XCTF攻防世界_Misc进阶区001

原创

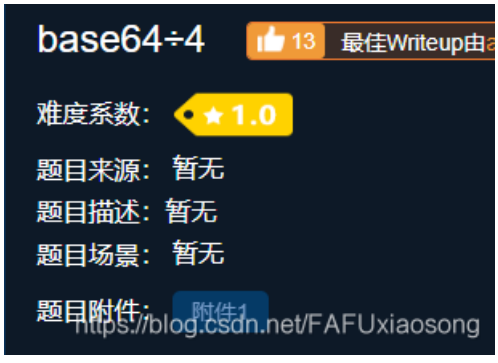
FAFU小宋 于 2020-10-30 00:03:25 发布 214 收藏 2

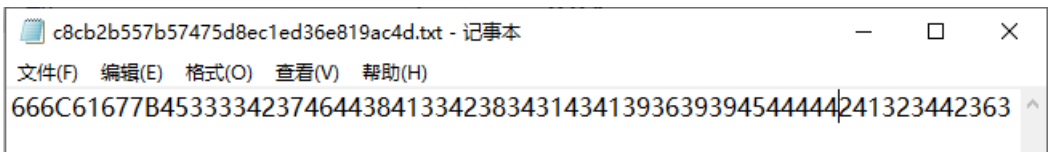
分类专栏： XCTF 文章标签： 安全

本文链接：https://blog.csdn.net/FAFUxiaosong/article/details/109374906

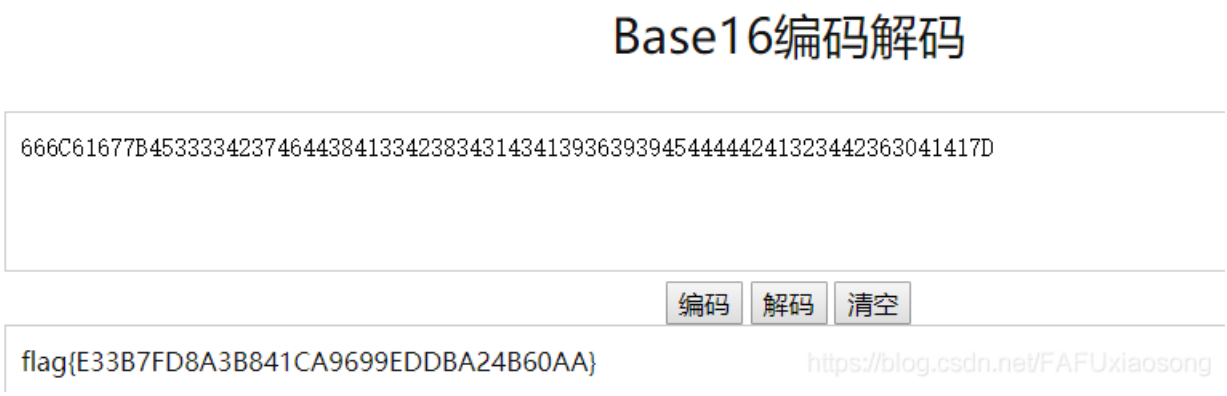
XCTF_Misc_高手进阶区

base64÷4





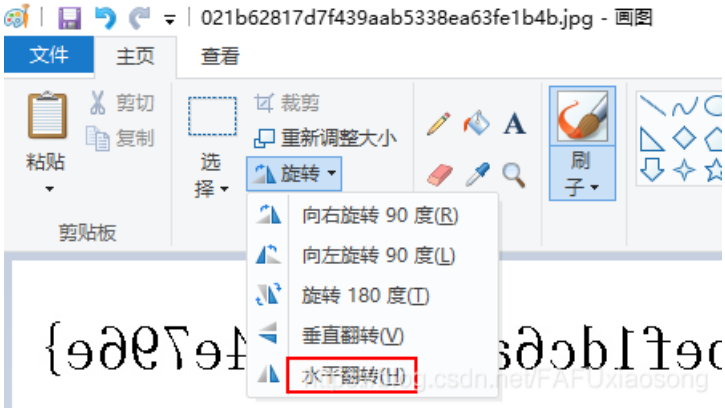
（1）根据题目提示，直接用base16解密得到flag

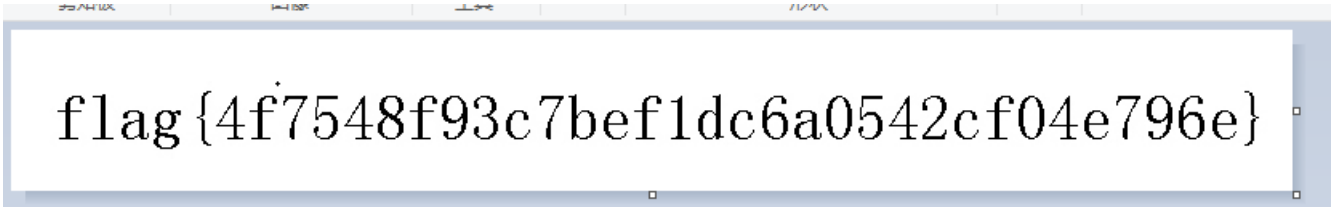


reverseMe

（1）打开附件，发现是一张图片

（2）用画图工具打开该图片，使用水平翻转，即可查看到flag

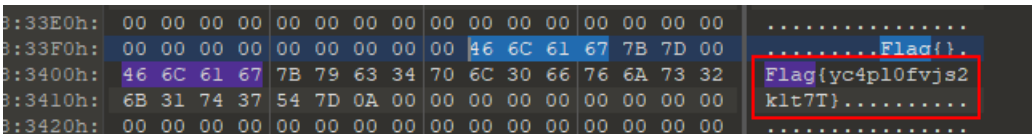




something_in_image

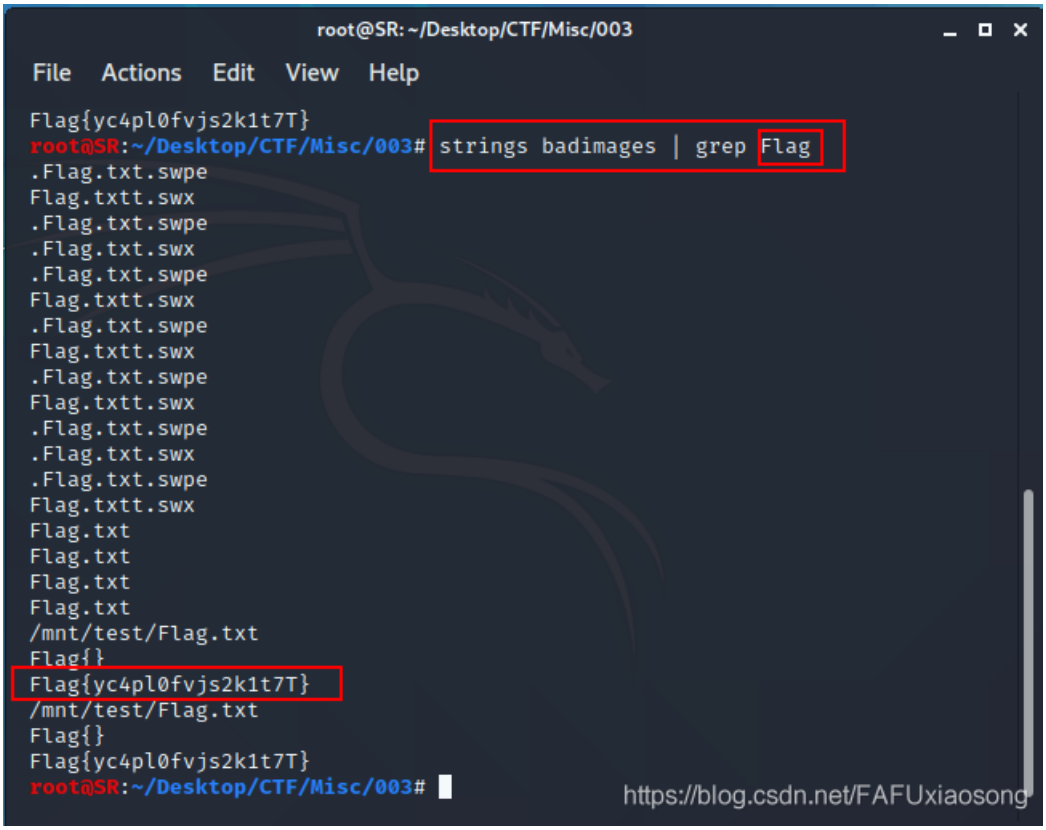
解法一：

（1）使用记事本或010 editor工具打开附件，Ctrl+F搜索flag



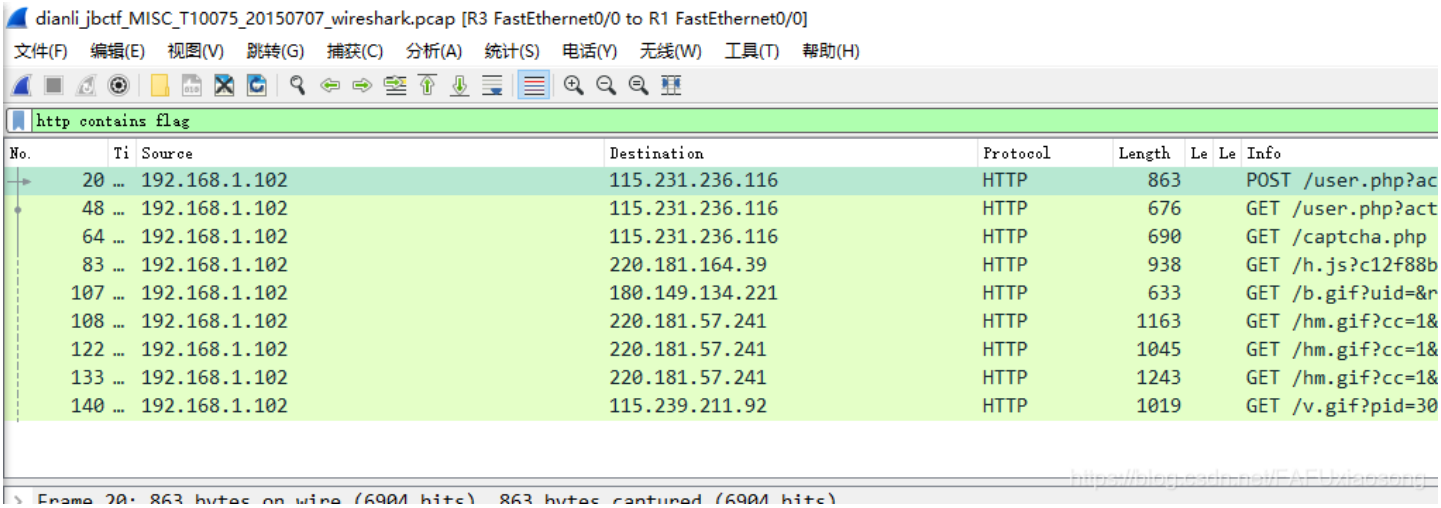
解法二：

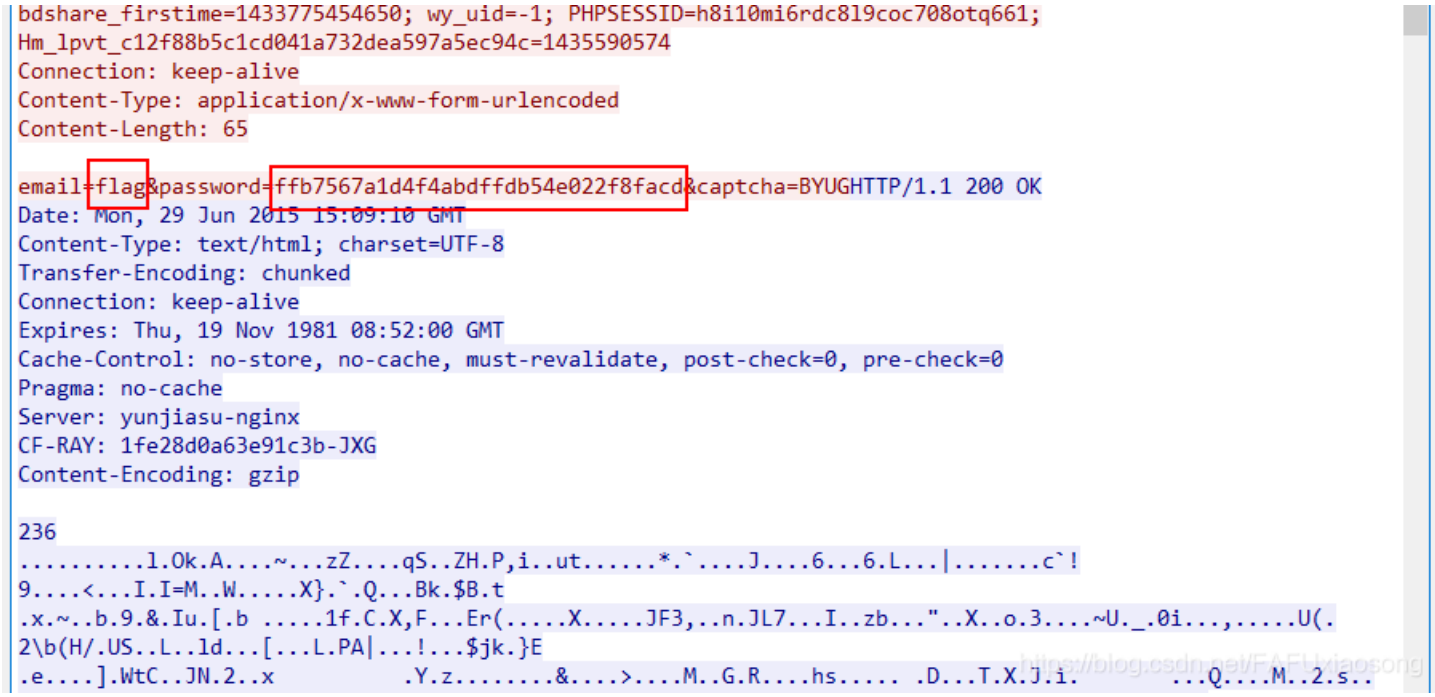
根据题目包含image，使用kali，命令：strings badimages | grep Flag （注，根据记事本中常出现的“Flag”，而不是“flag”进行搜索）



wireshark-1

（1）用wireshark打开附件，输入表达式http contains flag查找flag信息，根据题目，看到登录应该查找HTTP post请求，追踪筛选出的报文追踪tcp流，即可看到flag



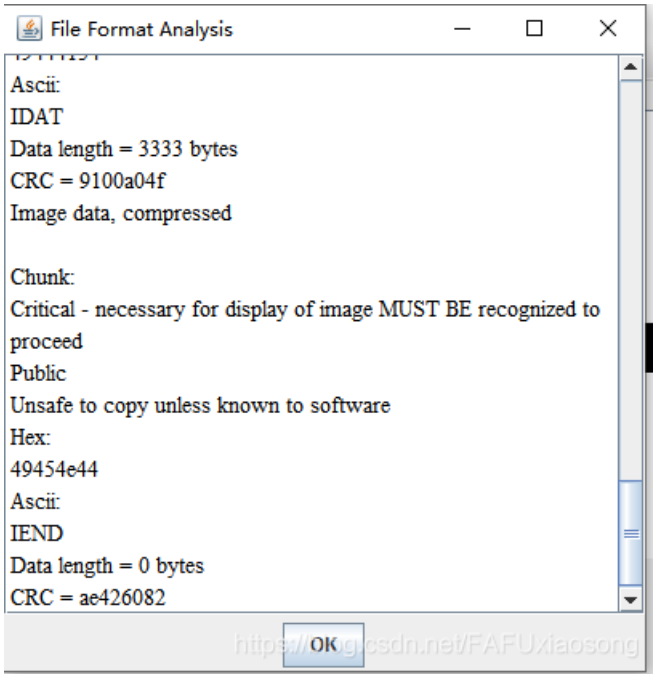


pure_color

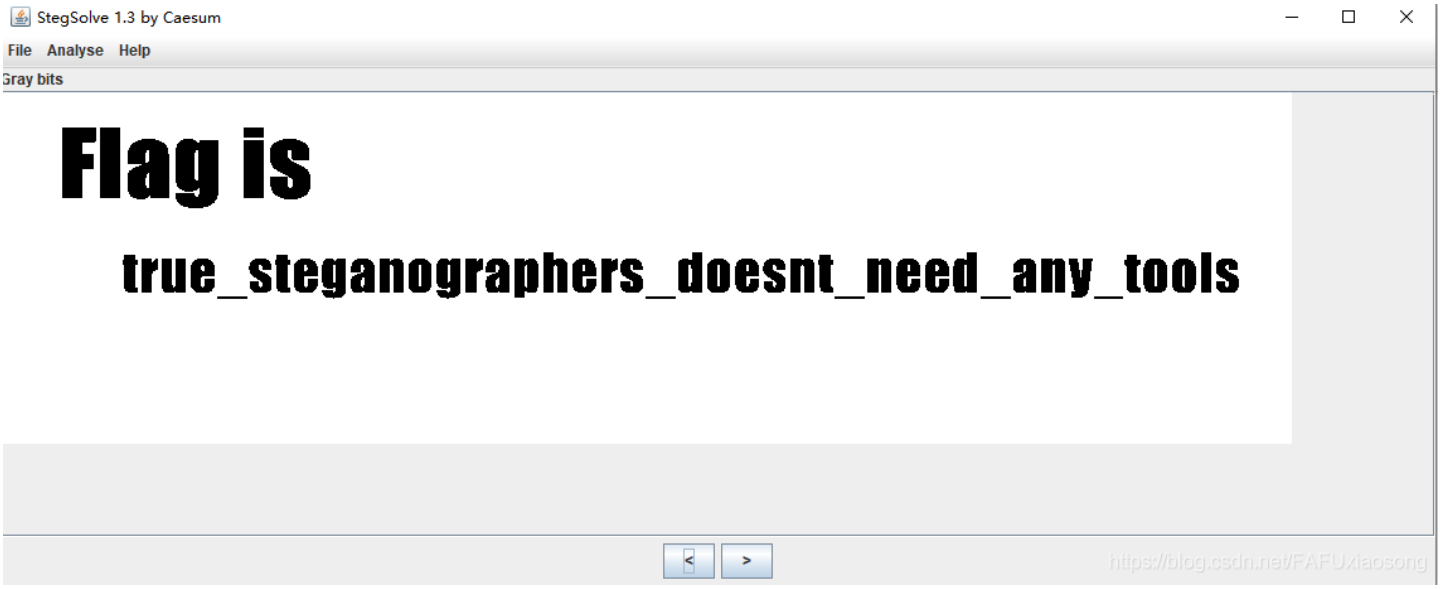
（1）附件是一张全白的png图片

（2）用notepad++打开没有发现flag信息

（3）用stegsolve打开，打开 File Format Analysis 没有flag的线索

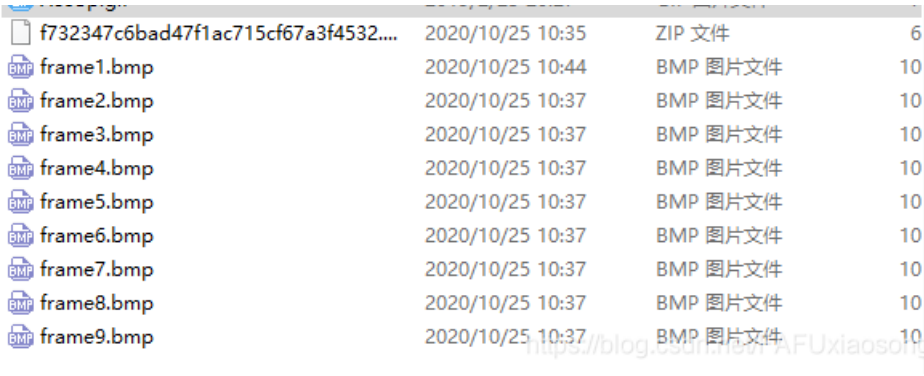


（4)最后在最底下的'<'按钮发现了flag



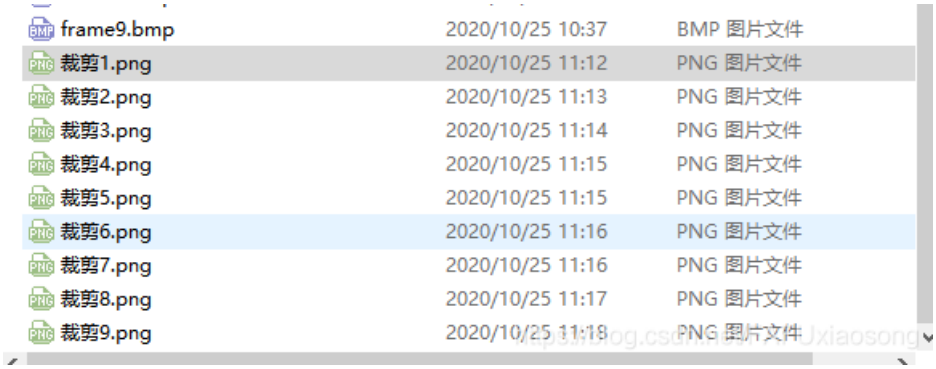
Aesop_secret

（1）下载附件得到一张gif的动图，通过stegsolve工具分解得到9张图片



（2）用画图工具分别将这九张图通过旋转、裁剪一系列操作，得到9张裁剪过的图，再将这9张裁剪过的图片拼在一起（这里耗费了较多时间），得到如下图

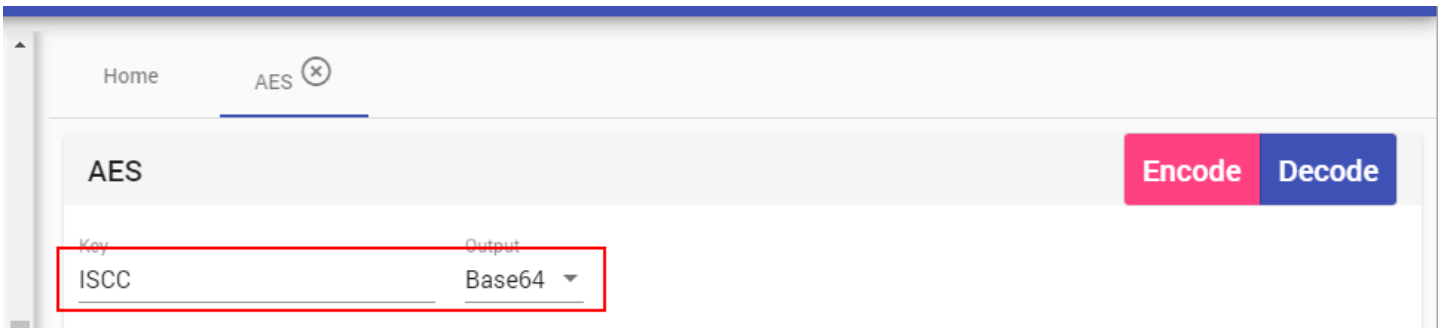
我的拼图



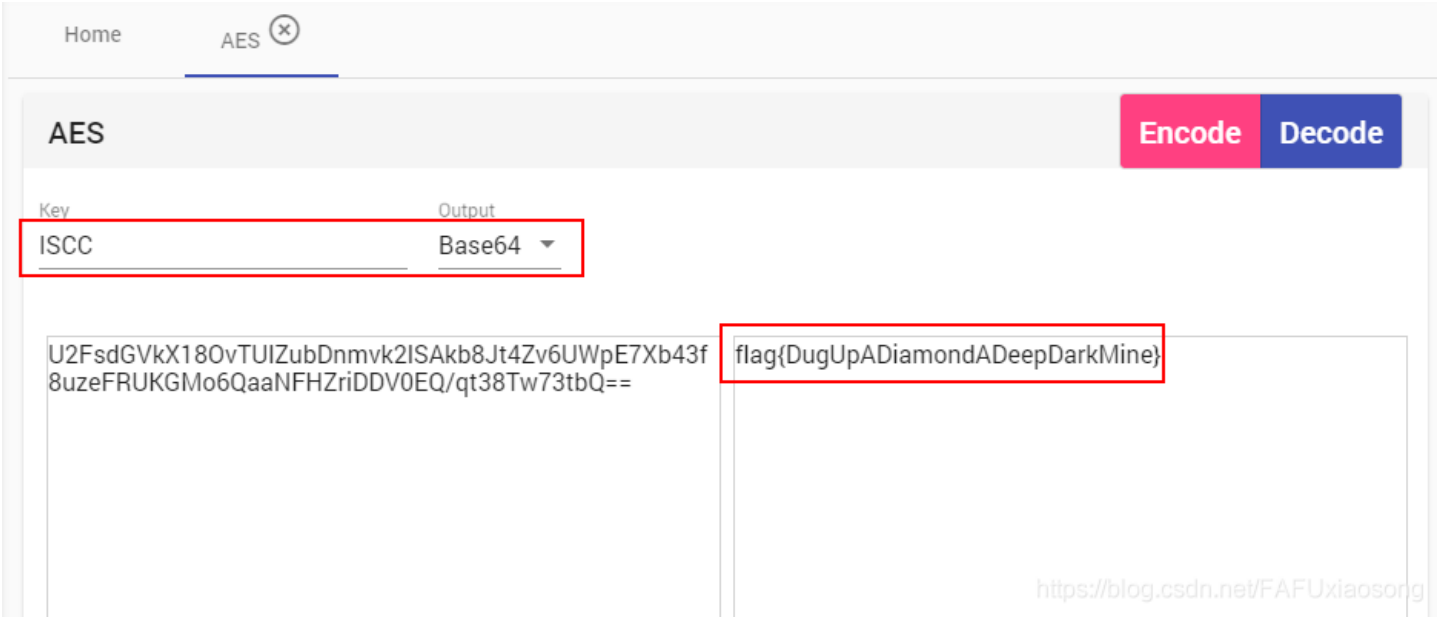
（3）图片显示为ISCC，显然不是flag，再用notepad++打开原来的gif动图，发现一串密文

```
00018c0  c4 16 6b ec b1 b2 06 04  00 3b 55 32 46 73 64 47  ?k毅?..;U2FsdG□h□
00018d0  56 6b 58 31 39 51 77 47  6b 63 67 44 30 66 54 6a  VkX19QwGkcgD0fTj
00018e0  5a 78 67 69 6a 52 7a 51  4f 47 62 43 57 41 4c 68  ZxgijRzQOGbCWALh
00018f0  34 73 52 44 65 63 32 77  36 78 73 59 2f 75 78 35  4sRDec2w6xsY/ux5
0001900  33 56 75 6a 2f 41 4d 5a  42 44 4a 38 37 71 79 5a  3Vuj/AMZBDJ87qyZ
0001910  4c 35 6b 41 66 31 66 6d  41 48 34 4f 65 31 33 49  L5kAf1fmAH4Oe13I
0001920  75 34 33 35 62 66 52 42  75 5a 67 48 70 6e 52 6a  u435bfRBuZgHpnRj
0001930  54 42 6e 35 2b 78 73 44  48 4f 4e 69 52 33 74 30  TBn5+xsDHONiR3t0
0001940  2b 4f 61 38 79 47 2f 74  4f 4b 4a 4d 4e 55 61 75  +Oa8yG/tOKJMNUau
0001950  65 64 76 4d 79 4e 34 76  34 51 4b 69 46 75 6e 77  edvMyN4v4QKiFunw
0001960  3d 3d 0d 0a
```

（4）用base64解码，无果，根据题目和上面图片的ISCC，联想到AES加密，ISCC则为密钥

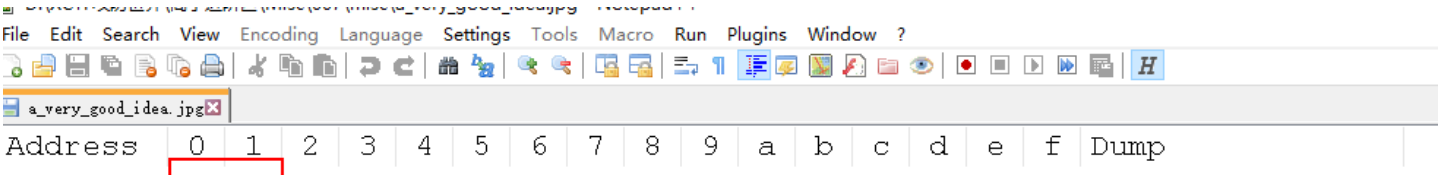


（5）解码后仍然是一串密文，再次解密，得到flag



a_good_idea

（1）打开附件，是一张.jpg格式的图片，用notepad++打开

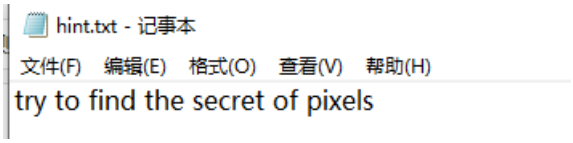


```
00000000 ff d8 ff e0 00 10 4a 46 49 46 00 01 01 01 00 78   ??.JFIF.....x□[
00000010 00 78 00 00 ff e1 00 66 45 78 69 66 00 00 4d 4d .x..  ?fExif..MM[
00000020 00 2a 00 00 00 08 00 05 01 00 00 03 00 00 00 01 .*..............
```

（2）确实以FF D8开头，寻找图片结尾FF D9，发现后面还有一个压缩文件

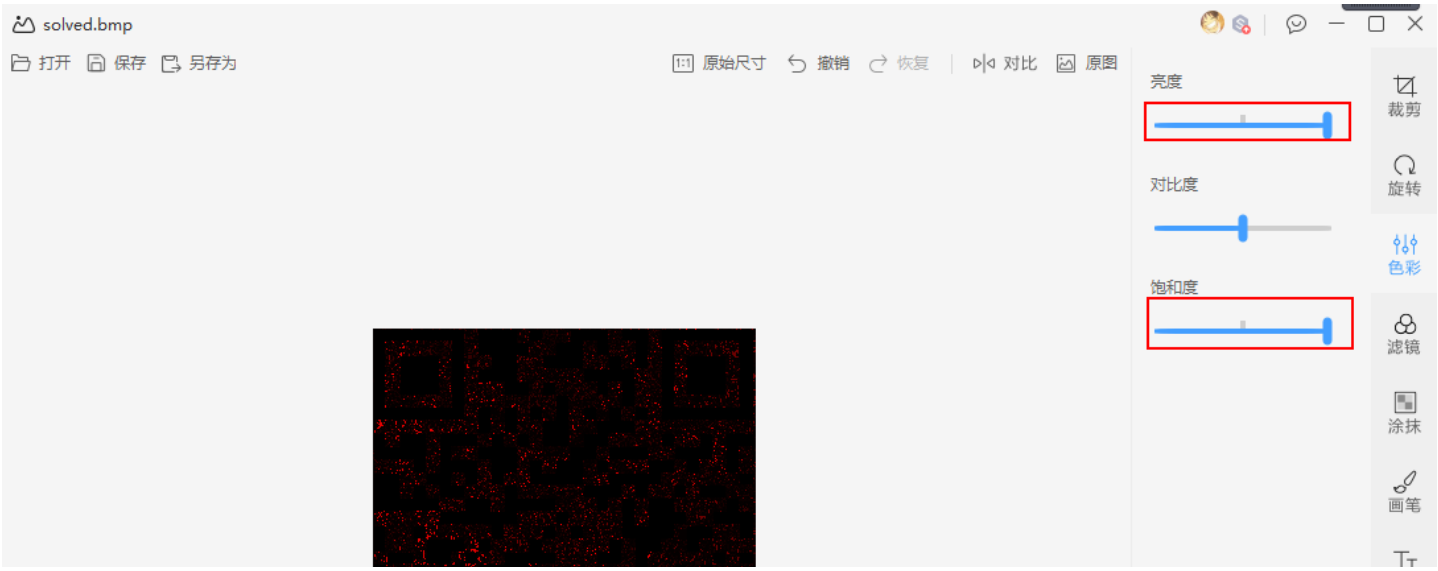
```
00007df0  f5 a9 f4 3f ca 8a 2b 94 d0 8b 5a ff 00 90 4d c7 酴?葬+咿媄  ..M?[
00007e00  fd 74 4f e4 d5 c5 dc 7f 0f d7 fa 0a 28 a0 52 d8 齮O澫跑..  .(.R?[
00007e10  e8 21 ff 00 54 bf 41 45 14 57 41 07 ff d9 50 4b ?  .T緢E.WA.  賦[
00007e20  03 04 0a 00 00 00 00 00 fc 03 72 4f 00 00 00 00 ........?rO....□
00007e30  00 00 00 00 00 00 00 00 05 00 00 00 6d 69 73 63 ............misc
```

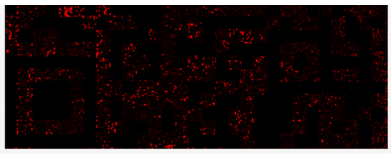
（3）将原图片后缀名改为.zip，解压后得到一个文件夹，里面是两张相同的图片和一个hint提示文本



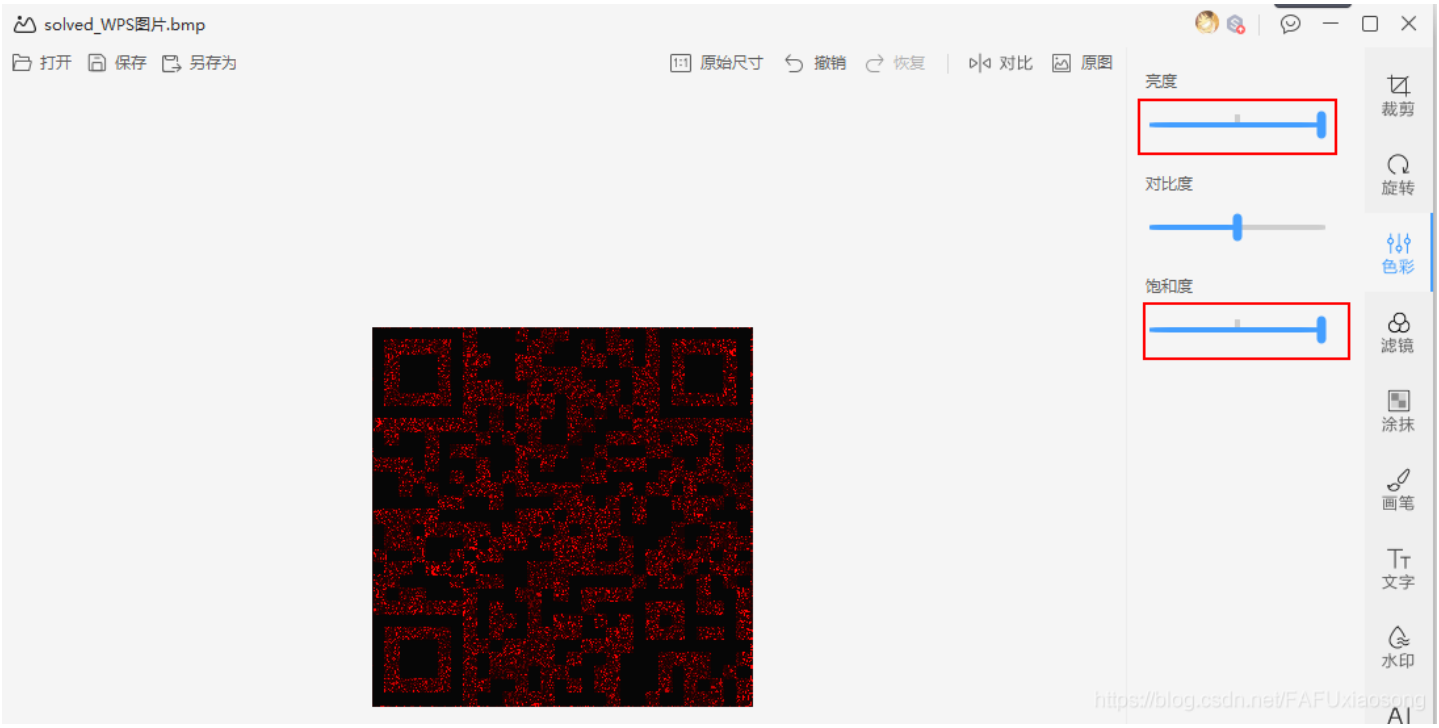
（4）hint为“试着找出像素的秘密”，用stegsolve工具合成这两张图片，发现一些红点，貌似是二维码

（5）用wps图片编辑，在“工具”中选择“色彩”，将亮度和饱和度调到最高

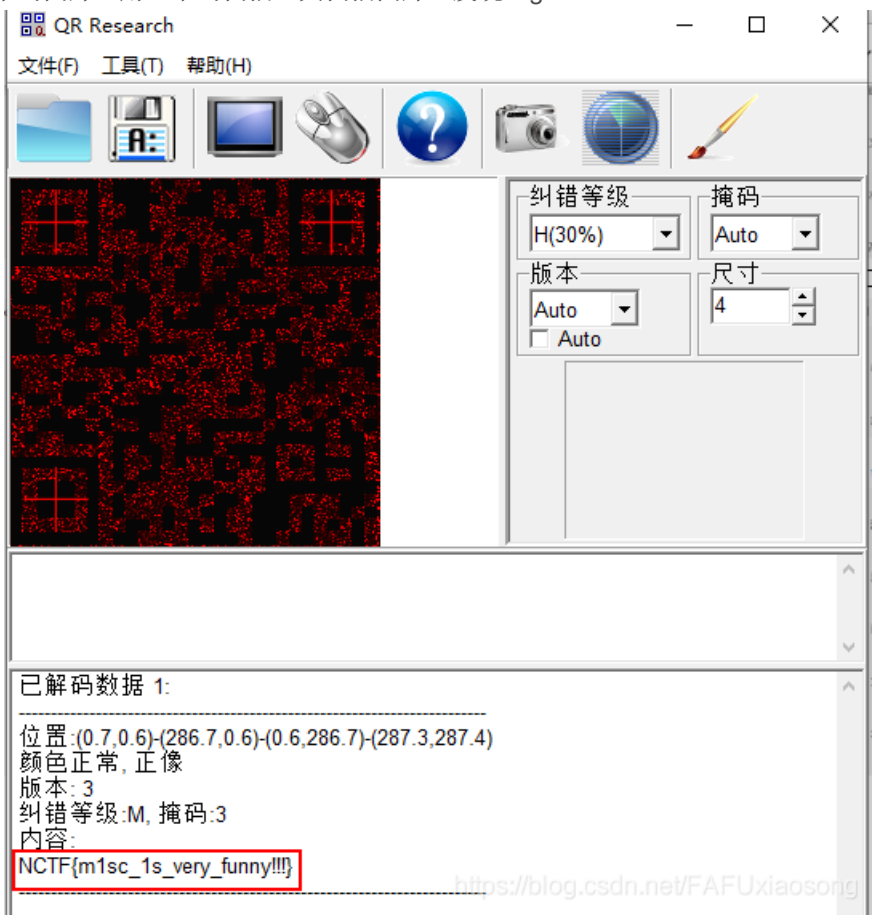




（6）发现还是不太明显，先保存这张图片，再用wps图片工具打开，再调一遍亮度和饱和度



（7）可以看到清晰的二维码图片，用二维码扫描工具扫描图片，发现flag



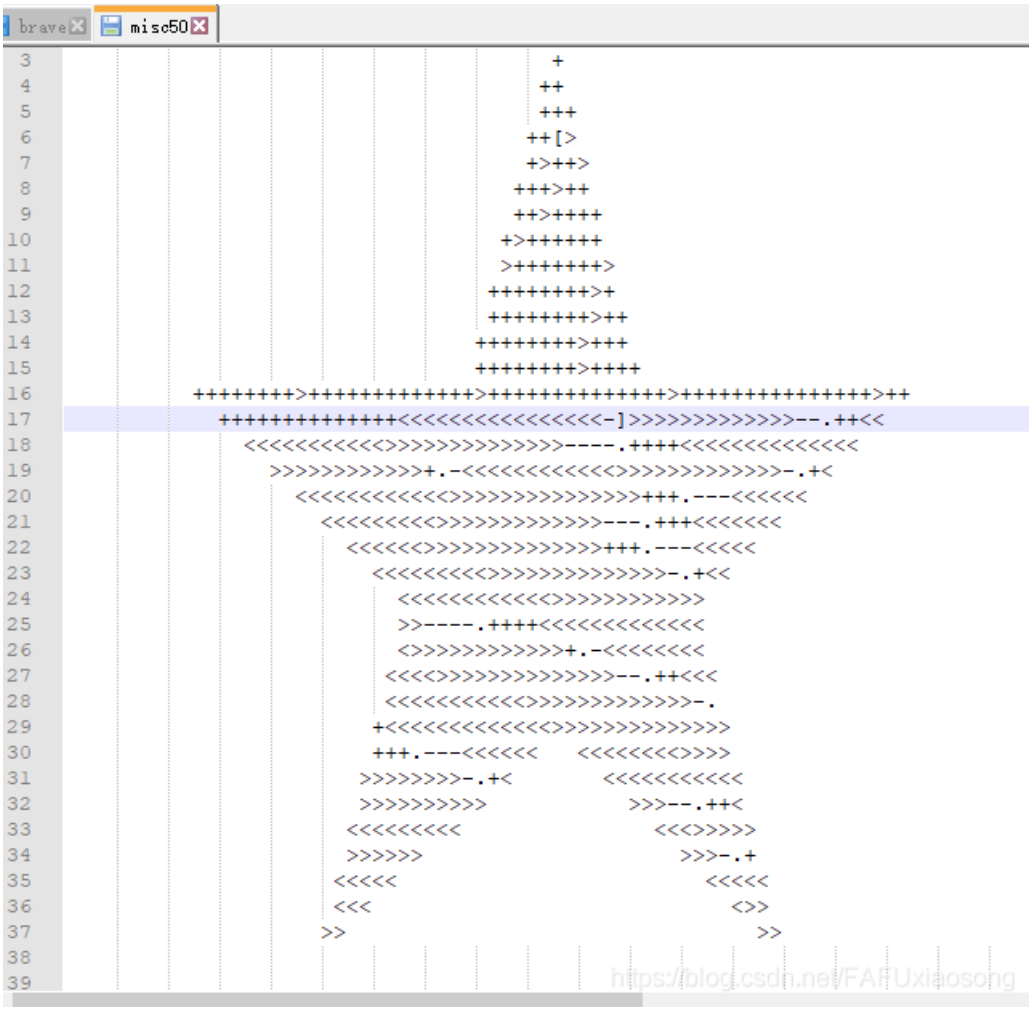
Training-Stegano-1

一张bmp格式的图片，直接用notepad++打开，即可看flag，注意提交时不用加flag{}

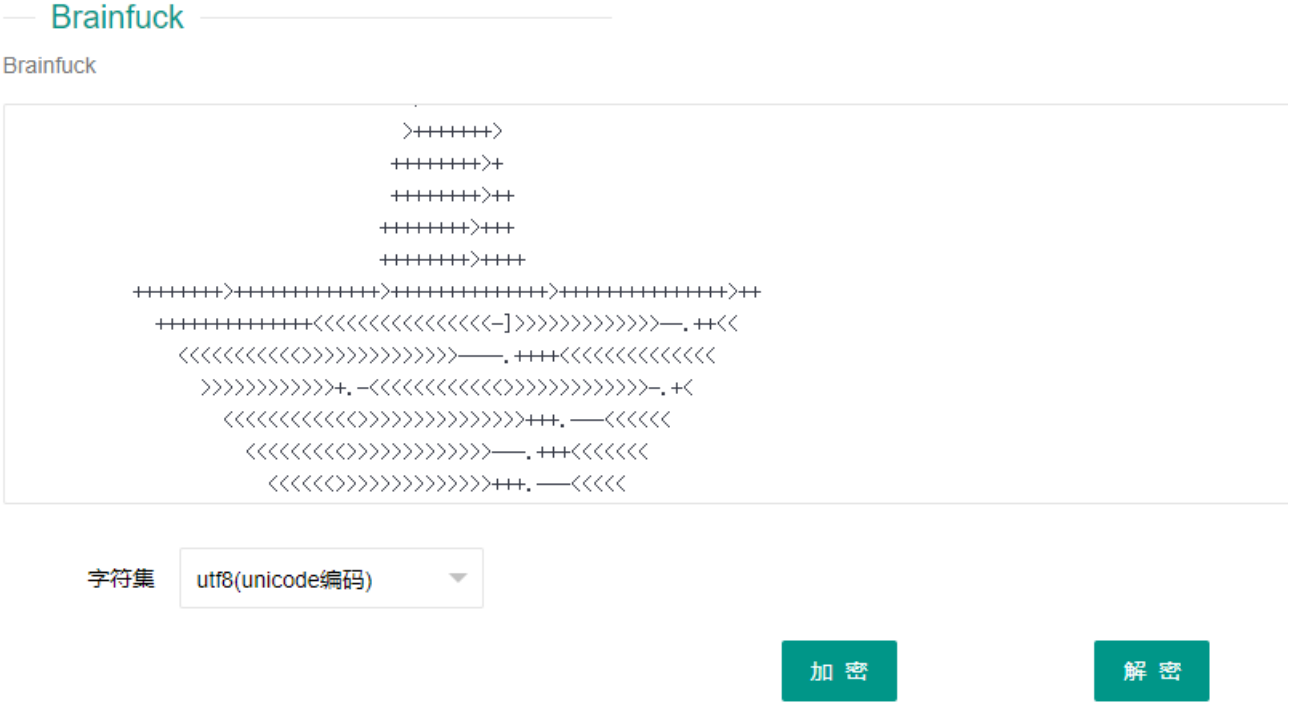
```
00000050  65 61 6c 65 64 3a 20 70 61 73 73 77 64 3a 73 74  ealed: passwd:st
00000060  65 67 61 6e 6f 49                                 eganoI
```

can_has_stdio?

（1）用notepad++打开附件



（2）发现是Brainfuck文件，用在线解析工具得到flaghttp://ctf.ssleye.com/brain.html



```
flag{esolangs_for_fun_and_profit}
```


#Brainfuck解析规则：

“>”: 指针位置右移一位

“<”: 指针位置左移一位

+: 指针所指向的位置里面的值加1

-: 指针所指向的位置里面的值减1

. :输出指针当前位置指向单元格中的内容

[: 如果指针指向的单元值为零，向后跳转到对应的]指令的次一指令处

]:如果指针指向的单元值不为零，向前跳转到对应的[指令的次一指令处(额…好像比较难理解,只是用于循环某一段代码而已,两个一般联合使用)