CTF.show：misc入门1-23

原创

FW_ENJOEY 于 2021-03-29 20:53:24 发布 1746 收藏 15

分类专栏： CTF.show CTF_MISC_Writeup

本文链接：https://blog.csdn.net/qq_46230755/article/details/115261625


CTF.show 同时被 2 个专栏收录

23 篇文章 4 订阅

订阅专栏


CTF_MISC_Writeup

24 篇文章 2 订阅

订阅专栏

八神爷爷出的题，很适合新手入门

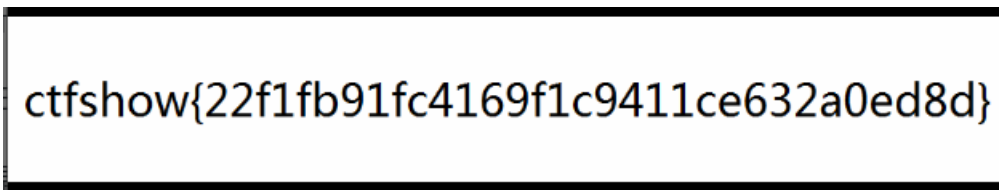
我只是记录一下自己做题过程，没啥技术含量

目录

图片篇(基础操作)

misc1

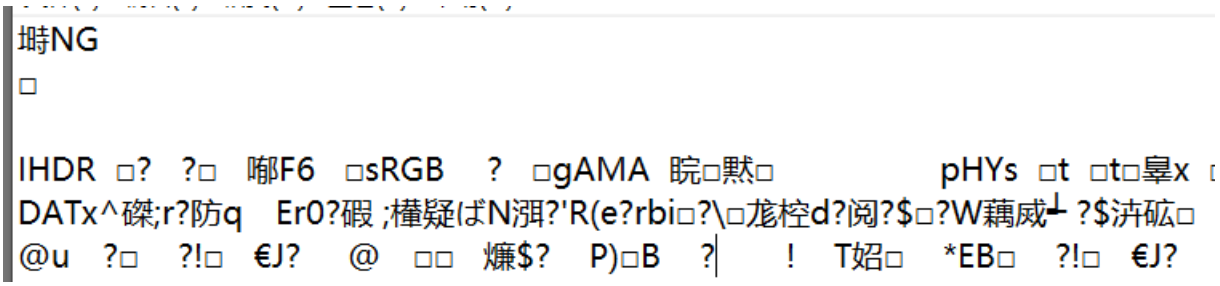
图片打开就是flag




```
ctfshow{22f1fb91fc4169f1c9411ce632a0ed8d}
```

misc2

打开图片是一串乱码，但是很明显存在一个png的文件头



修改文件名后缀为png，得到flag

ctfshow{6f66202f21ad22a2a19520cdd3f69e7b}

```
ctfshow{6f66202f21ad22a2a19520cdd3f69e7b}
```

misc3

题目给了一个BPG文件。BPG 是一种 JPEG 图像格式最有效压缩。 看图软件都可以打开。
这里可以学习一下BPG的文件格式
http://www.zhihu.com/question/27089508

这里下个查看bpg的软件
http://www.greenxf.com/soft/277387.html

ctfshow{aade771916df7cde3009c0e631f9910d}

```
ctfshow{aade771916df7cde3009c0e631f9910d}
```

misc4

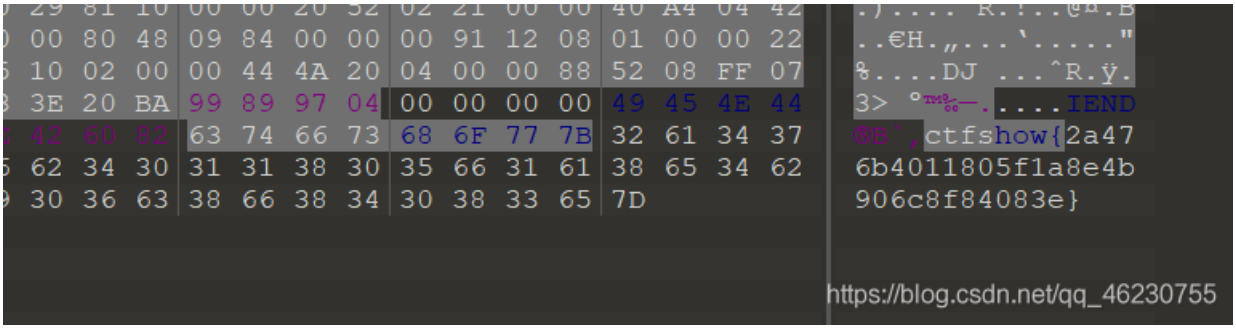
打开后存在6个txt文件，一样吧所有后缀都改成png得到flag

```
ctfshow{4314e2b15ad9a960e7d9d8fc2ff902da}
```

图片篇(信息附加)

misc5

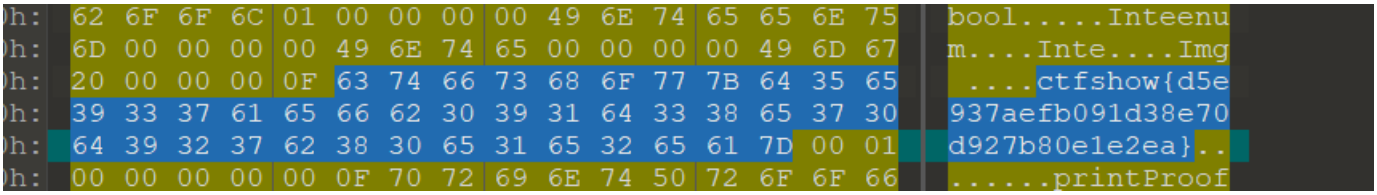
图片用010打开得到flag




```
ctfshow{2a476b4011805f1a8e4b906c8f84083e}
```

misc6

和misc5一样，010里面打开

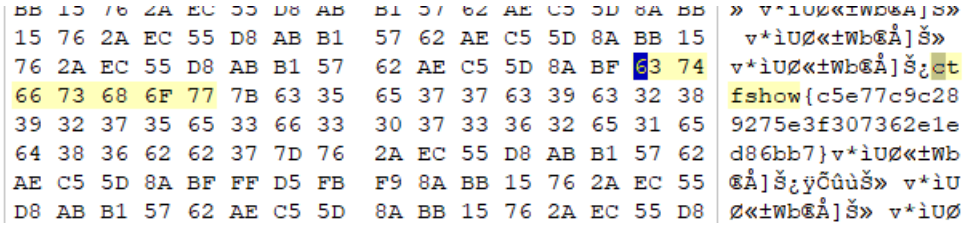



```
ctfshow{d5e937aefb091d38e70d927b80e1e2ea}
```

misc7

提示：`flag在图片文件信息中。`

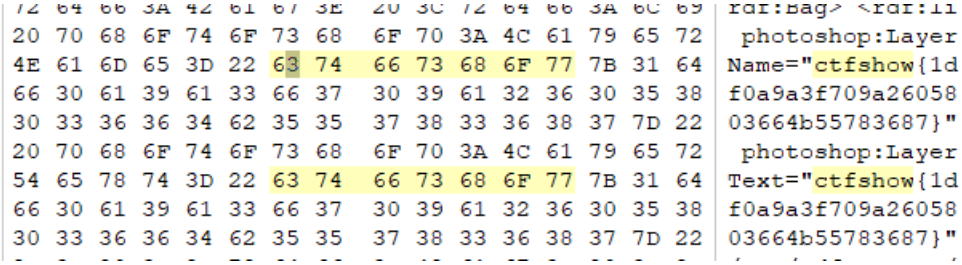
方法同上




```
ctfshow{c5e77c9c289275e3f307362e1ed86bb7}
```

misc8

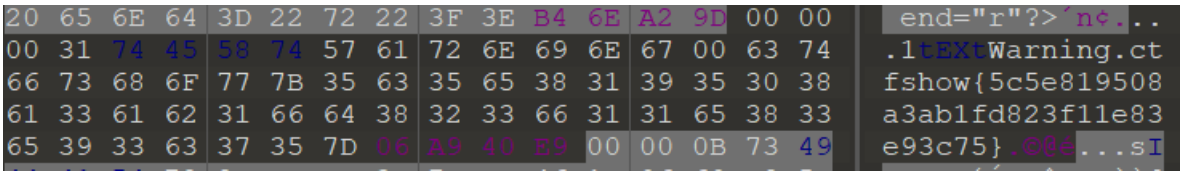
提示：`flag在图片文件中图片文件中。`



```
ctfshow{1df0a9a3f709a2605803664b55783687}
```

misc9

提示：`flag在图片块里。`
一样010打开

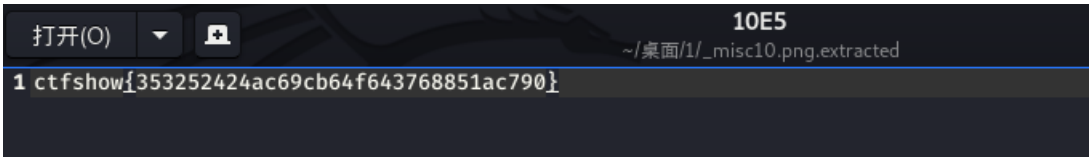



```
ctfshow{5c5e819508a3ab1fd823f11e83e93c75}
```

misc10

提示：`flag在图片数据里。`
binwalk一把梭

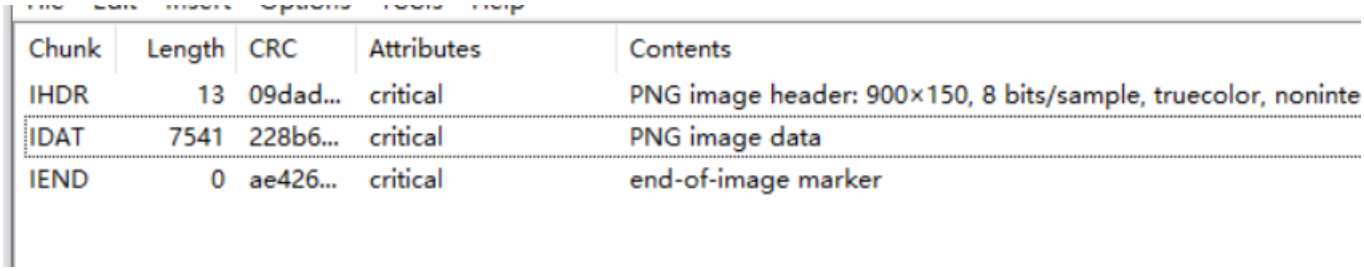
```
binwalk misc10.png -e
```



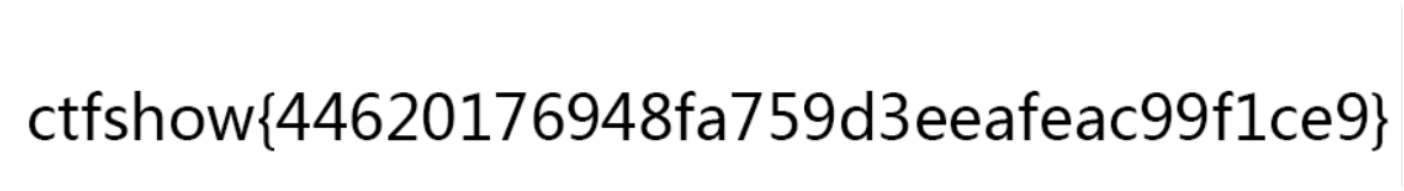

```
ctfshow{353252424ac69cb64f643768851ac790}
```

misc11

提示：`flag在另一张图里。`
放入tweakpng中，删除一个IDAT块，保存为新的图片。得到flag



Chunk	Length	CRC	Attributes	Contents
IHDR	13	09dad...	critical	PNG image header: 900×150, 8 bits/sample, truecolor, noninte
IDAT	7541	228b6...	critical	PNG image data
IEND	0	ae426...	critical	end-of-image marker

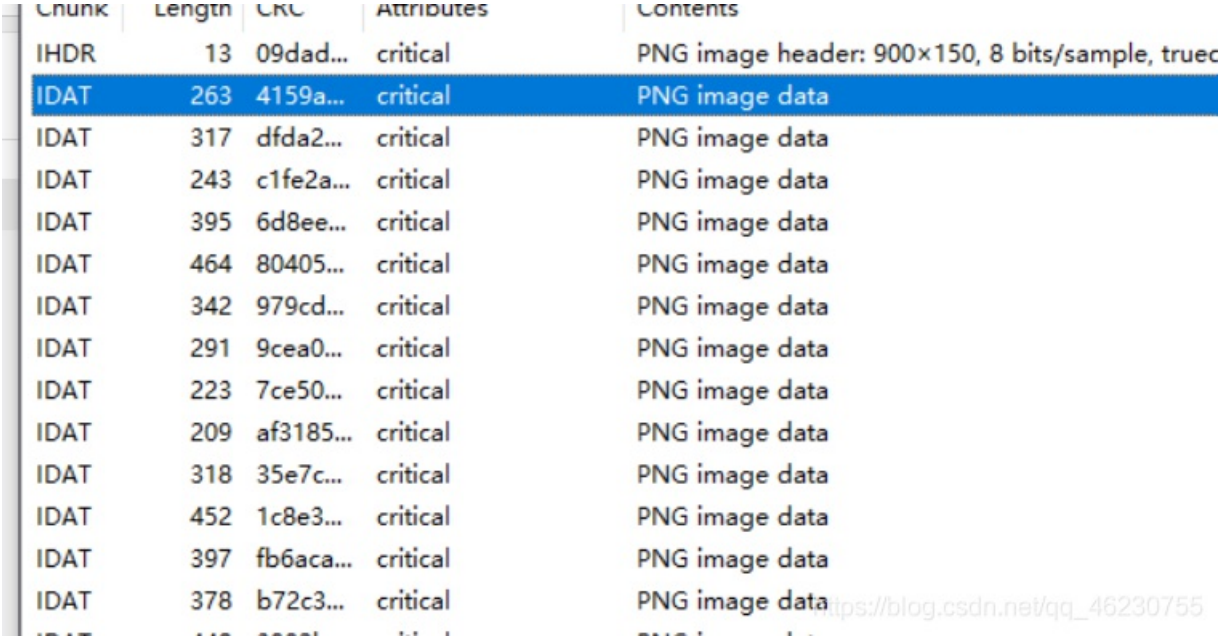



```
ctfshow{44620176948fa759d3eeafeac99f1ce9}
```

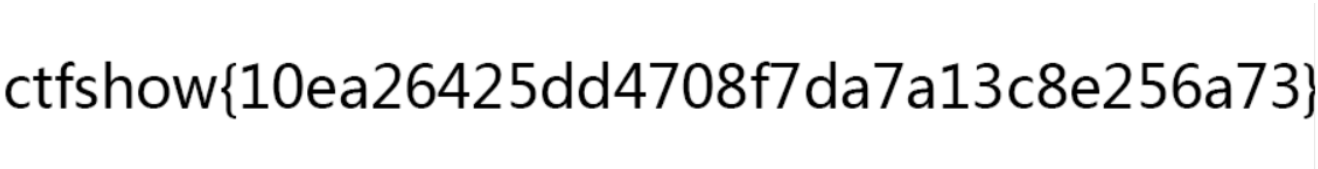
misc12

提示：`flag在另一张图里。`

做法与11一模一样，然后只是一个一个IDAT块删完然后保存下来得到flag。我映像中应该是删了5次。



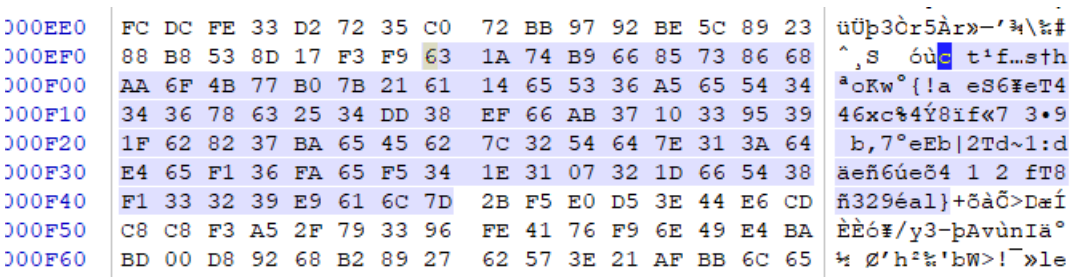
Chunk	Length	CRC	Attributes	Contents
IHDR	13	09dad...	critical	PNG image header: 900×150, 8 bits/sample, truec
IDAT	263	4159a...	critical	PNG image data
IDAT	317	dfda2...	critical	PNG image data
IDAT	243	c1fe2a...	critical	PNG image data
IDAT	395	6d8ee...	critical	PNG image data
IDAT	464	80405...	critical	PNG image data
IDAT	342	979cd...	critical	PNG image data
IDAT	291	9cea0...	critical	PNG image data
IDAT	223	7ce50...	critical	PNG image data
IDAT	209	af3185...	critical	PNG image data
IDAT	318	35e7c...	critical	PNG image data
IDAT	452	1c8e3...	critical	PNG image data
IDAT	397	fb6aca...	critical	PNG image data
IDAT	378	b72c3...	critical	PNG image data




```
ctfshow{10ea26425dd4708f7da7a13c8e256a73}
```

misc13

群里师傅讨论里学到的。看到这地方差不多是flag的样子，隔一个字节取一个数



```
00EE0  FC DC FE 33 D2 72 35 C0  72 BB 97 92 BE 5C 89 23  üÜþ3Òr5Àr»—'¾\‰#
00EF0  88 B8 53 8D 17 F3 F9 63  1A 74 B9 66 85 73 86 68  ˆ¸S  óùc t¹f…s†h
00F00  AA 6F 4B 77 B0 7B 21 61  14 65 53 36 A5 65 54 34  ªoKw°{!a eS6¥eT4
00F10  34 36 78 63 25 34 DD 38  EF 66 AB 37 10 33 95 39  46xc%4Ý8ïf«7 3•9
00F20  1F 62 82 37 BA 65 45 62  7C 32 54 64 7E 31 3A 64   b‚7ºeEb|2Td~1:d
00F30  E4 65 F1 36 FA 65 F5 34  1E 31 07 32 1D 66 54 38  äeñ6úeõ4 1 2 fT8
00F40  F1 33 32 39 E9 61 6C 7D  2B F5 E0 D5 3E 44 E6 CD  ñ329éal}+õàÕ>DæÍ
00F50  C8 C8 F3 A5 2F 79 33 96  FE 41 76 F9 6E 49 E4 BA  ÈÈó¥/y3–þAvùnIä°
00F60  BD 00 D8 92 68 B2 89 27  62 57 3E 21 AF BB 6C 65  ½ Ø’h²‰'bW>!¯»le
```

一开始的做法，硬看。错了。然后问了八神八神说硬找会找到错的

问了别的师傅，说去十六进制转字符后，隔一个字符删一下

631A74B96685738668AA6F4B77B07B216114655336A56554333465786125
34DD38EF66AB35103195381F628237BA6545347C3254647E373A64E465F
136FA66F5341E3107321D665438F1333239E9616C7D

字符串转16进制 >>

16进制转字符串 >>

结果互换

全部清空

ctfshow{ae6e3ea48f518b7e42d7de6f412f839a}


```
ctfshow{ae6e3ea48f518b7e42d7de6f412f839a}
```

misc14

提示：`flag在那张图里。`

```
root@kali:~/桌面/1# binwalk misc14.jpg

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, EXIF standard
12            0xC             TIFF image data, big-endian, offset of first image directory
: 8
1681          0x691           TIFF image data, big-endian, offset of first image directory
: 8
2103          0x837           JPEG image data, JFIF standard 1.01
```

binwalk发现存在两张图片
其中一张是JFIF的图片，直接在010里面搜JFIF找到文件头

```
00 48 00 00 00 01 FF D8 FF E0 00 10 4A 46 49   ..H....ÿØÿà..JFI
00 01 01 01 00 78 00 78 00 00 FF DB 00 43 00   F.....x.x..ÿÛ.C.
01 01 02 01 01 02 02 02 02 02 02 02 03 05      ..............
03 03 03 03 06 04 04 03 05 07 06 07 07 07 06   ..............
07 07 08 09 0B 09 08 08 0A 08 07 07 0A 0D 0A 0A   ..............
0C 0C 0C 0C 07 09 0E 0F 0D 0C 0E 0B 0C 0C 0C   ..............
DB 00 43 01 02 02 02 03 03 03 06 03 03 06 0C   ÿÛ.C...........
```

从这里开始复制下来新建为新的图片。打开后得到flag

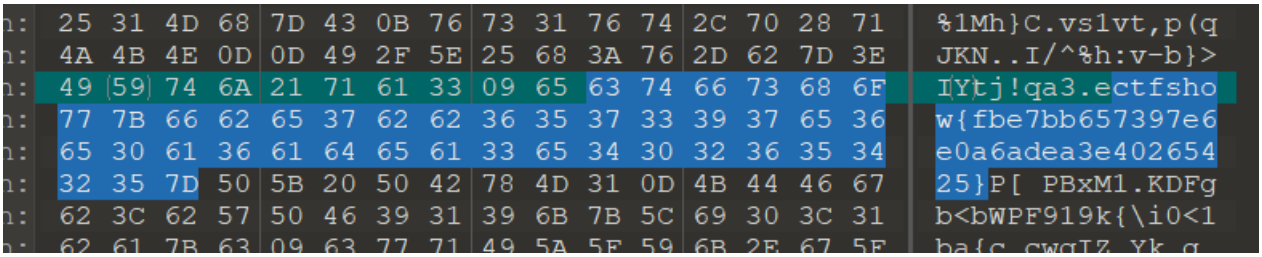
ctfshow{ce520f767fc465b0787cdb936363e694}

```
ctfshow{ce520f767fc465b0787cdb936363e694}
```

misc15

提示：`flag被跳过去了。`

直接打开010梭哈。



```
25 31 4D 68 7D 43 0B 76 73 31 76 74 2C 70 28 71   %1Mh}C.vs1vt,p(q
4A 4B 4E 0D 0D 49 2F 5E 25 68 3A 76 2D 62 7D 3E   JKN..I/^%h:v-b}>
49 59 74 6A 21 71 61 33 09 65 63 74 66 73 68 6F   IYtj!qa3.ectfsho
77 7B 66 62 65 37 62 62 36 35 37 33 39 37 65 36   w{fbe7bb657397e6
65 30 61 36 61 64 65 61 33 65 34 30 32 36 35 34   e0a6adea3e402654
32 35 7D 50 5B 20 50 42 78 4D 31 0D 4B 44 46 67   25}P[ PBxM1.KDFg
62 3C 62 57 50 46 39 31 39 6B 7B 5C 69 30 3C 31   b<bWPF919k{\i0<1
62 61 7B 63 09 63 77 71 49 5A 5F 59 6B 2E 67 5F   ba{c cwqIZ_Yk.g_
```

```
ctfshow{fbe7bb657397e6e0a6adea3e40265425}
```

misc16

提示：`flag在图片数据里。`

直接binwalk得到文件。

```
root@kali:~/桌面/1# binwalk misc16.png -e

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             PNG image, 900 x 150, 8-bit/color RGB, non-interlaced
41            0x29            Zlib compressed data, best compression
3540          0xDD4           LZMA compressed data, properties: 0x5D, dictionary size: 8388608 bytes, unco
essed size: -1 bytes
```

DD4中存在flag

DD4 - 记事本

文件(F)　编辑(E)　格式(O)　查看(V)　帮助(H)

ctfshow{a7e32f131c011290a62476ae77190b52}

```
ctfshow{a7e32f131c011290a62476ae77190b52}
```

misc17

提示：`flag在图片数据里。`

Sapphire师傅教的好啊！！

利用zsteg发现里面存在数据，我们要提取extradata：0的数据出来。

```
root@kali:~/桌面/1# zsteg misc17.png
[?] 3544 bytes of extra data after zlib stream
extradata:0         ..
    00000000: e1 1f 30 53 86 4f c5 a4  1b f5 e6 e5 c7 46 0a 92  |..0S.O.......F..|
    00000010: 9b ee 72 e7 c9 9e b9 a7  74 de 92 4d ad 61 5b 58  |..r.....t..M.a[X|
    00000020: f2 98 65 77 2b d2 d3 85  32 fc 08 83 86 1f 0f 1e  |..ew+...2.......|
    00000030: cb ab ac 9c 4b ca 02 20  e2 ce e4 ae 60 1a 2c c6  |....K.. ....`.,.|
    00000040: 7b c8 9a 77 31 2f 9e 67  db d9 3e 53 fe 17 a5 50  |{..w1/.g..>S...P|
    00000050: 20 e5 1d 8c d5 49 4e 52  a5 54 31 cb 8b c5 3b 09  | ....INR.T1...;.|
    00000060: a2 a6 fe 5b da 4f 9e 78  9c 5d 46 d6 e2 6b 6b 2a  |...[.O.x.]F..kk*|
    00000070: f2 62 0c ba 70 19 a0 27  f3 84 77 99 02 77 05 79  |.b..p..'..w..w.y|
    00000080: 5b 44 b7 79 b3 54 11 a1  f3 54 34 56 7e ff 55 d1  |[D.y.T...T4V~.U.|
    00000090: c6 39 90 c8 21 7f 26 39  44 58 78 c3 ed 37 4a 7c  |.9..!.&9DXx..7J||
    000000a0: 50 24 e8 79 7b 4b 9c fa  2a 2c bb e8 b9 fb 40 2c  |P$.y{K..*,....@,|
    000000b0: 50 05 21 4c 3b 29 65 b4  60 1c 27 bb 4c 16 bf f1  |P.!L;)e.`.'.L...|
    000000c0: 77 c0 55 04 5e 25 0e 18  1e 58 ab 0f 13 11 f2 3f  |w.U.^%...X.....?|
    000000d0: cf a0 32 b1 f5 a8 1b 99  a7 4b 46 89 cf 85 89 50  |..2......KF....P|
    000000e0: 88 20 8f 4f fd e2 97 55  68 73 b4 96 ba dd 25 a3  |. .O...Uhs....%.|
    000000f0: 83 72 3f 99 77 9e 0a 08  50 4f 11 8f 87 65 c0 29  |.r?.w...PO...e.)|
```



```
zsteg -E misc17.png 'extradata:0' > data文件名
```

我提取在2里面，然后就binwalk直接梭出来了。

```
root@kali:~/桌面/1# binwalk 2 -e

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
497           0x1F1           bzip2 compressed data, block size = 900k
```

ctfshow{0fe61fc42e8bbe55b9257d251749ae45}

```
ctfshow{0fe61fc42e8bbe55b9257d251749ae45}
```

misc18

提示：`flag在标题、作者、照相机和镜头型号里`。

属性	值
说明	
标题	ctfshow{32
主题	
分级	☆☆☆☆☆
标记	
备注	
来源	
作者	5d60c208f7
拍摄日期	

照相机型号	28ac17e5f0
光圈值	
曝光时间	
ISO 速度	
曝光补偿	
焦距	
最大光圈	
测光模式	
目标距离	
闪光灯模式	
闪光灯能量	
35mm 焦距	
高级照片	
镜头制造商	
镜头型号	2d4cf5a839}

```
ctfshow{325d60c208f728ac17e5f02d4cf5a839}
```

misc19

提示：`flag在主机上的文档名里。`

010打开发现字符串

```
08 00 08 00 63 74 66 73 68 6F 77 7B 64 66 64 63   ....ctfshow{dfdc
66 30 38 30 33 38 63 64 34 34 36 61 35 00 B8 54   f08038cd446a5.¸T
00 00 4E 63 00 00 95 0E 00 00 ED 02 00 00 80 FC   ..Nc..•...í...€ü
0A 00 10 27 00 00 80 FC 0A 00 10 27 00 00 41 64   ...'..€ü...'..Ad
6F 62 65 20 50 68 6F 74 6F 73 68 6F 70 20 43 43   obe Photoshop CC
20 32 30 31 39 20 28 57 69 6E 64 6F 77 73 29 00    2019 (Windows).
32 30 32 31 3A 30 33 3A 32 35 20 31 30 3A 33 35   2021:03:25 10:35
3A 31 38 00 65 62 35 30 37 38 32 66 38 64 33 36   :18.eb50782f8d36
30 35 64 7D 00 00 3C 3F 78 70 61 63 6B 65 74 20   05d}..<?xpacket
```

把两串拼接一下

```
ctfshow{dfdcf08038cd446a5eb50782f8d3605d}
```

misc20

提示：`flag在评论里。`

找个图片查看器直接查看图片信息

我用的是这https://exif.tuchong.com/

ExifByteOrder	Big-endian (Motorola, MM)
Comment	这图片也太难看了。来自：西替爱抚秀大括号西九七九六四必一诶易西爱抚零六易一弟七九西二一弟弟诶弟五九三易四二大括号
ImageWidth	900
ImageHeight	150

西替爱抚秀大括号西九七九六四必一诶易西爱抚零六易一弟七九西二一弟弟诶弟五九三易四二大括号

```
ctfshow{c97964b1aecf06e1d79c21ddad593e42}
```

misc21

提示：`flag在序号里。`

利用20题那个网站直接看

IFD0

X分辨率	3902939465
Y分辨率	2371618619
PageName	https://ctf.show/
X定位	1082452817
Y定位	2980145261
目标Printer	ctfshow{}

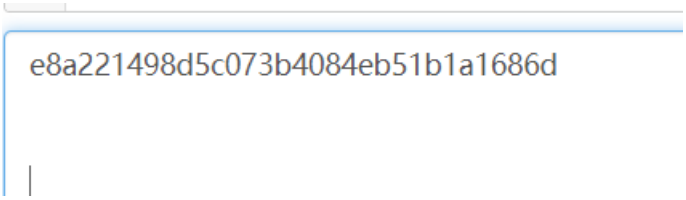
ExifIFD

Exif版本	0232
ComponentsConfiguration	Y, Cb, Cr, -
SecurityClassification	Top Secret
Flashpix版本	0100
色彩空间	Uncalibrated
序列号	686578285826597329

把序列号`686578285826597329`转字符串得到提示`hex(X&Ys)`

把XY分开哈希得到

```
print(hex(3902939465)[2:]+hex(2371618619)[2:]+hex(1082452817)[2:]+hex(2980145261)[2:])
```




```
ctfshow{e8a221498d5c073b4084eb51b1a1686d}
```

misc22

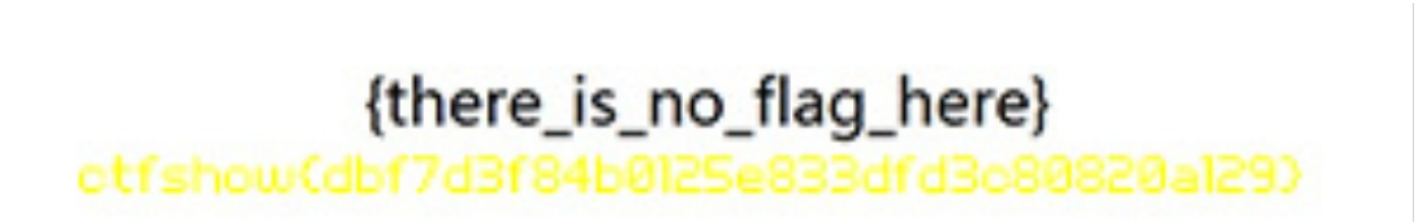
提示：`flag在图片里。`

在缩略图可以看见图片里有flag，这是一种缩略图隐写的方式，叫做thumbnail隐写。V3师傅博客讲的非常详细。

跳转V3师傅博客

我们利用exiftool工具导出图片

```
exiftool.exe -ThumbnailImage -b misc22.jpg > 1.jpg
```



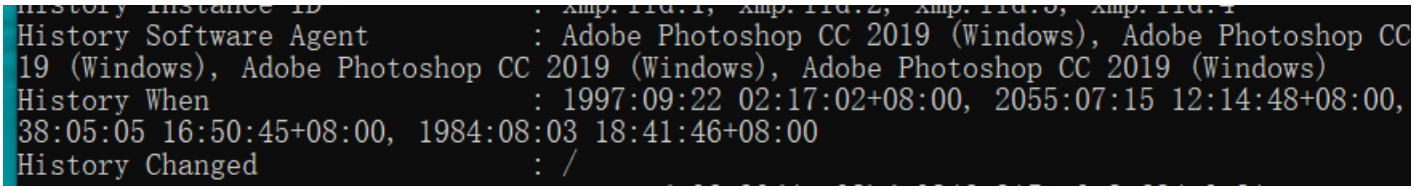

```
ctfshow{dbf7d3f84b0125e833dfd3c80820a129}
```

misc23

提示：`flag在时间里。`

利用exiftools查看时间

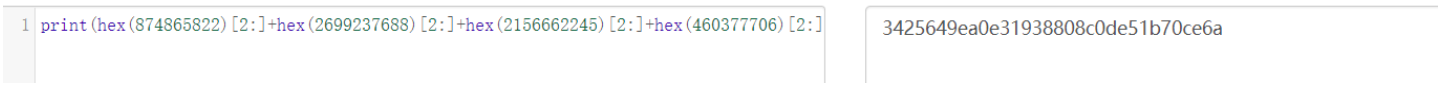
```
exiftool.exe misc23.psd
```



把四个时间转为时间戳再转hex得到flag

我是用这个网站转的


```
print(hex(874865822)[2:]+hex(2699237688)[2:]+hex(2156662245)[2:]+hex(460377706)[2:])
```




```
ctfshow{3425649ea0e31938808c0de51b70ce6a}
```