bugkuCTF—杂项—猫片(安恒)



根据题目提示



可以判断出来出来这很可能是一个LSB的图片隐写,下载下来



很明显是一个png的图片文件,只是被删掉了后缀名,我们将它补上

= 8 % _ **-** X



根据题目提示我们直接打开stegsolve的图片数据信息提取



根据题目提示我们选择LSB和BGR

	- 🗆 X								
Extract Preview									
fffe89504e470d0a 1a0a000000d4948 PNG. IH 4452000001180000 008c08020000008 DR. ID ec7edb0000059c49 444154789ceddd51 .~IDATxQ 6a1c3b1440c13864 ff5b761610145038 j.;0.8d [vP8 3792ecaadf37afdd eef141908bd43f7e 77. .A.?? 0000000000000000gf 3e56ffe1f3f3f37f									
1012047eedfe0fd3 739b95dd39c3f4dc~ s9									
Bit Planes	Order settings								
Alpha _ 7 _ 6 _ 5 _ 4 _ 3 _ 2 _ 1 _ 0	Extract By 💿 Row 🔾 Column								
Red 7 6 5 4 3 2 1 0	Bit Order 🔾 MSB First 💿 LSB First								
Green ☐ 7 ☐ 6 ☐ 5 ☐ 4 ☐ 3 ☐ 2 ☐ 1 ☑ 0	Bit Plane Order								
	⊖ RBG ⊖ BRG								
Preview Settings Include Hex Dump In Preview	⊖ gbr								
Preview Save Text Save Bin Cancel									

发现了里面隐藏了一张图片png我们将其提取出来,保存的格式bin(txt格式用winhex和010Editor打开都是乱码),然后我们修改文件的后缀为png,然后发现图片打不开,用winhex打开

🇱 WinHex - [999 - 副本.png]																						_		×
🚟 File Edit Search Navigatio	n View Tool	s S	pecia	list	Optio	ns V	Vindov	v He	elp														19.7	_ 8 ×
Case Data	🗅 7 🗐 😃 🕯	۵ 🔅	1	5	BD C	0 13 1	101 010	1	#	n 😘	MEX.	-	H) 🔶	•	33	in I	. 🔎	🕫 🔬 🍕		85 🔝 🖌	▶ .	>		
Fi <u>l</u> e E <u>d</u> it	999 - 副本.png	g																						
	Offset	0	1	2	3	4 5	6	7	8	9 10) 11	12	13 1	14 15		A	NSI 2	ASCII	^				funco	aistorod1
	00000000	FF	FE	89 :	50 4	E 47	0 0 0)a 1	A 0	A 00	00 (00	0D 4	19 48	ÿþ%I	PNG		IH				000 쾨★	- laure	gistereuj
	00000016	44	52	00 (0 00	1 18	00 0	0 0	0 8	C 08	8 02	00	00 0	80 00	DR		Œ					999 - mut.phi) 	- ND late -
	00000032	EC	7E	DB (0 00	0 05	9C 4	9 4	4 4	1 54	1 78	9C	ED E	DD 51	ì~Û	œ	IDAT:	xœíÝQ				C:\Users\Adm	inistrato	onDesktop
	00000048	6A	1C	3B 3	14 4	0 C1	38 6	54 E	F 5	в 76	5 16	10	14 5	50 38	j;	@Á8	dÿ[v	P8				File size:		150 KB
	00000064	37	92	EC 2	AA D	F 37	AF D	DD E	E F	1 41	90	8B	D4 3	3F 7E	7'ì'	∎ß7 [—]	ÝîñA	< Ô?~				1110 3120.	152 (600 bytes
	00000080	00	00	00 (0 00	0 00	C0 9	F 3	8E 5	6 FI	E1	F3	F3 F	73 7F		À	Ÿ>Vÿa	áóóó					155,0	boo bytes
	00000096	DE	C7	3F 1	FB F	8 58	FE 0	A (89 D	5 73	3 D8	FD	в9 г	D3 D7	ÞÇ?í	ìøΧþ	%Õs)	Øý¹Ó×				Default Edit M	lode	
	00000112	59	A9	9E (CF E	E FD	57 9	F F	BF C	D E#	A FE	7 F	FE E	27 FB	YCŽI	ΪĺÝW	Ÿ;Íê∣	o bcû				State:		original

发现了并不是我们预期的png文件头, PNG (png)的正常的文件头: 89504E47, 所以们将他前面的FFFF删掉保存退出。



° X _ □

额 半张二维码@@, CRC值出错, 很明显是高度出错引起的

Q Twea	akPNG				_		\times
File Ed	lit Insert	Optio	ns Tools Help				
Chunk	Length	CRC	Attributes	Contents			
	W	arning					×
			Incorrect crc for IH	DR chunk (is 08ec7edb, sho	uld be 286	ic213a)]
۲						确定] >

这个时候我们用一个py脚本算出正确的宽度和高度

```
1 import os
2 import binascii
3 import struct
4 crcbp = open("999.png","rb").read() #文件名
5 for i in range(1024):
6
      for j in range(1024):
          data = crcbp[12:16] + struct.pack('>i',i) + struct.pack('>i',j) +
7
crcbp[24:29]
          crc32 = binascii.crc32(data) & 0xfffffff
8
9
          if crc32 == 0x08ec7edb: #当前的CRC值
10
              print i,j
              print "hex", hex(i), hex(j) #输出宽度和高度
11
```

在有python环境的kali下面跑一下脚本

root@kali:~/桌面/python跑脚本/图片隐写(png)# python calWidthHeight.py 280_280 hex 0x118_0x118 **root@kalsi**t~/桌面/python跑脚本/图片隐写(png)# A

在winhex里面修改保存即可。

999 - 副本.png

Offset	0	1	2	3	4	5	6	7		8	9	10	11	12	13	14	15
00000000	89	50	4E	47	0D	0A	1A	0A	(00	00	00	0D	49	48	44	52
00000016	00	00	01	18	00	00	01	18		0 <mark>8</mark>	02	00	00	00	28	6C	21
00000032	3A	00	00	05	9C	49	44	41		54	78	9C	$^{\rm ED}$	DD	51	6A	1C
	<u></u>				0.0							~ ~		5.0	~~	0.7	~~

发现一个问题这个二维码和我们平时见到的不太一样,正常正方形中间应该是黑色的。所以还要用画图工具进行反色





反色后的图片用QR二维码扫一下



发现一个百度云链接,我也是醉了,还没出来 --

打开链接下载一个压缩包

高
修
20

mmp flag竟然不在里面,真TM。。。无语。。现在呢??我无从下手了,开始参考网上大佬的wp

度娘搜了下,发现另一位老铁写的writeup

https://www.jianshu.com/p/abc44c54857a

最后根据hint里面的提示"NTFS",根据大佬的说法,这是一种流隐写,需要用到工具

ntfstreamseditor,然而。。这里还有一个坑就是,这压缩文件一定要用winrar来解压才会产生这样的效果

🗼 WinRAR:诊断信息	-		×
信息 <mark>↑</mark> E:\网络安全\安恒杯2018-1月赛\mics\flag.rar: 无法设置 拒绝访问。	E:\网络	安全\安恒	亘杯201:
★闭(C) 中断操作(B) 复制到剪	贴板(<u>L</u>)		>
错误总数:1			:

接着用ntfstreamseditor,查看解压的文件夹里面的数据流,然后把他导出来,而且更可恶的是解压的时候必须使用的是winrar来解压(我是在win7的虚拟机里面做的,

不然的话扫不出文件夹里面的数据流,,真的是坑爹到了极致@@

得到一个pyc文件,也就是py编译后的文件,因此需要扔到网上去在线反编译一下

	flag	flag.rar		
StreamsEditor2				
NtfsStream	ISEd Ntfs数据流	itor ^{处理工具}	http	://blog.sina.cor advne
ア 捜索 💎 编辑 📄 记录 ? 信息				
选择搜索类型		数据流名称匹配		
● 全部NTFS磁盘 C:\Users\Administrator\D	eskton	* -	2 搜索	停止
	esktop (
授系结果:共口";用呐0.000 s	*	提达名称	未小学生	可经度(0-5)
C:\Users\Administrator\Desktop\flag\flag.txt:flag.	.pyc fla	ig.pyc	755	1
🔀 刪除 — 🛛 惧 导出 —> 附加 +/ 🕏	异入 <	(前备份 >>	还原<<	

这里推荐一个网站,可以反编译py,https://tool.lu/pyc/



根据他这个加密的脚本再写出一个解密的脚本,运行一下就可以得到flag了

```
1 def decode():
 2
       ciphertext = [
 3
       '96',
 4
       '65',
 5
       '93',
       '123',
 6
 7
       '91',
 8
       '97',
 9
       '22',
10
       '93',
       '70',
11
       '102',
12
       '94',
13
       '132',
14
       '46',
15
16
       '112',
17
       '64',
       '97',
18
19
       '88',
       '80',
20
21
       '82',
22
       '137',
23
       '90',
24
       '109',
25
       '99',
       '112']
26
27
       ciphertext.reverse()
       flag = ''
28
29
       for i in range(len(ciphertext)):
30
           if i % 2 == 0:
                s = int(ciphertext[i]) - 10
31
32
           else:
                s = int(ciphertext[i]) + 10
33
           s=chr(i^s)
34
           flag += s
35
36
       return flag
37
38 def main():
39
       flag = decode()
       print(flag)
40
41
42 if name == ' main ':
43
       main()
```

拿到这个网上去跑一下就出来了



吐槽一番:

我只能想吐一口血在出题人脸上,没写过py,这个脚本和还是参考网上大佬写的,好刚嘎,刚刚那个软件 ntfstreamseditor也是花了很长时间才找到的,网上这种软件都快绝种了。。这个题目解题的一个转折点就是 从数据流中提取出来那个已经编译过的py程序,我们需要拿去反编译,

然后呢?就是要看得懂这个加密脚本了,再然后呢,就是写解密脚本了,这个是真的没写过@@,一波三折, +写博客=花了4个小时 - 感觉dei找个时间去学一下py --人生苦短,我用python。

转载于:https://www.cnblogs.com/liuzeyu12a/p/10033384.html