ctfshow 做题 MISC入门 模块1-10

原创

Goodric 于 2021-08-15 11:26:17 发布

分类专栏： 做题 文章标签： ctfshow misc

ctfshow 做题 MISC入门 模块 1-10

---

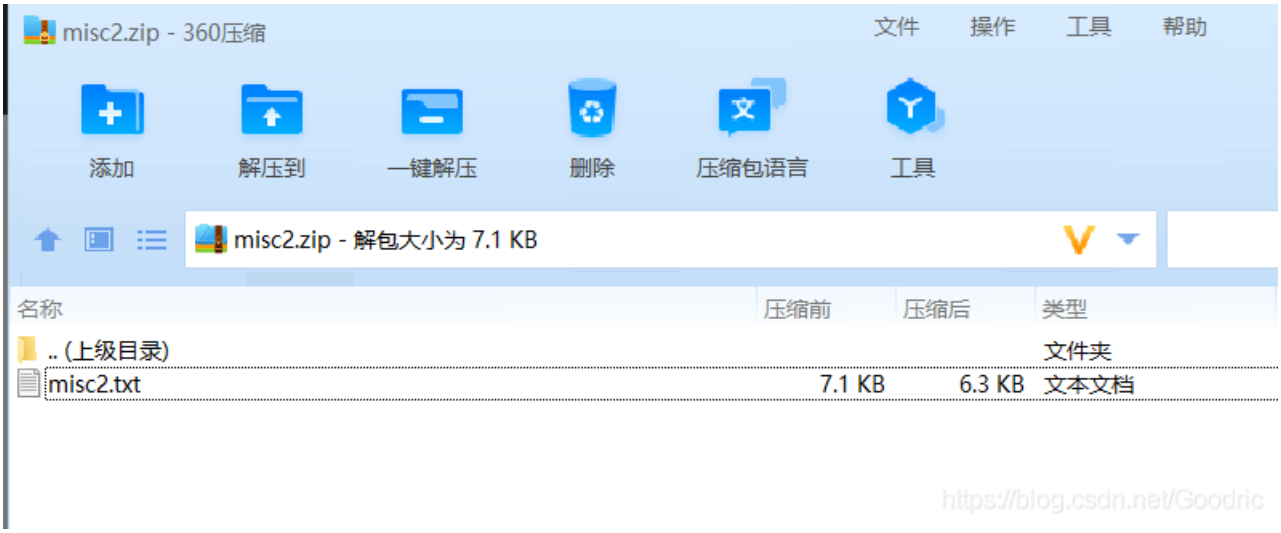
misc1

下载附件压缩包，里面为一张图片，直接得到 flag 。
ctfshow{22f1fb91fc4169f1c9411ce632a0ed8d}

---

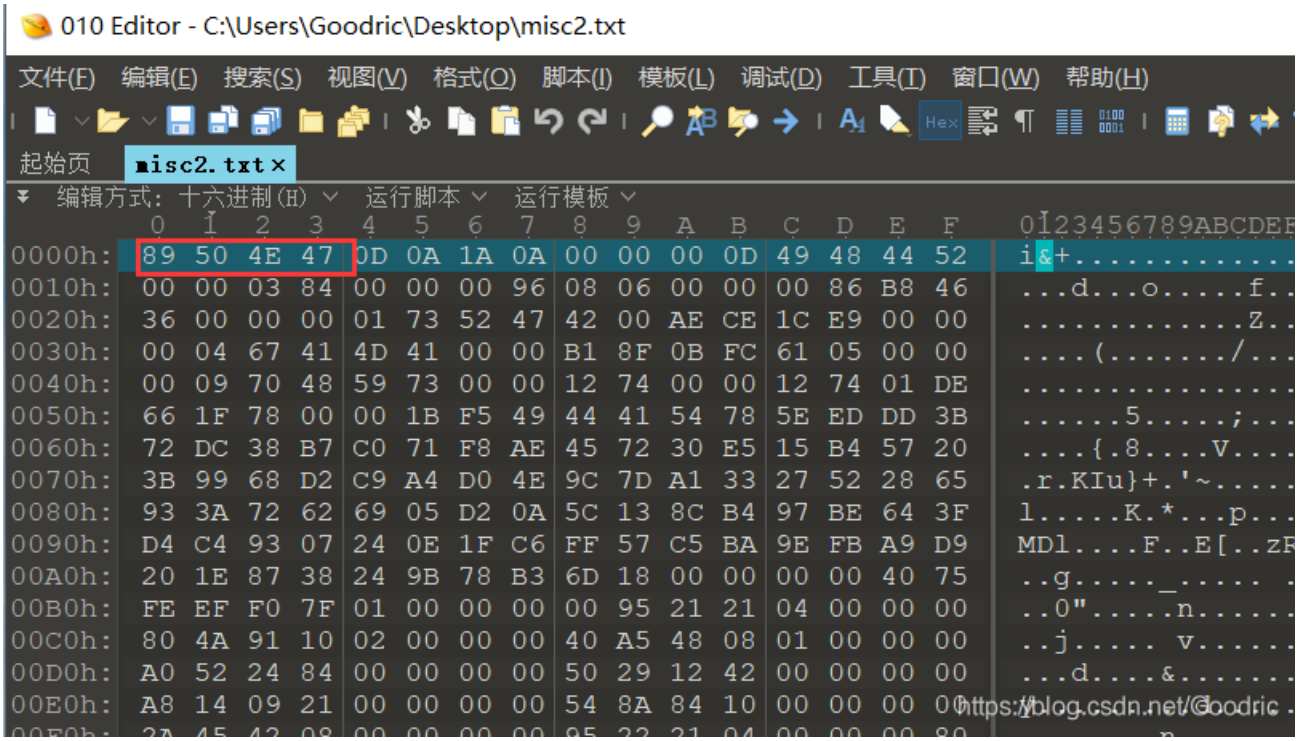
misc2

下载的压缩包附件里有一个 txt 文件。

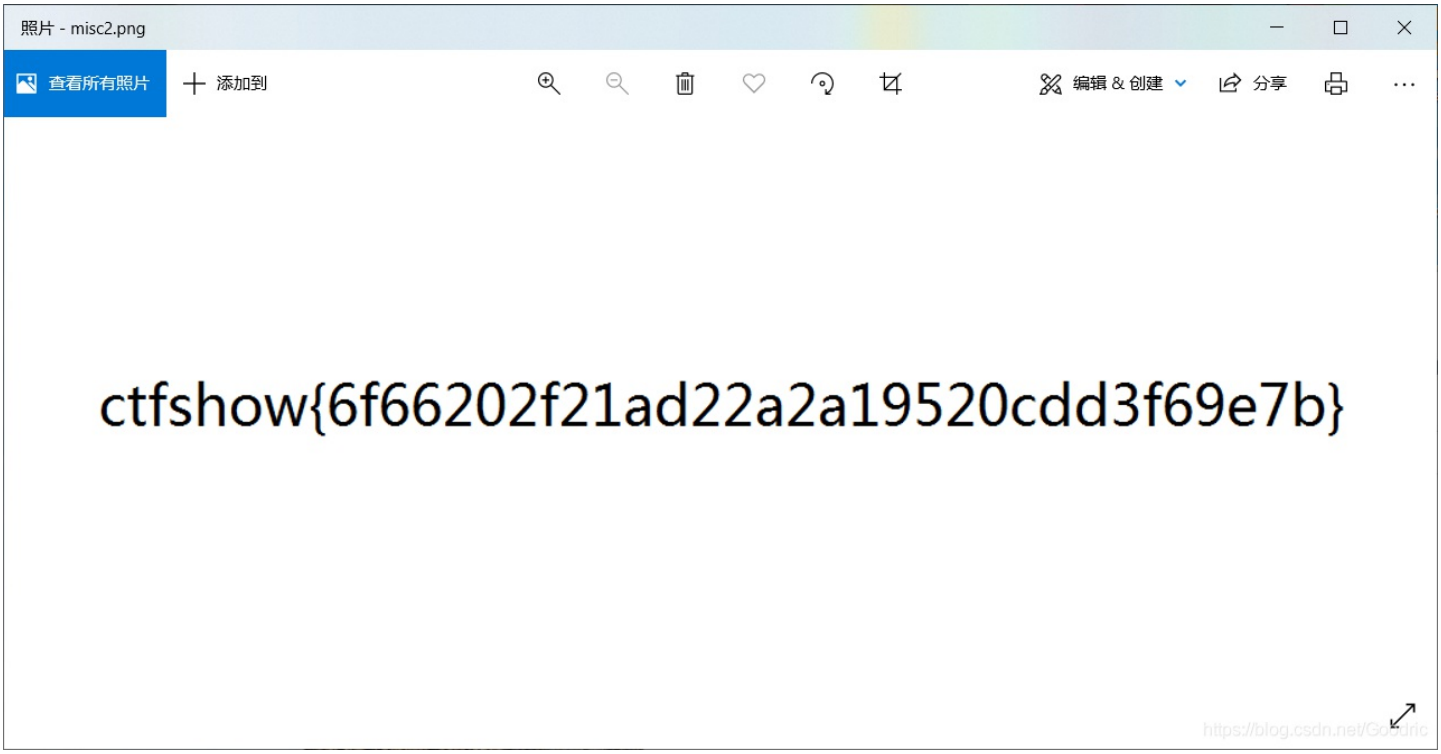


先直接打开文本模式查看，未发现有用消息。

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。



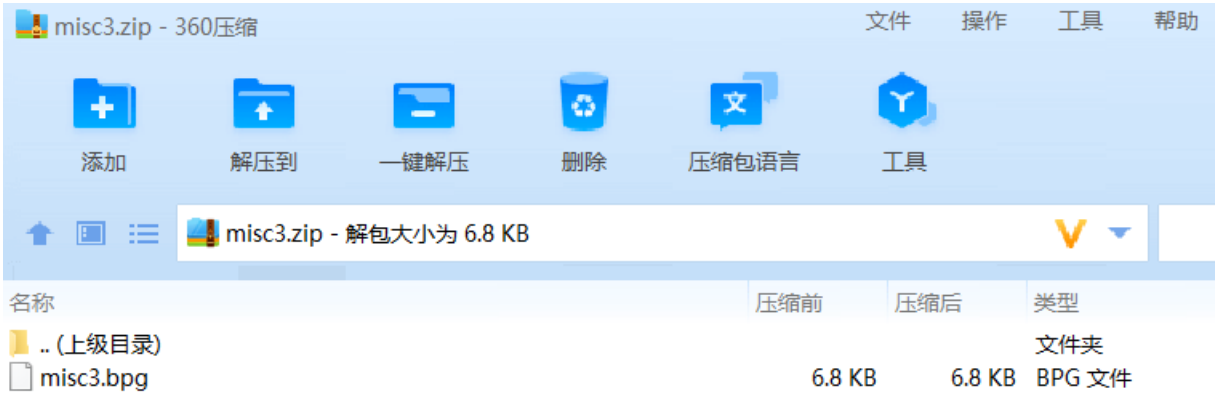
把文件后缀改为 .png 打开为一张图片，得到 flag 。
ctfshow{6f66202f21ad22a2a19520cdd3f69e7b}



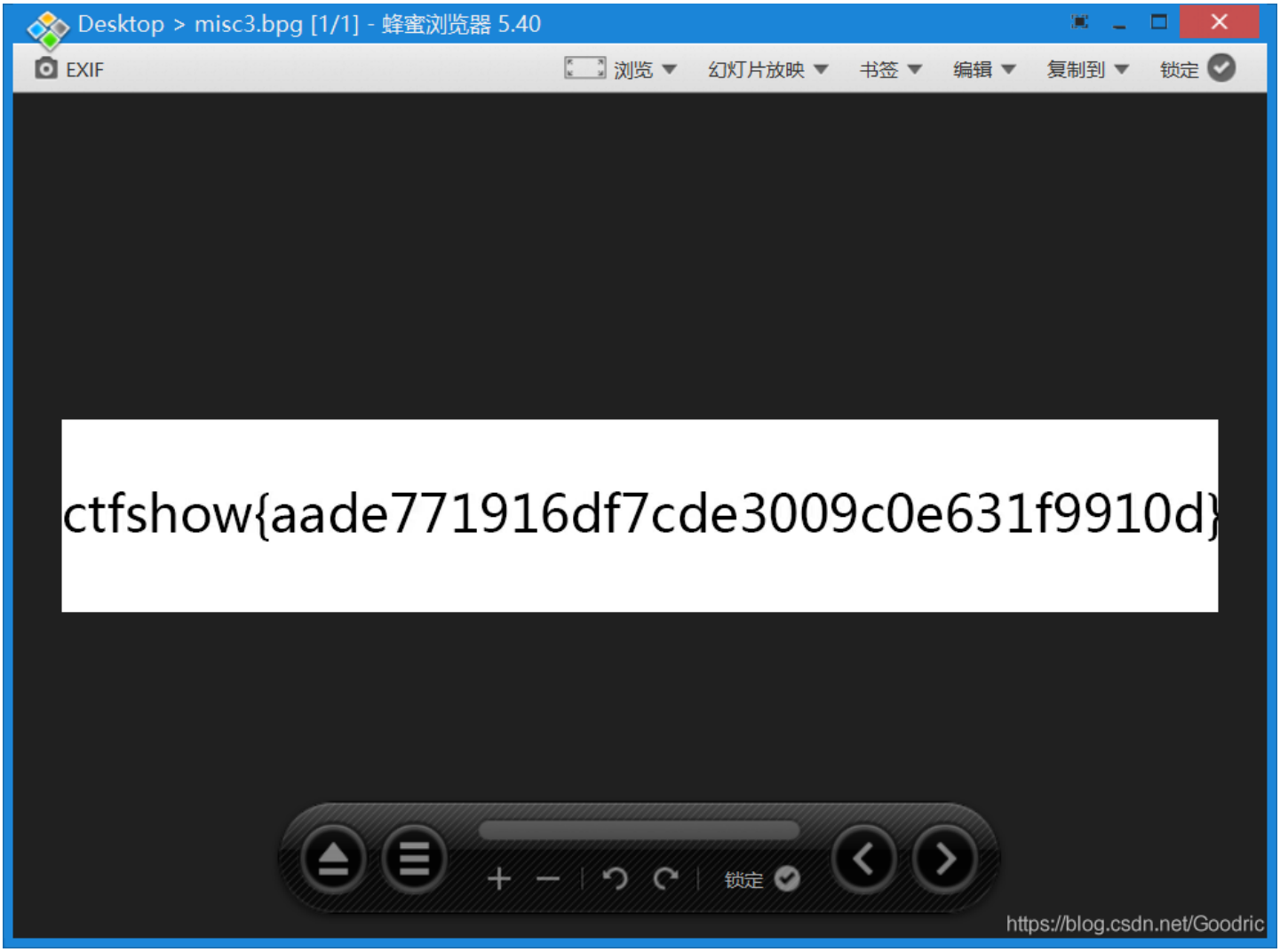
---

misc3

下载的压缩包附件里有一个文件 misc3.bpg （打开方式未知）



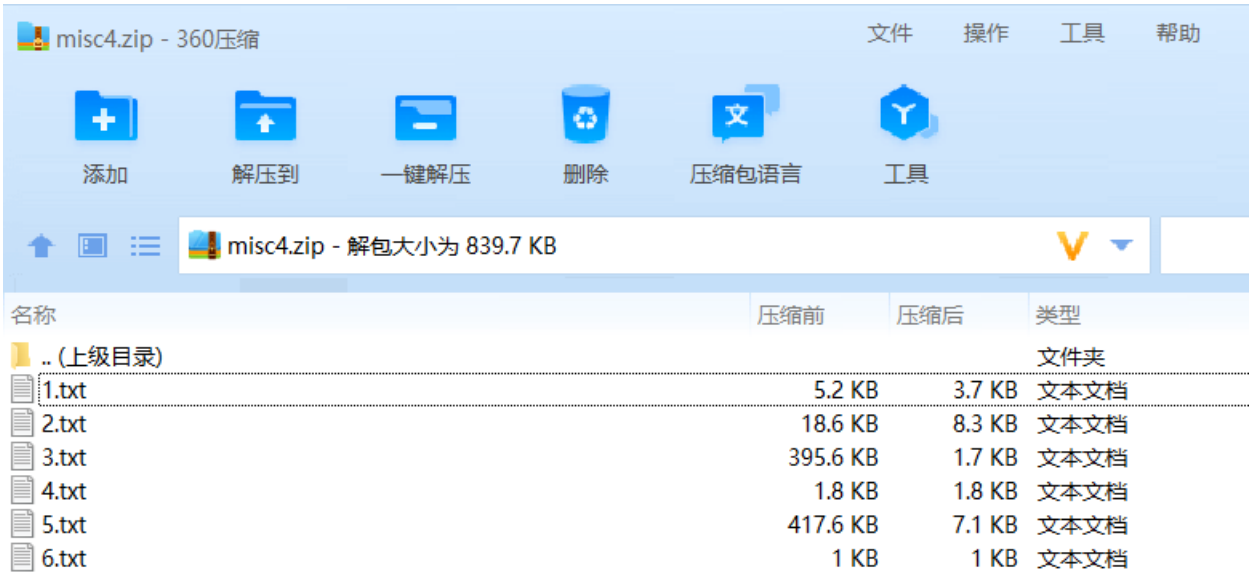
常规先用文本打开查看，未发现有效信息。
百度查了 bpg 格式的后缀就是一种图片的格式，这种格式的图片可以用许多图片查看器查看。
这里尝试下载一个图片查看器 Honeyview 。
安装好图片查看软件之后，misc3.bpg 文件得以打开。得到 flag 。
ctfshow{aade771916df7cde3009c0e631f9910d}



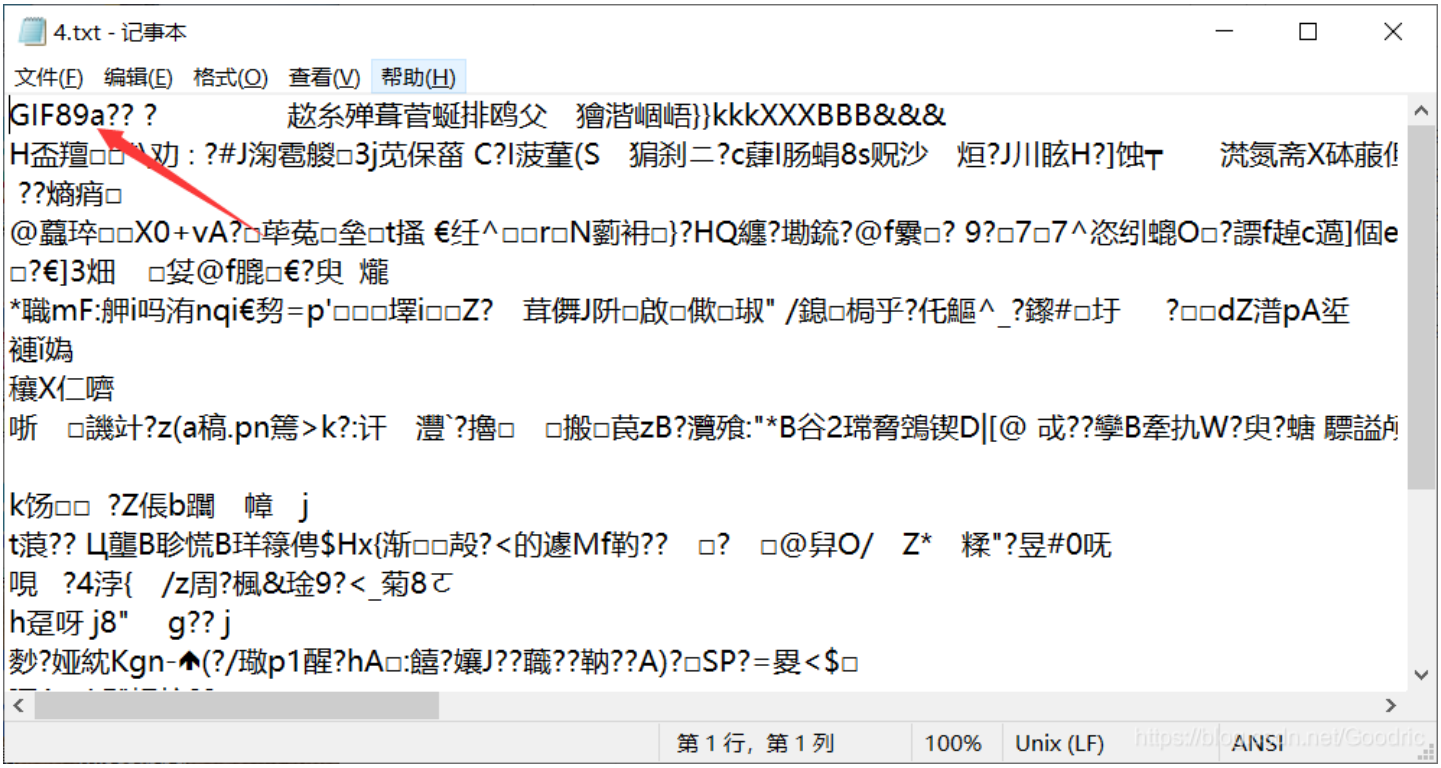
---

misc4

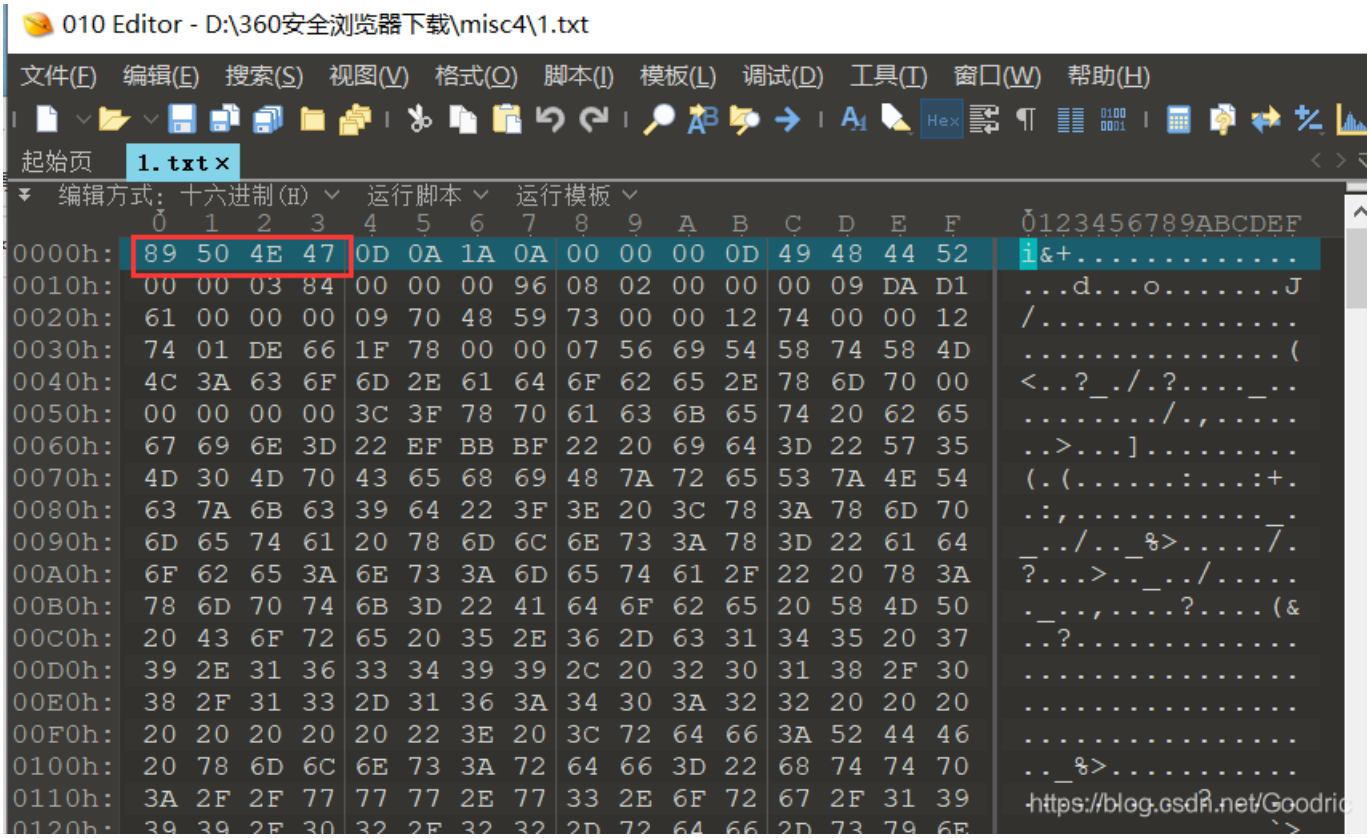
下载附件压缩包，里面有 6 个 txt 文本文件。



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。

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

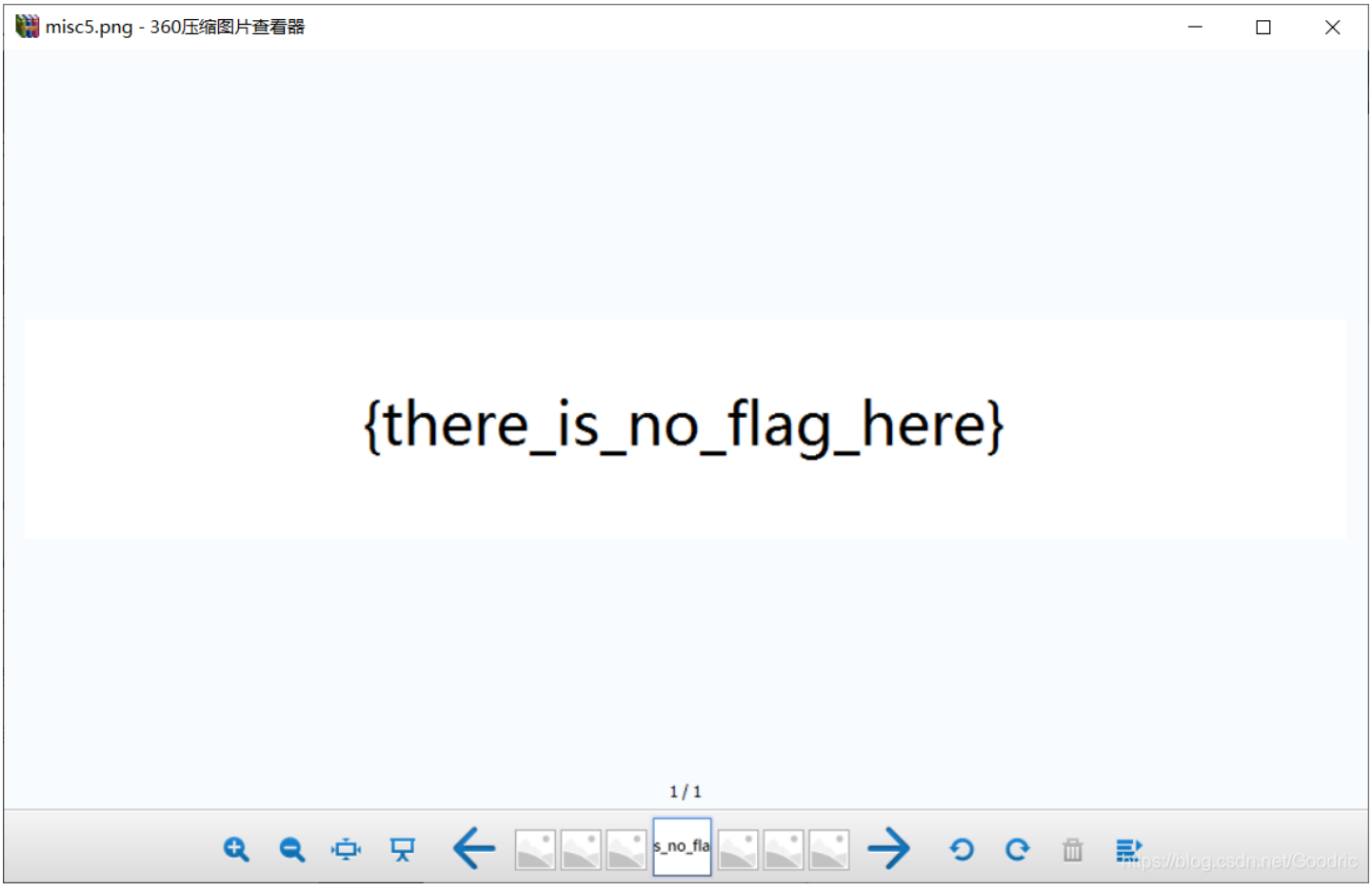
常见文件的文件头：

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

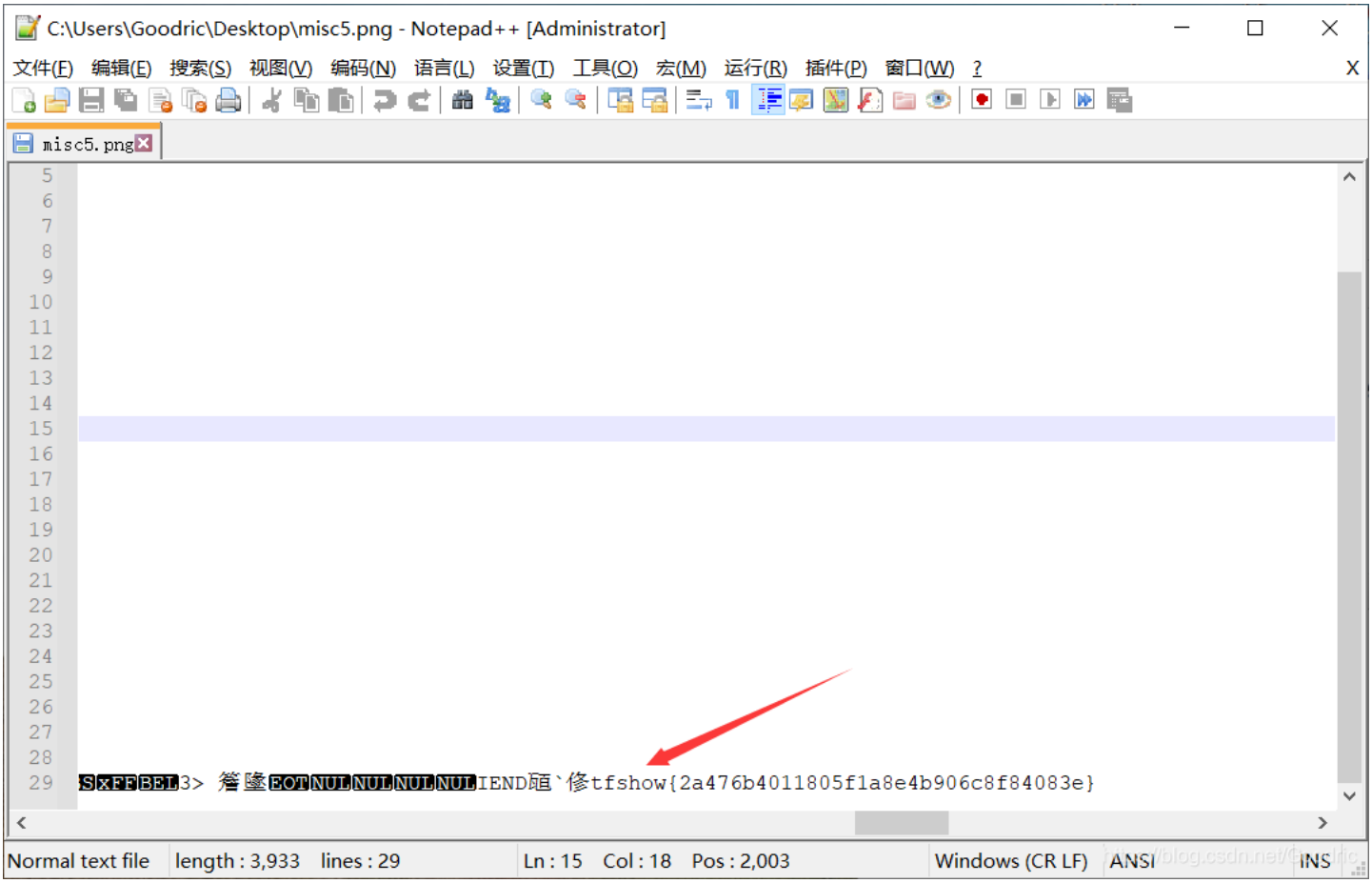
---

misc5

下载附件为一张图片，但是显示了一个错误 flag 。



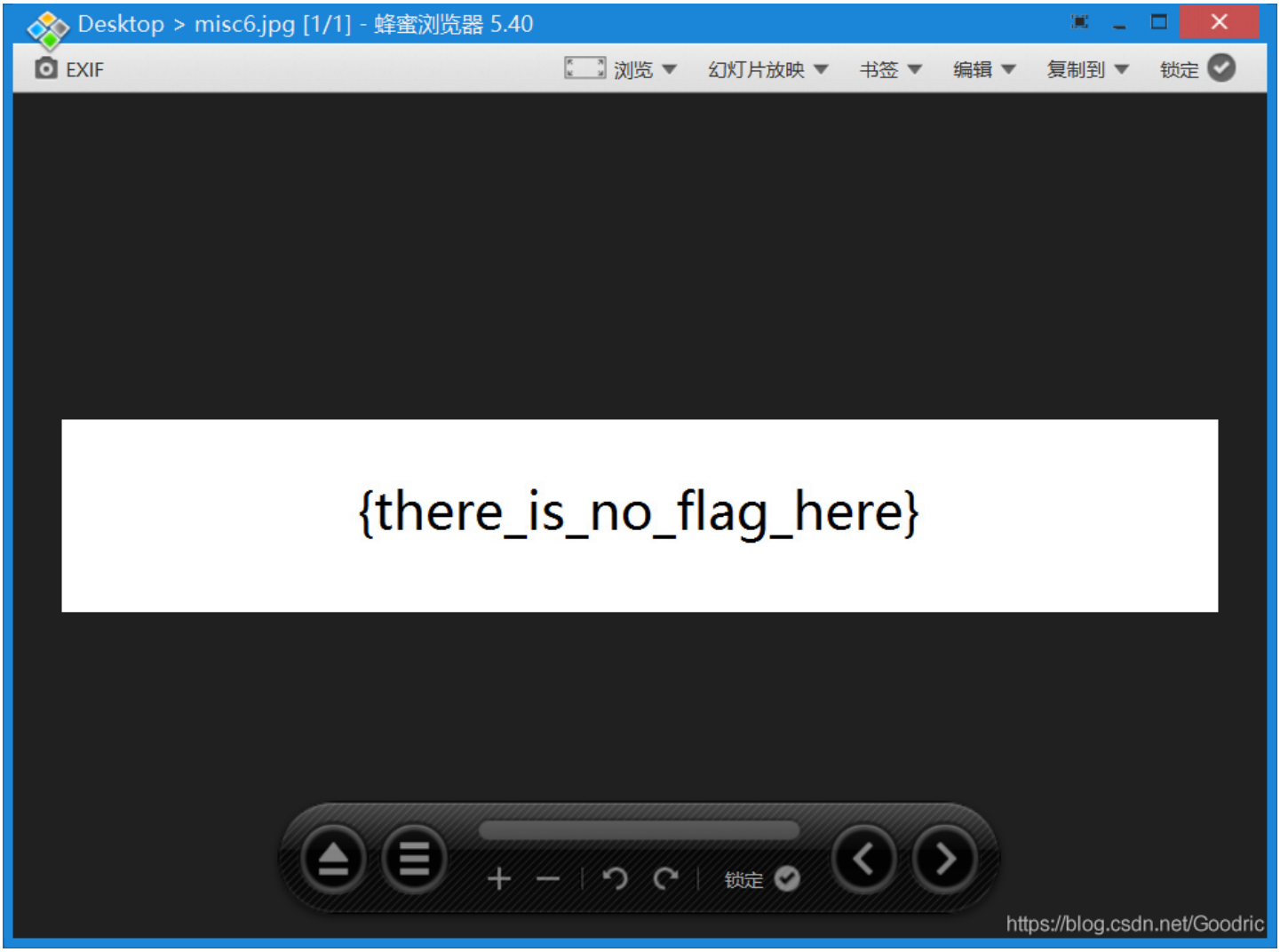
常规文本打开，在末尾得到 flag 。



---

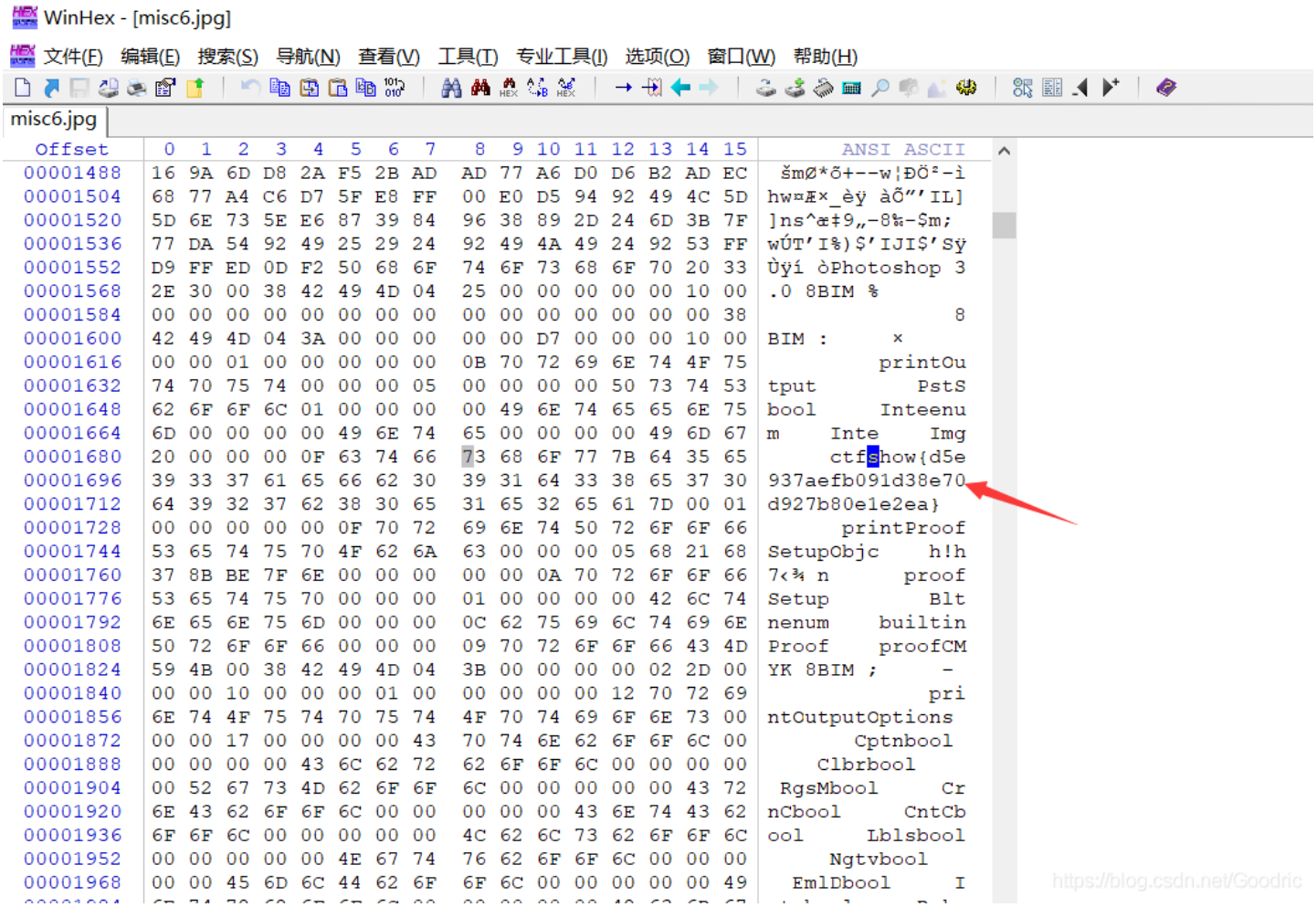
misc6

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。

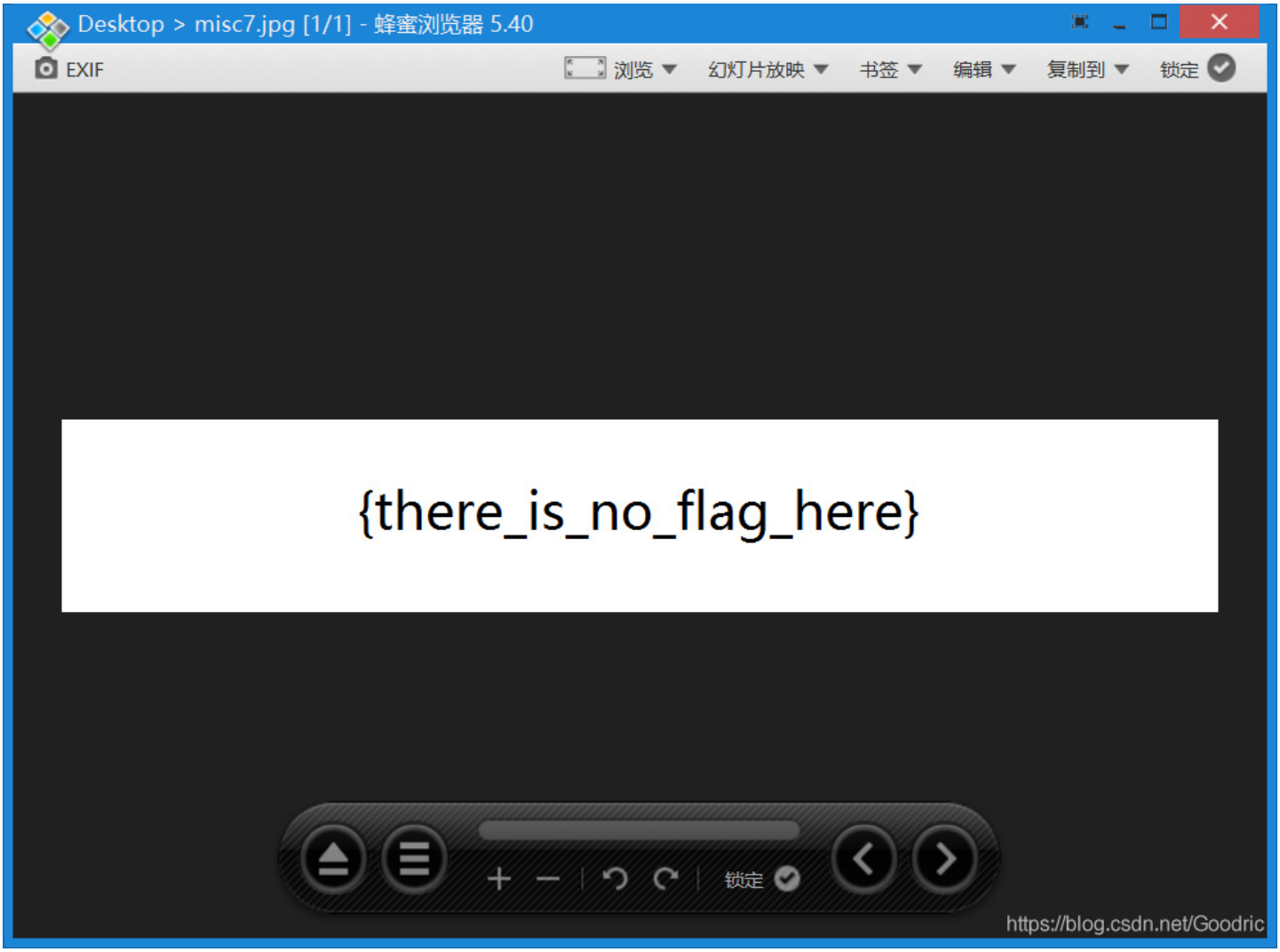
用 winhex 打开搜索关键字 ctfshow 直接得到 flag 。



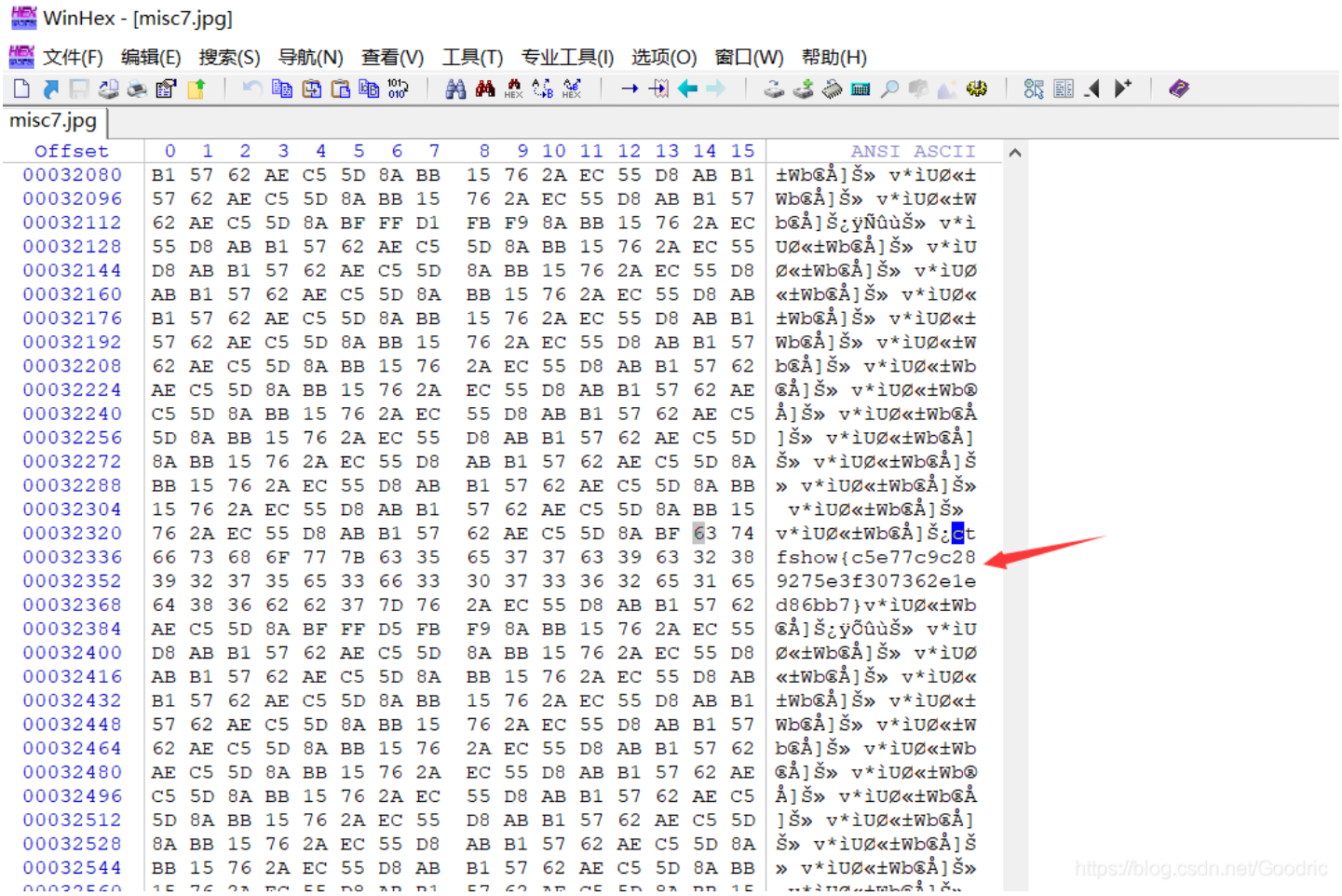
---

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

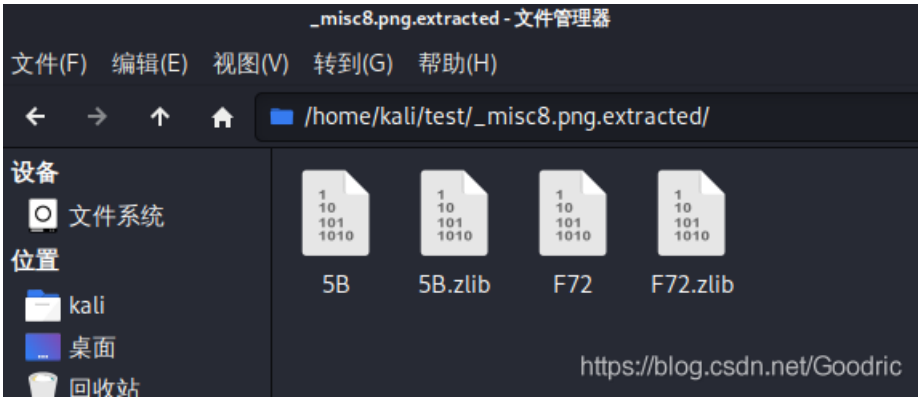
题目描述： flag在图片文件中图片文件中。
下载附件还是前面那张图片的样子。
根据题目描述，在图片文件中，尝试用 binwalk 或 foremost 对其进行分离。
先尝试用 binwalk 工具：
扫描文件是否可分离

```
binwalk misc8.png
```

分离出前面扫描的结果。

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

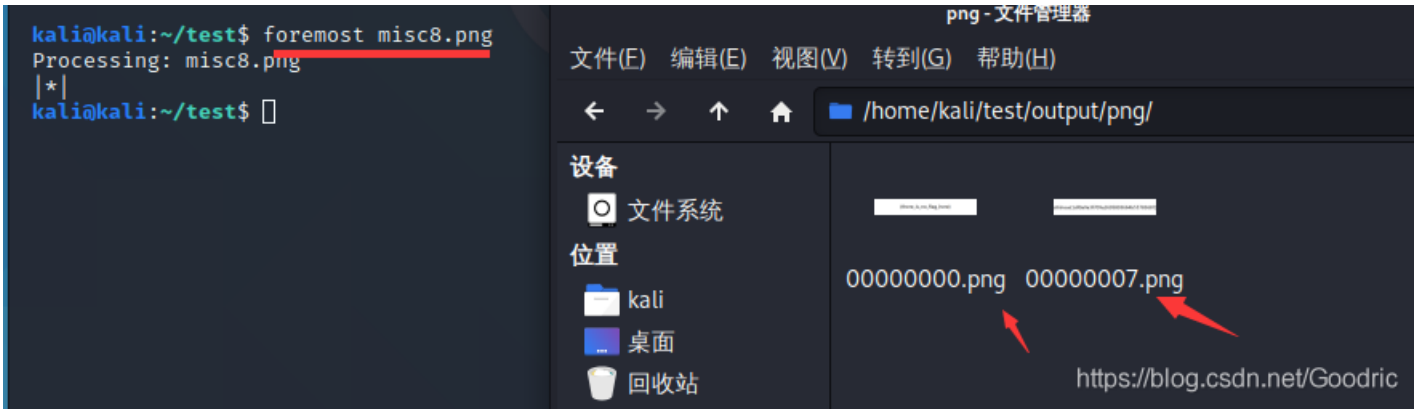
看分离出的结果感觉无法利用。



再尝试用 foremost：

```
foremost misc8.png
```

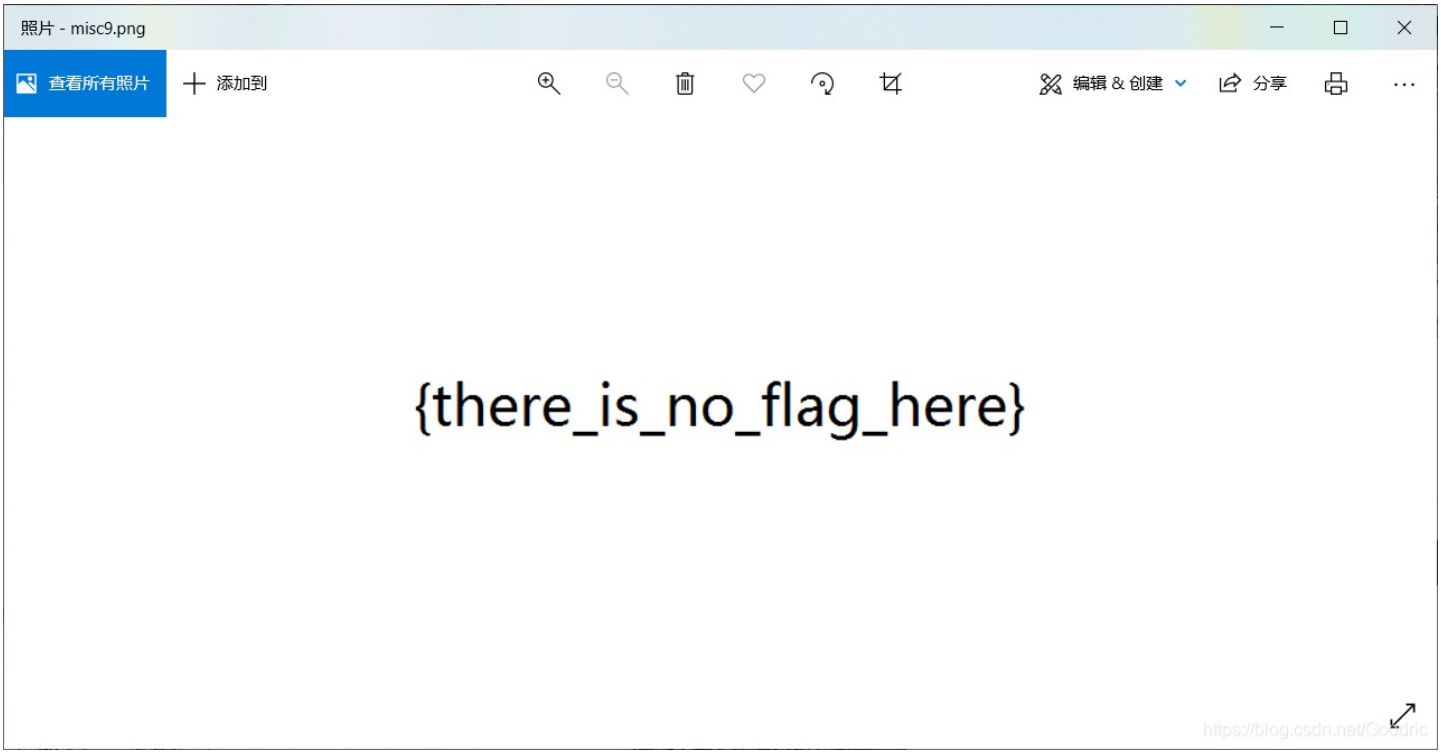
分离结果得到两张图片，其中一张打开即显示 flag 。
ctfshow{1df0a9a3f709a2605803664b55783687}



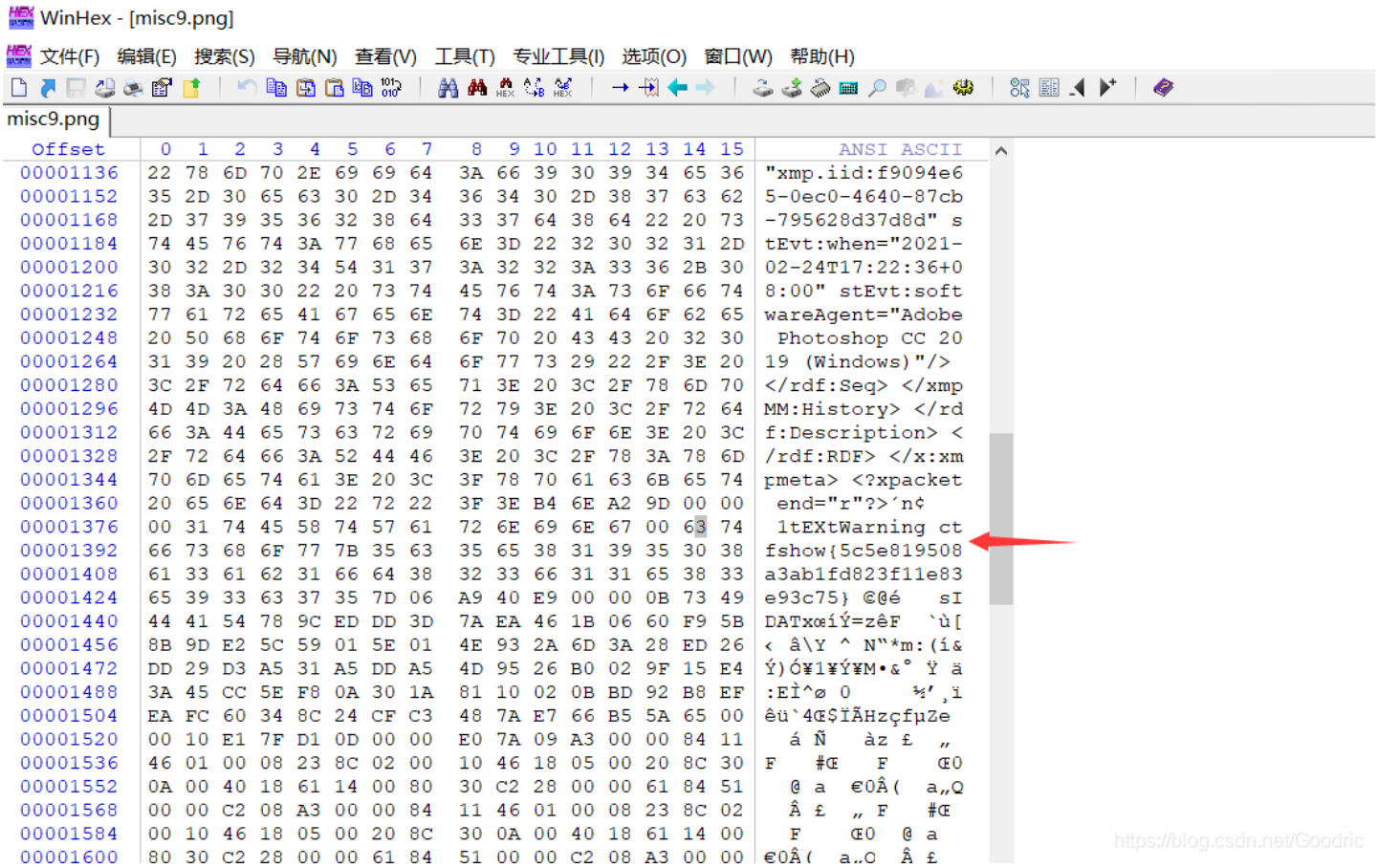
---

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。

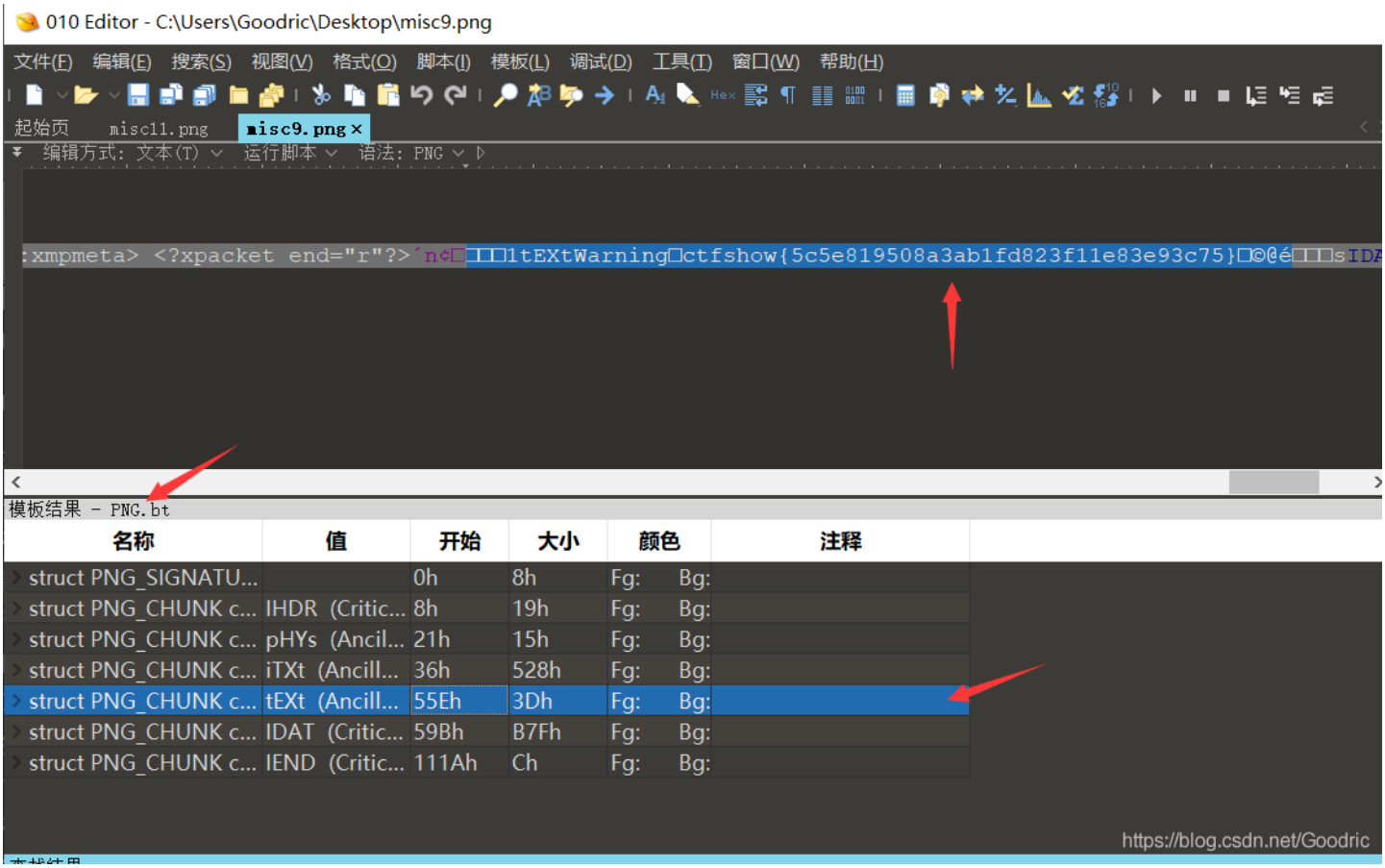


说是在图片块里，但是图片块是什么，暂时让我无法理解，有答案说叫数据块。
不过这里还是只要用 winhex 打开即可看到 flag 。



Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	ANSI ASCII
00001136	22	78	6D	70	2E	69	69	64	3A	66	39	30	39	34	65	36	"xmp.iid:f9094e6
00001152	35	2D	30	65	63	30	2D	34	36	34	30	2D	38	37	63	62	5-0ec0-4640-87cb
00001168	2D	37	39	35	36	32	38	64	33	37	64	38	64	22	20	73	-795628d37d8d" s
00001184	74	45	76	74	3A	77	68	65	6E	3D	22	32	30	32	31	2D	tEvt:when="2021-
00001200	30	32	2D	32	34	54	31	37	3A	32	32	3A	33	36	2B	30	02-24T17:22:36+0
00001216	38	3A	30	30	22	20	73	74	45	76	74	3A	73	6F	66	74	8:00" stEvt:soft
00001232	77	61	72	65	41	67	65	6E	74	3D	22	41	64	6F	62	65	wareAgent="Adobe
00001248	20	50	68	6F	74	6F	73	68	6F	70	20	43	43	20	32	30	Photoshop CC 20
00001264	31	39	20	28	57	69	6E	64	6F	77	73	29	22	2F	3E	20	19 (Windows)"/>
00001280	3C	2F	72	64	66	3A	53	65	71	3E	20	3C	2F	78	6D	70	</rdf:Seq> </xmp
00001296	4D	4D	3A	48	69	73	74	6F	72	79	3E	20	3C	2F	72	64	MM:History> </rd
00001312	66	3A	44	65	73	63	72	69	70	74	69	6F	6E	3E	20	3C	f:Description> <
00001328	2F	72	64	66	3A	52	44	46	3E	20	3C	2F	78	3A	78	6D	/rdf:RDF> </x:xm
00001344	70	6D	65	74	61	3E	20	3C	3F	78	70	61	63	6B	65	74	pmeta> <?xpacket
00001360	20	65	6E	64	3D	22	72	22	3F	3E	B4	6E	A2	9D	00	00	end="r"?>´n¢
00001376	00	31	74	45	58	74	57	61	72	6E	69	6E	67	00	63	74	1tEXtWarning ct
00001392	66	73	68	6F	77	7B	35	63	35	65	38	31	39	35	30	38	fshow{5c5e819508
00001408	61	33	61	62	31	66	64	38	32	33	66	31	31	65	38	33	a3ab1fd823f11e83
00001424	65	39	33	63	37	35	7D	06	A9	40	E9	00	00	0B	73	49	e93c75} ©@é sI
00001440	44	41	54	78	9C	ED	DD	3D	7A	EA	46	1B	06	60	F9	5B	DATxœíÝ=zêF `ù[
00001456	8B	9D	E2	5C	59	01	5E	01	4E	93	2A	6D	3A	28	ED	26	‹ â\Y ^ N"*m:(í&
00001472	DD	29	D3	A5	31	A5	DD	A5	4D	95	26	B0	02	9F	15	E4	Ý)Ó¥1¥Ý¥M•&° Ÿ ä
00001488	3A	45	CC	5E	F8	0A	30	1A	81	10	02	0B	BD	92	B8	EF	:EÌ^ø 0 ½'¸ï
00001504	EA	FC	60	34	8C	24	CF	C3	48	7A	E7	66	B5	5A	65	00	êü`4Œ$ÏÃHzçfµZe
00001520	00	10	E1	7F	D1	0D	00	00	E0	7A	09	A3	00	00	84	11	á Ñ àz £ „
00001536	46	01	00	08	23	8C	02	00	10	46	18	05	00	20	8C	30	F #Œ F Œ0
00001552	0A	00	40	18	61	14	00	80	30	C2	28	00	00	61	84	51	@ a €0Â(a„Q
00001568	00	00	C2	08	A3	00	00	84	11	46	01	00	08	23	8C	02	Â £ „ F #Œ
00001584	00	10	46	18	05	00	20	8C	30	0A	00	40	18	61	14	00	F Œ0 @ a
00001600	80	30	C2	28	00	00	61	84	51	00	00	C2	08	A3	00	00	€0Â(a„Q Â £

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 。



---

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。

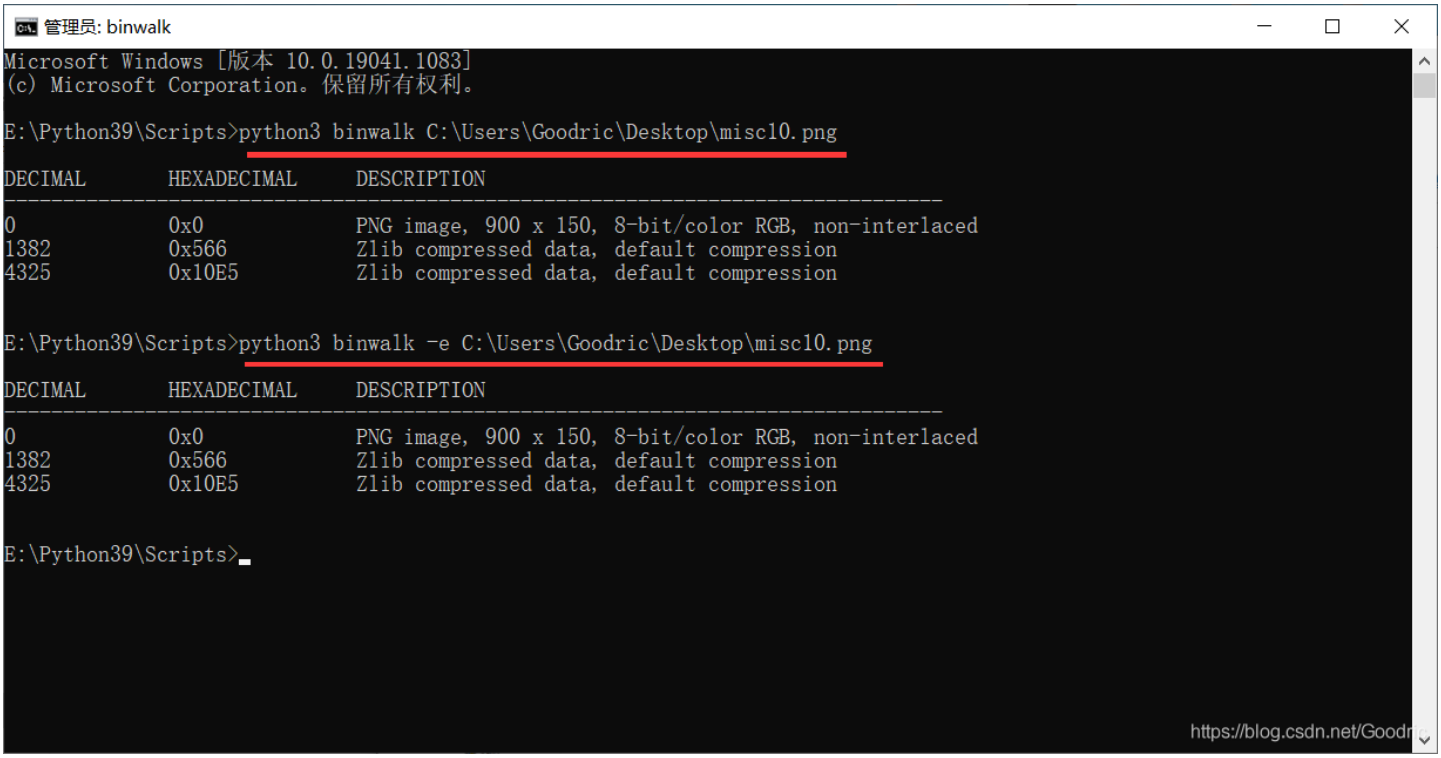
我在 Windows 下用 binwalk 对图片进行扫描提取。（在 kali 下自带）
关于 binwalk 在 Windows 下的使用可参考：
https://blog.csdn.net/Goodric/article/details/117845492?spm=1001.2014.3001.5501

扫描

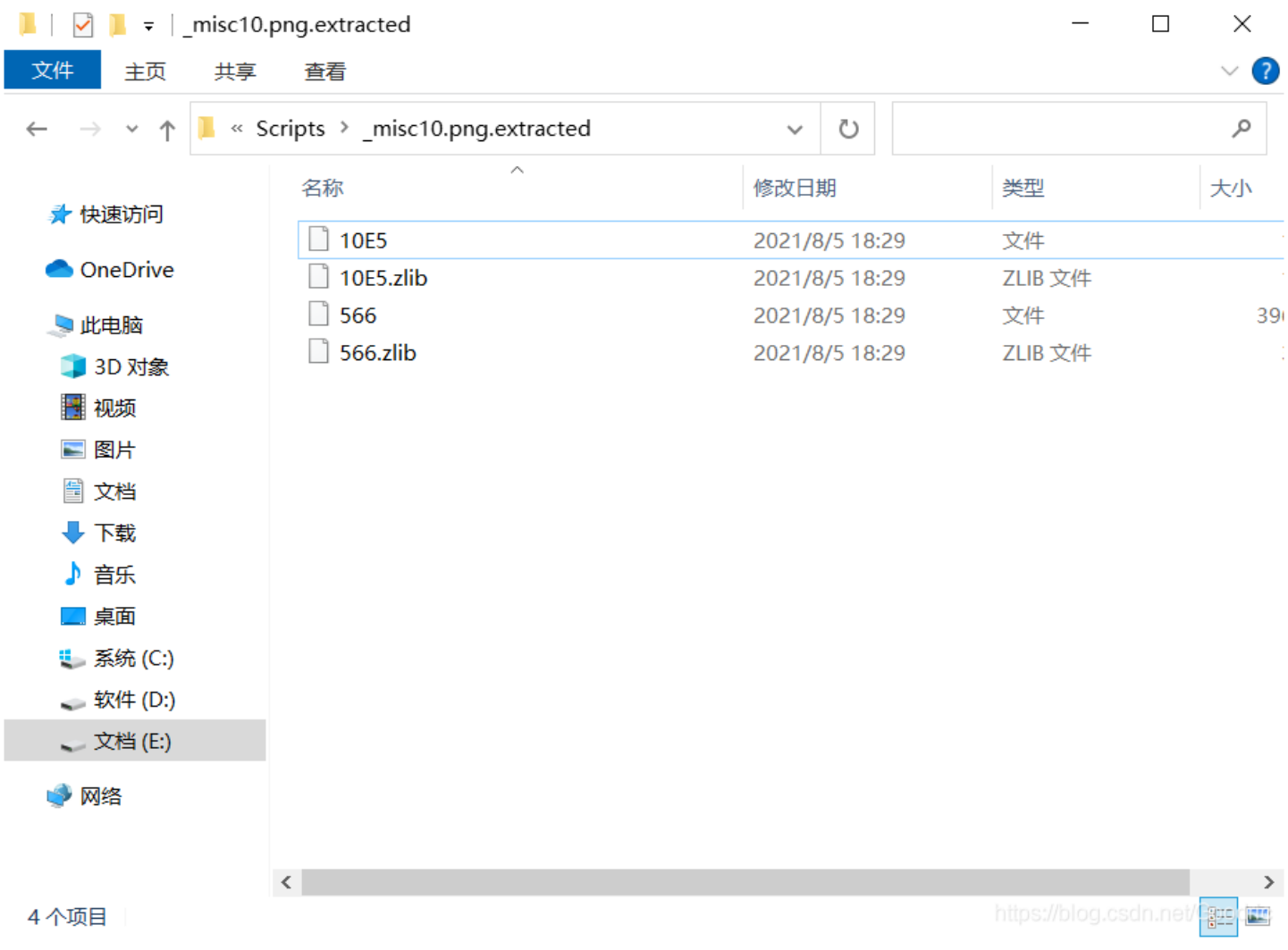
```
python3 binwalk C:\Users\Goodric\Desktop\misc10.png
```

提取分离

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



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 。