

CTF-i春秋网鼎杯第一场misc部分writeup

转载

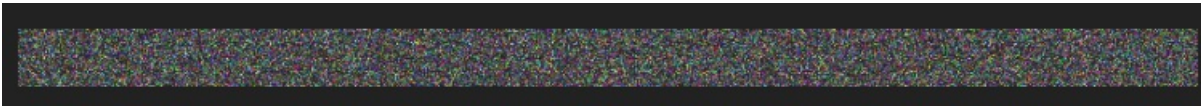
[weixin_33766805](#) 于 2018-08-23 14:14:00 发布 741 收藏 4
原文链接: <http://www.cnblogs.com/pureqh/p/9523185.html>
版权

CTF-i春秋网鼎杯第一场misc部分writeup

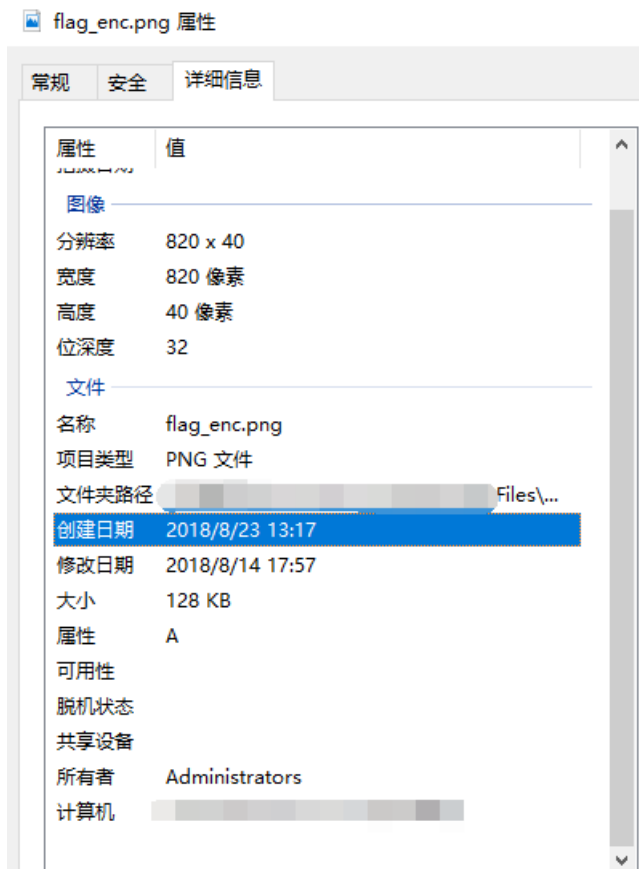
最近因为工作原因报名了网鼎杯，被虐了几天后方知自己还是太年轻！分享一下自己的解题经验吧

minified

题目：



一张花屏，png的图片，老方法，先看属性



什么也没。

再扔到winhexv里看看

Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	ANSI ASCII
00000000	89	50	4E	47	0D	0A	1A	0A	00	00	00	0D	49	48	44	52	PNG IHDR
00000016	00	00	03	34	00	00	00	28	08	06	00	00	00	28	8E	BF	4 (Žž
00000032	0A	00	01	00	00	49	44	41	54	78	9C	00	FF	7F	00	80	IDATxœ ý €
00000048	00	76	79	F5	A0	FC	D6	A6	3D	0E	26	E6	85	AC	38	97	vyð üÖ;= &æ...-8-
00000064	27	9C	39	60	80	84	B6	B2	84	2A	85	D4	30	8C	2B	B0	'œ9`€,,q*,*...ÔOG+°
00000080	33	82	F8	DE	6F	84	01	3D	1E	1A	D0	D4	16	E4	54	00	3,øBo,, = ðÔ äT
00000096	CB	EC	FC	5B	4B	7C	B6	A5	2D	4C	7A	1B	A2	48	04	35	Ëiü[K]q¥-Lz çH 5
00000112	AC	F6	0E	63	CE	08	D4	2C	A7	8A	BF	E8	EE	A8	EA	0A	-ö cî Ô,ššžèi`è
00000128	81	CA	4E	6C	2D	36	CE	FA	2D	F0	81	39	20	9A	C2	0A	ËN1-6îú-8 9 šĂ
00000144	51	36	59	C2	48	F6	31	D9	78	24	7F	2D	0E	26	5C	FF	Q6YĂHö1Üx\$ - &ÿ
00000160	A3	BE	E0	83	27	06	44	D6	63	B6	FE	25	55	AA	A5	F2	š%âf' DÖcqp%U*¥ò
00000176	04	62	4B	63	D2	4E	B4	CD	8B	B8	9E	EC	85	2E	FC	5E	bKcòN`í<,žì...ü^
00000192	3B	BE	4A	1F	51	4A	CA	E0	09	AA	9B	C4	EA	A0	97	8D	;%J QJËà *>Ăè -
00000208	36	E4	9D	43	3C	20	FD	43	2C	D2	7A	D6	89	02	6F	7A	6ä C< ýC,òzÖ% oz
00000224	0E	D6	E2	8C	01	12	3E	1D	5C	14	A1	10	38	0E	06	16	šm.š x \ š -šé

头文件正常，确实是png，尝试搜索flag关键字也没有收获。

继续扔到kali里用binwalk分析一下

```
root@kali: ~/Desktop# binwalk flag_enc.png
DECIMAL      HEXADECIMAL  DESCRIPTION
-----
0            0x0         PNG image, 820 x 40, 8-bit/color RGBA, non-interlaced
41          0x29       Zlib compressed data, default compression
```

也没有发现异常的

会不会是高度隐写呢，直接在kali双击打开png图片，发现可以正常打开(注:被修改过高度的图片无法在kali中直接打开，会显示无法载入图像)

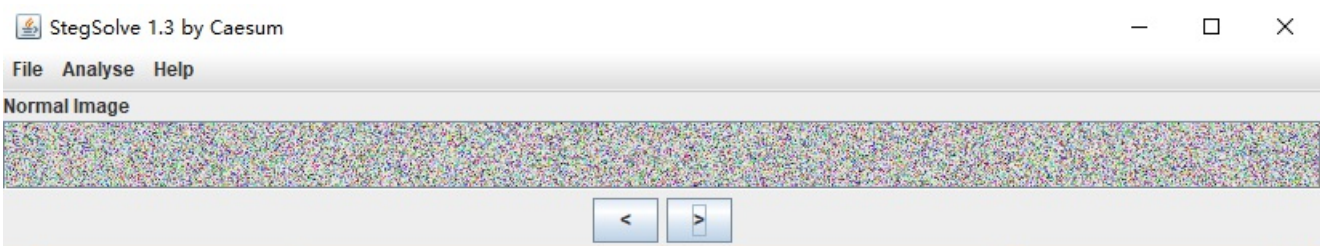
那继续分析IDAT块，IDAT是png图片中储存图像像数数据的块，不清楚的可以去补充一下关于png图片格式知识

我们使用pngcheck分析图片的IDAT块

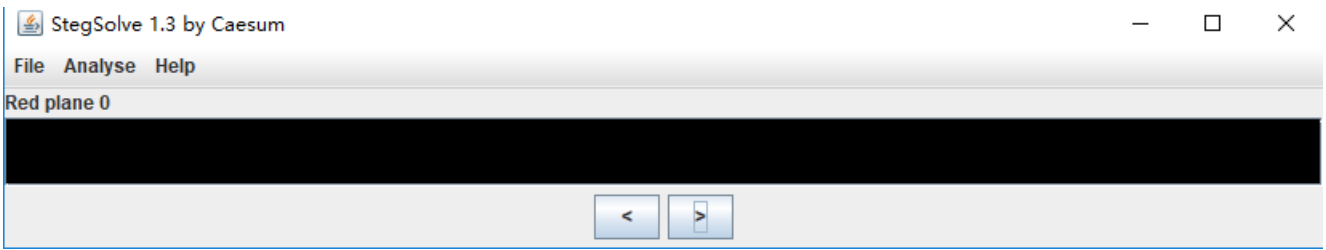
```
C:\pngcheck-2.3.0-win32>pngcheck.exe -v flag_enc.png
File: flag_enc.png (131418 bytes)
 chunk IHDR at offset 0x0000c, length 13
   820 x 40 image, 32-bit RGB+alpha, non-interlaced
 chunk IDAT at offset 0x00025, length 65536
   zlib: deflated, 32K window, default compression
 chunk IDAT at offset 0x10031, length 65536
 chunk IDAT at offset 0x2003d, length 265
 chunk IEND at offset 0x20152, length 0
No errors detected in flag_enc.png (5 chunks, -0.2% compression).
```

好吧也没有异常

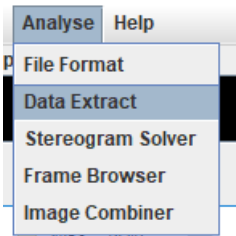
最后拿到Stegsolve跑一下吧



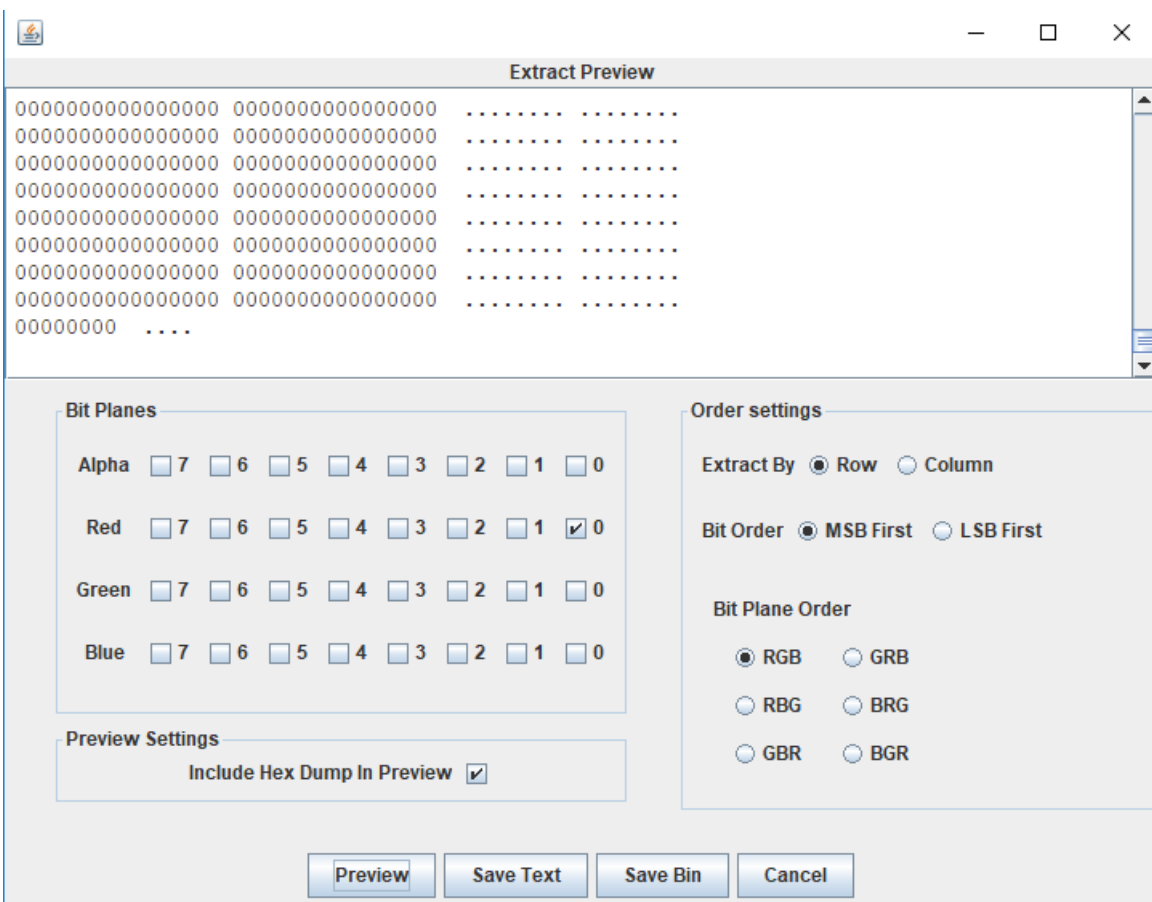
点击向右箭头发现Red plane 0居然是空的，本应该是和花屏一样的图片居然全黑，肯定有问题，而且其他的0通道都是有内容的



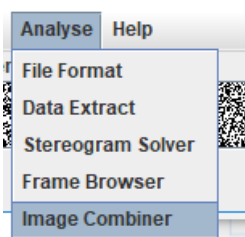
点击Analyse - Data Extract, 查看图片通道



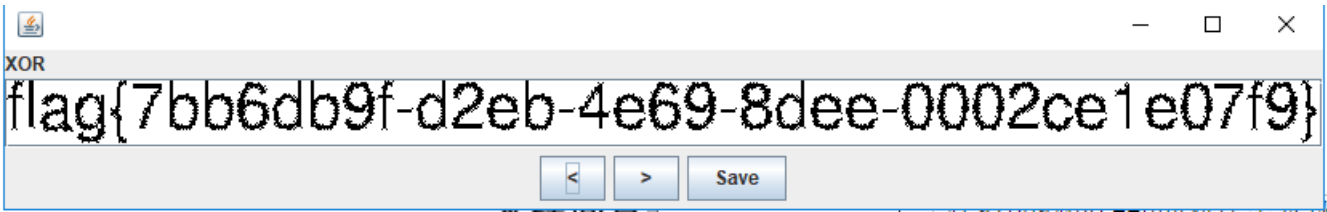
选择 0通道发现payload是LSB 隐写



分别把 alpha, green 和 blue 的 0 通道另存为再进行异或处理

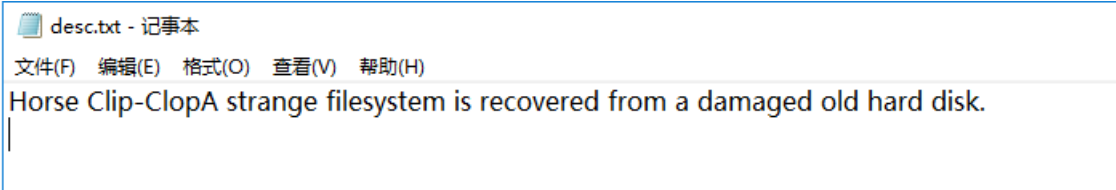
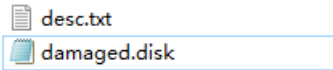


在alpha和green的对比中发现flag如下图所示;



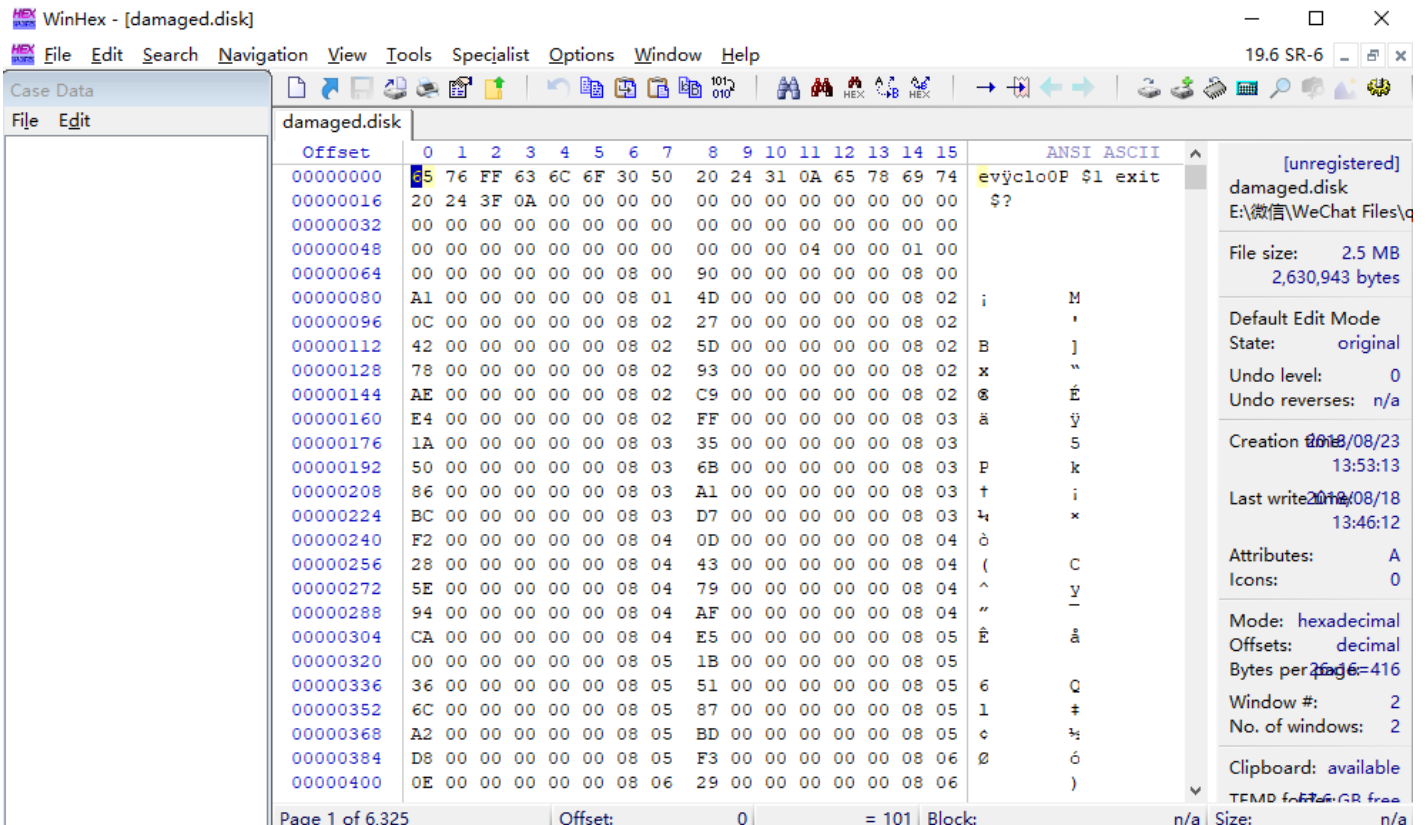
clip

打开题目发现是 .disk 文件，这个应该是linux磁盘文件

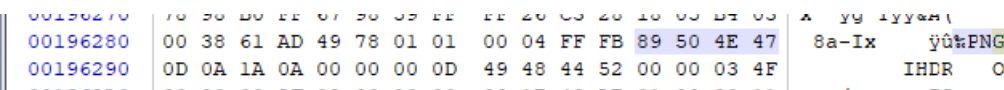


但是题目提示说词频是损坏的，那肯定不去kali试了，那分析一下文件吧

用winhexv打开



好吧，不认识这是什么，继续向下找，发现了和上面不一样的Hex数据!里面可能有东西



在 winhex 中的第 196280 行发现了 png 的文件头(注:png 16 进制文件头以 89504E47 开头)

还有一个IHDR 的 png 图片

001988A0	01 00 04 FF FB 47 0D 0A 1A 0A 00 00 00 0D 49 48	yúG IH
001988B0	44 52 00 00 03 4F 00 00 00 2F 08 00 00 00 00 1E	DR C /
001988C0	49 AE 01 00 00 11 12 49 44 41 54 78 01 EC D3 31	IË IDATx ió1
001988D0	01 00 00 08 04 A1 EF 5F FA 2C E1 08 1D 58 5F 80	i_i_ú,á X_€
001988E0	05 F8 04 3E 81 4F 80 4F E0 13 F8 04 2C E0 F7 D3	ø > C€Oà ø ,à=Ó
001988F0	B5 77 16 40 72 55 4D 1B 7E D6 25 EE B6 1B 57 DC	uw @rUM ~C%iŕ WÜ

剪切出来加png头

得到两张图片

图片1:



图片2:



对两张图片进行ps拼接，得到flag

flag{0b008070-eb72-4b99-abed-092075d72a40}

原创文章，转载请标明出处：<https://www.cnblogs.com/pureqh>

转载于：<https://www.cnblogs.com/pureqh/p/9523185.html>