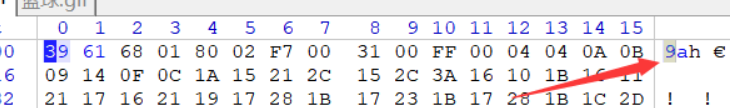
1、签到题（BJDCTF）

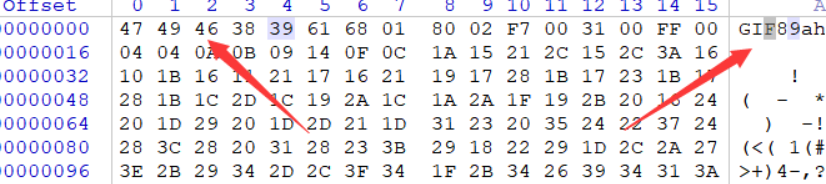
下载附件后得到一个压缩包，解压后得到第二个压缩包，但是第二个压缩包有提示提示，无法打开，用winhex打开发现并不是压缩包，而是张png图片，打开后得到一张二维码，扫码得到flag

2、鸡你太美

附件里是两个gif，但第二个无法显示，winhex打开发现缺了文件格式头部缺失（gif的文件头部为GIF89a）



补全头部

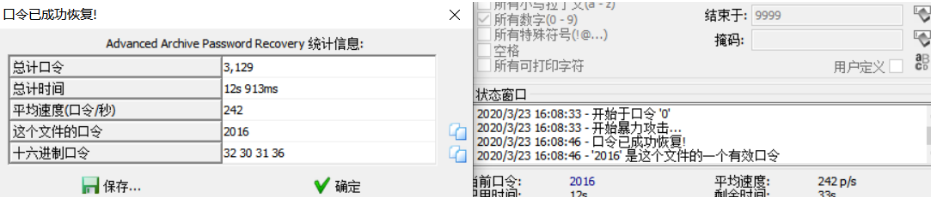


再打开gif就可以看到flag了

3、Just a rar

附件是一个压缩包，需要密码，提示是4位数

直接用ARCHPR暴力破解



解压后得到一张图片，用文本直接打开（这里直接用txt打开）



得到flag，注意要把中间夹带的空格去掉

4、你猜我是个啥

附件zip无法解压，winhex查看，发现是png，修改格式后打开后是张二维码，扫码，提示flag不在这里。



查看网页源码也没有有用的信息。回头看看winhex内容，在文本末尾找到flag

5.一叶障目

png图片打不开 可能是png的CRC校验问题，图片的宽高被改了导致无法打开，网上找了个自动修复脚本：

import zlib  
import struct  
#读文件  
file = '1.png'  
fr = open(file,'rb').read()  
data = bytearray(fr[12:29])  
crc32key = eval(str(fr[29:33]).replace('\\x','').replace("b'",'0x').replace("'",''))  
#crc32key = 0xCBD6DF8A #补上0x，copy hex value  
#data = bytearray(b'\x49\x48\x44\x52\x00\x00\x01\xF4\x00\x00\x01\xF1\x08\x06\x00\x00\x00') #hex下copy grep hex  
n = 4095 #理论上0xffffffff,但考虑到屏幕实际，0x0fff就差不多了  
for w in range(n):#高和宽一起爆破  
  width = bytearray(struct.pack('>i', w))#q为8字节，i为4字节，h为2字节  
  for h in range(n):  
      height = bytearray(struct.pack('>i', h))  
      for x in range(4):  
          data[x+4] = width[x]  
          data[x+8] = height[x]  
          #print(data)  
      crc32result = zlib.crc32(data)  
      if crc32result == crc32key:  
          print(width,height)  
          #写文件  
          newpic = bytearray(fr)  
          for x in range(4):  
              newpic[x+16] = width[x]  
              newpic[x+20] = height[x]  
          fw = open(file+'.png','wb')#保存副本  
          fw.write(newpic)  
          fw.close  
          # return None

6.藏藏藏

附加是图片，先用binwalk扫一下，发现末尾有zip。但用dd分离出后发现zip打不开。具体原因未知，换成foremost来分离就正常了（懵）

得到二维码，扫一下得到flag

7.认真你就输了

下载后解压是名为10.xls的文件，打开发现包含flag.txt和一个doc文件，用foremost打开，得到一个类似doc文档的压缩包根据第一行的路径找到flag.txt即得到flag