PNG隐写入门赛

原创

Atkxor 于 2021-04-23 17:07:39 发布 285 收藏 3

分类专栏： CTF WriteUp 文章标签： png

本文链接：https://blog.csdn.net/qq_46150940/article/details/114821304

版权

CTF 同时被 2 个专栏收录

39 篇文章 2 订阅

订阅专栏

WriteUp

15 篇文章 0 订阅

订阅专栏

目录标题

题目说明

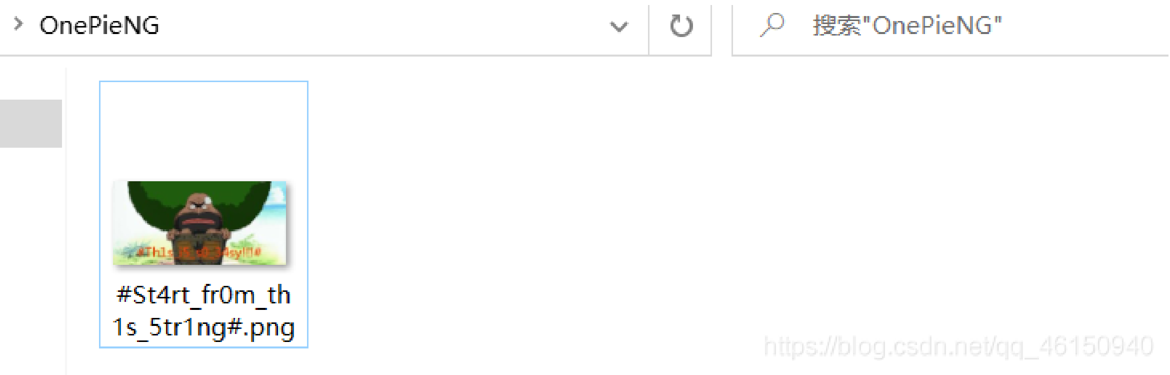
```
附件下载：https://ctfshow.lanzous.com/iDEiVmr8m4d

说明：

0、本场比赛共有18题，但只有1个附件文件（见第1题），所有flag均可以从附件中获取；
1、所有的flag开头和结尾均为#，中间由字母、数字或下划线组成；
2、本场比赛不使用任何*可以*设置密码的隐写方法，包括可以将密码留空的隐写方法；
3、原理类似的隐写方法在确保不互相干扰的前提下可能会以多种方式使用；
4、如果从附件提取的隐写信息为字符串形式，可能需要转码得到指定格式的结果；
5、如果从附件提取的隐写信息为另一张图片，该图片不会再包含隐写信息，即不存在套娃隐写；
6、所使用的字体均为微软雅黑，若有字符无法分辨，请与字体对比查看；
7、取得类似#abcd_1234#的字符串后，请计算其MD5值（包含头尾的#号）；
8、每道题目都给出了一段MD5值，请找到MD5值匹配的题目后，将flag包上ctfshow{}格式提交。
One PieNG 1  342f08112d4ffb0577f49e89a2a18fa2
One PieNG 2  d64fc33636dda50babdde6b775d8cf10
One PieNG 3  8b8bc8c6aa81e7b955660fba3575af63
One PieNG 4  c35bc750588f620f49e83493f4125bfd
One PieNG 5  91848bee27655dc0da45006f467a59fb
One PieNG 6  335b63183f19e4fe1b9bd734af81403e
One PieNG 7  e18d9aa18b35ae3a702875beab14cc86
One PieNG 8  8d4ae0eed967e9936ee5373f0f58829c
One PieNG 9  9734a5d18504ef6a31c2c104b224f0df
One PieNG 10 cec1969402261bd550f1b3d0c0ccc655
One PieNG 11 3e703086b0e2585eff041cbd186f1bd4
One PieNG 12 fba2e6b912ab1a308c6b1438da31fbb8
One PieNG 13 23e4464f1b458a062fb13e155a72f999
One PieNG 14 d325d41389ddb0c3fdec30e51565fda3
One PieNG 15 ad9d95f270d91aed3ba2203487bf01cd
One PieNG 16 7dc6506ac3d4c7a99587c9b3cbf43798
One PieNG 17 170cee5e9bd6dd81021d8533490a4b8b
One PieNG 18 5f6b859726bd17bd5fb4905c4420b269
```

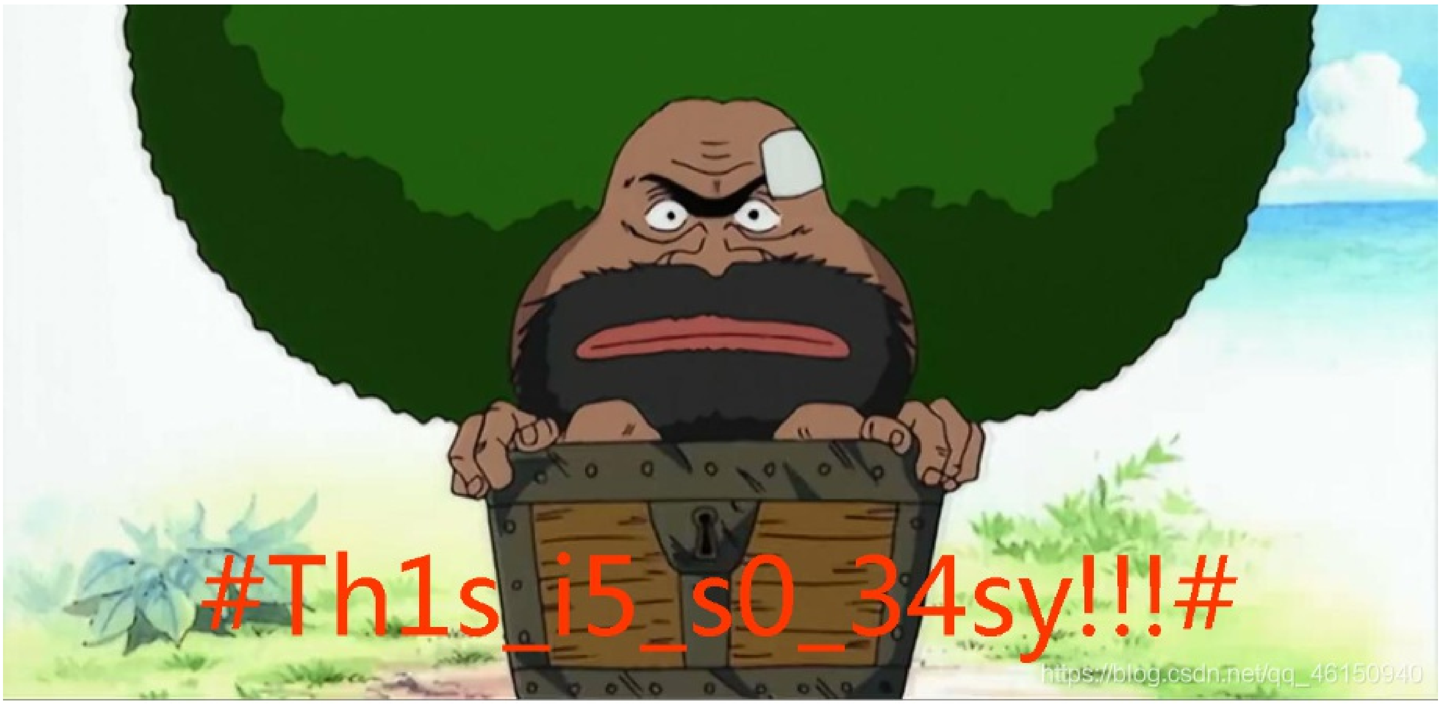
One PieNG 1

文件名称



#St4rt_fr0m_th1s_5tr1ng#.png

One PieNG 2



One PieNG 3

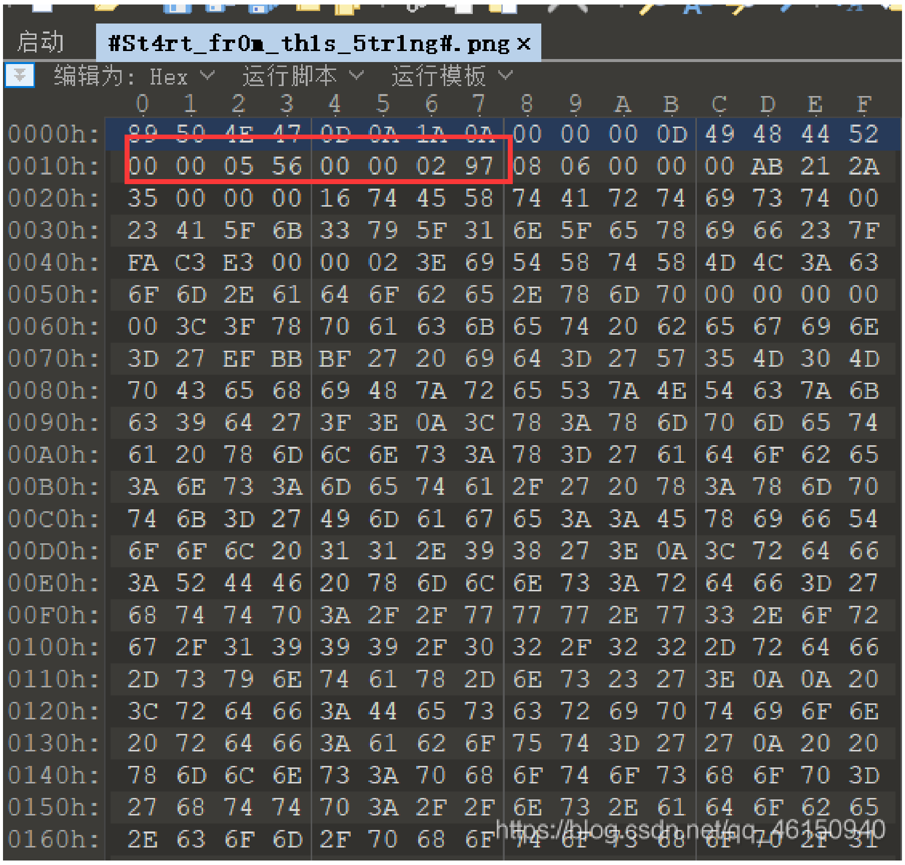
使用python脚本爆破图片高度

```
import os
import binascii
import struct
misc = open("ctfshow.png","rb").read()
#print(misc[0x0c:0x0f+1])
# 爆破高

crc32_bytes = misc[0x1d:0x20+1]# 读出bytes
crc32_hex_eval = eval('0x' + crc32_bytes.hex())#bytes串 -> hex串  ->  值
print(crc32_hex_eval)
for i in range(4096):
    data = misc[0x0c:0x0f+1] + misc[0x10:0x13+1] + struct.pack('>i',i)+ misc[0x18:0x1c+1]  #IHDR数据
    crc32 = binascii.crc32(data) & 0xffffffff
    if crc32 == crc32_hex_eval : #IHDR块的crc32值
        print(i)
        print("height_hex:"+ hex(i))
```

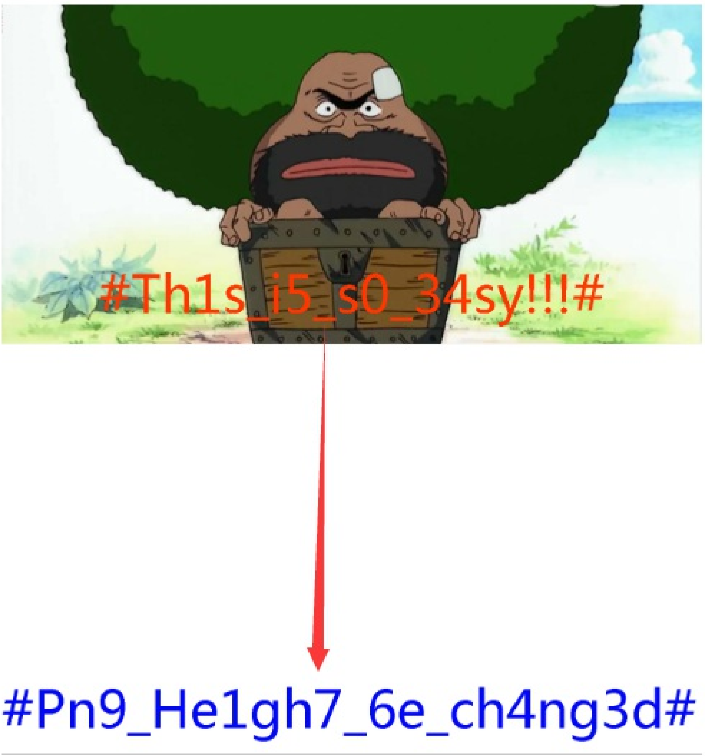
运行脚本得到

```
2871077429
1463
height_hex:0x5b7
```



```
        0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F
0000h: 89 50 4E 47 0D 0A 1A 0A 00 00 00 0D 49 48 44 52
0010h: 00 00 05 56 00 00 02 97 08 06 00 00 00 AB 21 2A
0020h: 35 00 00 00 16 74 45 58 74 41 72 74 69 73 74 00
0030h: 23 41 5F 6B 33 79 5F 31 6E 5F 65 78 69 66 23 7F
0040h: FA C3 E3 00 00 02 3E 69 54 58 74 58 4D 4C 3A 63
0050h: 6F 6D 2E 61 64 6F 62 65 2E 78 6D 70 00 00 00 00
0060h: 00 3C 3F 78 70 61 63 6B 65 74 20 62 65 67 69 6E
0070h: 3D 27 EF BB BF 27 20 69 64 3D 27 57 35 4D 30 4D
0080h: 70 43 65 68 69 48 7A 72 65 53 7A 4E 54 63 7A 6B
0090h: 63 39 64 27 3F 3E 0A 3C 78 3A 78 6D 70 6D 65 74
00A0h: 61 20 78 6D 6C 6E 73 3A 78 3D 27 61 64 6F 62 65
00B0h: 3A 6E 73 3A 6D 65 74 61 2F 27 20 78 3A 78 6D 70
00C0h: 74 6B 3D 27 49 6D 61 67 65 3A 3A 45 78 69 66 54
00D0h: 6F 6F 6C 20 31 31 2E 39 38 27 3E 0A 3C 72 64 66
00E0h: 3A 52 44 46 20 78 6D 6C 6E 73 3A 72 64 66 3D 27
00F0h: 68 74 74 70 3A 2F 2F 77 77 77 2E 77 33 2E 6F 72
0100h: 67 2F 31 39 39 39 2F 30 32 2F 32 32 2D 72 64 66
0110h: 2D 73 79 6E 74 61 78 2D 6E 73 23 27 3E 0A 0A 20
0120h: 3C 72 64 66 3A 44 65 73 63 72 69 70 74 69 6F 6E
0130h: 20 72 64 66 3A 61 62 6F 75 74 3D 27 27 0A 20 20
0140h: 78 6D 6C 6E 73 3A 70 68 6F 74 6F 73 68 6F 70 3D
0150h: 27 68 74 74 70 3A 2F 2F 6E 73 2E 61 64 6F 62 65
0160h: 2E 63 6F 6D 2F 70 68 6F 74 6F 73 68 6F 70 2F 31
```

将0297修改为05b7，可以得到



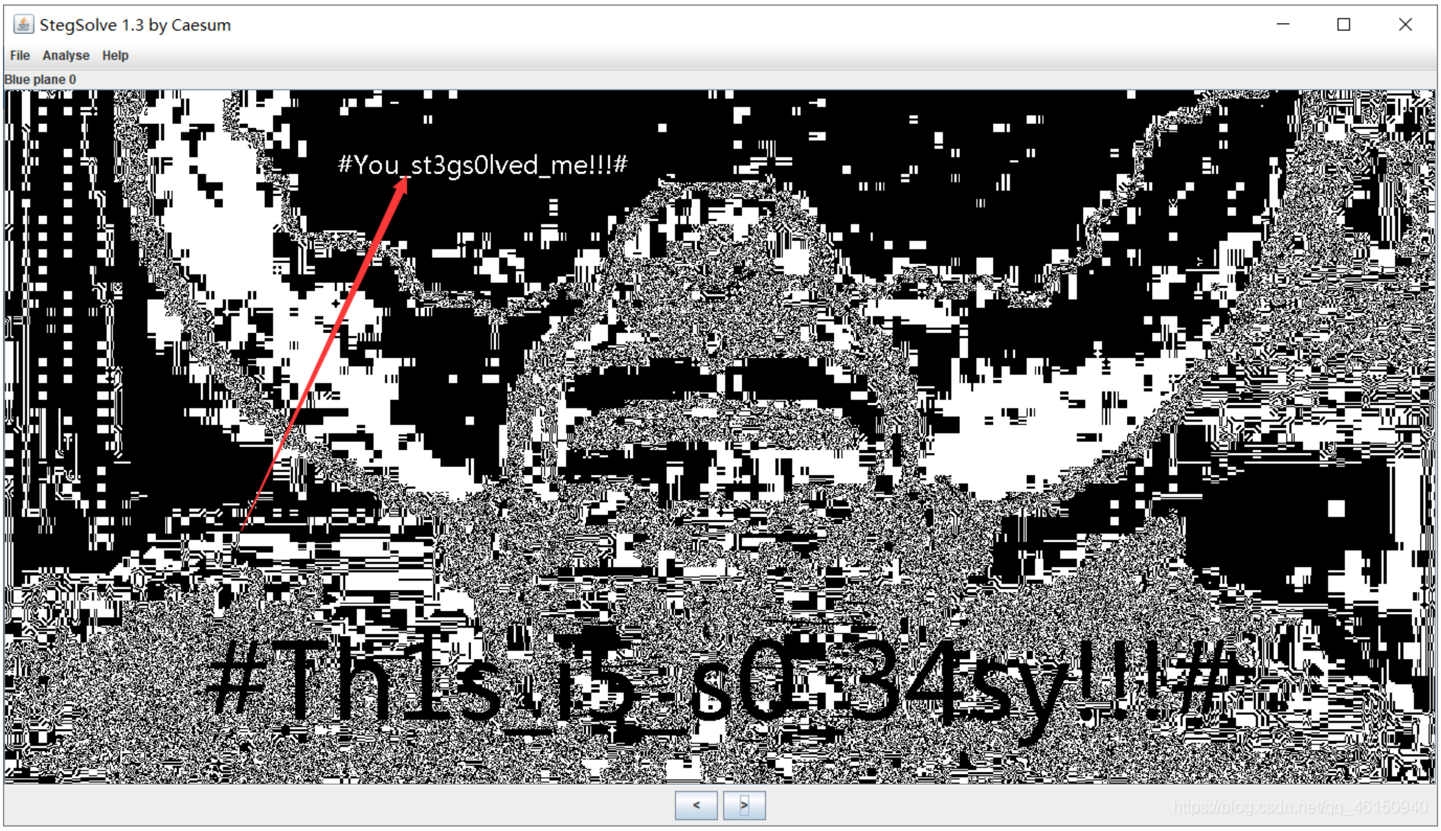
One PieNG 4

上面脚本计算出，i=1463，直接把高改为1463

#Th1s_i5_s0_34sy!!!#
#Pn9_He1gh7_6e_ch4ng3d#
#M4yb3_we_sh0uld_9o_d33per#

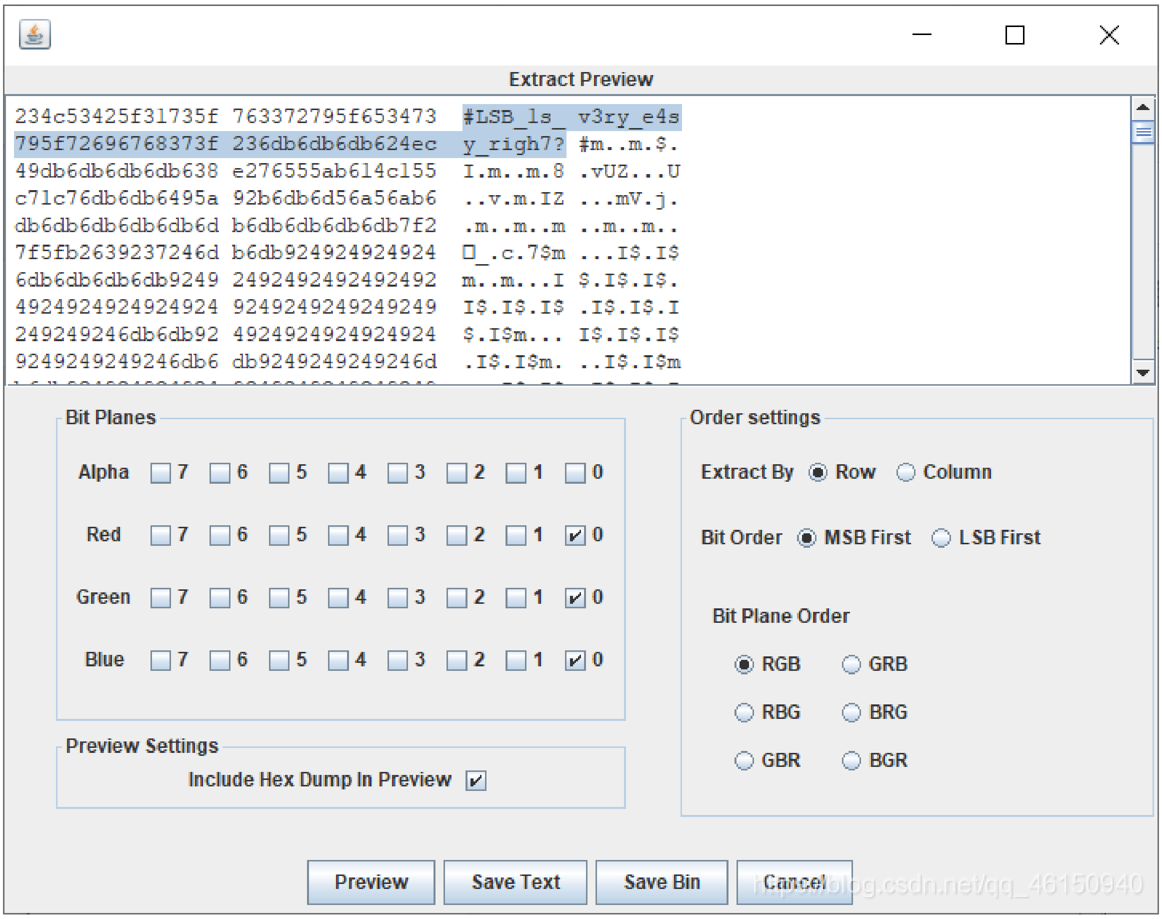
One PieNG 5

Blue通道最低位



One PieNG 6

使用stegsolve的data extract模块，除Alpha均勾选0通道



Extract Preview

```
234c53425f31735f 763372795f653473  #LSB_1s_ v3ry_e4s
795f72696768373f 236db6db6db624ec  y_righ7? #m..m.$.
49db6db6db6db638 e276555ab614c155  I.m..m.8 .vUZ...U
c71c76db6db6495a 92b6db6d56a56ab6  ..v.m.IZ ...mV.j.
db6db6db6db6db6d b6db6db6db6db7f2  .m..m..m ..m..m..
7f5fb2639237246d b6db924924924924  □_.c.7$m ...I$.I$
6db6db6db6db9249 2492492492492492  m..m...I $.I$.I$.
4924924924924924 9249249249249249  I$.I$.I$ .I$.I$.I
249249246db6db92 4924924924924924  $.I$m... I$.I$.I$
9249249249246db6 db9249249249246d  .I$.I$m. ..I$.I$m
```

Bit Planes

Alpha	□ 7	□ 6	□ 5	□ 4	□ 3	□ 2	□ 1	□ 0
Red	□ 7	□ 6	□ 5	□ 4	□ 3	□ 2	□ 1	☑ 0
Green	□ 7	□ 6	□ 5	□ 4	□ 3	□ 2	□ 1	☑ 0
Blue	□ 7	□ 6	□ 5	□ 4	□ 3	□ 2	□ 1	☑ 0

Preview Settings

Include Hex Dump In Preview ☑

Order settings

Extract By ● Row ○ Column

Bit Order ● MSB First ○ LSB First

Bit Plane Order

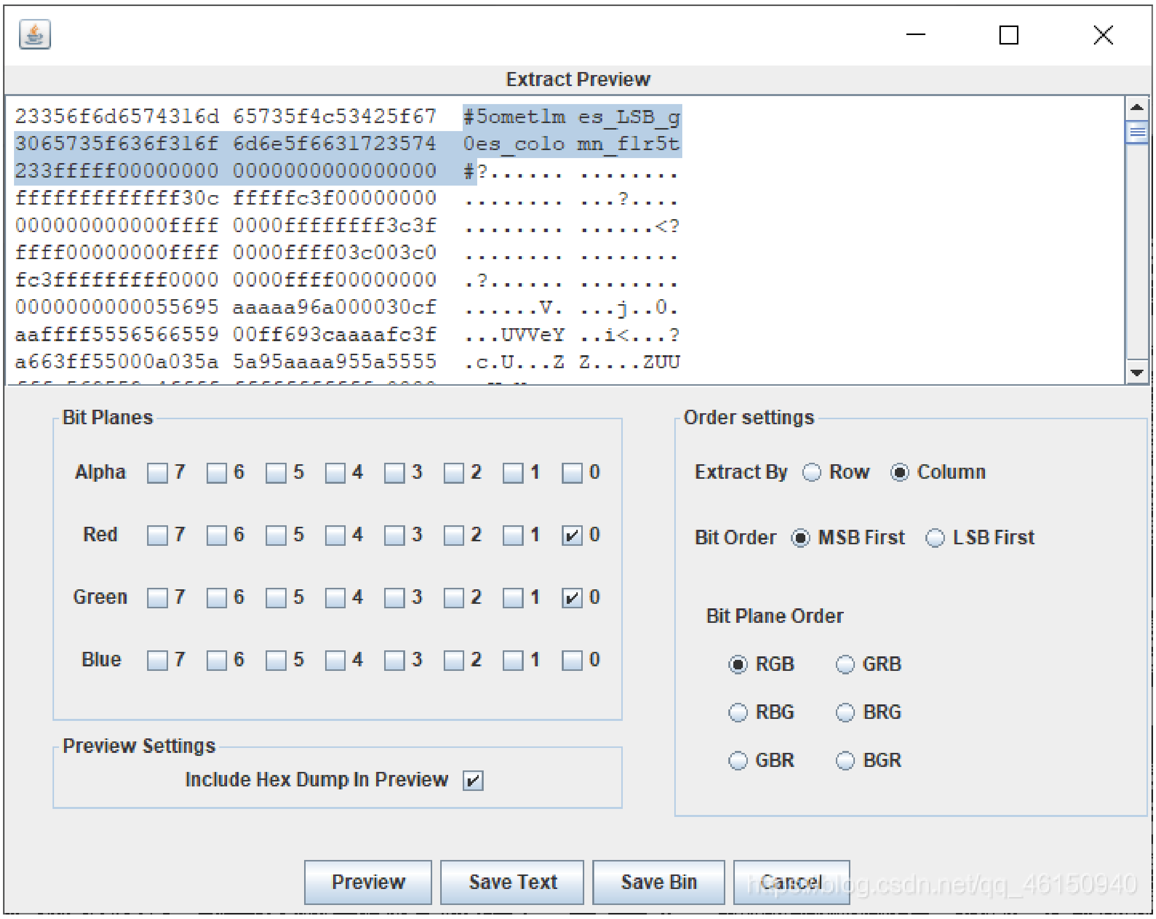
● RGB ○ GRB

○ RBG ○ BRG

○ GBR ○ BGR

One PieNG 7

勾选Red通道和Green通道的最低位0，然后选择Column



```
23356f6d6574316d 65735f4c53425f67  #5omet1m es_LSB_g
3065735f636f316f 6d6e5f6631723574  0es_co1o mn_f1r5t
233fffff00000000 0000000000000000  #?...... ........
fffffffffffff30c fffffc3f00000000  ........ ...?....
000000000000ffff 0000ffffffff3c3f  ........ ......<?
ffff00000000ffff 0000ffff03c003c0  ........ ........
fc3fffffffff0000 0000ffff00000000  .?...... ........
0000000000055695 aaaaa96a000030cf  ......V. ...j..0.
aaffff5556566559 00ff693caaaafc3f  ...UVVeY ..i<...?
a663ff55000a035a 5a95aaaa955a5555  .c.U...Z Z....ZUU
```

Bit Planes

Alpha	□ 7	□ 6	□ 5	□ 4	□ 3	□ 2	□ 1	□ 0
Red	□ 7	□ 6	□ 5	□ 4	□ 3	□ 2	□ 1	☑ 0
Green	□ 7	□ 6	□ 5	□ 4	□ 3	□ 2	□ 1	☑ 0
Blue	□ 7	□ 6	□ 5	□ 4	□ 3	□ 2	□ 1	□ 0

Preview Settings

Include Hex Dump In Preview ☑

Order settings

Extract By ○ Row ● Column

Bit Order ● MSB First ○ LSB First

Bit Plane Order

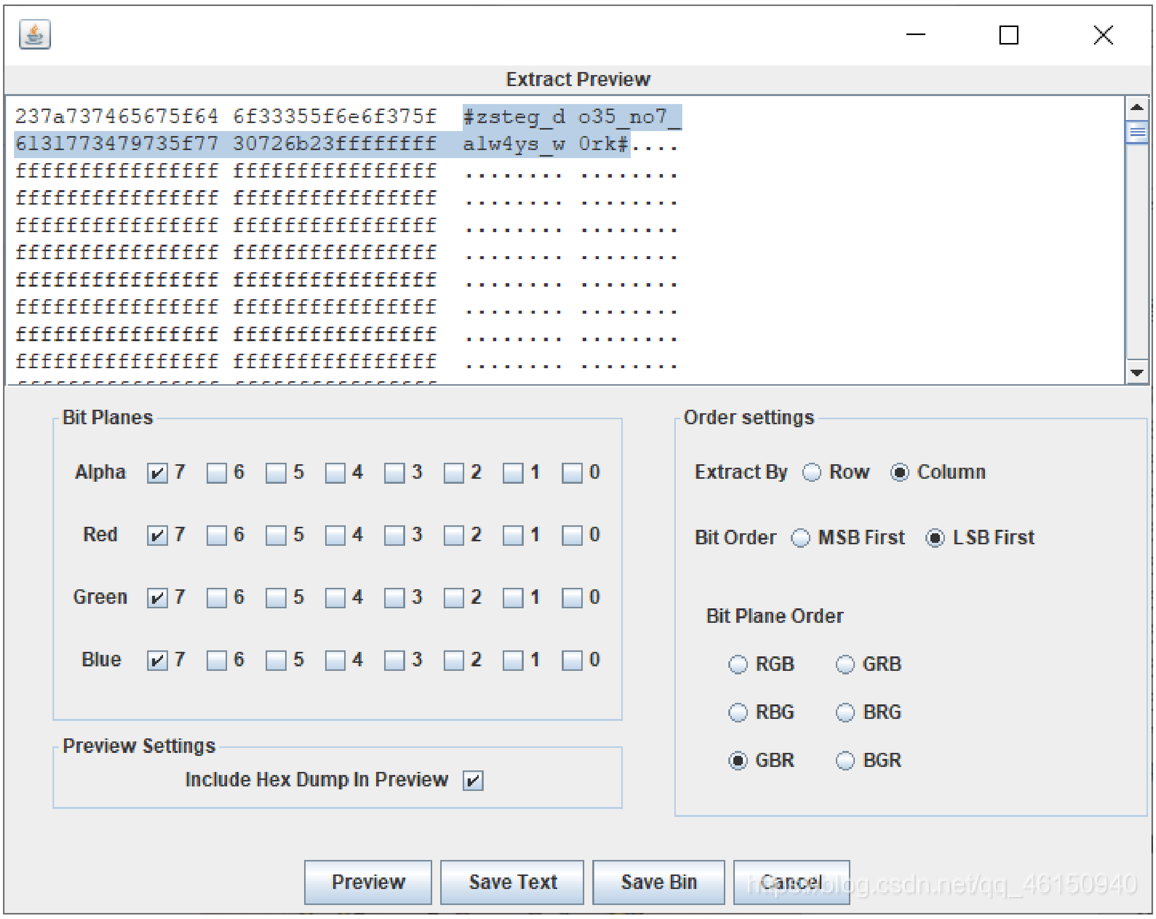
● RGB ○ GRB

○ RBG ○ BRG

○ GBR ○ BGR

One PieNG 8

全部勾选最高位，选择GBR



Extract Preview

```
237a737465675f64 6f33355f6e6f375f  #zsteg_d o35_no7_
6131773479735f77 30726b23ffffffff  alw4ys_w 0rk#....
ffffffffffffffff ffffffffffffffff  ........ ........
ffffffffffffffff ffffffffffffffff  ........ ........
ffffffffffffffff ffffffffffffffff  ........ ........
ffffffffffffffff ffffffffffffffff  ........ ........
ffffffffffffffff ffffffffffffffff  ........ ........
ffffffffffffffff ffffffffffffffff  ........ ........
ffffffffffffffff ffffffffffffffff  ........ ........
ffffffffffffffff ffffffffffffffff  ........ ........
```

Bit Planes

Alpha ☑ 7 □ 6 □ 5 □ 4 □ 3 □ 2 □ 1 □ 0

Red ☑ 7 □ 6 □ 5 □ 4 □ 3 □ 2 □ 1 □ 0

Green ☑ 7 □ 6 □ 5 □ 4 □ 3 □ 2 □ 1 □ 0

Blue ☑ 7 □ 6 □ 5 □ 4 □ 3 □ 2 □ 1 □ 0

Preview Settings

Include Hex Dump In Preview ☑

Order settings

Extract By ○ Row ● Column

Bit Order ○ MSB First ● LSB First

Bit Plane Order

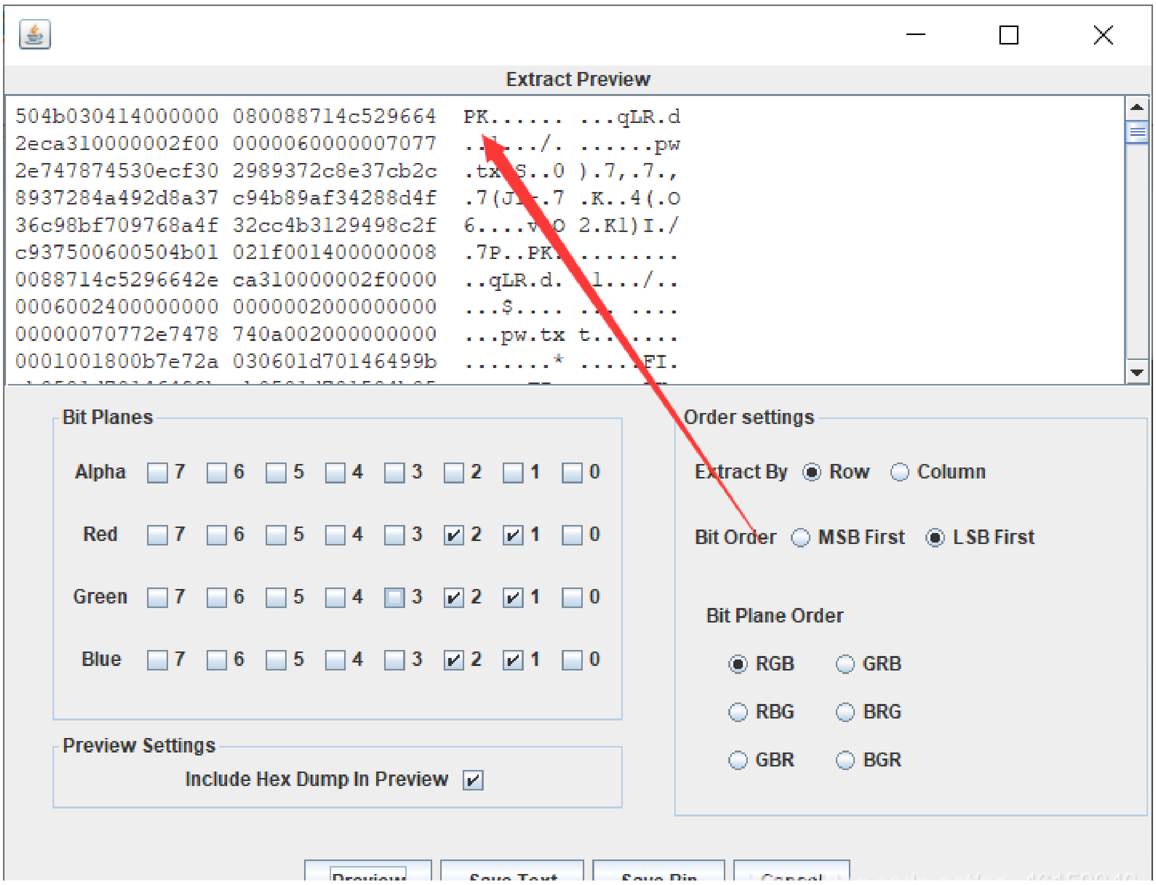
○ RGB ○ GRB

○ RBG ○ BRG

● GBR ○ BGR

One PieNG 9

除Alpha通道外均勾选1、2位，发现PK压缩包，另存为a.zip



Extract Preview

```
504b030414000000 080088714c529664  PK...... ...qLR.d
2eca310000002f00 0000060000007077  ..1.../. ......pw
2e747874530ecf30 2989372c8e37cb2c  .txtS..0 ).7,.7.,
8937284a492d8a37 c94b89af34288d4f  .7(JI-.7 .K..4(.O
36c98bf709768a4f 32cc4b3129498c2f  6....v.O 2.K1)I./
c937500600504b01 021f001400000008  .7P..PK. ........
0088714c5296642e ca310000002f0000  ..qLR.d. .1.../..
0006002400000000 0000002000000000  ...$.... ... ....
00000070772e7478 740a002000000000  ...pw.tx t.. ....
0001001800b7e72a 030601d70146499b  .......* .....FI.
```

Bit Planes

Alpha □ 7 □ 6 □ 5 □ 4 □ 3 □ 2 □ 1 □ 0

Red □ 7 □ 6 □ 5 □ 4 □ 3 ☑ 2 ☑ 1 □ 0

Green □ 7 □ 6 □ 5 □ 4 □ 3 ☑ 2 ☑ 1 □ 0

Blue □ 7 □ 6 □ 5 □ 4 □ 3 ☑ 2 ☑ 1 □ 0

Preview Settings

Include Hex Dump In Preview ☑

Order settings

Extract By ● Row ○ Column

Bit Order ○ MSB First ● LSB First

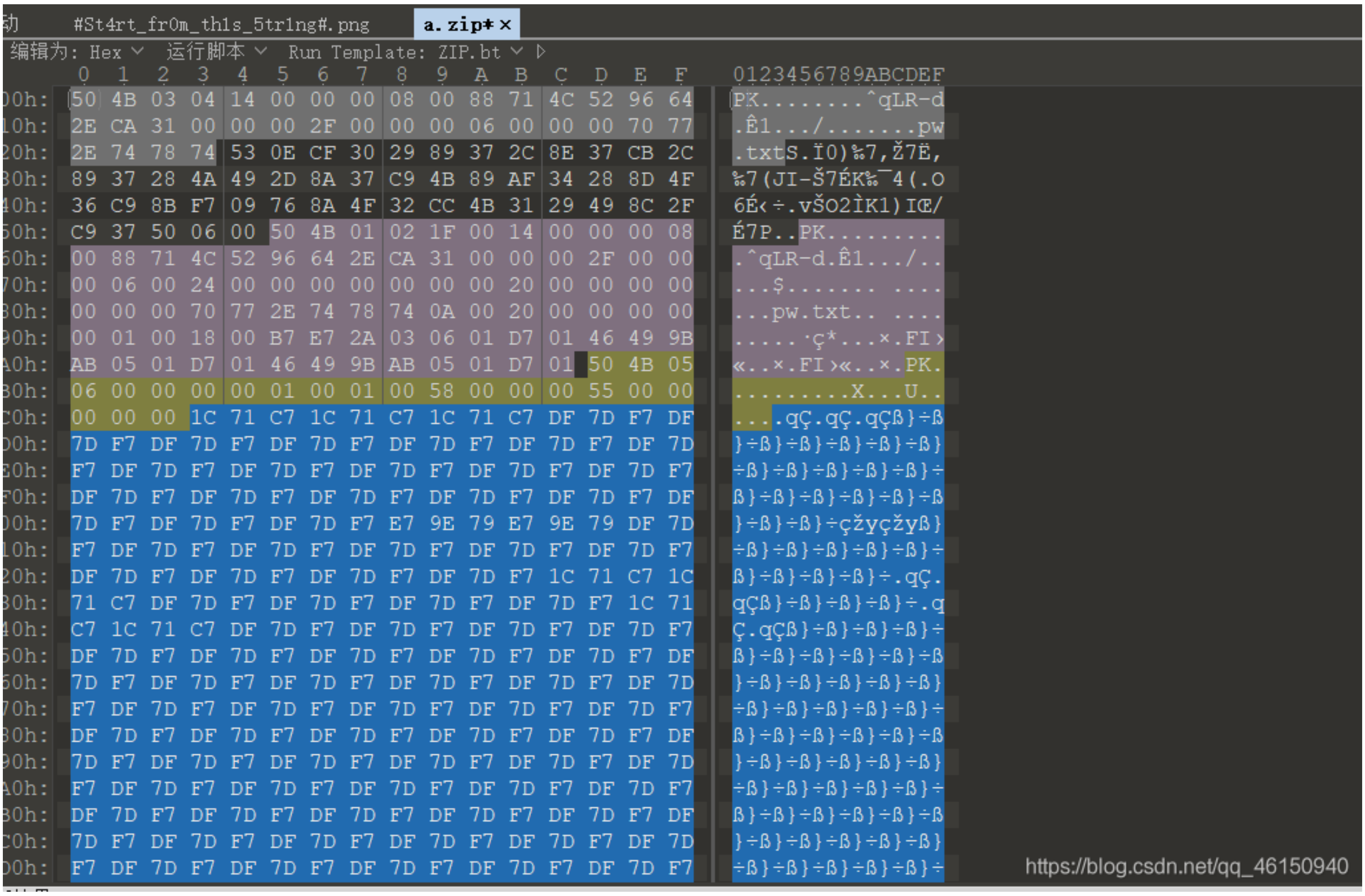
Bit Plane Order

● RGB ○ GRB

○ RBG ○ BRG

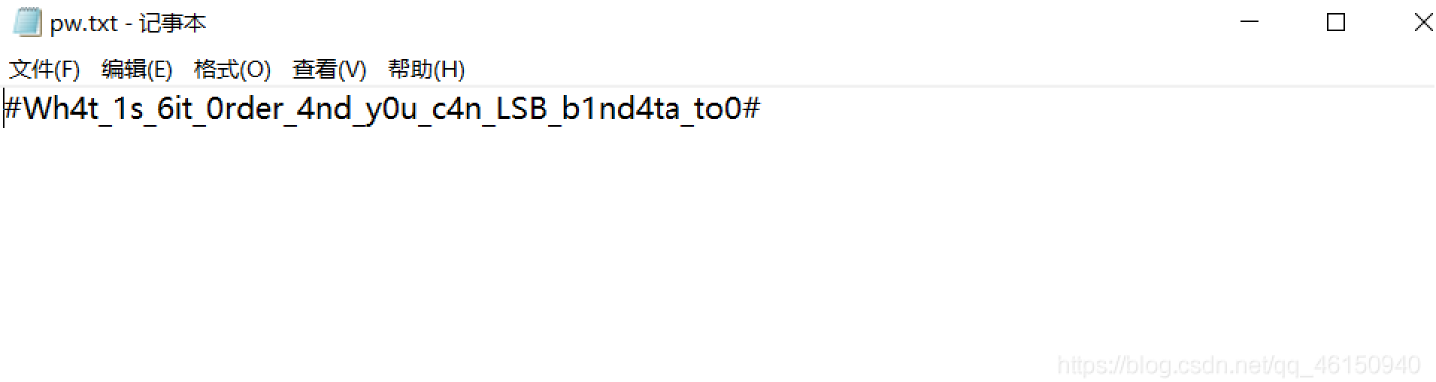
○ GBR ○ BGR

删去多余的部分



```
       0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F    0123456789ABCDEF
 0h:  50 4B 03 04 14 00 00 00 08 00 88 71 4C 52 96 64    PK........^qLR-d
10h:  2E CA 31 00 00 00 2F 00 00 00 06 00 00 00 70 77    .Ê1.../.......pw
20h:  2E 74 78 74 53 0E CF 30 29 89 37 2C 8E 37 CB 2C    .txtS.Ï0)‰7,Ž7Ë,
30h:  89 37 28 4A 49 2D 8A 37 C9 4B 89 AF 34 28 8D 4F    ‰7(JI-Š7ÉK‰¯4(.O
40h:  36 C9 8B F7 09 76 8A 4F 32 CC 4B 31 29 49 8C 2F    6É‹÷.vŠO2ÌK1)IŒ/
50h:  C9 37 50 06 00 50 4B 01 02 1F 00 14 00 00 00 08    É7P..PK.........
60h:  00 88 71 4C 52 96 64 2E CA 31 00 00 00 2F 00 00    .^qLR-d.Ê1.../..
70h:  00 06 00 24 00 00 00 00 00 00 00 20 00 00 00 00    ...$....... ....
80h:  00 00 00 70 77 2E 74 78 74 0A 00 20 00 00 00 00    ...pw.txt.. ....
90h:  00 01 00 18 00 B7 E7 2A 03 06 01 D7 01 46 49 9B    .....·ç*...×.FI›
A0h:  AB 05 01 D7 01 46 49 9B AB 05 01 D7 01 50 4B 05    «..×.FI›«..×.PK.
B0h:  06 00 00 00 00 01 00 01 00 58 00 00 00 55 00 00    .........X...U..
C0h:  00 00 00 1C 71 C7 1C 71 C7 1C 71 C7 DF 7D F7 DF    ....qÇ.qÇ.qÇß}÷ß
D0h:  7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D    }÷ß}÷ß}÷ß}÷ß}÷ß}
E0h:  F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7    ÷ß}÷ß}÷ß}÷ß}÷ß}÷
F0h:  DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF    ß}÷ß}÷ß}÷ß}÷ß}÷ß
00h:  7D F7 DF 7D F7 DF 7D F7 E7 9E 79 E7 9E 79 DF 7D    }÷ß}÷ß}÷çžyçžyß}
10h:  F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7    ÷ß}÷ß}÷ß}÷ß}÷ß}÷
20h:  DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 1C 71 C7 1C    ß}÷ß}÷ß}÷ß}÷.qÇ.
30h:  71 C7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 1C 71    qÇß}÷ß}÷ß}÷ß}÷.q
40h:  C7 1C 71 C7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7    Ç.qÇß}÷ß}÷ß}÷ß}÷
50h:  DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF    ß}÷ß}÷ß}÷ß}÷ß}÷ß
60h:  7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D    }÷ß}÷ß}÷ß}÷ß}÷ß}
70h:  F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7    ÷ß}÷ß}÷ß}÷ß}÷ß}÷
80h:  DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF    ß}÷ß}÷ß}÷ß}÷ß}÷ß
90h:  7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D    }÷ß}÷ß}÷ß}÷ß}÷ß}
A0h:  F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7    ÷ß}÷ß}÷ß}÷ß}÷ß}÷
B0h:  DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF    ß}÷ß}÷ß}÷ß}÷ß}÷ß
C0h:  7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D    }÷ß}÷ß}÷ß}÷ß}÷ß}
D0h:  F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7 DF 7D F7    ÷ß}÷ß}÷ß}÷ß}÷ß}÷
```

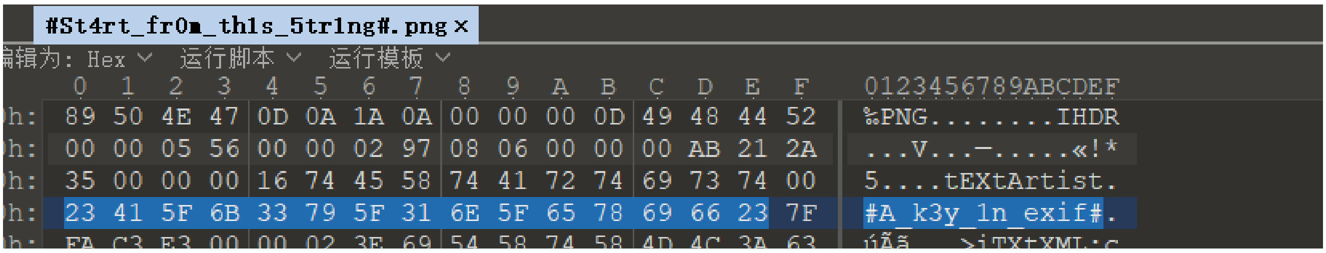
解压压缩包得到



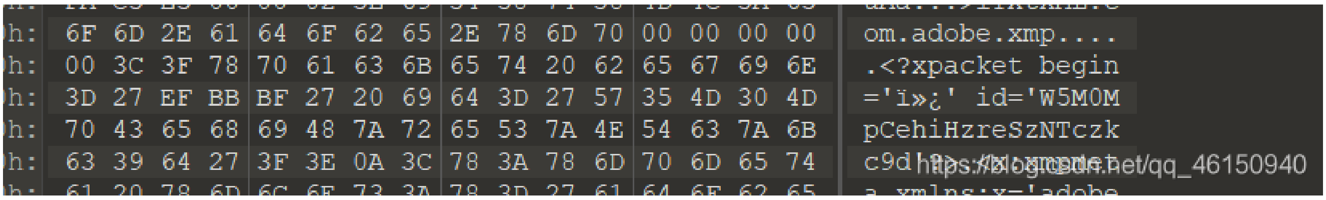
#Wh4t_1s_6it_0rder_4nd_y0u_c4n_LSB_b1nd4ta_to0#

One PieNG 10

使用010 editor查看十六进制

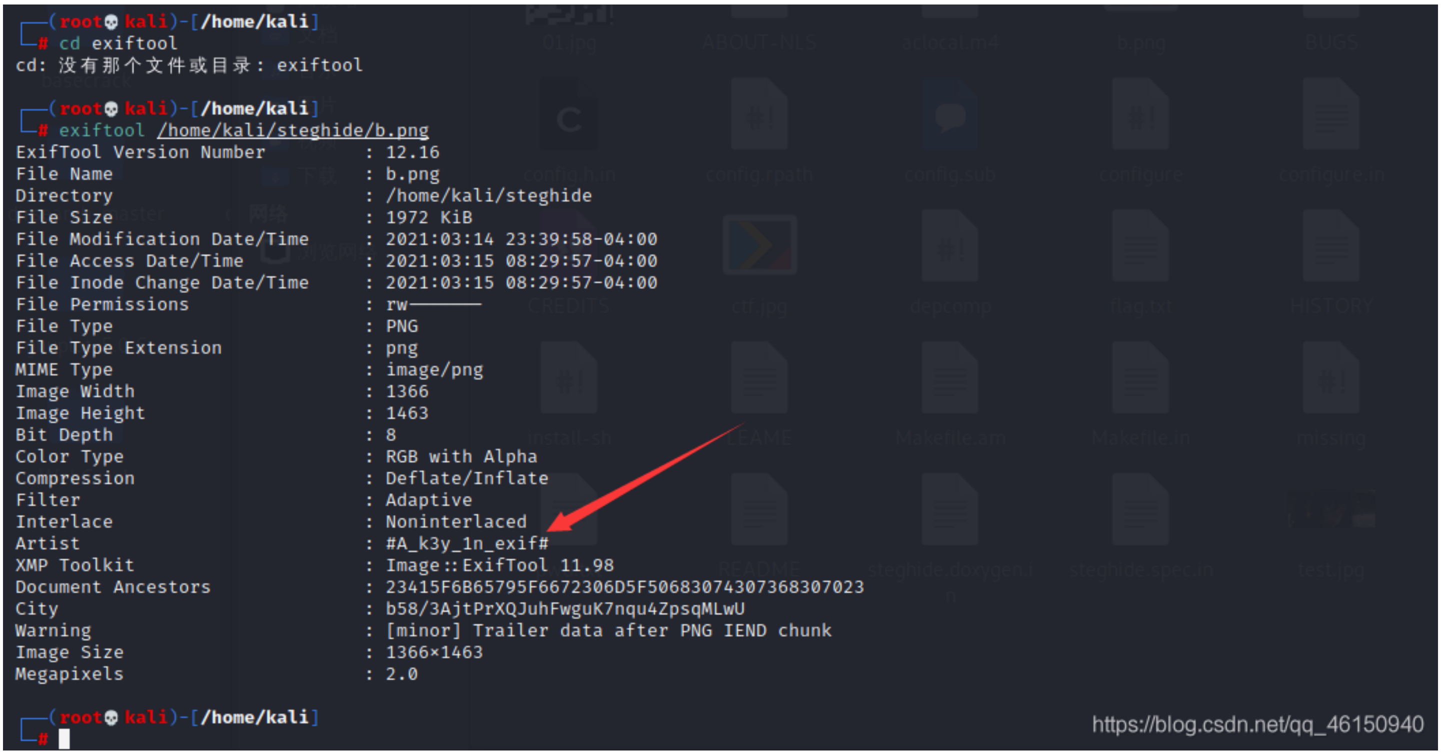


```
     0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F    0123456789ABCDEF
h:  89 50 4E 47 0D 0A 1A 0A 00 00 00 0D 49 48 44 52    ‰PNG........IHDR
h:  00 00 05 56 00 00 02 97 08 06 00 00 00 AB 21 2A    ...V...—.....«!*
h:  35 00 00 00 16 74 45 58 74 41 72 74 69 73 74 00    5....tEXtArtist.
h:  23 41 5F 6B 33 79 5F 31 6E 5F 65 78 69 66 23 7F    #A_k3y_1n_exif#.
h:  FA C3 E3 00 00 02 3E 69 54 58 74 58 4D 4C 3A 63    úÃã..>iTXtXML:c
```



```
h:  6F 6D 2E 61 64 6F 62 65 2E 78 6D 70 00 00 00 00   om.adobe.xmp....
h:  00 3C 3F 78 70 61 63 6B 65 74 20 62 65 67 69 6E   .<?xpacket begin
h:  3D 27 EF BB BF 27 20 69 64 3D 27 57 35 4D 30 4D   ='ï»¿' id='W5M0M
h:  70 43 65 68 69 48 7A 72 65 53 7A 4E 54 63 7A 6B   pCehiHzreSzNTczk
h:  63 39 64 27 3F 3E 0A 3C 78 3A 78 6D 70 6D 65 74   c9d'?>.<x:xmpmet
```

使用exiftool工具也可以得到

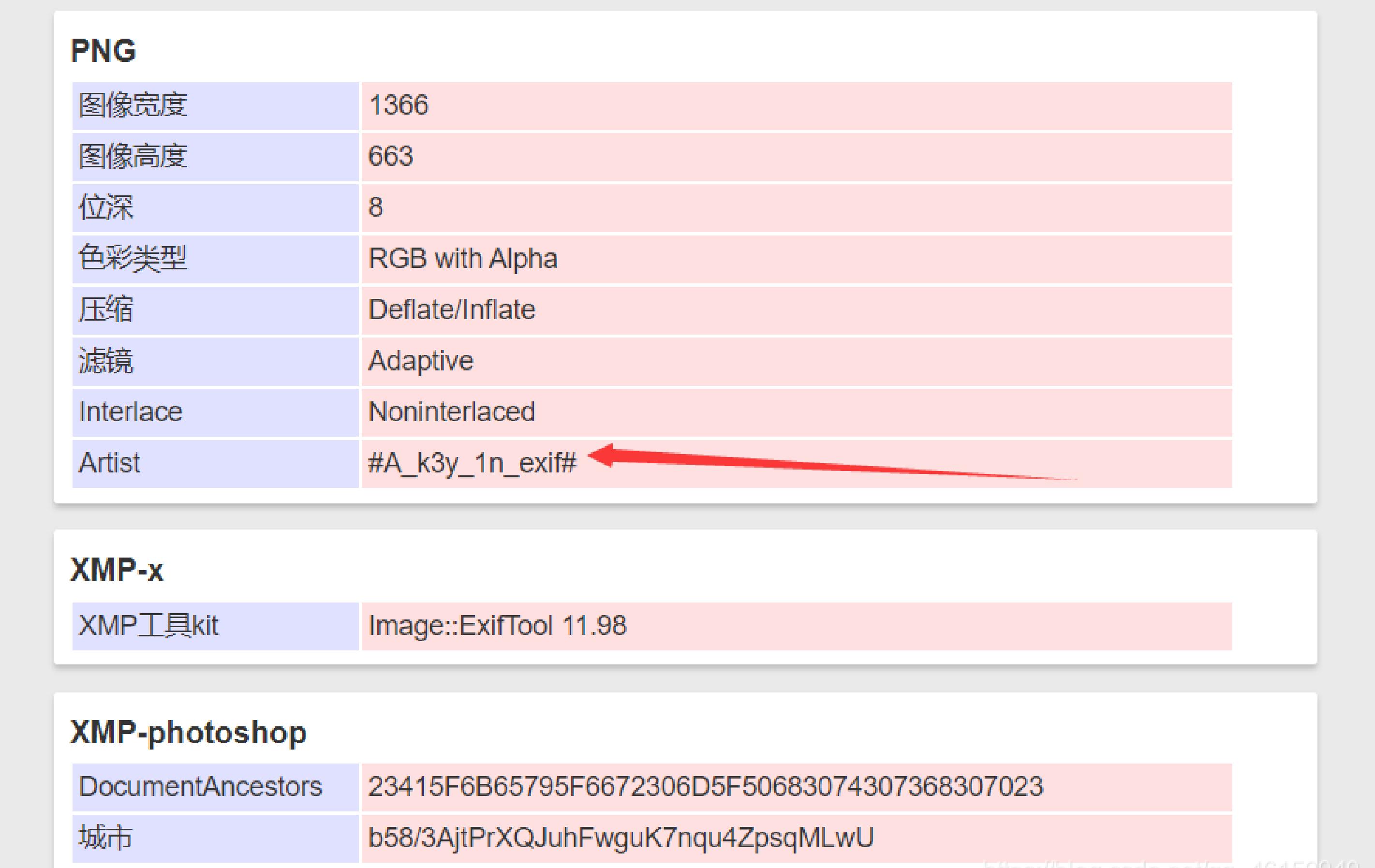


```
┌──(root💀kali)-[/home/kali]
└─# cd exiftool
cd: 没有那个文件或目录: exiftool

┌──(root💀kali)-[/home/kali]
└─# exiftool /home/kali/steghide/b.png
ExifTool Version Number         : 12.16
File Name                       : b.png
Directory                       : /home/kali/steghide
File Size                       : 1972 KiB
File Modification Date/Time     : 2021:03:14 23:39:58-04:00
File Access Date/Time           : 2021:03:15 08:29:57-04:00
File Inode Change Date/Time     : 2021:03:15 08:29:57-04:00
File Permissions                : rw-------
File Type                       : PNG
File Type Extension             : png
MIME Type                       : image/png
Image Width                     : 1366
Image Height                    : 1463
Bit Depth                       : 8
Color Type                      : RGB with Alpha
Compression                     : Deflate/Inflate
Filter                          : Adaptive
Interlace                       : Noninterlaced
Artist                          : #A_k3y_1n_exif#
XMP Toolkit                     : Image::ExifTool 11.98
Document Ancestors              : 23415F6B65795F6672306D5F50683074307368307023
City                            : b58/3AjtPrXQJuhFwguK7nqu4ZpsqMLwU
Warning                         : [minor] Trailer data after PNG IEND chunk
Image Size                      : 1366x1463
Megapixels                      : 2.0

┌──(root💀kali)-[/home/kali]
└─# 
```

也可以使用在线EXIF查看器



PNG

图像宽度	1366
图像高度	663
位深	8
色彩类型	RGB with Alpha
压缩	Deflate/Inflate
滤镜	Adaptive
Interlace	Noninterlaced
Artist	#A_k3y_1n_exif#

XMP-x

XMP工具kit	Image::ExifTool 11.98

XMP-photoshop

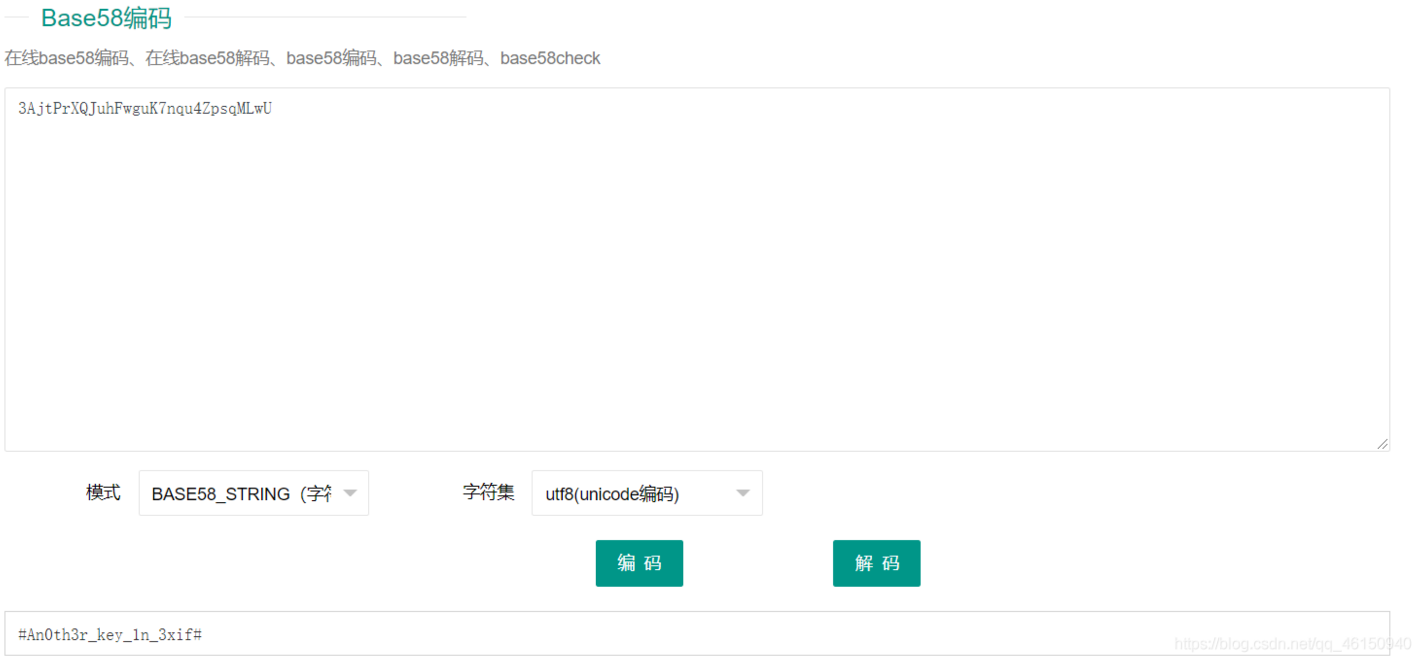
DocumentAncestors	23415F6B65795F6672306D5F50683074307368307023
城市	b58/3AjtPrXQJuhFwguK7nqu4ZpsqMLwU

One PieNG 11

上面使用在线EXIF查看器，可以发现DocumentAncestors栏有可疑字符串

```
b58/3AjtPrXQJuhFwguK7nqu4ZpsqMLwU
```

Base58解码/后面的内容


Base58编码
在线base58编码、在线base58解码、base58编码、base58解码、base58check
编 码
解 码
https://blog.csdn.net/qq_46150940

```
3AjtPrXQJuhFwguK7nqu4ZpsqMLwU
```

模式 BASE58_STRING（字

字符集 utf8(unicode编码)

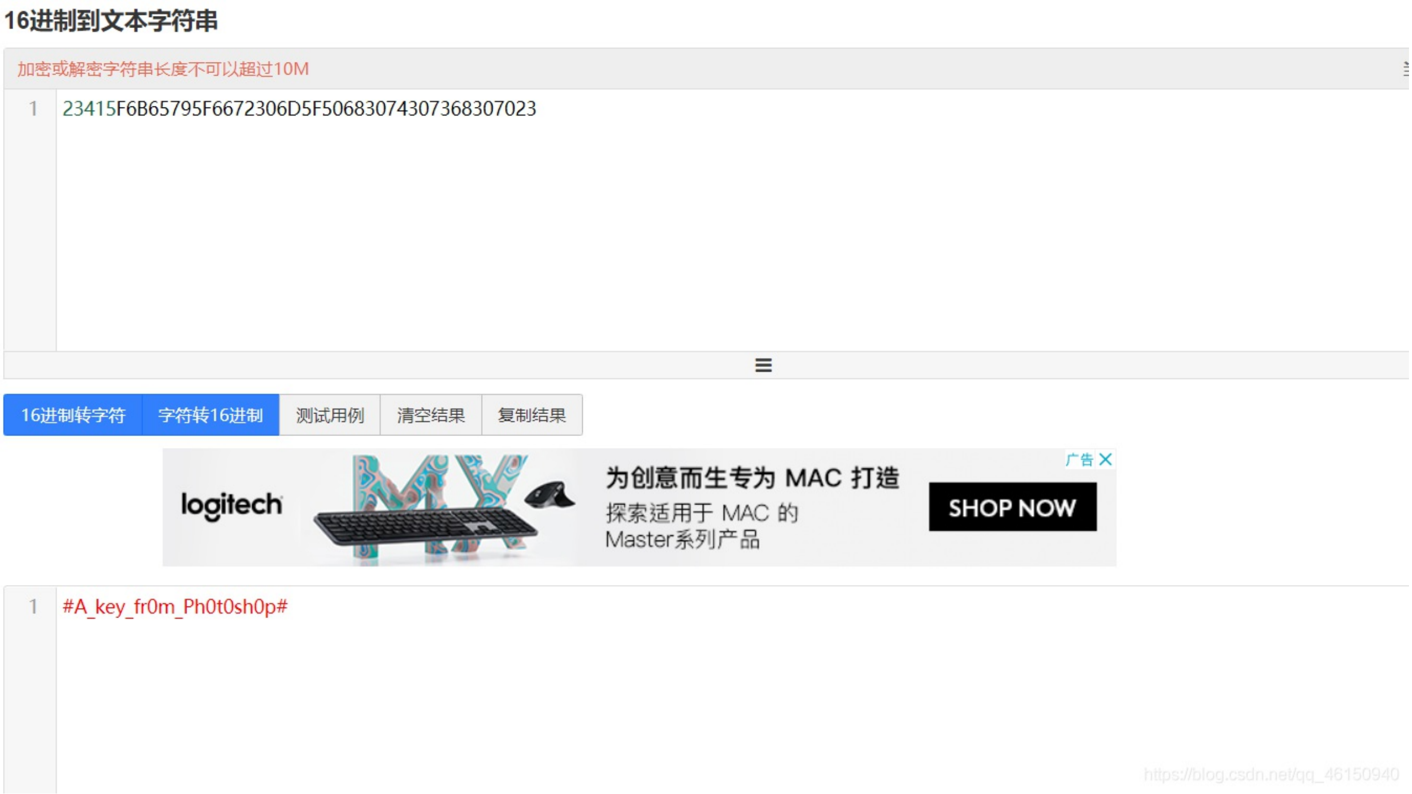
```
#An0th3r_key_1n_3xif#
```

One PieNG 12

同样city一栏中有十六进制字符串

```
23415F6B65795F6672306D5F50683074307368307023
```

16进制转字符串



16进制到文本字符串

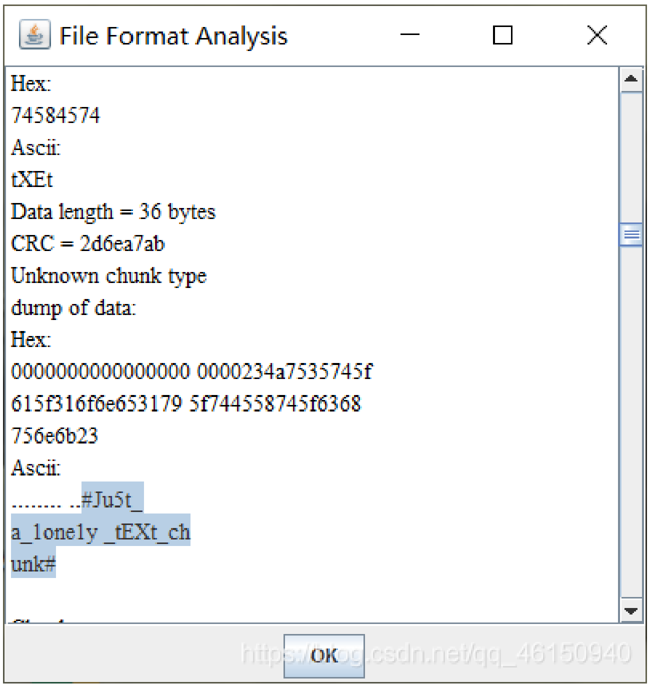
加密或解密字符串长度不可以超过10M

```
23415F6B65795F6672306D5F50683074307368307023
```

```
#A_key_fr0m_Ph0t0sh0p#
```

One PieNG 13

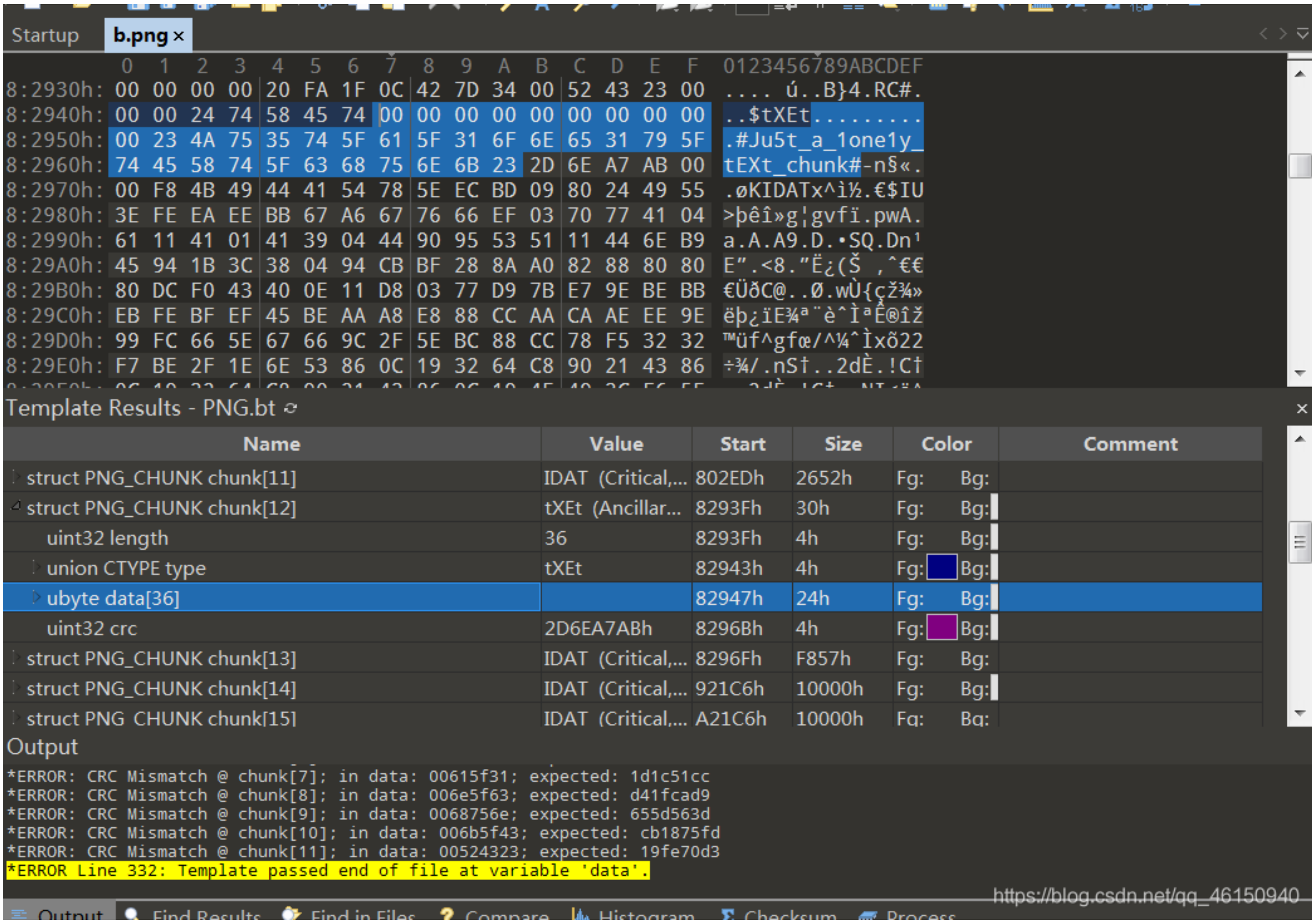
在stegsolve主页面，选择File Format



```
Hex:
74584574
Ascii:
tXEt
Data length = 36 bytes
CRC = 2d6ea7ab
Unknown chunk type
dump of data:
Hex:
0000000000000000 0000234a7535745f
615f316f6e653179 5f744558745f6368
756e6b23
Ascii:
........ ..#Ju5t_
a_1one1y _tEXt_ch
unk#
```

套神的方法：

010查看变量窗口（打开方式：视图–检查器窗口–变量，需要下载png模板，点击模板–模板储存库–png模板）



```
           0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F   0123456789ABCDEF
8:2930h: 00 00 00 00 20 FA 1F 0C 42 7D 34 00 52 43 23 00  .... ú..B}4.RC#.
8:2940h: 00 00 24 74 58 45 74 00 00 00 00 00 00 00 00 00  ..$tXEt.........
8:2950h: 00 23 4A 75 35 74 5F 61 5F 31 6F 6E 65 31 79 5F  .#Ju5t_a_1one1y_
8:2960h: 74 45 58 74 5F 63 68 75 6E 6B 23 2D 6E A7 AB 00  tEXt_chunk#-n§«.
8:2970h: 00 F8 4B 49 44 41 54 78 5E EC BD 09 80 24 49 55  .øKIDATx^ì½.€$IU
8:2980h: 3E FE EA EE BB 67 A6 67 76 66 EF 03 70 77 41 04  >þêî»g¦gvfï.pwA.
8:2990h: 61 11 41 01 41 39 04 44 90 95 53 51 11 44 6E B9  a.A.A9.D.•SQ.Dn¹
8:29A0h: 45 94 1B 3C 38 04 94 CB BF 28 8A A0 82 88 80 80  E”.<8.”Ë¿(Š ‚ˆ€€
8:29B0h: 80 DC F0 43 40 0E 11 D8 03 77 D9 7B E7 9E BE BB  €ÜðC@..Ø.wÙ{çž¾»
8:29C0h: EB FE BF EF 45 BE AA A8 E8 88 CC AA CA AE EE 9E  ëþ¿ïE¾ª¨èˆÌªÊ®îž
8:29D0h: 99 FC 66 5E 67 66 9C 2F 5E BC 88 CC 78 F5 32 32  ™üf^gfœ/^¼ˆÌxõ22
8:29E0h: F7 BE 2F 1E 6E 53 86 0C 19 32 64 C8 90 21 43 86  ÷¾/.nS†..2dÈ.!C†
```

Template Results - PNG.bt

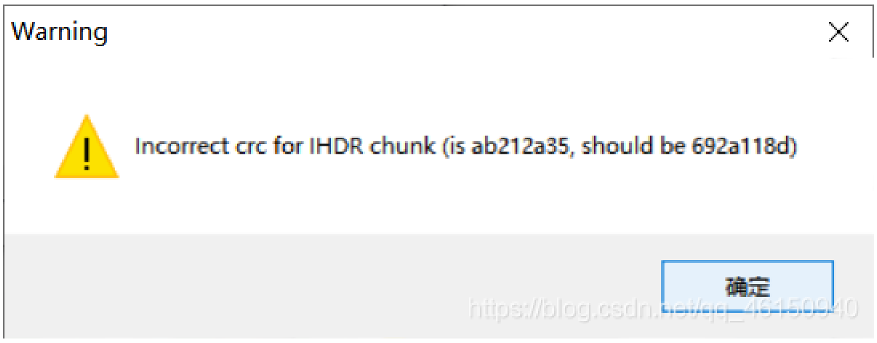
Name	Value	Start	Size	Color	Comment
struct PNG_CHUNK chunk[11]	IDAT (Critical,...	802EDh	2652h	Fg: Bg:	
struct PNG_CHUNK chunk[12]	tXEt (Ancillar...	8293Fh	30h	Fg: Bg:	
uint32 length	36	8293Fh	4h	Fg: Bg:	
union CTYPE type	tXEt	82943h	4h	Fg: Bg:	
ubyte data[36]		82947h	24h	Fg: Bg:	
uint32 crc	2D6EA7ABh	8296Bh	4h	Fg: Bg:	
struct PNG_CHUNK chunk[13]	IDAT (Critical,...	8296Fh	F857h	Fg: Bg:	
struct PNG_CHUNK chunk[14]	IDAT (Critical,...	921C6h	10000h	Fg: Bg:	
struct PNG_CHUNK chunk[15]	IDAT (Critical,...	A21C6h	10000h	Fg: Bg:	

Output

```
*ERROR: CRC Mismatch @ chunk[7]; in data: 00615f31; expected: 1d1c51cc
*ERROR: CRC Mismatch @ chunk[8]; in data: 006e5f63; expected: d41fcad9
*ERROR: CRC Mismatch @ chunk[9]; in data: 0068756e; expected: 655d563d
*ERROR: CRC Mismatch @ chunk[10]; in data: 006b5f43; expected: cb1875fd
*ERROR: CRC Mismatch @ chunk[11]; in data: 00524323; expected: 19fe70d3
*ERROR Line 332: Template passed end of file at variable 'data'.
```

One PieNG 14

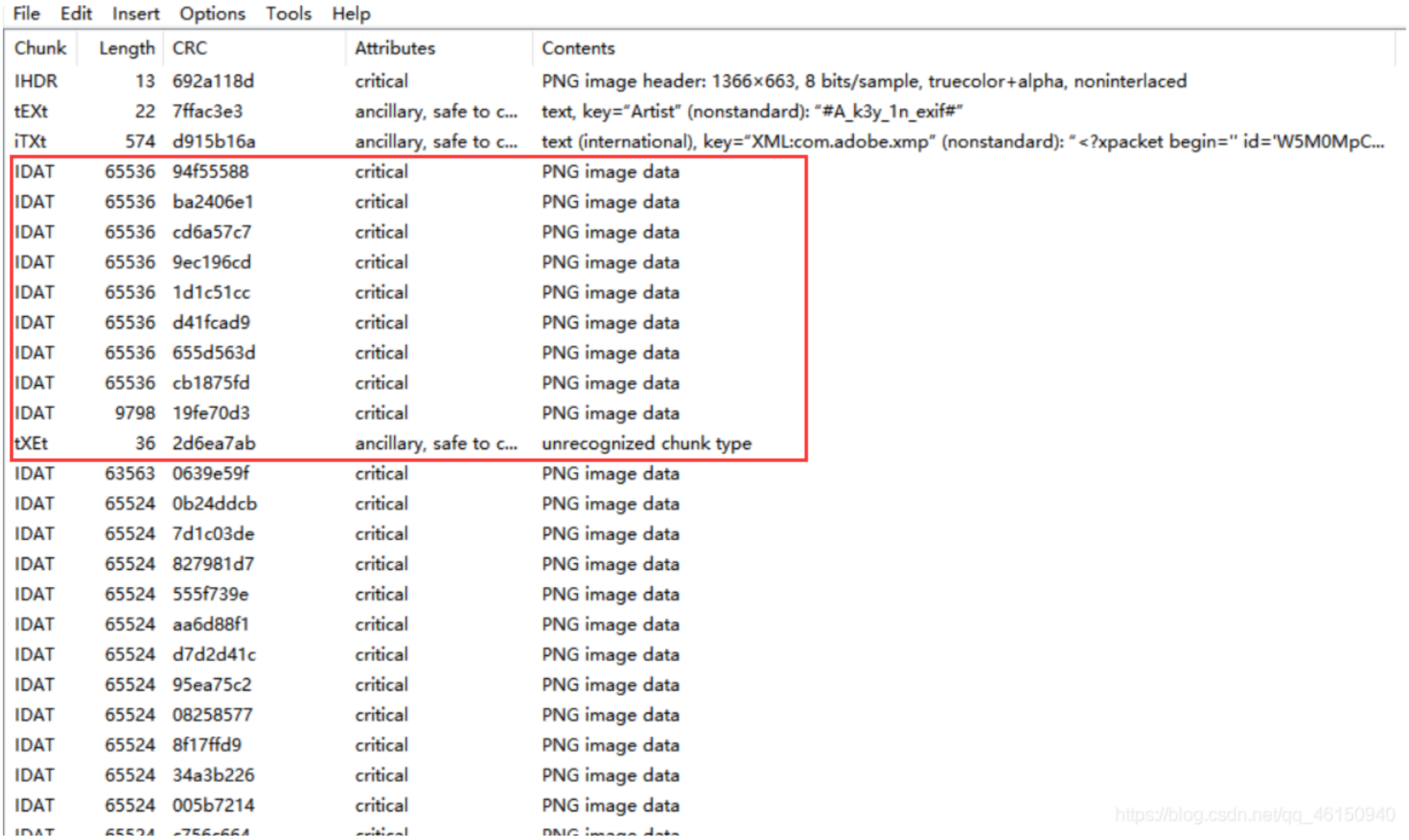
使用tweakpng工具打开图片，连着十个警告



Warning

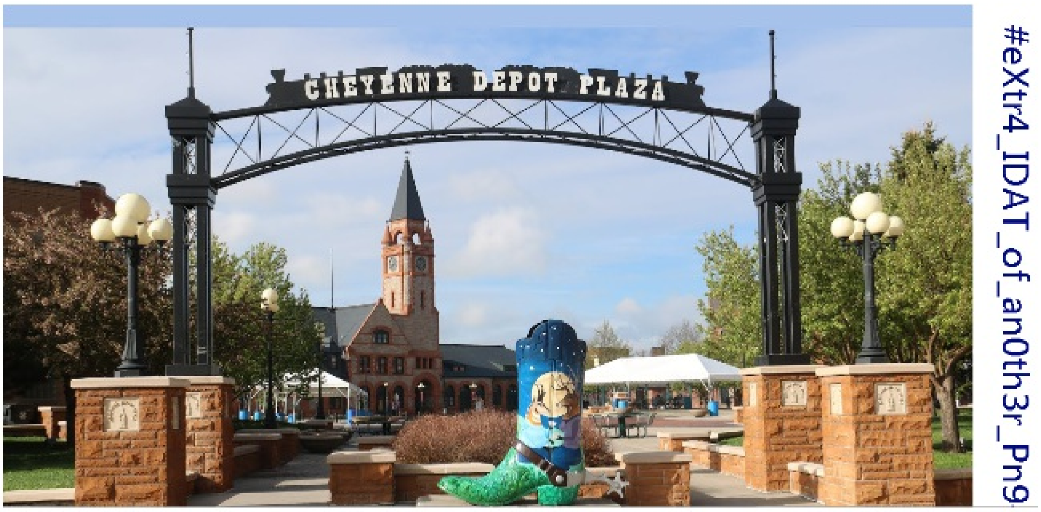
Incorrect crc for IHDR chunk (is ab212a35, should be 692a118d)

删去九个IDAT以及tXEt



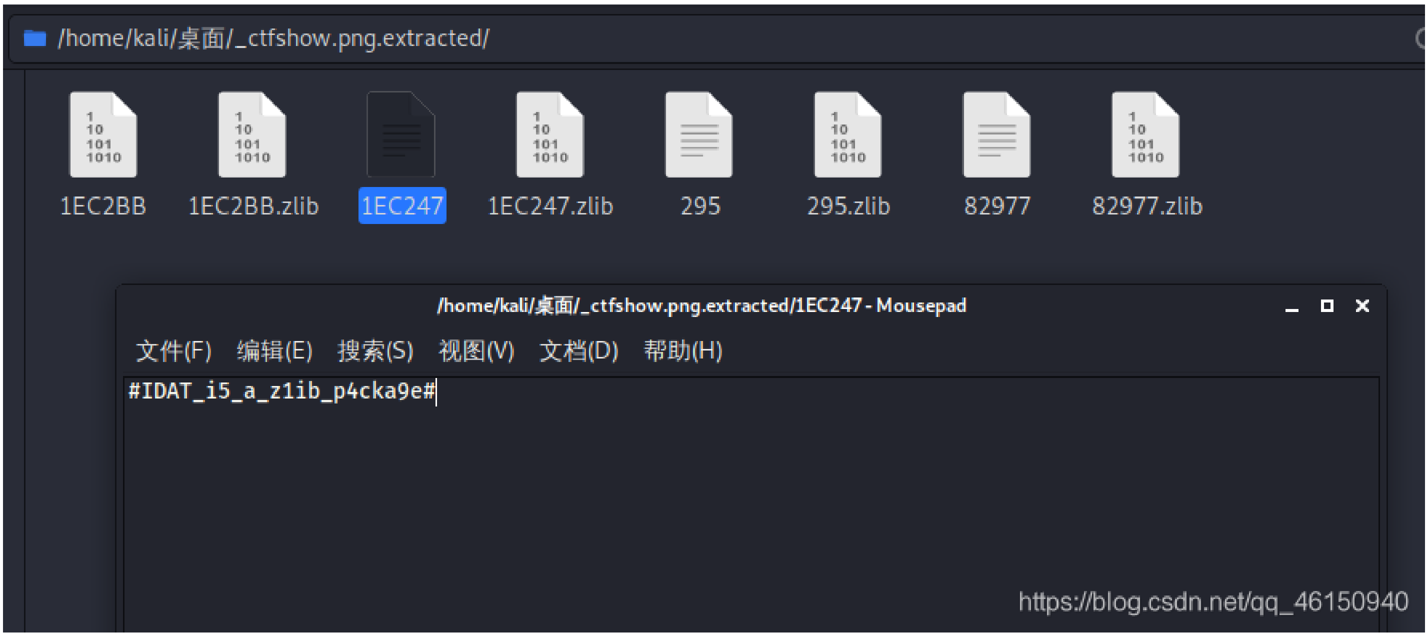
Chunk	Length	CRC	Attributes	Contents
IHDR	13	692a118d	critical	PNG image header: 1366×663, 8 bits/sample, truecolor+alpha, noninterlaced
tEXt	22	7ffac3e3	ancillary, safe to c...	text, key="Artist" (nonstandard): "#A_k3y_1n_exif#"
iTXt	574	d915b16a	ancillary, safe to c...	text (international), key="XML:com.adobe.xmp" (nonstandard): "<?xpacket begin='' id='W5M0MpC...
IDAT	65536	94f55588	critical	PNG image data
IDAT	65536	ba2406e1	critical	PNG image data
IDAT	65536	cd6a57c7	critical	PNG image data
IDAT	65536	9ec196cd	critical	PNG image data
IDAT	65536	1d1c51cc	critical	PNG image data
IDAT	65536	d41fcad9	critical	PNG image data
IDAT	65536	655d563d	critical	PNG image data
IDAT	65536	cb1875fd	critical	PNG image data
IDAT	9798	19fe70d3	critical	PNG image data
tXEt	36	2d6ea7ab	ancillary, safe to c...	unrecognized chunk type
IDAT	63563	0639e59f	critical	PNG image data
IDAT	65524	0b24ddcb	critical	PNG image data
IDAT	65524	7d1c03de	critical	PNG image data
IDAT	65524	827981d7	critical	PNG image data
IDAT	65524	555f739e	critical	PNG image data
IDAT	65524	aa6d88f1	critical	PNG image data
IDAT	65524	d7d2d41c	critical	PNG image data
IDAT	65524	95ea75c2	critical	PNG image data
IDAT	65524	08258577	critical	PNG image data
IDAT	65524	8f17ffd9	critical	PNG image data
IDAT	65524	34a3b226	critical	PNG image data
IDAT	65524	005b7214	critical	PNG image data
IDAT	65524	c756c664	critical	PNG image data

另存为flag.png



One PieNG 15

使用binwalk分离图片


/home/kali/桌面/_ctfshow.png.extracted/

1EC2BB
1EC2BB.zlib
1EC247
1EC247.zlib
295
295.zlib
82977
82977.zlib
/home/kali/桌面/_ctfshow.png.extracted/1EC247 - Mousepad
文件(F) 编辑(E) 搜索(S) 视图(V) 文档(D) 帮助(H)

```
#IDAT_i5_a_z1ib_p4cka9e#
```

One PieNG 16

pngdebug检测图片，第4-12共9个IDAT块都报错，查看发现这些错误的CRC32值都是00开头，且后三个字节都在ASCII可打印字符范围内

```
...

0x0000028D      chunk-length=0x00010000 (65536)
0x00000291      chunk-type='IDAT'
0x00010295      crc-code=0x00234831
>> (CRC CHECK)  crc-computed=0x94F55588         =>      CRC FAILED


0x00010299      chunk-length=0x00010000 (65536)
0x0001029D      chunk-type='IDAT'
0x000202A1      crc-code=0x0064655F
>> (CRC CHECK)  crc-computed=0xBA2406E1         =>      CRC FAILED


0x000202A5      chunk-length=0x00010000 (65536)
0x000202A9      chunk-type='IDAT'
0x000302AD      crc-code=0x00683378
>> (CRC CHECK)  crc-computed=0xCD6A57C7         =>      CRC FAILED


0x000302B1      chunk-length=0x00010000 (65536)
0x000302B5      chunk-type='IDAT'
0x000402B9      crc-code=0x00643437
>> (CRC CHECK)  crc-computed=0x9EC196CD         =>      CRC FAILED


0x000402BD      chunk-length=0x00010000 (65536)
0x000402C1      chunk-type='IDAT'
0x000502C5      crc-code=0x00615F31
>> (CRC CHECK)  crc-computed=0x1D1C51CC         =>      CRC FAILED


0x000502C9      chunk-length=0x00010000 (65536)
0x000502CD      chunk-type='IDAT'
0x000602D1      crc-code=0x006E5F63
>> (CRC CHECK)  crc-computed=0xD41FCAD9         =>      CRC FAILED


0x000602D5      chunk-length=0x00010000 (65536)
0x000602D9      chunk-type='IDAT'
0x000702DD      crc-code=0x0068756E
>> (CRC CHECK)  crc-computed=0x655D563D         =>      CRC FAILED


0x000702E1      chunk-length=0x00010000 (65536)
0x000702E5      chunk-type='IDAT'
0x000802E9      crc-code=0x006B5F43
>> (CRC CHECK)  crc-computed=0xCB1875FD         =>      CRC FAILED


0x000802ED      chunk-length=0x00002646 (9798)
0x000802F1      chunk-type='IDAT'
0x0008293B      crc-code=0x00524323
>> (CRC CHECK)  crc-computed=0x19FE70D3         =>      CRC FAILED

...
```

这几个异常的CRC值提取出来

```
0x00234831
0x0064655F
0x00683378
0x00643437
0x00615F31
0x006E5F63
0x0068756E
0x006B5F43
0x00524323
```

整理一下得到

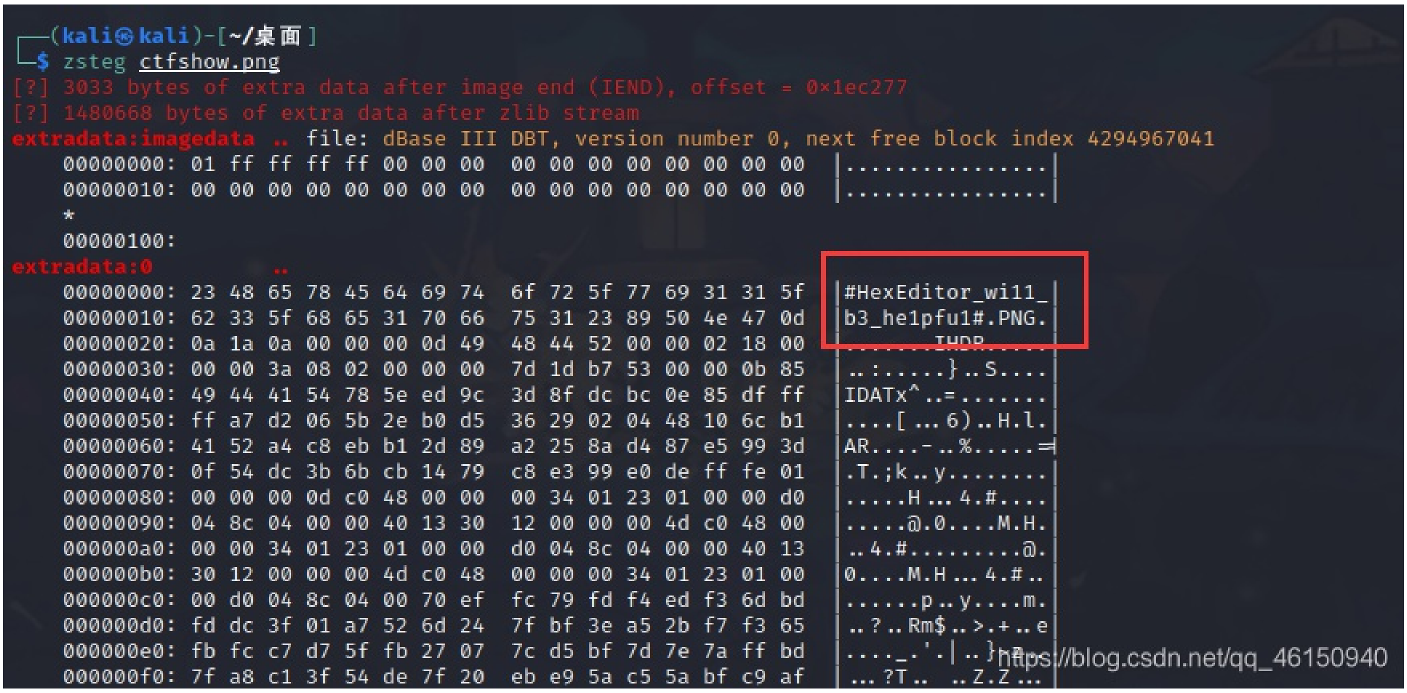
```
23483164655F6833786434376 15F316E5F6368756E6B5F43524323
```

然后十六进制转字符串

```
#H1de_h3xd47a_1n_chunk_CRC#
```

One PieNG 17

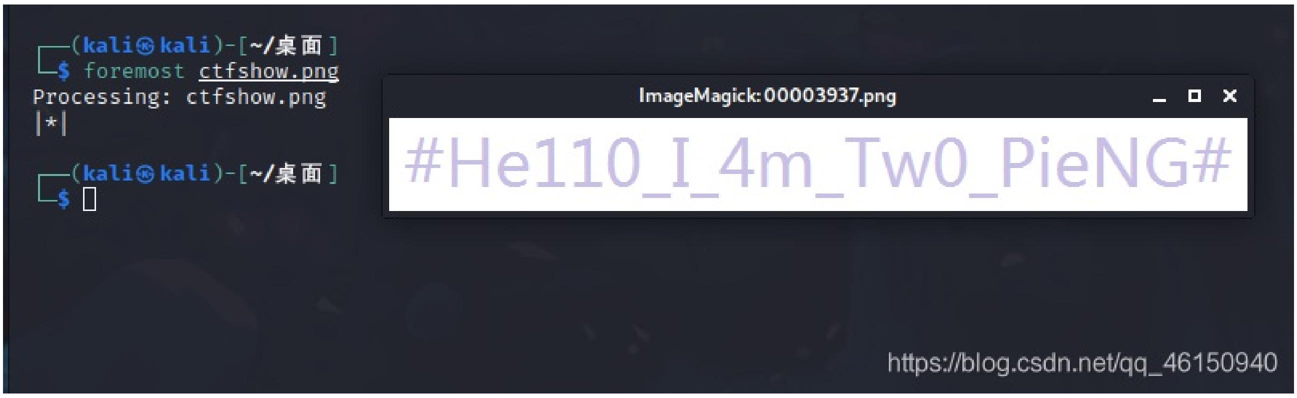
使用zsteg工具检测



```
┌──(kali㉿kali)-[~/桌面]
└─$ zsteg ctfshow.png
[?] 3033 bytes of extra data after image end (IEND), offset = 0×1ec277
[?] 1480668 bytes of extra data after zlib stream
extradata:imagedata .. file: dBase III DBT, version number 0, next free block index 4294967041
    00000000: 01 ff ff ff ff 00 00 00  00 00 00 00 00 00 00 00  |................|
    00000010: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
    *
    00000100:
extradata:0         ..
    00000000: 23 48 65 78 45 64 69 74  6f 72 5f 77 69 31 31 5f  |#HexEditor_wi11_|
    00000010: 62 33 5f 68 65 31 70 66  75 31 23 89 50 4e 47 0d  |b3_he1pfu1#.PNG.|
    00000020: 0a 1a 0a 00 00 00 0d 49  48 44 52 00 00 02 18 00  |.......IHDR.....|
    00000030: 00 00 3a 08 02 00 00 00  7d 1d b7 53 00 00 0b 85  |..:.....}..S....|
    00000040: 49 44 41 54 78 5e ed 9c  3d 8f dc bc 0e 85 df ff  |IDATx^..=.......|
    00000050: ff a7 d2 06 5b 2e b0 d5  36 29 02 04 48 10 6c b1  |....[...6)..H.l.|
    00000060: 41 52 a4 c8 eb b1 2d 89  a2 25 8a d4 87 e5 99 3d  |AR....-..%.....=|
    00000070: 0f 54 dc 3b 6b cb 14 79  c8 e3 99 e0 de ff fe 01  |.T.;k..y........|
    00000080: 00 00 00 0d c0 48 00 00  00 34 01 23 01 00 00 d0  |.....H...4.#....|
    00000090: 04 8c 04 00 00 40 13 30  12 00 00 00 4d c0 48 00  |.....@.0....M.H.|
    000000a0: 00 00 34 01 23 01 00 00  d0 04 8c 04 00 00 40 13  |..4.#.........@.|
    000000b0: 30 12 00 00 00 4d c0 48  00 00 00 34 01 23 01 00  |0....M.H...4.#..|
    000000c0: 00 d0 04 8c 04 00 70 ef  fc 79 fd f4 ed f3 6d bd  |......p..y....m.|
    000000d0: fd dc 3f 01 a7 52 6d 24  7f bf 3e a5 2b f7 f3 65  |..?..Rm$..>.+..e|
    000000e0: fb fc c7 d7 5f fb 27 07  7c d5 bf 7d 7e 7a ff bd  |...._.'.|..}~z..|
    000000f0: 7f a8 c1 3f 54 de 7f 20  eb e9 5a c5 5a bf c9 af  |...?T.. ..Z.Z...|
```

One PieNG 18

foremost分离图片



```
┌──(kali㉿kali)-[~/桌面]
└─$ foremost ctfshow.png
Processing: ctfshow.png
|*|

┌──(kali㉿kali)-[~/桌面]
└─$ 
```

ImageMagick: 00003937.png

#He110_I_4m_Tw0_PieNG#

总结：主要是各种工具的运用，有些工具没见过，还是太菜了

参考：https://blog.csdn.net/qq_42880719/article/details/114825260