

# ctfshow 做题 MISC入门 模块1-10

原创

Goodric 于 2021-08-15 11:26:17 发布 436 收藏 2

分类专栏: [做题](#) 文章标签: [ctfshow misc](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/Goodric/article/details/119711283>

版权



[做题](#) 专栏收录该内容

24 篇文章 0 订阅

订阅专栏

## ctfshow 做题 MISC入门 模块 1-10

### misc1

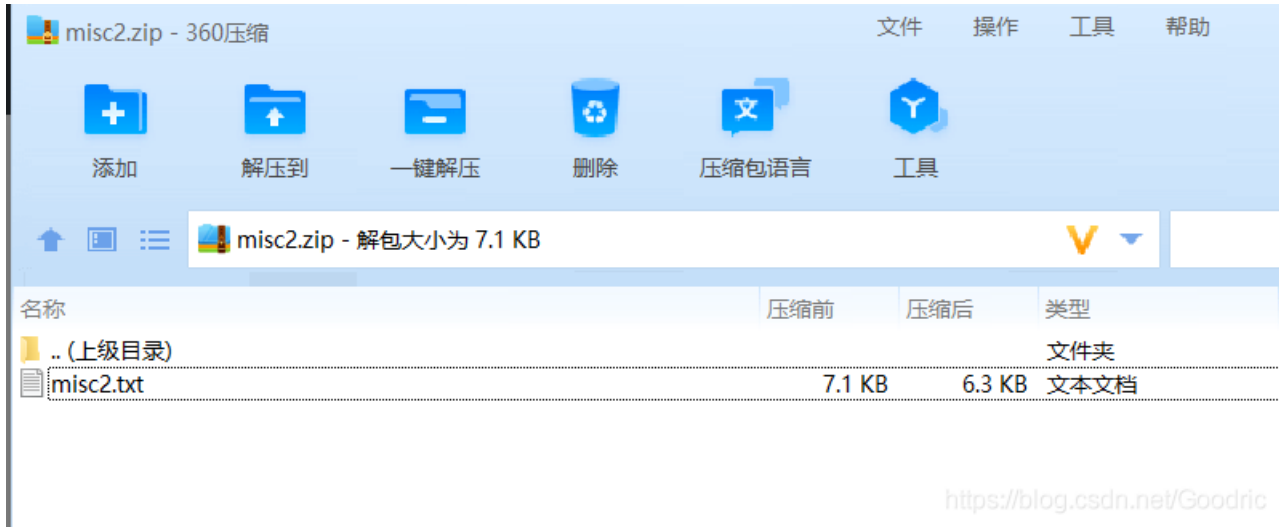
下载附件压缩包, 里面为一张图片, 直接得到 flag 。

ctfshow{22f1fb91fc4169f1c9411ce632a0ed8d}



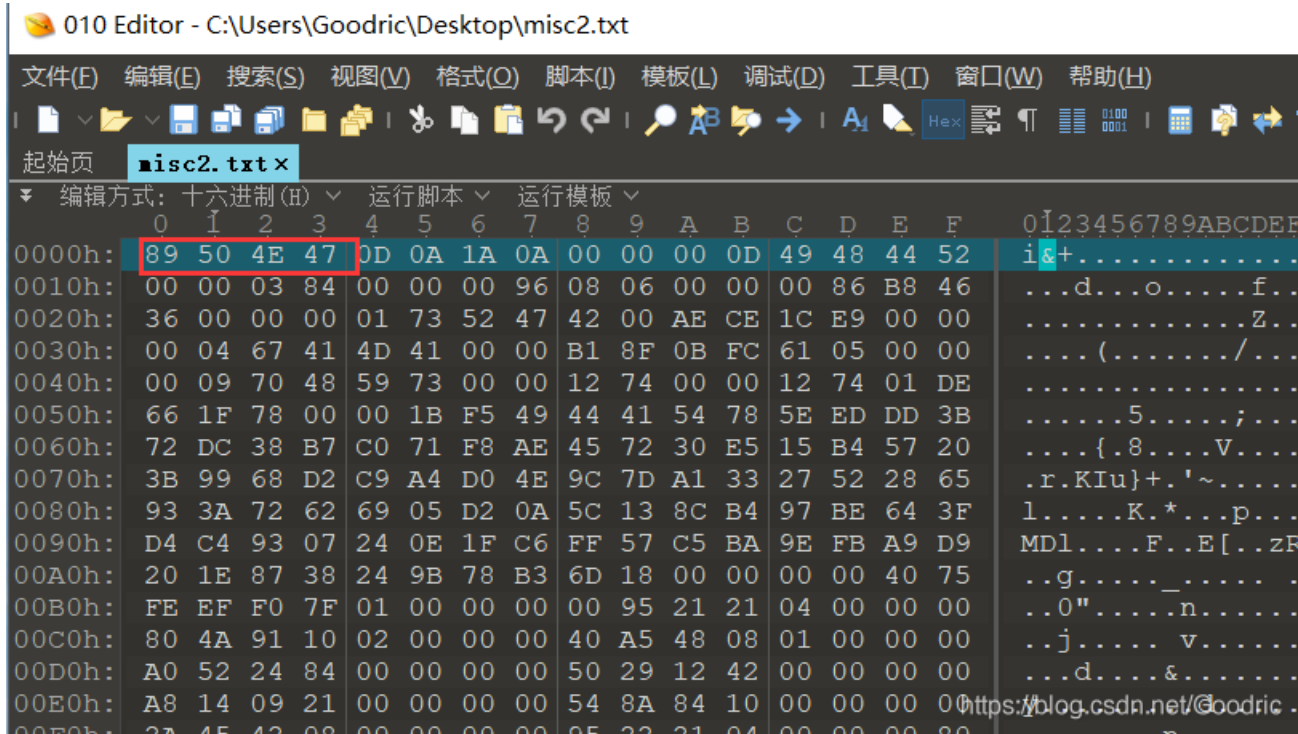
## misc2

下载的压缩包附件里有一个 txt 文件。

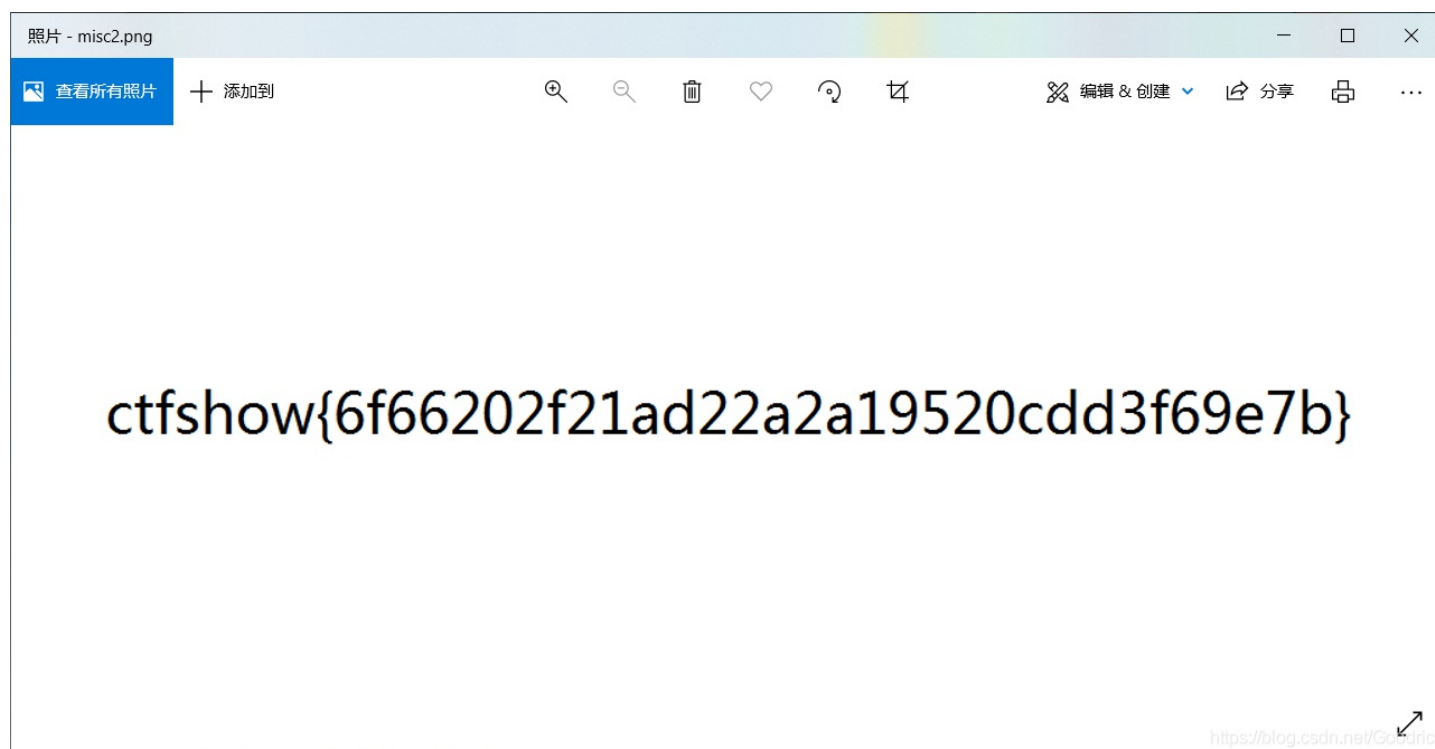


先直接打开文本模式查看，未发现有用消息。

用 010editor 打开查看 16 进制，发现文件头：**89 50 4E 47** 为 .png 文件的文件头。



把文件后缀改为 .png 打开为一张图片，得到 flag。  
ctfshow{6f66202f21ad22a2a19520cdd3f69e7b}



## misc3

下载的压缩包附件里有一个文件 misc3.bpg（打开方式未知）



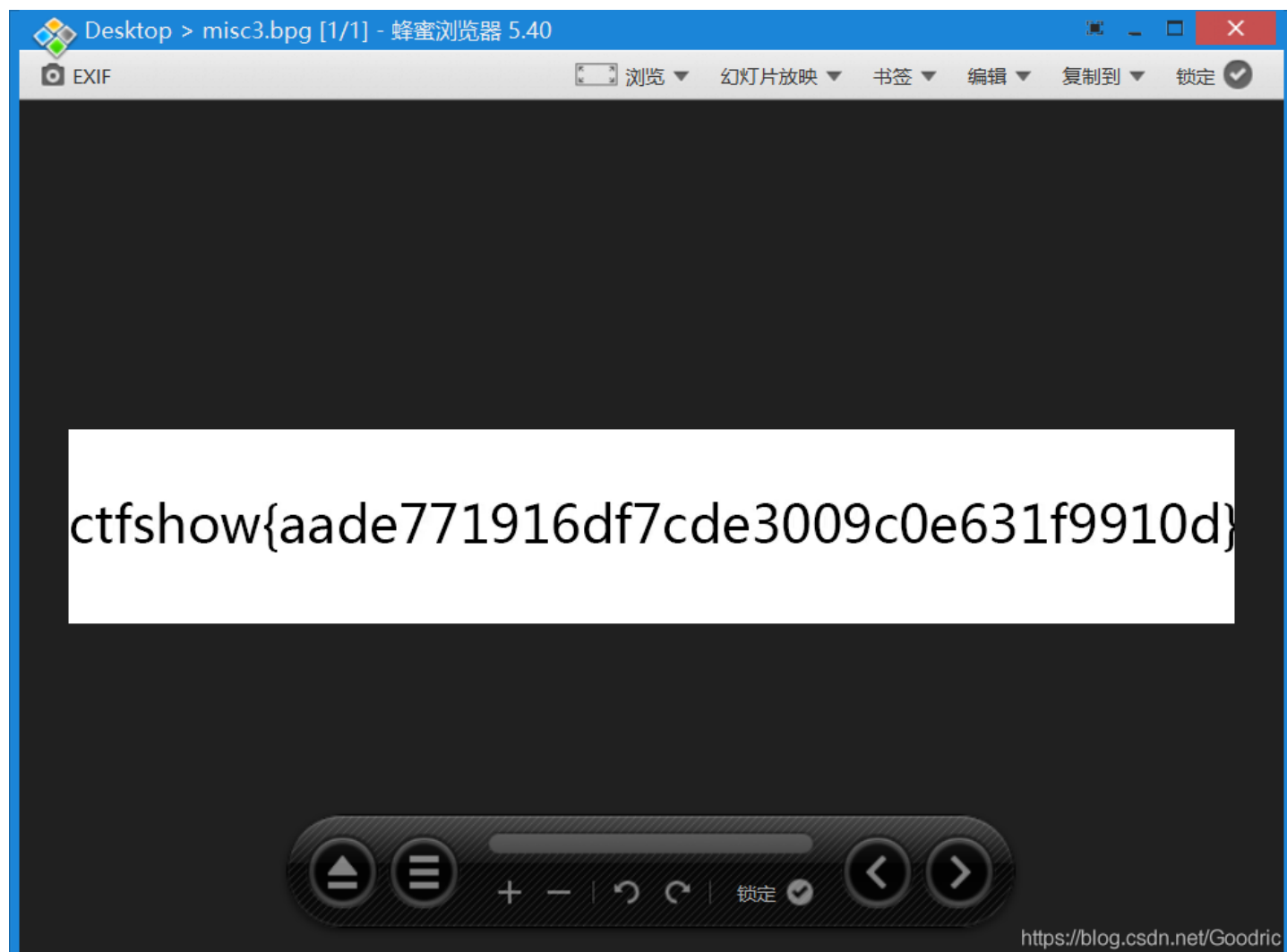
常规先用文本打开查看，未发现有效信息。

百度查了 bpg 格式的后缀就是一种图片的格式，这种格式的图片可以用许多图片查看器查看。

这里尝试下载一个图片查看器 Honeyview 。

安装好图片查看软件之后，misc3.bpg 文件得以打开。得到 flag 。

ctfshow{aade771916df7cde3009c0e631f9910d}

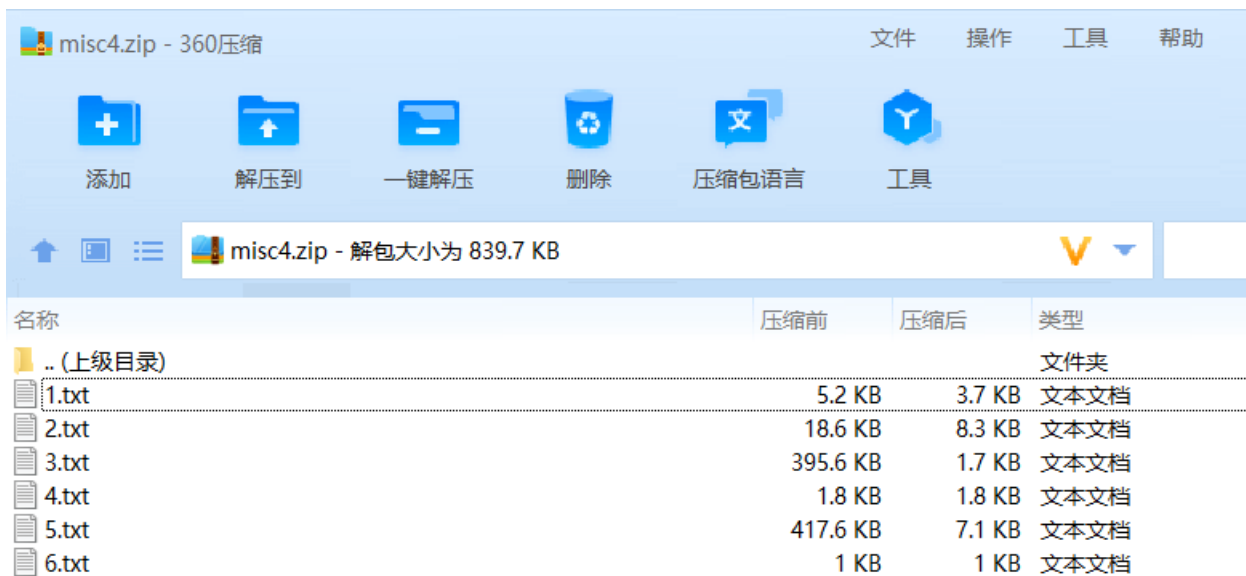


---

---

misc4

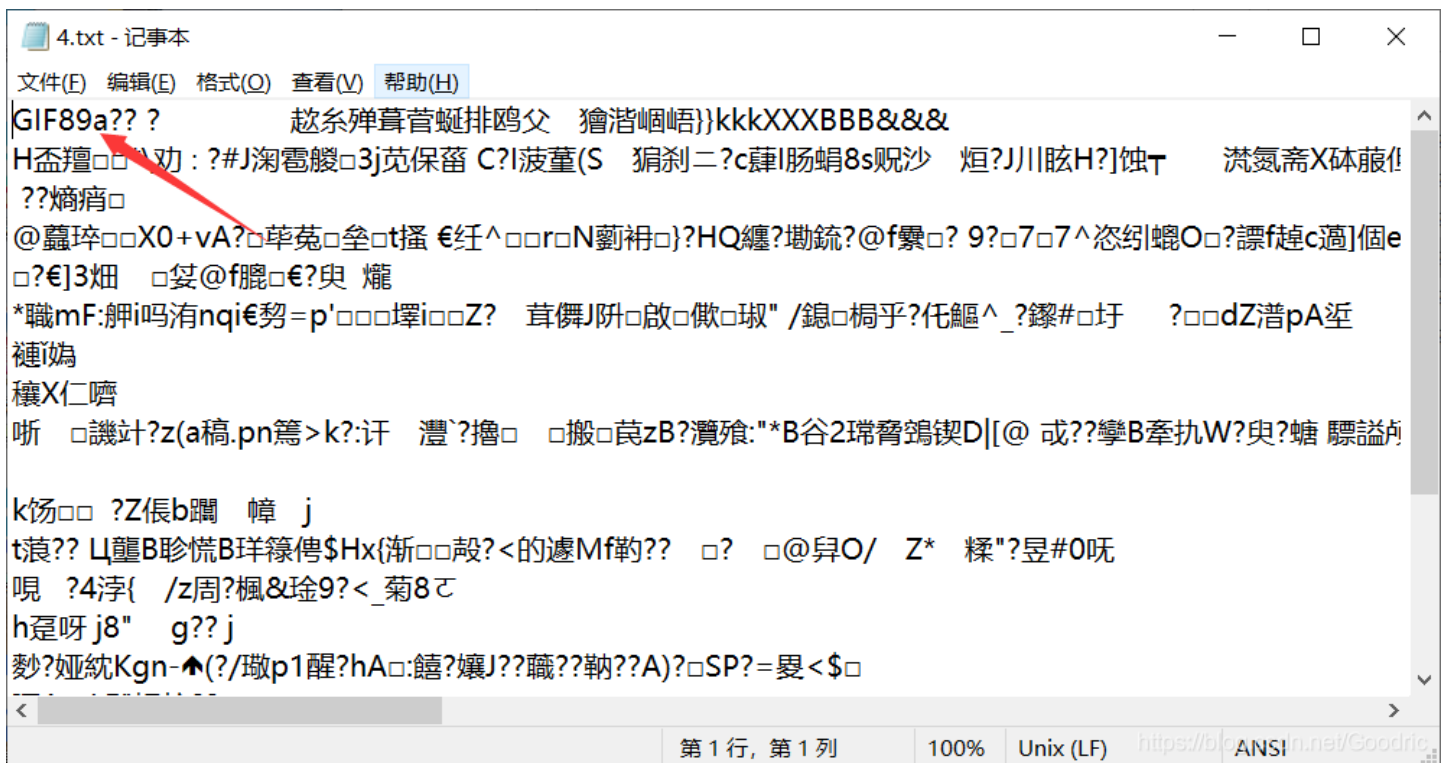
下载附件压缩包，里面有 6 个 txt 文本文件。



<https://blog.csdn.net/Goodric>

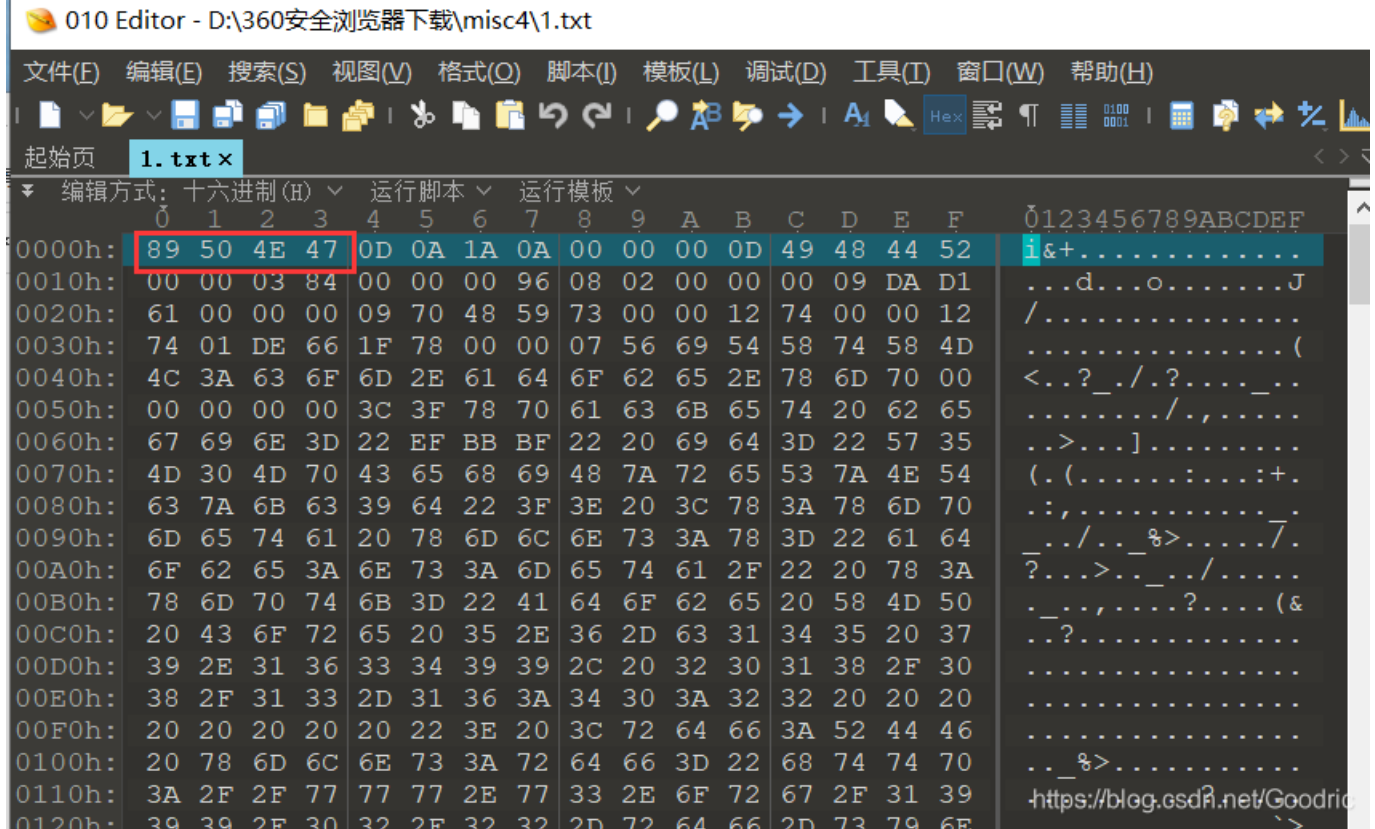
直接文本格式打开，里面都是一堆乱码，没看到有效信息。

但是当打开到 4.txt 时，看到自己见过的东西，GIF 图片的文件头，而其它文件文本格式开头也都能看得出是某种图片的文件头格式。



接下来用 010 editor 查看 16 进制，对应某个准确的文件后缀。

如 1.txt 的文件头 89 50 4E 47 为 png 图片的文件头。



依此查看，把每个文件改成正确的后缀。

名称	修改日期	类型
1.png	2021/2/4 17:25	PNG 文件
2.jpg	2021/2/4 17:25	Image (jpg) File
3.bmp	2021/2/4 17:26	Image (bmp) File
4.gif	2021/2/4 17:27	GIF 文件
5.tif	2021/2/4 17:28	Image (tif) File
6.ani	2021/3/25 0:49	ANI 文件

<https://blog.csdn.net/Goodric>

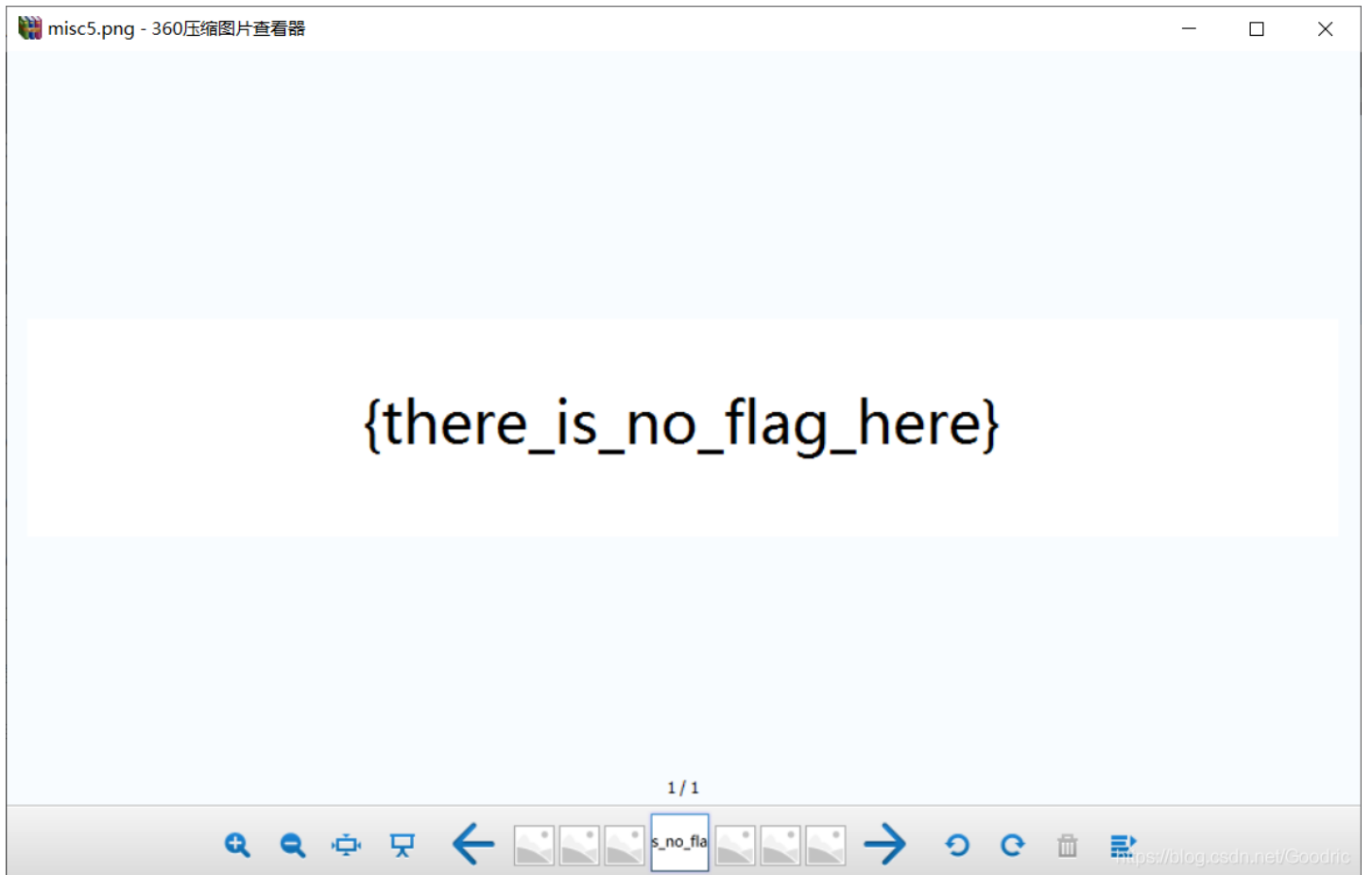
对照每张图片组合好 flag : ctfshow{4314e2b15ad9a960e7d9d8fc2ff902da}

常见文件的文件头:

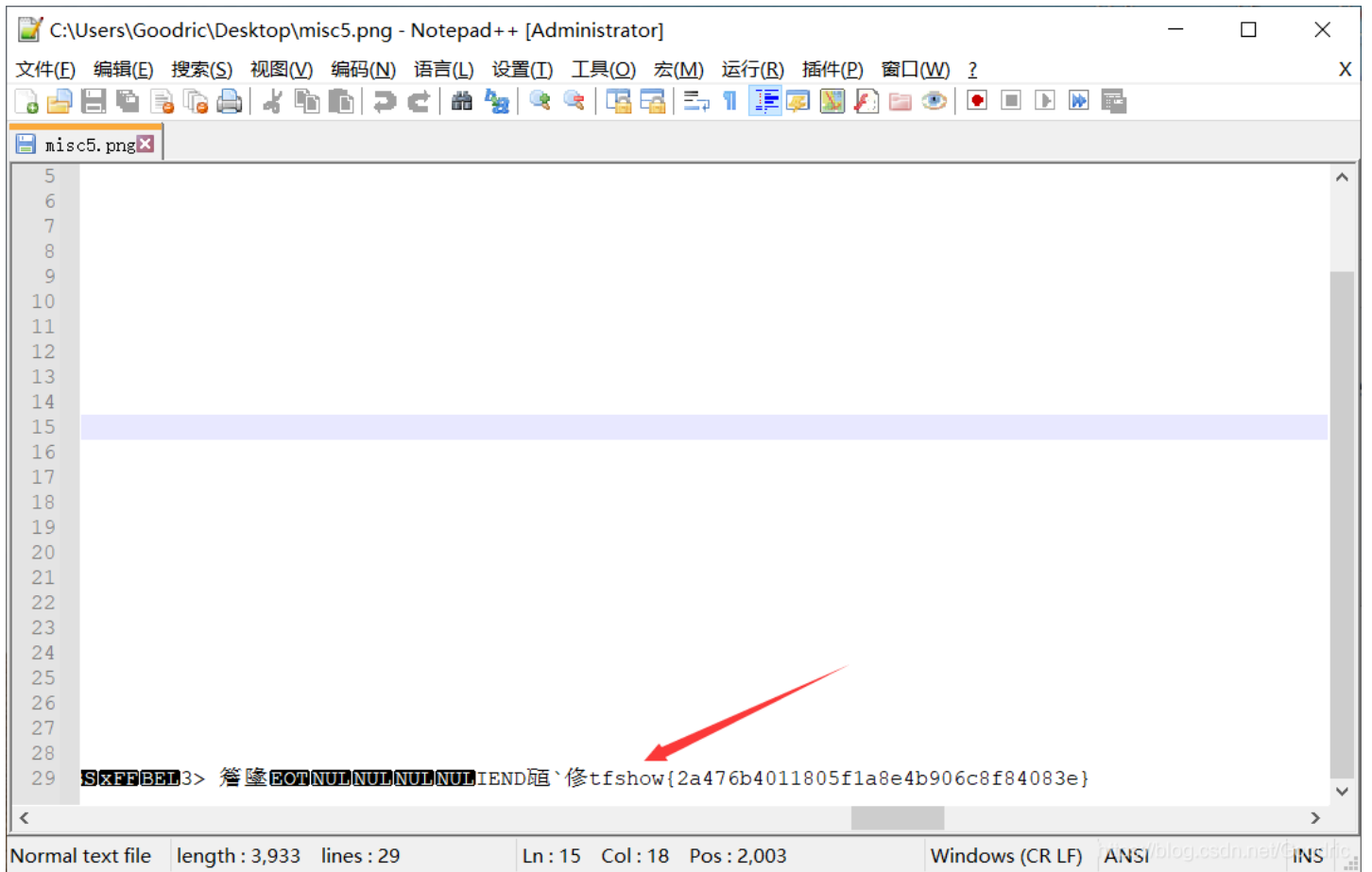
JPEG (jpg), 文件头: FFD8FFE1  
PNG (png), 文件头: 89504E47  
GIF (gif), 文件头: 47494638  
TIFF (tif), 文件头: 49492A00  
CAD (dwg), 文件头: 41433130  
Adobe Photoshop (psd), 文件头: 38425053  
Rich Text Format (rtf), 文件头: 7B5C727466  
MS Word/Excel (xls.or.doc), 文件头: D0CF11E0  
ZIP Archive (zip), 文件头: 504B0304  
RAR Archive (rar), 文件头: 52617221  
BMP, 文件头: 42 4D  
ANI, 文件头: 52 49 46 46

## misc5

下载附件为一张图片，但是显示了一个错误 flag。



常规文本打开，在末尾得到 flag 。



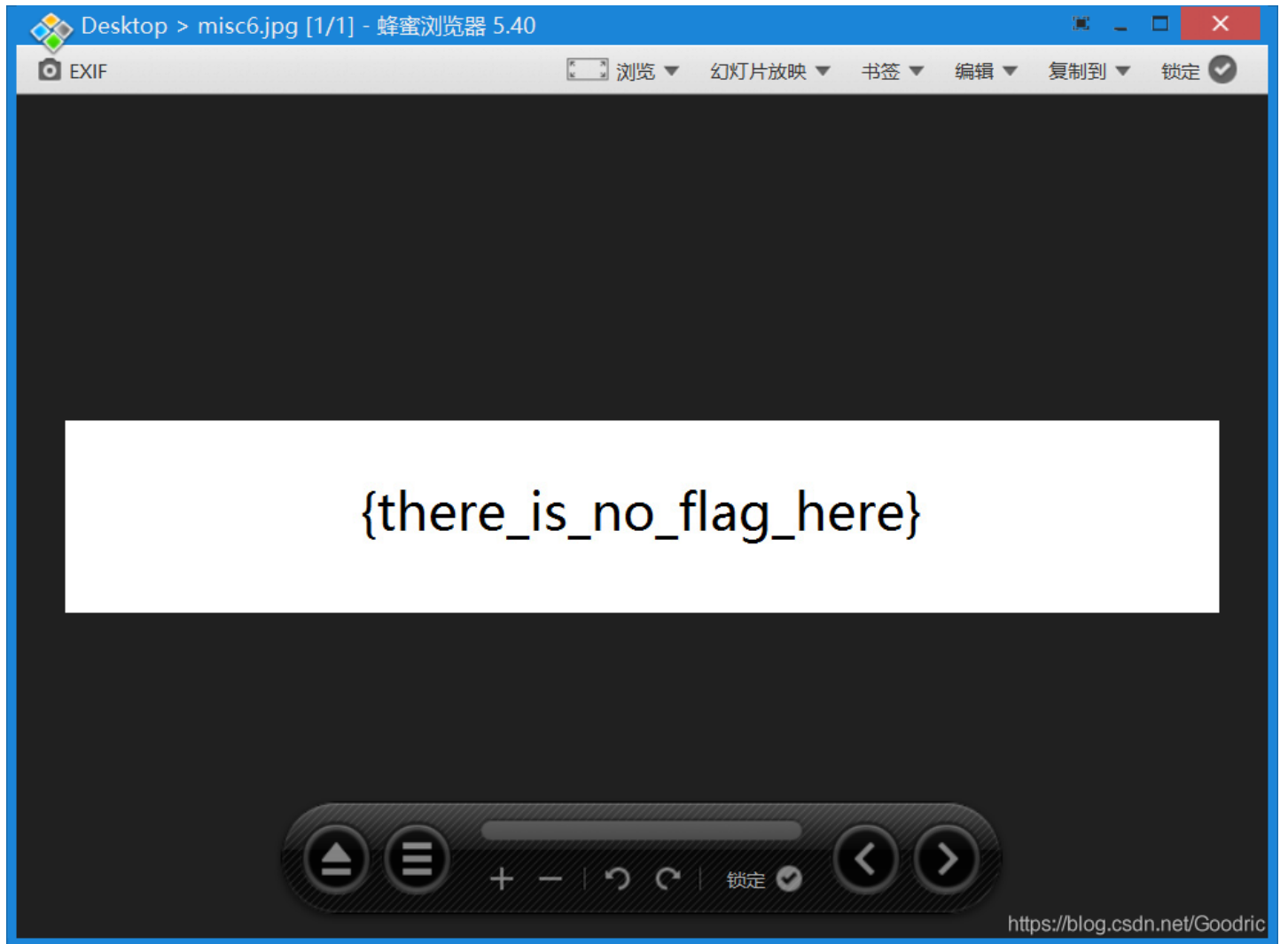
```
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29 S&FFBET3> 答案BOTNULNULNULNULIEND脰`修tfshow{2a476b4011805f1a8e4b906c8f84083e}
```

Normal text file | length : 3,933 | lines : 29 | Ln : 15 | Col : 18 | Pos : 2,003 | Windows (CR LF) | ANSI | /blog.csdn.net/ | INS

## misc6



下载附件，和前一题一样，是一张图片，但是 flag 信息被隐藏。



文本打开无有效信息。

用 winhex 打开搜索关键字 ctfshow 直接得到 flag 。

WinHex - [misc6.jpg]

文件(E) 编辑(E) 搜索(S) 导航(N) 查看(V) 工具(I) 专业工具(I) 选项(O) 窗口(W) 帮助(H)

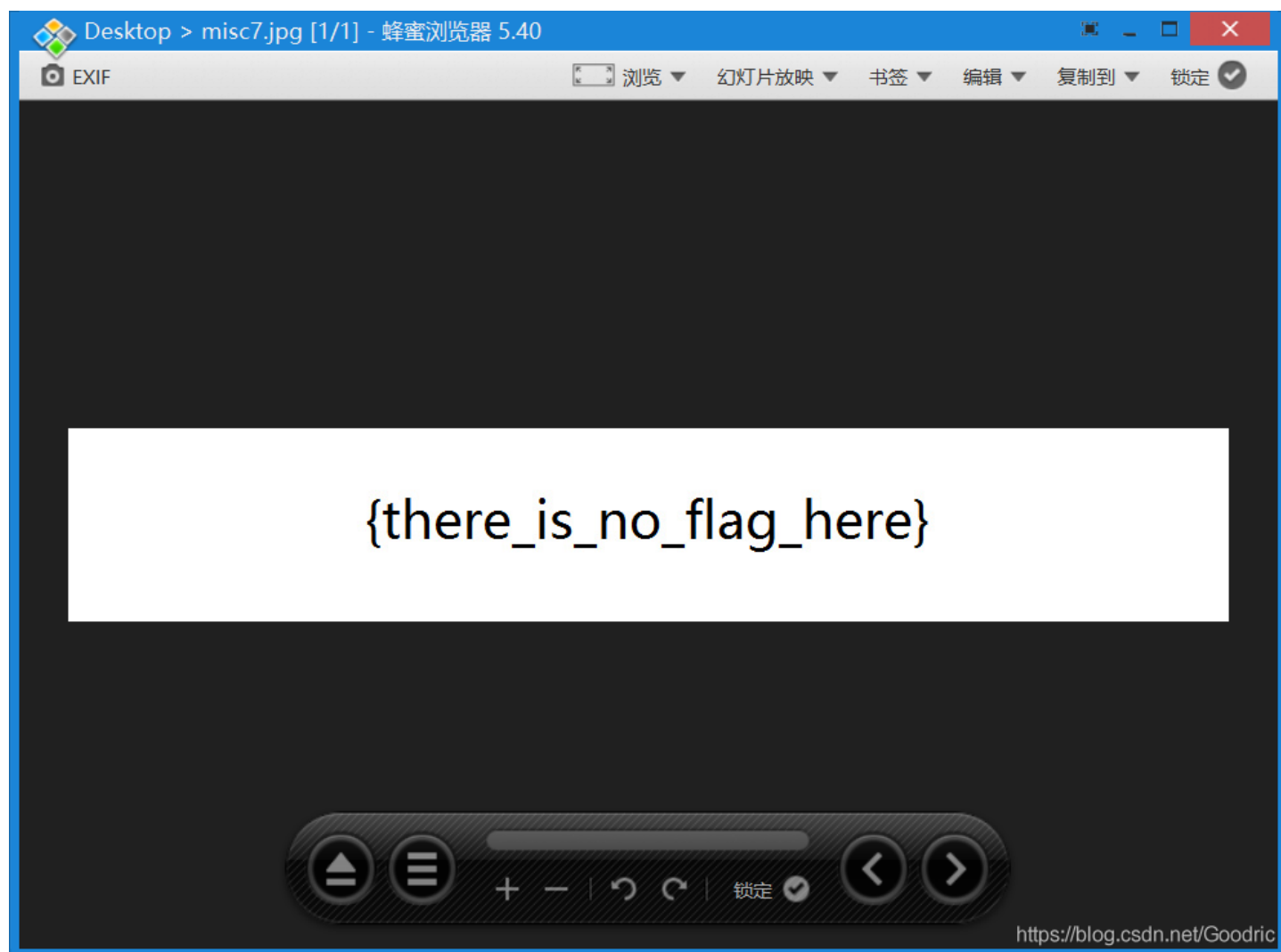
Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	ANSI ASCII
00001488	16	9A	6D	D8	2A	F5	2B	AD	AD	77	A6	D0	D6	B2	AD	EC	šmø*õ+--w ĐÖ°-i
00001504	68	77	A4	C6	D7	5F	E8	FF	00	E0	D5	94	92	49	4C	5D	hwæx_èý àõ""IL]
00001520	5D	6E	73	5E	E6	87	39	84	96	38	89	2D	24	6D	3B	7F	]ns^æ±9,,-8%-šm;
00001536	77	DA	54	92	49	25	29	24	92	49	4A	49	24	92	53	FF	wÚT'I%)Š'IJIS'Sý
00001552	D9	FF	ED	0D	F2	50	68	6F	74	6F	73	68	6F	70	20	33	Ûýí òPhotoshop 3
00001568	2E	30	00	38	42	49	4D	04	25	00	00	00	00	00	10	00	.0 8BIM %
00001584	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	38	8
00001600	42	49	4D	04	3A	00	00	00	00	00	D7	00	00	00	10	00	BIM : x
00001616	00	00	01	00	00	00	00	00	0B	70	72	69	6E	74	4F	75	printOu
00001632	74	70	75	74	00	00	00	05	00	00	00	00	50	73	74	53	tput PstS
00001648	62	6F	6F	6C	01	00	00	00	00	49	6E	74	65	65	6E	75	bool Inteenu
00001664	6D	00	00	00	00	49	6E	74	65	00	00	00	00	49	6D	67	m Inte Img
00001680	20	00	00	00	0F	63	74	66	73	68	6F	77	7B	64	35	65	ctfshow{d5e
00001696	39	33	37	61	65	66	62	30	39	31	64	33	38	65	37	30	937aefb091d38e70
00001712	64	39	32	37	62	38	30	65	31	65	32	65	61	7D	00	01	d927b80e1e2ea}
00001728	00	00	00	00	0F	70	72	69	6E	74	50	72	6F	6F	66	66	printProof
00001744	53	65	74	75	70	4F	62	6A	63	00	00	00	05	68	21	68	SetupObjc h!h
00001760	37	8B	BE	7F	6E	00	00	00	00	00	0A	70	72	6F	6F	66	7<¼ n proof
00001776	53	65	74	75	70	00	00	00	01	00	00	00	00	42	6C	74	Setup Blt
00001792	6E	65	6E	75	6D	00	00	00	0C	62	75	69	6C	74	69	6E	nenum builtin
00001808	50	72	6F	6F	66	00	00	00	09	70	72	6F	6F	66	43	4D	Proof proofCM
00001824	59	4B	00	38	42	49	4D	04	3B	00	00	00	00	02	2D	00	YK 8BIM ; -
00001840	00	00	10	00	00	00	01	00	00	00	00	00	12	70	72	69	pri
00001856	6E	74	4F	75	74	70	75	74	4F	70	74	69	6F	6E	73	00	ntOutputOptions
00001872	00	00	17	00	00	00	00	43	70	74	6E	62	6F	6F	6C	00	Cptnbool
00001888	00	00	00	00	43	6C	62	72	62	6F	6F	6C	00	00	00	00	Clbrbool
00001904	00	52	67	73	4D	62	6F	6F	6C	00	00	00	00	00	43	72	RgsMbool Cr
00001920	6E	43	62	6F	6F	6C	00	00	00	00	00	43	6E	74	43	62	nCbool CntCb
00001936	6F	6F	6C	00	00	00	00	00	4C	62	6C	73	62	6F	6F	6C	ool Lblsbool
00001952	00	00	00	00	00	4E	67	74	76	62	6F	6F	6C	00	00	00	Ngtvbool
00001968	00	00	45	6D	6C	44	62	6F	6F	6C	00	00	00	00	00	49	EmlDbool I

<https://blog.csdn.net/Goodric>

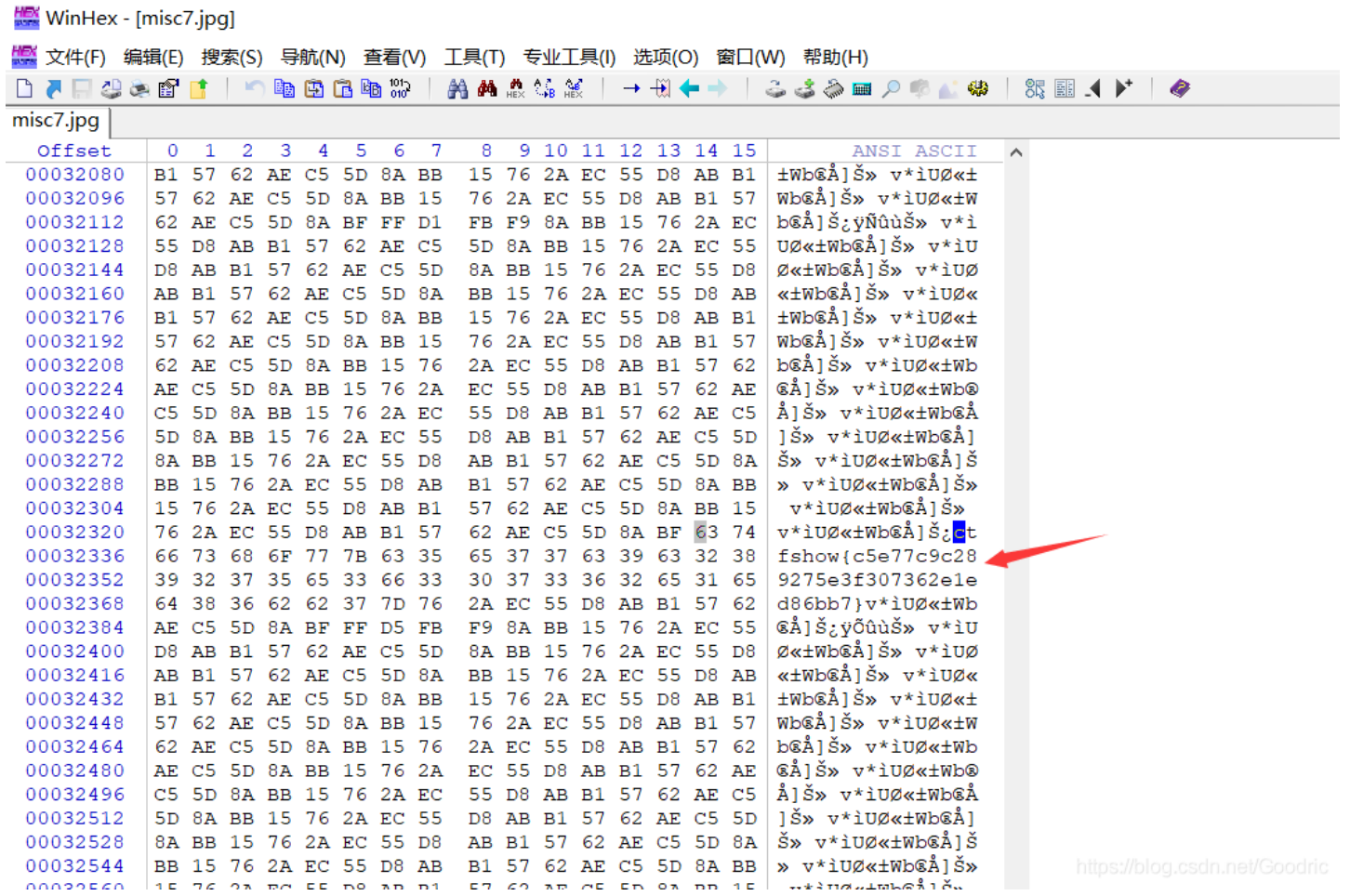
misc7

下载附件，还是这张图片。

题目还有一句提示：flag在图片文件信息中。



还是可以用 winhex 打开，搜索关键字得到 flag。



## misc8

题目描述： flag在图片文件中图片文件中。

下载附件还是前面那张图片的样子。

根据题目描述，在图片文件中，尝试用 binwalk 或 foremost 对其进行分离。

先尝试用 binwalk 工具：

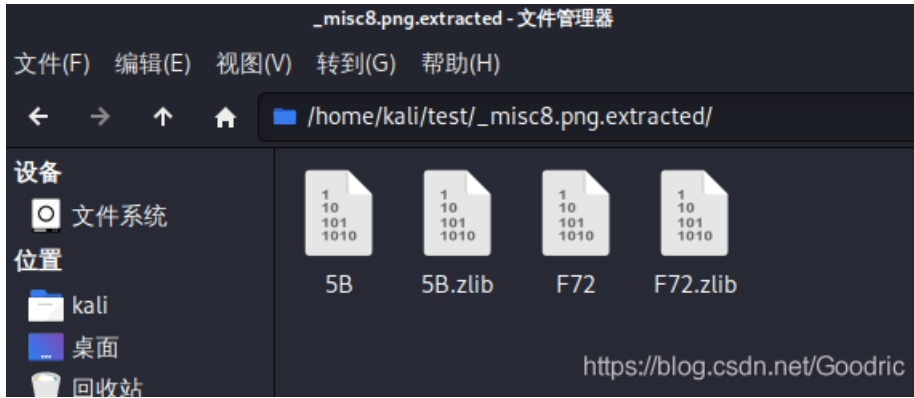
扫描文件是否可分离

```
binwalk misc8.png
```

分离出前面扫描的结果。

```
binwalk -e misc8.png
```

看分离出的结果感觉无法利用。

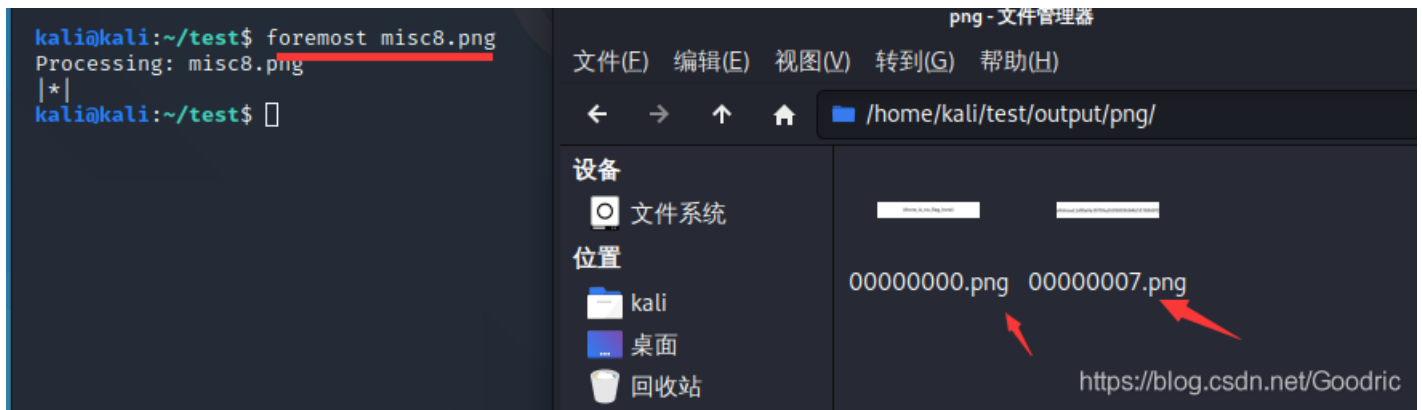


再尝试用 foremost :

```
foremost misc8.png
```

分离结果得到两张图片，其中一张打开即显示 flag 。

```
ctfshow{1df0a9a3f709a2605803664b55783687}
```

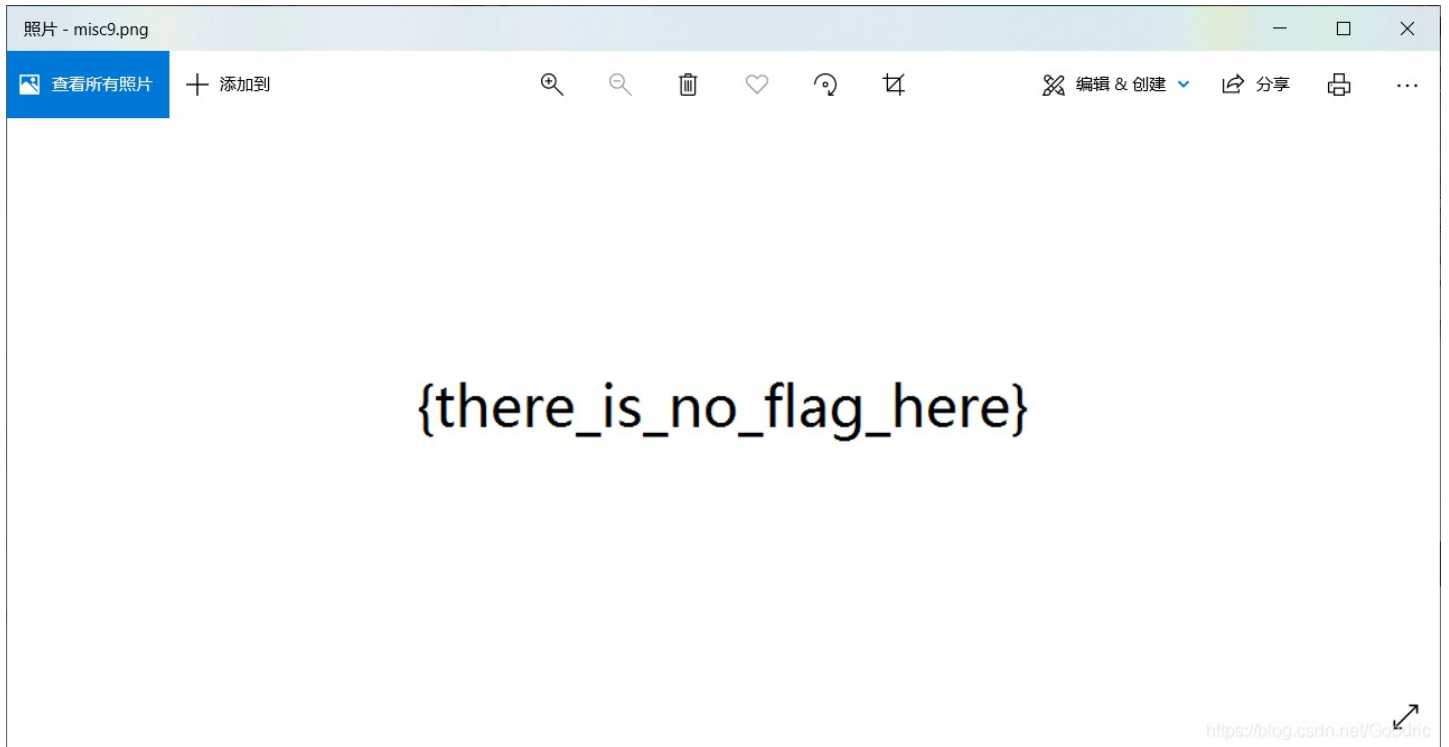


—  
—

## misc9

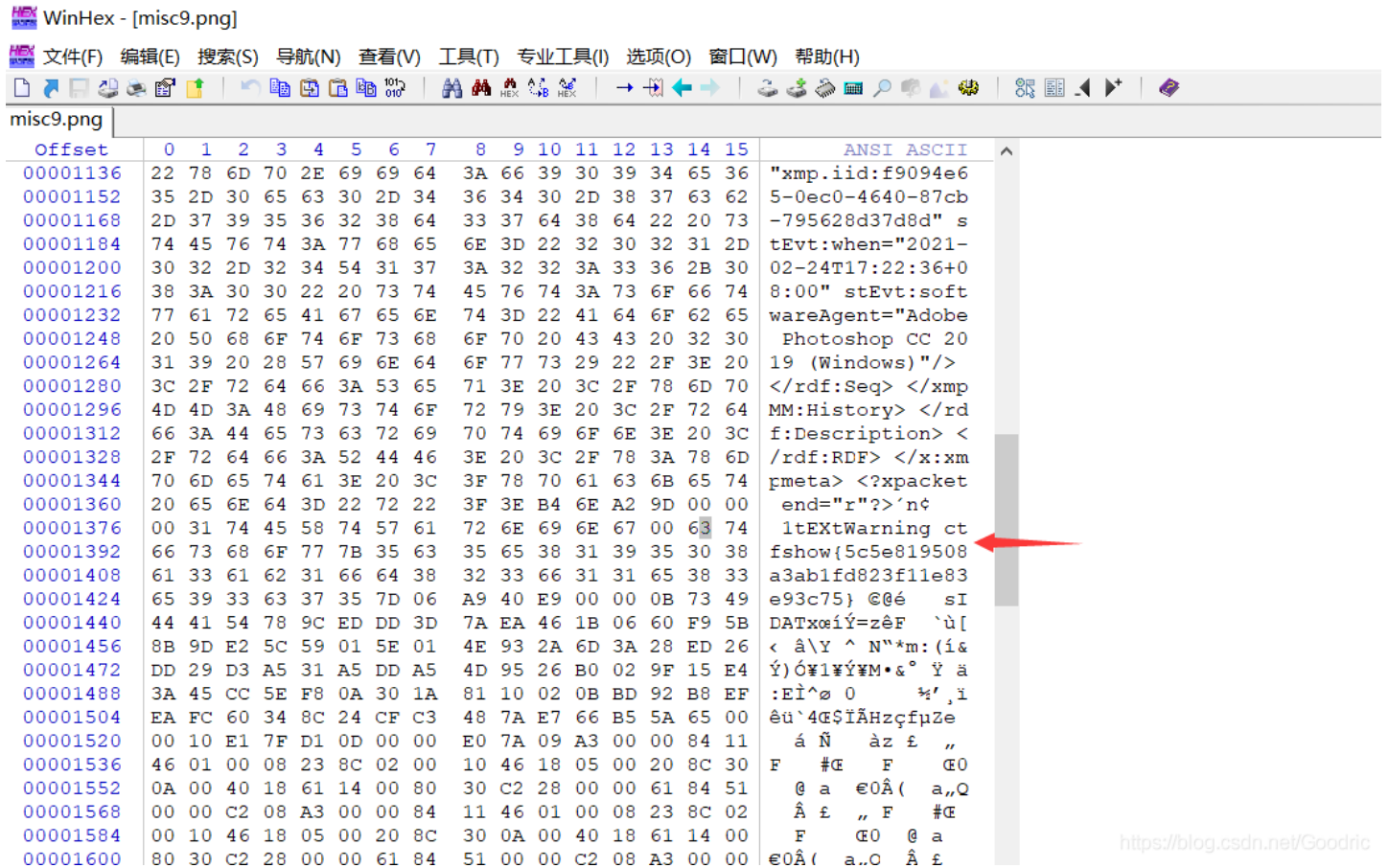
题目描述：flag在图片块里。

下载附件和前面一样，是那张图片的样子。

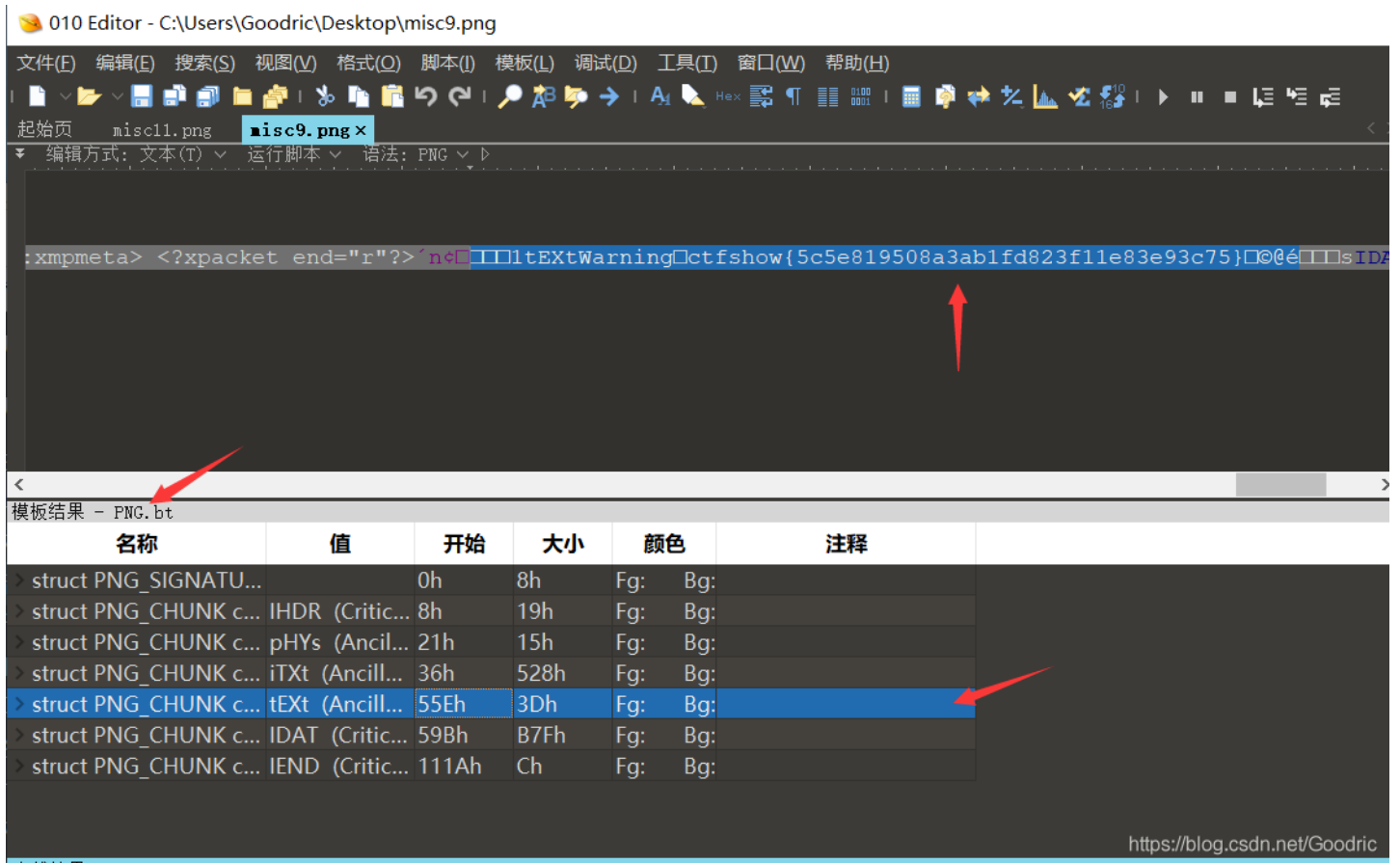


说是在图片块里，但是图片块是什么，暂时让我无法理解，有答案说叫数据块。

不过这里还是只要用 winhex 打开即可看到 flag 。



从数据块角度的话，用 010editor 打开文件，运行模板 png.bit，编辑方式改为文本，点击下面对应存在的数据，看到文本对应显示了 flag。



010 Editor - C:\Users\Goodric\Desktop\misc9.png

文件(F) 编辑(E) 搜索(S) 视图(V) 格式(O) 脚本(I) 模板(L) 调试(D) 工具(T) 窗口(W) 帮助(H)

起始页 misc11.png misc9.png x

编辑方式: 文本(T) 运行脚本 语法: PNG

```
:xmpmeta> <?xpacket end="r"?'> n<[ ]tEXtWarning[ ]ctfshow{5c5e819508a3ab1fd823f11e83e93c75}[ ]@@é[ ]sIDZ
```

模板结果 - PNG.bt

名称	值	开始	大小	颜色	注释
> struct PNG_SIGNATU...		0h	8h	Fg: Bg:	
> struct PNG_CHUNK c... IHDR (Criti...		8h	19h	Fg: Bg:	
> struct PNG_CHUNK c... pHYS (Ancil...		21h	15h	Fg: Bg:	
> struct PNG_CHUNK c... iTXt (Ancill...		36h	528h	Fg: Bg:	
> struct PNG_CHUNK c... tEXt (Ancill...		55Eh	3Dh	Fg: Bg:	
> struct PNG_CHUNK c... IDAT (Criti...		59Bh	B7Fh	Fg: Bg:	
> struct PNG_CHUNK c... IEND (Criti...		111Ah	Ch	Fg: Bg:	

<https://blog.csdn.net/Goodric>

## misc10

题目描述: flag在图片数据里。

附件还是那张图片。

只去大致得了解了一下图片的数据这方面的知识，还不是很懂。

不过有得知 binwalk 可以把这题一些数据直接提取出来。

我在 Windows 下用 binwalk 对图片进行扫描提取。(在 kali 下自带)

关于 binwalk 在 Windows 下的使用可参考:

<https://blog.csdn.net/Goodric/article/details/117845492?spm=1001.2014.3001.5501>

扫描

```
python3 binwalk C:\Users\Goodric\Desktop\misc10.png
```

提取分离

```
python3 binwalk -e C:\Users\Goodric\Desktop\misc10.png
```

```
管理员: binwalk
Microsoft Windows [版本 10.0.19041.1083]
(c) Microsoft Corporation. 保留所有权利。

E:\Python39\Scripts>python3 binwalk C:\Users\Goodric\Desktop\misc10.png

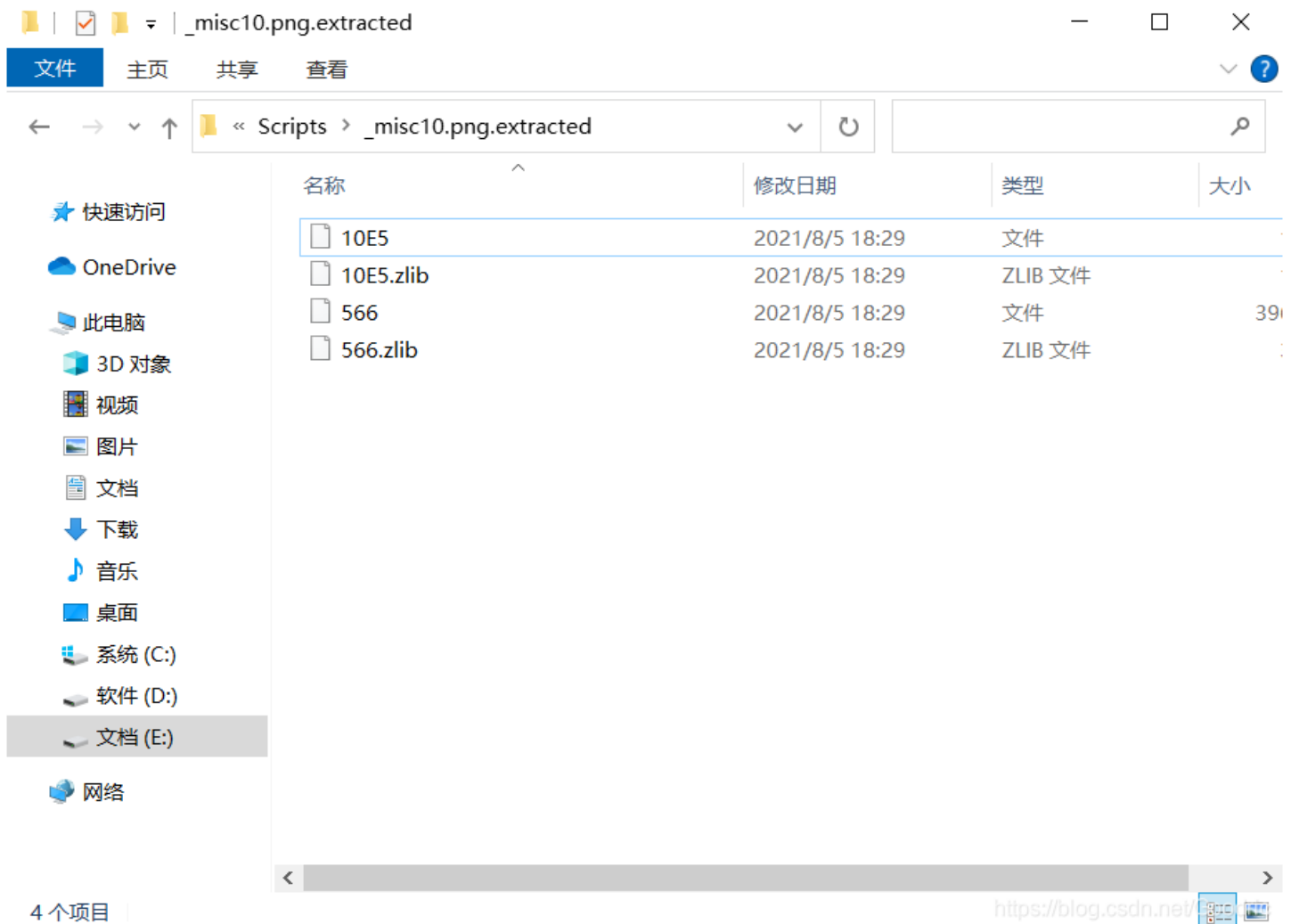
DECIMAL      HEXADECIMAL    DESCRIPTION
-----
0            0x0            PNG image, 900 x 150, 8-bit/color RGB, non-interlaced
1382        0x566         Zlib compressed data, default compression
4325        0x10E5        Zlib compressed data, default compression

E:\Python39\Scripts>python3 binwalk -e C:\Users\Goodric\Desktop\misc10.png

DECIMAL      HEXADECIMAL    DESCRIPTION
-----
0            0x0            PNG image, 900 x 150, 8-bit/color RGB, non-interlaced
1382        0x566         Zlib compressed data, default compression
4325        0x10E5        Zlib compressed data, default compression

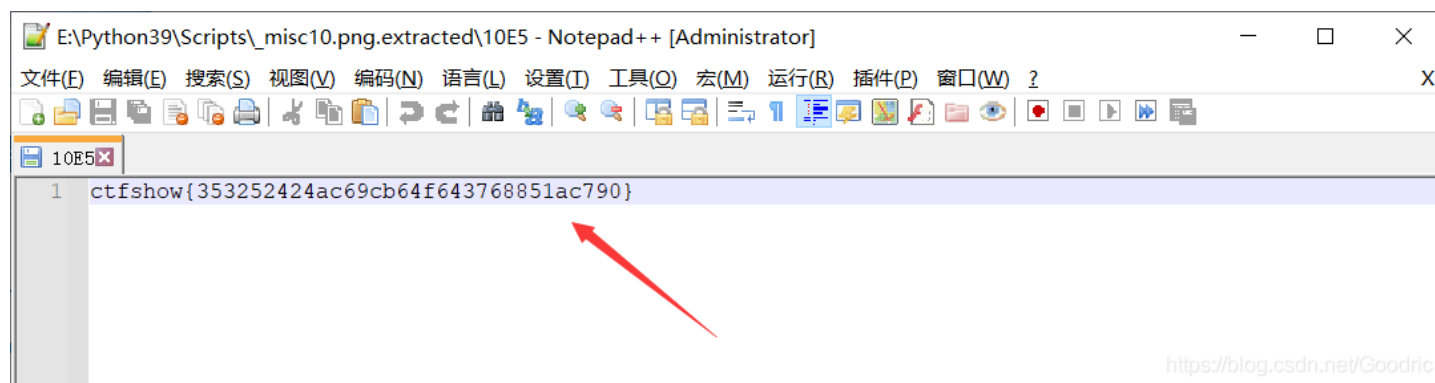
E:\Python39\Scripts>_
```

在 python 安装目录的 /Script 下就会得到提取后的文件夹。





文本格式打开打开第一个文件 10E5 得到 flag。



E:\Python39\Scripts\\_misc10.png.extracted\10E5 - Notepad++ [Administrator]

文件(F) 编辑(E) 搜索(S) 视图(V) 编码(N) 语言(L) 设置(I) 工具(O) 宏(M) 运行(R) 插件(P) 窗口(W) ?

10E5

```
1 ctshow{353252424ac69cb64f643768851ac790}
```

<https://blog.csdn.net/Goodric>



[创作打卡挑战赛](#) >

[赢取流量/现金/CSDN周边激励大奖](#)