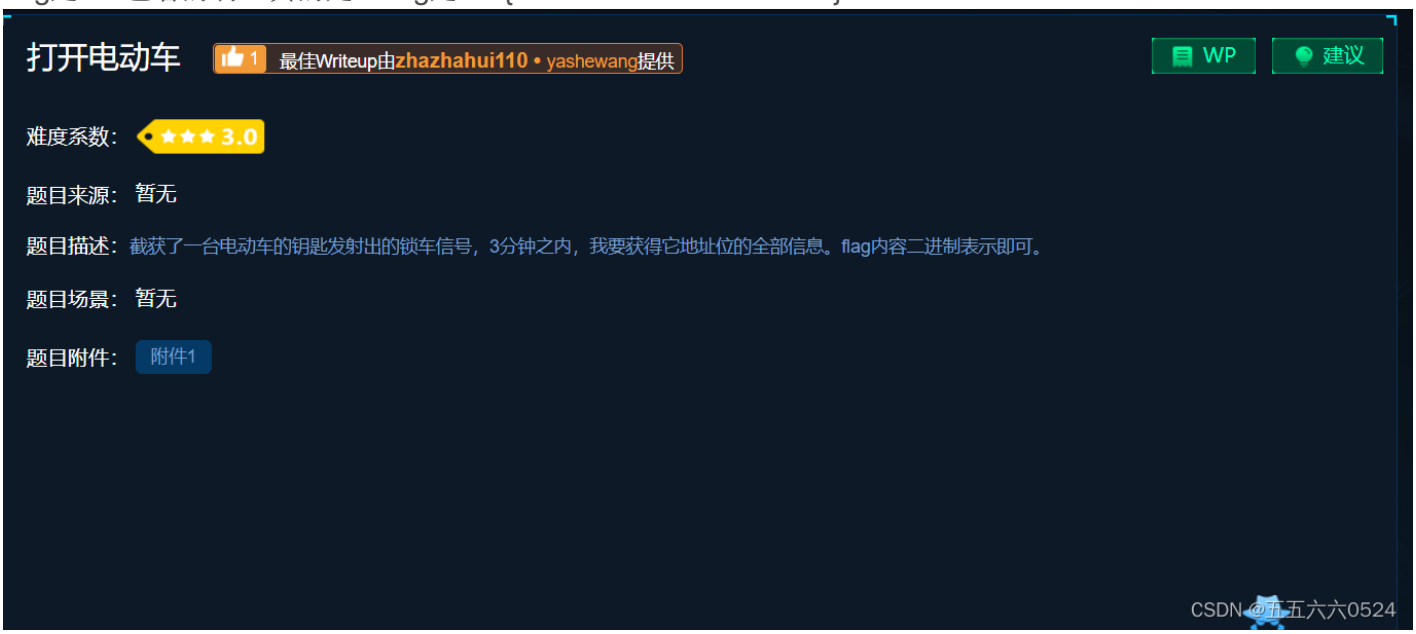
用Audacity打开，短的是0，长的是1



得到

```
0   01110100101010100110 00100011101001010101001100010
```

攻防世界 Misc高手进阶区 4分题 打开电动车_闵行小鱼塘-CSDN博客 根据他的博客，钥匙信号(PT224X) = 同步引导码(8bit) + 地址位(20bit) + 数据位(4bit) + 停止码(1bit)，中间的20位就是答案，但是题目长这样，也没提示flag是sctf包着的啊，真的是，flag是sctf{01110100101010100110}

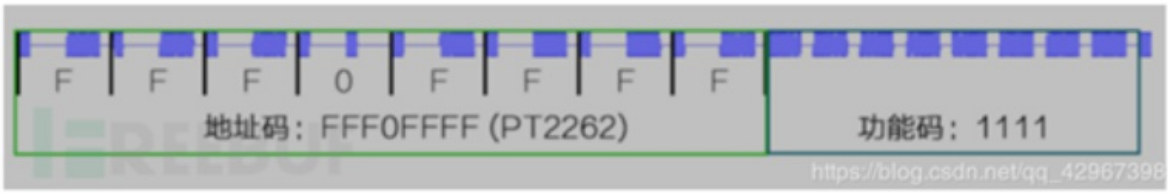


附官方wp

有题目提示说地址位是flag

查询得到：信号分别是同步引导码，地址位和数据位，最后一个就是停止码

又经过查询信息得到：



一个是PT2242的，前面4bit表示同步码，中间的20bit表示地址码，后面的4bit表示功能码，后面最后一个是停止码

一个是PT2262的，前面4bit表示同步码，中间的8bit表示地址码，后面的4bit表示功能码，后面最后一个是停止码

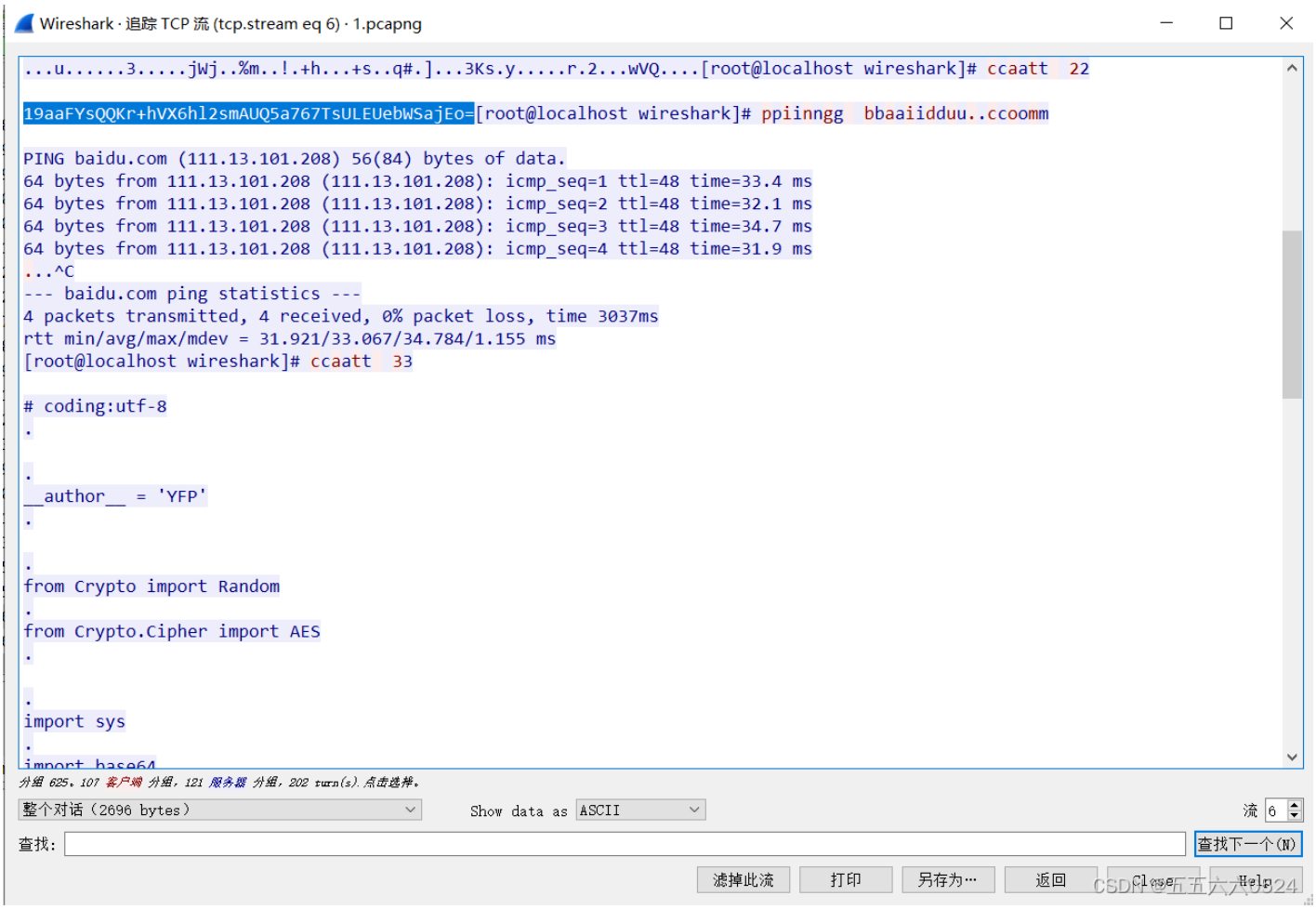
只不过表达的方式不同，这道题就是PT2242的，

所以flag就是：`sctf{01110100101010100110}`

CSDN @五五六六0524

75、3-1

下载得到一个文件，扔进kali里查看，发现是rar，修改文件名后解压，里面有一个pcapng流量包，wireshark搜索发现里面有flag.txt和flag.rar，http导出flag.rar，加密的，那解压密码应该还在流量包里，在tcp.stream eq 6里面查到base64、python脚本



在编译器里总报错，不太会修改，在kali里运行脚本：python3 +文件名

```
PS C:\Users\86139\Desktop\crc3图片宽高破解> & C:/Users/86139/AppData/Local/Programs/Python/Python37/python.exe c:/Users/86139/Desktop/crc3图片宽高破解/1.py
Traceback (most recent call last):
  File "c:/Users/86139/Desktop/crc3图片宽高破解/1.py", line 24, in <module>
    example = encrypt(str)
  File "c:/Users/86139/Desktop/crc3图片宽高破解/1.py", line 20, in encrypt
    aes = AES.new(IV, AES.MODE_CBC, IV)
  File "C:\Users\86139\AppData\Local\Programs\Python\Python37\lib\site-packages\Crypto\Cipher\AES.py", line 232, in new
    return _create_cipher(sys.modules[__name__], key, mode, *args, **kwargs)
  File "C:\Users\86139\AppData\Local\Programs\Python\Python37\lib\site-packages\Crypto\Cipher\__init__.py", line 79, in _create_cipher
    return modes[mode](factory, **kwargs)
  File "C:\Users\86139\AppData\Local\Programs\Python\Python37\lib\site-packages\Crypto\Cipher\_mode_cbc.py", line 274, in _create_cbc_cipher
    cipher_state = factory._create_base_cipher(kwargs)
  File "C:\Users\86139\AppData\Local\Programs\Python\Python37\lib\site-packages\Crypto\Cipher\AES.py", line 103, in _create_base_cipher
    result = start_operation(c_uint8_ptr(key),
  File "C:\Users\86139\AppData\Local\Programs\Python\Python37\lib\site-packages\Crypto\Util\_raw_api.py", line 138, in c_uint8_ptr
    raise TypeError("Object type %s cannot be passed to C code" % type(data))
TypeError: Object type <class 'str'> cannot be passed to C code
PS C:\Users\86139\Desktop\crc3图片宽高破解>
```


关于这个问题的解决，参考了下面两篇博客，大佬yyds

【攻防世界AD】Misc进阶区：3-1_yoyoko_chan的博客-CSDN博客

加密：python crypto AES : Object type cannot be passed to C code_chenran187906的博客-CSDN博客

```
from Crypto import Random
from Crypto.Cipher import AES
import sys

import base64

IV = 'QWERTYUIOPASDFGH'.encode('utf-8')#修改1

def decrypt(encrypted):
  aes = AES.new(IV, AES.MODE_CBC, IV)
  return aes.decrypt(encrypted)

def encrypt(message):
  length = 16
  count = len(message)
  padding = length - (count % length)
  message = message + '\0' * padding
  aes = AES.new(IV, AES.MODE_CBC, IV)
  return aes.encrypt(message.encode('utf-8'))#修改2

str = 'this is a test'
example = encrypt(str)
print(decrypt(example))
s='19aaFYsQQKr+hVX6hl2smAUQ5a767TsULEUebWSajEo='
flag=base64.b64decode(s)
print(decrypt(flag))
```

解压密码No_One_Can_Decrypt_Me，解压得到flag为WDCTF{Seclab_CTF_2017}

```
宽高破解/1.py
b'this is a test\x00\x00'
b'passwd={No One Can Decrypt Me}\x00\x00'
```

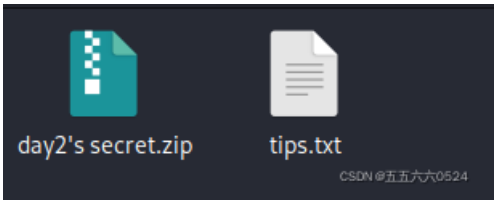

76、4-1

得到一张图

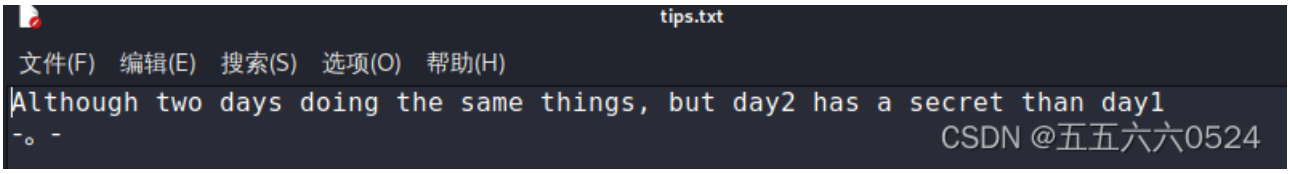
扔进kali里binwalk一下，里面有东西

```
┌──(kali㉿kali)-[~/桌面]
└─$ binwalk 1.png

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0×0             PNG image, 487 x 742, 8-bit/color RGBA, non-interlaced
41            0×29            Zlib compressed data, default compression
415520        0×65720         Zip archive data, at least v2.0 to extract, compressed size: 74,
uncompressed size: 78, name: tips.txt
415632        0×65790         Zip archive data, at least v1.0 to extract, compressed size: 6594
34, uncompressed size: 659434, name: day2's secret.zip
1075091       0×106793        End of Zip archive, footer length: 22
1075302       0×106866        End of Zip archive, footer length: 22
```


foremost提取出来一个压缩包，解压得到



txt里面提示

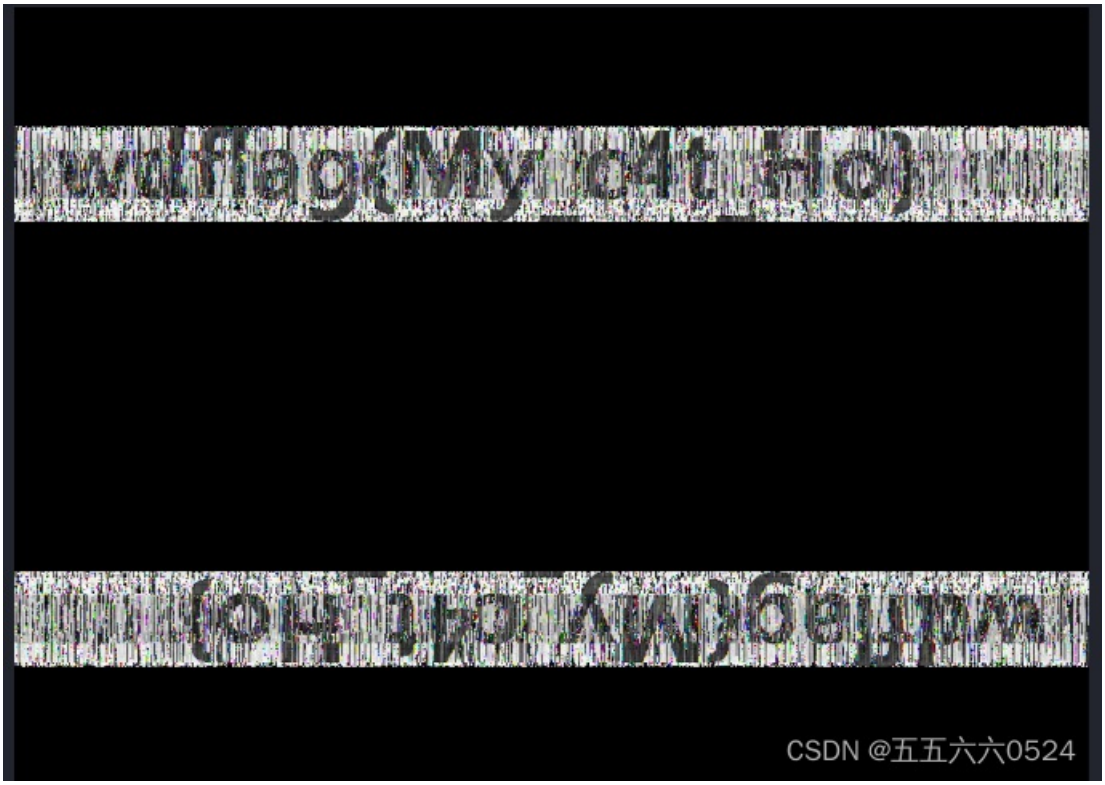


剩下的一个压缩包解压得到两张一模一样的图，猜测是盲水印

【CTF】图片隐写术 · 盲水印 - 双份浓缩馥芮白 - 博客园，感谢大佬哇，第一次成功使用盲水印

python bwmforpy3.py decode day1.png day2.png flag.png --oldseed

成功提取出来图片，得到flag：wdflag{My_c4t_Ho}



77、5-1

工具xortool

CTF-Xortool,windows上的安装与使用_半岛铁盒的博客-CSDN博客_xortool

python xortool.py C:\Users\86139\Desktop\1

```
C:\Users\86139\Desktop\xortool-for-Windows-master\xortool>python xortool.py C:\Users\86139\Desktop\1
The most probable key lengths:
   2:   12.2%
   5:   11.9%
   9:   9.8%
  13:   22.2%
  20:   6.8%
  22:   6.2%
  26:   12.8%
  30:   4.6%
  39:   7.8%
  52:   5.7%
Key-length can be 3*n
Most possible char is needed to guess the key!
```


python xortool.py C:\Users\86139\Desktop\1 -l 13 -c 00

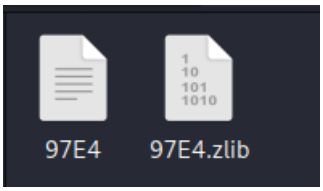
```
C:\Users\86139\Desktop\xortool-for-Windows-master\xortool>python xortool.py C:\Users\86139\Desktop\1 -l 13 -c 00
Traceback (most recent call last):
  File "xortool.py", line 398, in <module>
    main()
  File "xortool.py", line 80, in main
    key_char_used) = guess_probable_keys_for_chars(ciphertext, try_chars)
  File "xortool.py", line 264, in guess_probable_keys_for_chars
    keys = guess_keys(text, c)
  File "xortool.py", line 286, in guess_keys
    key_possible_bytes[offset].append(chr(ord(char) ^ most_char))
TypeError: ord() expected string of length 1, but int found
```


这一步真的蒙了，不知道为什么会出错

78、picture2

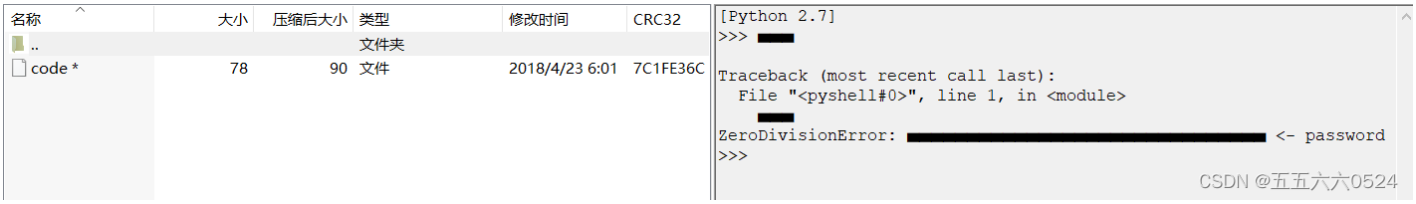
binwalk发现有隐写，但又不是zsteg隐写，-e提取出来

```
└─$ binwalk 1.png

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, JFIF standard 1.01
38884         0x97E4          Zlib compressed data, default compression


┌──(kali㉿kali)-[~/桌面]
└─$ zsteg 1.png
[!] #<ZPNG::NotSupported: Unsupported header "\xFF\xD8\xFF\xE0\x00\x
```

第一个文档里面是base64，解密后发现十六进制是压缩包格式，txt前面是KP，应该是反了

得到的压缩包加密的，有提示



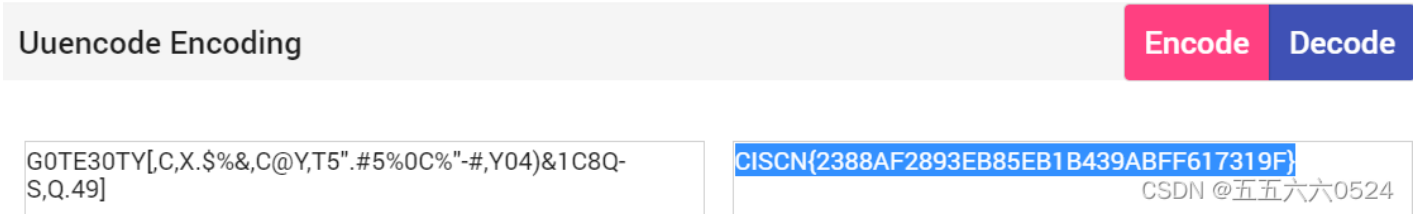
错误异常有三种，一个个试，第二种对了

ZeroDivisionError: division by zero

ZeroDivisionError: integer division or modulo by zero

ZeroDivisionError: float division by zero CSDN @五五六六0524

解压得code文件，里面是密文，解得CISCN{2388AF2893EB85EB1B439ABFF617319F}



78、test.pyc

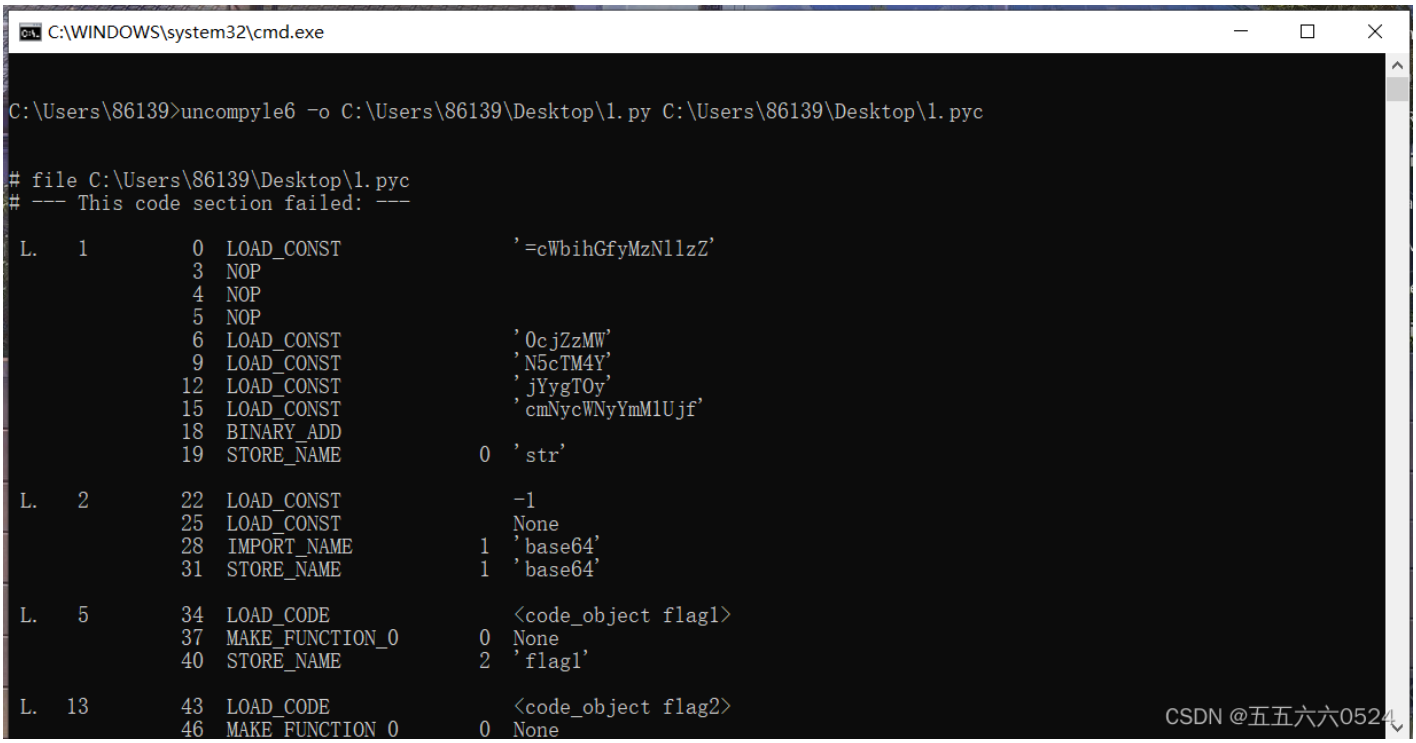
反编译不完全，问题出在flag3上

```
#!/usr/bin/env python
# visit https://tool.lu/pyc/ for more information
str = 'jYygTOy' + 'cmNycWNyYmM1Ujf'
import base64

def flag1():
    code = str[::-3]
    result = ''
    for i in code:
        ss = ord(i) - 1
        result += chr(ss)

    print(result[::-1])


def flag2():
    code = str[::-2]
    result = ''
    for i in code:
        ss = ord(i) - 1
        result += chr(ss)

    print(result[::-2])


def flag3():
    pass
# WARNING: Decompyle incomplete

flag1()
```

参考【愚公系列】2021年12月 攻防世界-进阶题-MISC-075(test.pyc)_愚公搬代码的博客-CSDN博客用的是uncompyle6

> uncompyle6是一个原生python的跨版本反编译器和fragment反编译器，是decompyle、uncompyle、uncompyle2等的接替者。
>
> uncompyle6可将python字节码转换回等效的python源代码
>
> 安装：pip install uncompyle6
>
> 反编译使用：uncompyle6 -o . pyc文件名

uncompyle6 -o C:\Users\86139\Desktop\1.py C:\Users\86139\Desktop\1.pyc



从上到下复制下来然后反转fjU1MmYyNWcyNmcyOTgyYjY4MTc5NWMzZjc0ZzlINzMyfGhibWc=

base64解密得到~552f25g26g2982b681795c3f74g9e732|hbmg

反转gmbh|237e9g47f3c597186b2892g62g52f255~

gmbh和flag就差一位，凯撒加密不对

gmbh|237e9g47f3c597186b2892g62g52f255~

位移 1 加密 解密

flag|237d9f47e3b597186a2892f62f52e255~

ascii码移一位，flag{126d8f36e2b486075a1781f51f41e144}

```
c='gmbh|237e9g47f3c597186b2892g62g52f255~'
d=''
for i in range(0,len(c)):
 k = chr(ord(c[i])-1)
 d+=k
print(d)
```

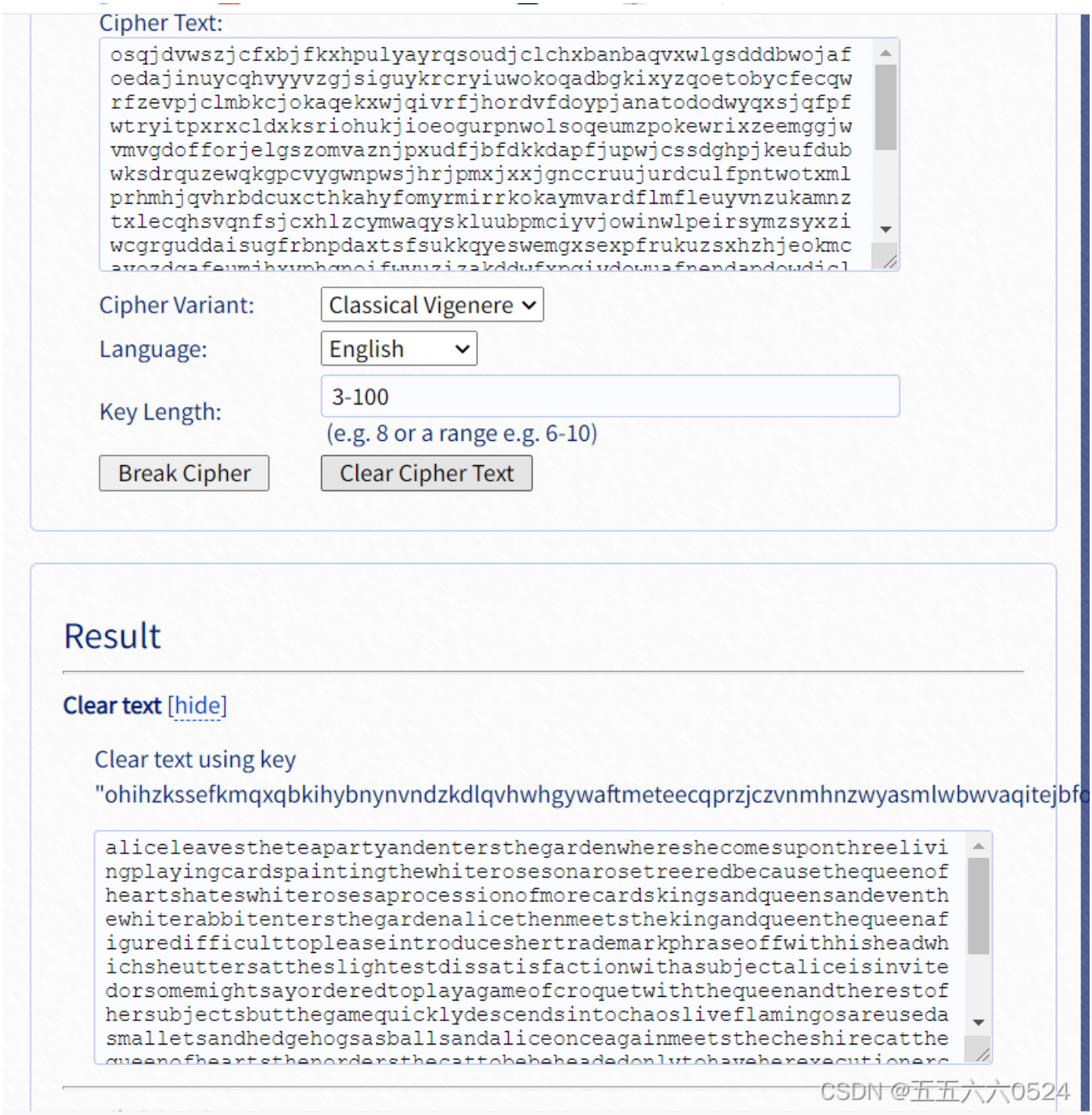
79、challenge_how_many_Vigenère

参考：攻防世界 Misc高手进阶区 7分题 challenge_how_many_Vigenère_思源湖的鱼的博客-CSDN博客

根据题目，知道是维吉尼亚密码，但是没有key，没法直接解密得到明文

enn第一次遇到需要爆破得维吉尼亚密码，Vigenere Solver - www.guballa.de

key：ohihzkssefkmqxqbkihybnynvndzkdlqvhwhgywaftmeteecqprzjczvnmhnzwyasmlwbwvaqitejbfofycejjlcbpk



爱丽丝梦游仙境，但意译

Alice in Wonderland不对，得直译Alice's Adventures in Wonderland，LCTF{osqjdcsvzjxfkoutsvdmoqcegnqc}

Vigenère Cipher（维吉尼亚密码）

Encode Decode

Keyword

ohihzkssefkmqxqbkihybnynvndzk

AlicesAdventuresinWonderland

osqjdcsvzjxfkoutsvdmoqcegnqc