XCTF 攻防世界 MISC杂项 新手练习区

原创

天问_Herbert555 于 2020-01-26 21:52:55 发布 2050 收藏 3

分类专栏： # 各平台题目 文章标签： 信息安全

文章目录

1，stegano(AB转换为摩斯电码)

日期：2020/02/03

---

control+A全选pdf文档，ctrl+C复制，ctrl+B粘贴到记事本里。发现有连续的A 和 B，用' . '代替A,用' - '代替B。（涨知识了）

```
import re
import sys
a='BABA BBB BA BBA ABA AB B AAB ABAA AB B AA BBB BA AAA BBAABB AABA ABAA AB BBA BBBAAA ABBBB BA AAAB ABBBB AAAAA
 ABBBB BAAA ABAA AAABB BB AAABB AAAAA AAAAA AAAAB BBA AAABB'
b=re.findall('.{1}',a)
for i in b:
    if i=='A':
        sys.stdout.write('.')
    if i==' ':
        print ' ',
    if i=='B':
        sys.stdout.write('-')

#-.-. --- -. --. .-. .- - ..- .-.. .- - .. --- -. ... --..-- ..-. .-.. .- --. ---... .---- -. ...- .---- ..... .
---- -... .-.. ...-- -- ...-- ..... ..... ....- --. ...--
```

解码后：`congratulations,flag:1nv151bl3m3554g3`

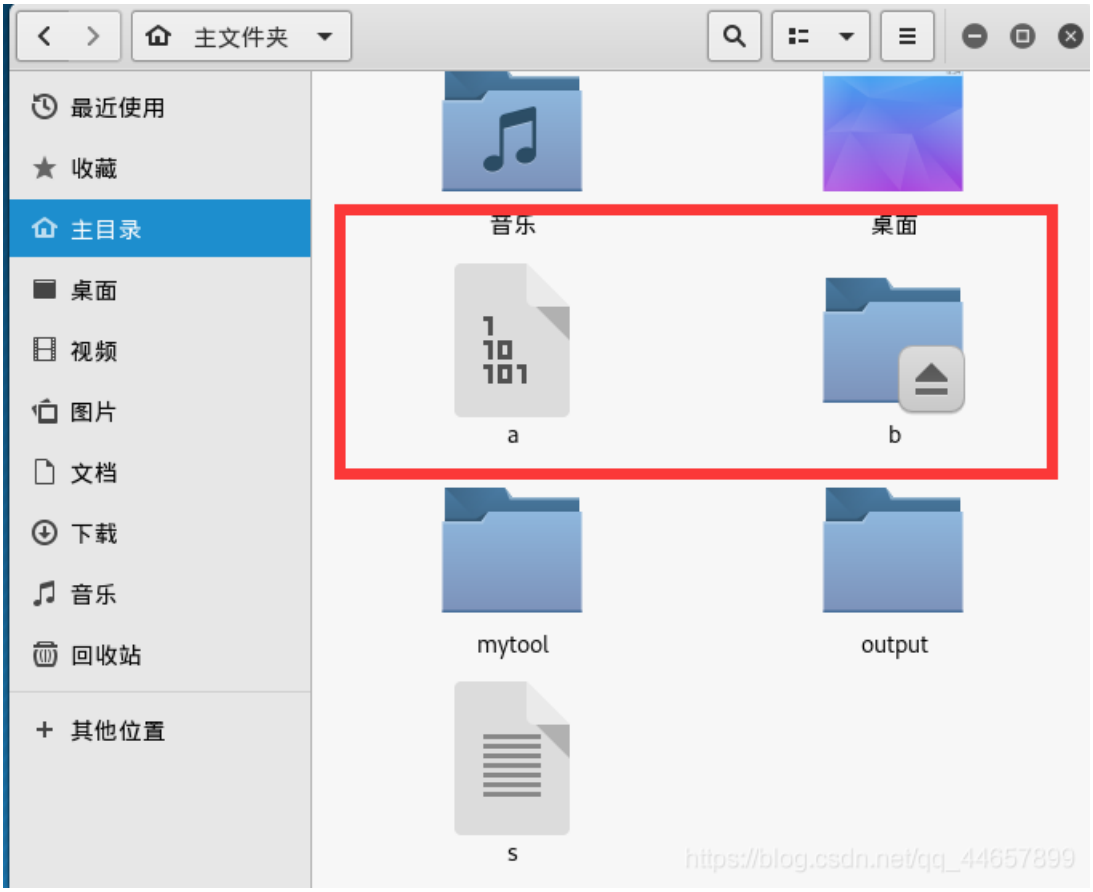
2，ext3（挂载-mount命令）

日期：2020/02/03

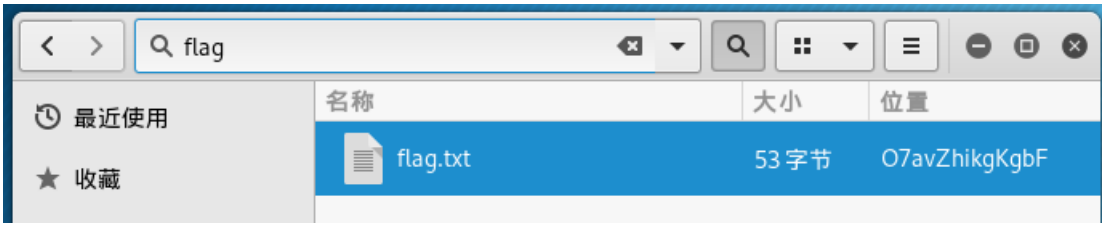
根据题目知道这是一个linux系统光盘，然后要挂载到linux里看看里面有什么东西。

挂载命令：mount 文件 目录

例如这道题我把附件重命名为a，再在该目录下创建一个b文件夹，然后`mount a b`就可以了



打开文件夹，里面有很多文件，搜索flag，找到一个flag.txt。



打开base64解码得到flag。

3，base64stego（伪加密，base64隐写）

日期：2020/02/02

打开后是一个加密的文本文件，首先用ARCHPR 破解压缩密码尝试暴力破解，

但是报错了，上网也没有找到解决方法，只有试试其他方法了。

然后尝试一下是不是伪加密，搜索50 4B 01 02 ，把第九位，第十位改为0。保存，成功解压出stego.txt。

```
00000D40  3D 21 E4 58 69 B4 1A C2  22 F0 15 35 09 8B 6F 08  =!äXi´ Â"ð 5 ‹o
00000D50  43 87 FF 11 88 42 D5 58  EA E7 E0 2C 55 5A C2 07  C‡ÿ ˆBÕXêçà,UZÂ
00000D60  5B C6 D6 E0 F4 32 AF 96  F7 82 D5 7B EB 4A 32 F0  [ÆÖàô2¯–÷‚Õ{ëJ2ð
00000D70  DE 4E E0 23 FF 0B A2 06  72 6D 01 D0 A6 22 43 79  ÞNà#ÿ ¢ rm Ð¦"Cy
00000D80  CA 97 62 46 75 9D AF 77  EF 23 39 B0 8A B0 91 F3  Ê—bFu ¯wï#9°Š°‘ó
00000D90  A9 CD 8A 49 16 6B B9 BF  D6 7C 86 23 12 F8 07 34  ©ÍŠI k¹¿Ö|†# ø 4
00000DA0  94 9A 4E E5 70 96 1A EA  5F 44 FE 46 89 DA 17 AF  ”šNåp– ê_DþF‰Ú ¯
00000DB0  63 42 87 8D D5 FF 68 D6  7A CE 8C 71 B1 9A 2B 20  cB‡ ÕÿhÖzÎŒq±š+
00000DC0  64 C2 55 3C 88 57 B3 35  F3 5E D7 B9 53 7C C6 48  dÂU<ˆW³5ó^×¹S|ÆH
00000DD0  5D F5 34 27 7C 3E 25 33  56 89 72 D2 3D 43 9C C8  ]õ4'|>%3V‰rÒ=CœÈ
00000DE0  14 2D DE 6A EC B9 36 4F  18 ED EC 71 DA E5 FB FA   -Þjì¹6O íìqÚåûú
00000DF0  B5 8E 01 5B 68 F9 8F 24  74 78 50 F1 8E E7 E3 0B  µŽ [hù $txPñŽçã
00000E00  36 7A C7 00 3A B1 B6 F5  2F AD E8 CC FC DB F8 0F  6zÇ :±¶õ/-èÌüÛø
00000E10  50 4B 01 02 3F 03 14 03  00 00 08 00 68 BF 9B 48  PK  ?       h¿›H
00000E20  FE 32 7D 4B E9 0D 00 00  B5 1B 00 00 09 00 24 00  þ2}Ké   µ      $
00000E30  00 00 00 00 00 00 20 80  ED 81 00 00 00 00 73 74        €í      st
00000E40  65 67 6F 2E 74 78 74 0A  00 20 00 00 00 00 00 01  ego.txt
00000E50  00 18 00 80 0B 49 BF 9D  A0 D1 01 80 A7 42 38 B7     € I¿  Ñ €§B8·
00000E60  2F D4 01 00 11 AA 37 B7  2F D4 01 50 4B 05 06 00  /Ô   ª7·/Ô PK
00000E70  00 00 00 01 00 01 00 5B  00 00 00 10 0E 00 00 00         [
```


打开文件里面有很多字符。

很多段落之后有=号，猜测是base64加密，但是解码失败。结合题目先rot13再base64还是失败。

然后看题解才知道是base64隐写，原理：神奇的base64隐写

```python
# -*- coding: cp936 -*-
b64chars = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/'
with open('D:\desktop\stego.txt', 'rb') as f:
    bin_str = ''
    for line in f.readlines():
        stegb64 = ''.join(line.split())
        rowb64 =  ''.join(stegb64.decode('base64').encode('base64').split())
        offset = abs(b64chars.index(stegb64.replace('=','')[-1])-b64chars.index(rowb64.replace('=','')[-1]))
        equalnum = stegb64.count('=') #no equalnum no offset
        if equalnum:
            bin_str += bin(offset)[2:].zfill(equalnum * 2)
        print ''.join([chr(int(bin_str[i:i + 8], 2)) for i in xrange(0, len(bin_str), 8)]) #8 位一组
```

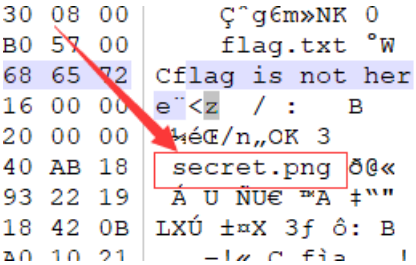
```
Base_sixty_four_point_fiv
Base_sixty_four_point_fiv
Base_sixty_four_point_fiv
Base_sixty_four_point_five
Base_sixty_four_point_five
Base_sixty_four_point_five
Base_sixty_four_point_five
```

4，SimpleRAR（rar文件格式）

日期：2020/02/02

下载好文件之后打开有个flag.txt，里面只有一句flag is not here。

然后拖到winhex里面看到里面有个secret.jpg，于是用foremost尝试分离出来，但是分了半天什么都没分离出来？？



看了看题解才知道要把7A为74，，，因为要把子块(7A)改成文件块(74)。

原理：CTF解题技能之压缩包分析基础篇

png的图片压缩的文件头有问题，文件块的HEAD_TYPE应该是0×74而不是0x7A。

2.2.1 标记块（MARK_HEAD）

HEAD_CRC	2 字节	总是 0x6152
HEAD_TYPE	1 字节	头类型 0x72
HEAD_FLAGS	2 字节	总是 0x1a21
HEAD_SIZE	2 字节	块大小 = 0x0007,即 7 个字节

Test 文件：HEAD_CRC：

0	1	2	3	4	5	6
52	61	72	21	1A	07	00

HEAD_TYPE：

0	1	2	3	4	5	6
52	61	72	21	1A	07	00

HEAD_FLAGS：

0	1	2	3	4	5	6
52	61	72	21	1A	07	00

HEAD_SIZE：

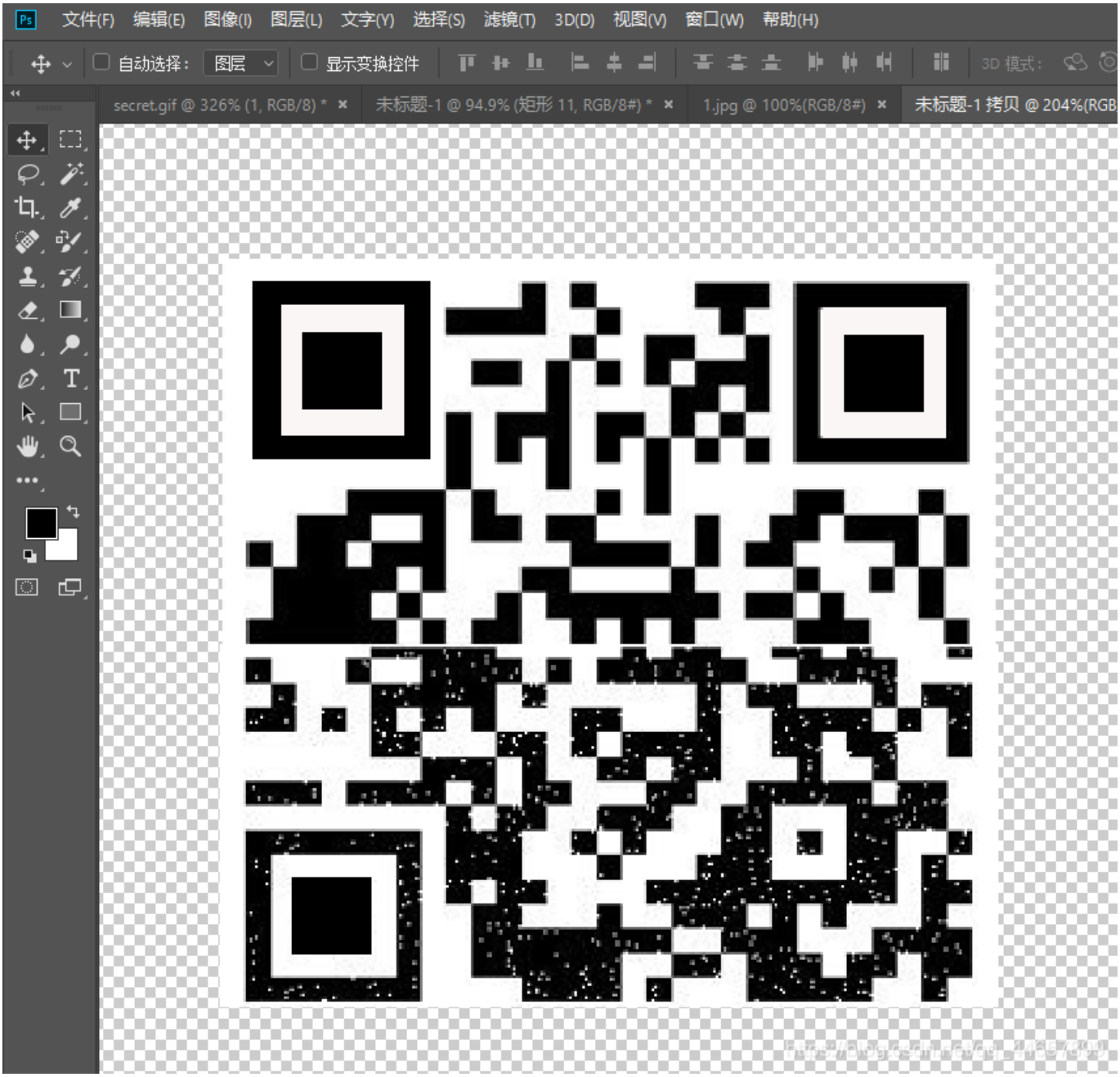
0	1	2	3	4	5	6
52	61	72	21	1A	07	00

所以这里标记块的大小固定是 7 个字节，且是一个固定的字节序列。

然后成功解压出来一张白图片，放到Stegsolve里看到半张二维码

题目提示了ps双图层，于是把另外一张图层保存出来然后放到stegsolve里找到另外半张二维码。

放到ps里面p到一起，然后加一个二维码定位点，搞了半小时，，，



然后扫描二维码得到flag。

5，掀桌子（两位16进制减128，脑洞）

日期：2020/02/01

题目描述：
菜狗截获了一份报文如下
c8e9aca0c6f2e5f3e8c4efe7a1a0d4e8e5a0e6ece1e7a0e9f3baa0e8eafae3f9e4eafae2eae4e3eaebfaebe3f5e7e9f3e4e3e8eaf9eaf3e2e4e6f2，生气地掀翻了桌子(ノ °□°）ノ ︵ ┻━┻

这道题完全看writeup，刚开始看都是f以前的，试了试16进制无果，看了题解才知道要两个两个拆出来，再减去128。

```
from __future__ import print_function
import re

a = 'c8e9aca0c6f2e5f3e8c4efe7a1a0d4e8e5a0e6ece1e7a0e9f3baa0e8eafae3f9e4eafae2eae4e3eaebfaebe3f5e7e9f3e4e3e8eaf9eaf3e2e4e6f2'
b = re.findall(r'.{2}',a)
flag = ''
for i in b:
    print(chr(int(i,16)-128),end='')
```

前面的一个 r 表示字符串为非转义的原始字符串，让编译器忽略反斜杠，也就是忽略转义字符。

---

python2 print 不换行方法:

- 1，后面加一个逗号，但是有空格。
- 2，使用 sys.stdout.write。
- 3，from **future** import print_function使用python3的print函数。

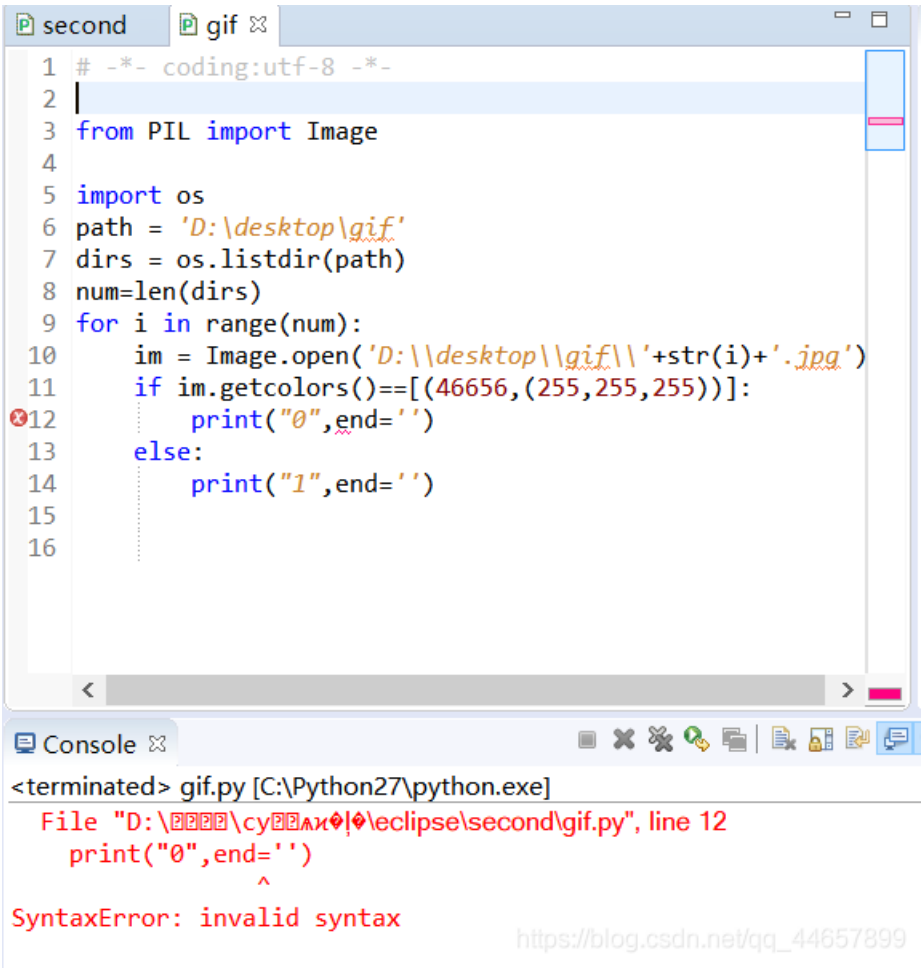
6，gif（脚本图片取色转为二进制）

日期：2020/01/31

打开后是黑白图片，联想到是对应的0，1。

然后学看看别人的脚本学习学习，

```
from PIL import Image
import os
path = 'D:\desktop\gif'
dirs = os.listdir(path)
num=len(dirs)
for i in range(num):
    im = Image.open('D:\\desktop\\gif\\'+str(i)+'.jpg')
    if im.getcolors()==[(46656,(255,255,255))]:
        print("0",end='')
    else:
        print("1",end='')
```



但是编译的时候会报错，是因为我用的python2，print(“0”,end=’’)是python3的用法，要在前面加一个`from __future__ import print_function`

这样就可以在2.X中使用3.X中的print函数了。

还有如果不在print后面加end=' '的话会出现换行的情况，因为print默认输出语句后换行。

然后将得到的字符串每8个一组求出ascii值,转换为字符，这里可以用脚本也可以用在线工具
在线二进制转换ascii值

```
from __future__ import print_function #python2必须加
s='01100110011011000110000101100111011110110100011001110101010011100101111101100111011010010100011001111101'
for i in range(len(s)/8):
    print(chr(int(s[i*8:i*8+8],2)),end='')
    #flag{FuN_giF}
```

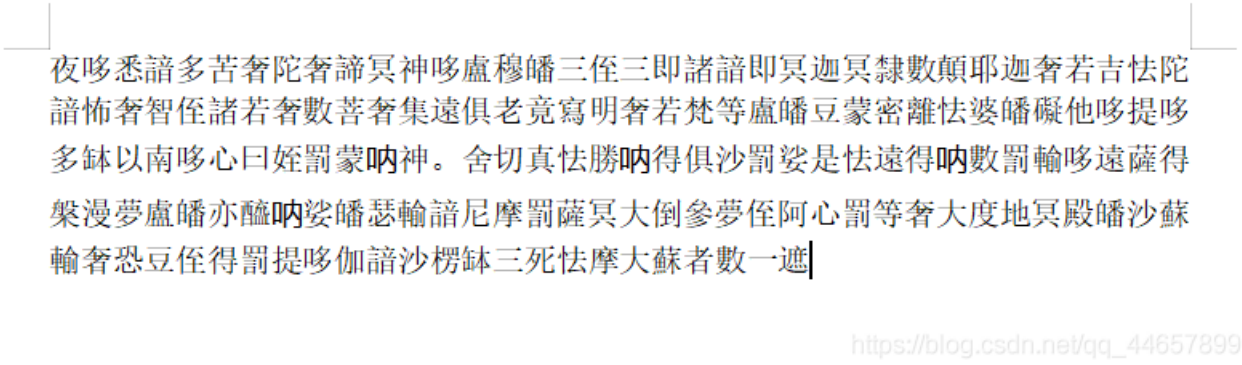
7，坚持60s（简单反编译）

使用jd-gui打开搜索得到flag

8，如来十三掌（多重加密）

日期：2020/01/31

下载好文件后打开，猜测是佛曰加密，解密后得到字符串：
MzkuM3gvMUAwnzuvn3cgozMlMTuvqzAenJchMUAeqzWenzEmLJW9

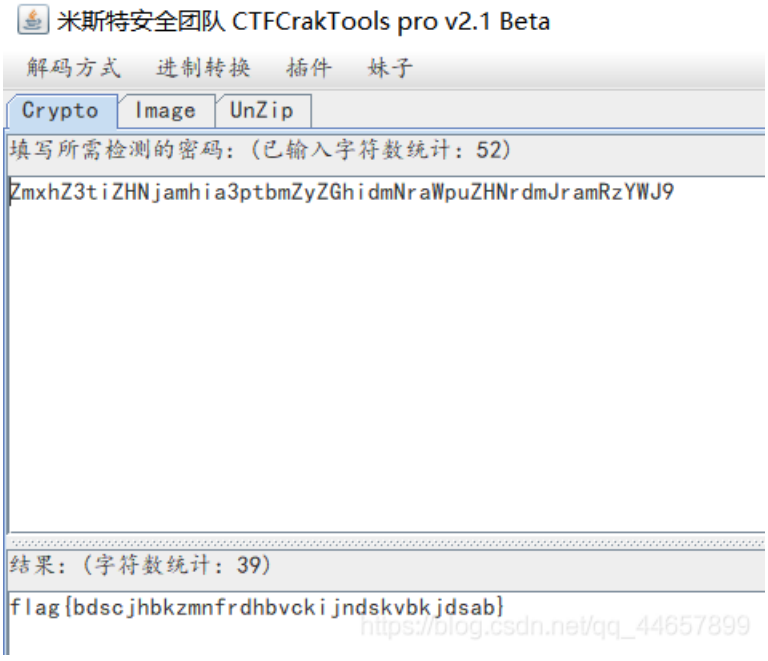


然后尝试base64解码，结果解码不了。然后试了试很多种加密都已失败告终。

看了看writeup结果是rot13 + base64…

然后用CTFCrackTools解码得到：ZmxhZ3tiZHNjamhia3ptbmZyZGhidmNraWpuZHNrdmJramRzYWJ9

再base64解密flag{bdscjhbkzmnfrdhbvckijndskvbkjdsab}



这道题感觉最重要的是审题，因为我没有好好审题，所以只是试了试rot13加密，还有忘记了双重加密。

9，功夫再高也怕菜刀（文件头的考察）

日期：2020/01/30

首先下载文件是流量包，用foremost 分离出来一个有密码的flag.txt。然后尝试在流量包里找密码。

使用wireshark打开流量包，查找flag，发现一张图片，选择第1150个，右键，追踪流 -> TCP 流

从 FFD8FF 开始到 FFD9 复制出来，保存为jpg格式，打开图片。