攻防世界Misc新手练习区WriteUp大全





CTF WriteUp 专栏收录该内容

28 篇文章 3 订阅 订阅专栏

文章目录

写这篇博客的目的 解题过程
1. this is flag
2.ext3
3.give_you_flag
3.give_you_flag
4.pdf
5.SimpleRAR
6.坚持60s
7.gif
8.掀桌子
9.如来十三掌
10.base64stego
11.功夫再高也怕菜刀
12.stego

总结

写这篇博客的目的

对于CTF中的Misc来说,做题经验显得十分重要,而做题经验的获得很大一部分取决于刷题量。为了避免大家在刷题过程到处搜 WriteUp浪费时间,现在把我的一些做题方法分享出来,希望对大家有帮助。当然,大家有更好的解决方法欢迎在评论区留言, 互相学习,共同进步。

解题过程

1. this is flag

点开这个题目直接显示flag

2.ext3

strings linux | grep flag

在linux这个文件中搜索flag字符串



发现存在flag.txt文件, ext3文件是一种Linux日志文件,所以把它挂载到linux系统上 把这个文件挂载到mnt目录,然后进入此目录,使用ls列出所有文件,查看flag

mount linux /mnt cd /mnt ls cd 07avZhikgbf/ cat flag.txt

linux操作截图



b.得到flag.txt中是base64编码的。使用

base64 -d flag.txt

解码

3.give_you_flag

a.附件为一个gif b.放入stegsolvs一帧一帧的查看一下(stegsolve中analyse下的frame browser) c.发现二维码



4.pdf

a.根据提示说图片下面什么都没有,怀疑图片下面隐藏着另一个图层 b.把pdf放进linux中,隐藏的部分鼠标会有变化,把它选中 c.flag就出现了



5.SimpleRAR

a.得到压缩包后打开得知压缩包内用文件头损坏



https://blog.csdn.net/CliffordF

损坏

的是一个图片,并且发现里面还有一个flag.txt b.打开这个txt

 flag.txt - 记事本
 文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
 flag is not here

并没有搞头,目光聚集到这个损坏的png上 c.查资料得知被压缩后的png头为A8 3C 74,在winhex中查找一下这个png头,看看有没有改变 d.发现压缩包中文件头是A8 3C 7A

 :0
 43
 66
 6C
 61
 67
 20
 69
 73
 20
 6E

 :0
 65
 83
 3C
 7A
 20
 90
 2F
 00
 3A
 15

 :0
 02
 BC
 E9
 8C
 2F
 6E
 84
 4F
 4B
 1D

 :0
 00
 73
 65
 63
 72
 65
 74
 2E
 70
 6E

e.修改7A为74。

f.成功解压出图片,但是发现是一个白色的图片。根据提示是ps双图层,把它放入ps中分离图层,后缀为png但是拖入ps的过程 中被提示这不是一个png文件。在2345看图王中打开发现存在两个帧,那么这就是一个gif了,改后缀为gif g.然后把它拖入ps中



把这两个图层分别保存

f.保存后把这个纯白的图片放入stegsolve中。发现两个图中都可以发现半个二维码,剩下的工作就是把两个二维码拼接成一个二 维码

g.

🛓 StegSolve 1.3 by	-	×	🧯 🛃 StegSolve 1.3 by 🗕 🗆 🗙
File Analyse Help			File Analyse Help
Red plane 0			-Red plane 0
			· · · · · · · · · · · · · · · · · · ·



但是第二块二维码定位点缺失,只要第一个二维码中的定位点补过去就可以 f.补图+扫描

🔐 QR Research			_		\times
文件(F) 工具(T) 帮助(H)					
	۵ 😯	E			
		纠错等级— H(30%) 版本— Auto ▼ Auto		3円 uto です	-
Ι					^
					~
					~
位置:(26.0,28.9)-(290.7,26.8)-(30.0,29 颜色正常,正像 版本:3 纠错等级:H, 掩码:4 内容:	1.2)-(294.9,288.6) L	og			
flag{yanji4n_bu_we1shi}					
					~
解码完成		agold\\ b qiin	eseirari	ai/Ghin	oreiFt

6.坚持60s

a.下载下来是一个.jar

b.看文件类型,这就需要看出这个程序的基本的对象,就要用到java反编译器,这里使用jd-gui c.



7.gif

a.下载附件后打开发现是一些黑白图片 b.一共104块,104÷8=13,也就刚好是13个八位的二进制数 c.白色为0,黑色为1,依靠手输肯定是行不通的。附上一个python脚本

```
from PIL import Image
flag_dic=""
for i in range(0,104):
    img = Image.open("F:\\ctf文件\\b3ba561080fb4a9d9c1f019e298e218b\\gif\\%d.jpg"%i)
    clrs = img.getcolors() #clrs 包含 [("num of occurences", "color"),...]
    if clrs == [(46656, (12, 12, 0))]:
      flag_dic += "1"
    else:
        flag_dic += "1"
    else:
        flag_dic += "0"
print(flag_dic)
flag = ""
for q in range(int(len(flag_dic)/8)):
    flag += chr(int(flag_dic[q*8:(q+1)*8],2))
print(flag)
```

脚本的话是自己写的,比较菜,师傅们可以交流一下,咱们共同进步。代码逻辑比较简单,有疑问的可以评论区留言。 利用这个python脚本可以直接获得flag。

原理就是黑色代表1, 白色代表0, 然后是8个为一组的二进制。转换为字符串就是flag了。



8.掀桌子

a.初步判定是十六进制直接转换 b.发现不行 c.将ascii码减去128转换



9.如来十三掌

a.看字符直接与佛论禅

b.得到这一串 MzkuM3gvMUAwnzuvn3cgozMIMTuvqzAenJchMUAeqzWenzEmLJW9 c.由如来十三掌想到rot13,然后进行rot13解码,最后base64解码得到flag

10.base64stego

这个题目说简单也不简单,说难也不难,我对这个题目的理解还没有到达分享的地步,在这里分享一个师傅的博客。

在这里只把python的脚本贴出来做一个保存。

```
import base64
b64chars = 'ABCDEFGHIJKLMN0PQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/'
with open('F:\\ctf文件\\rebuilt.787798f51ce441dd9961405c1aff8893\\stego.txt', 'rb') as f:
    flag = ''
    bin_str = ''
    for line in f.readlines():
        stegb64 = str(line, "utf-8").strip("\n")
        rowb64 = str(base64.b64encode(base64.b64decode(stegb64)), "utf-8").strip("\n")
        offset = abs(b64chars.index(stegb64.replace('=', '')[-1]) - b64chars.index(rowb64.replace('=', '')[-1]))
        equalnum = stegb64.count('=')  # no equalnum no offset
        if equalnum:
            bin_str += bin(offset)[2:].zfill(equalnum * 2)
            # flag += chr(int(bin(offset)[2:].zfill(equalnum * 2), 2))
            # print(flag) 这样写得不出正确结果
        print([chr(int(bin_str[i:i + 8], 2)) for i in range(0, len(bin_str), 8)])
```

11.功夫再高也怕菜刀

a.在流量包中搜索flag b.追踪几个tcp数据流后发现存在FFD8/FFD9,这是png图片文件的开头和结尾

·····			Content-type: application/x-www-torm-urlencoded
(G) 捕获(C) 分析(A) 统	\$计(S) 电话(Y) 无统	浅(W) 工具(T) 幕	Content-Length: 204999
९ 🗢 🗢 🕾 🗿 🛓 🖥	📃 ର୍ ର୍ ର୍		22-0012
			<pre>dbase6 decode(\$ POST[action])):&action=OGluaV9z7X0oTmRpc3BsYX1f7X1vb31zTiwiMCTpO0Bz7XRfd(</pre>
		分组详情	CgwKTtAc2V0X21hZ21jX3F1b3R1c19ydW50aW11KDAp02VjaG8oIi0%2BfCIp0zskZj1iYXN1NjRfZGVjb2R1KCRfL
	Destination	Protection	XSk7JGM9JF9QT1NUWyJ6MiJd0yRjPXN0c19yZXBsYWN1KCJcciIsIiIsJGMpOyRjPXN0c19yZXBsYWN1KCJcbiIsIi
ource	Jestination		WY9III7Zm9yKCRpPTA7JGk8c3RybGVuKCRjKTskaSs9MikkYnVmLj11cmxkZWNvZGUoIIUILnN1YnN0cigkYywkaSw
72.100.45.05	102 100 23.120		hAZndyaXR1KGZvcGVuKCRmLCJ3IiksJGJ1Zik%2FIjEiOiIwIik7O2VjaG8oInw8LSIpO2RpZSgpOw%3D
92.100.23.120	192.100.45.05		%3D&z1=RDpcd2FtcDY0XHd3d1x1cGxvYWRcNjY2Ni5qcGc
92.100.45.05	192.100.25.120		%30&z2=FFD8FFE000104A46494600010101007800780000FFDB0043000101010101010101010101010101010101
92.168.43.83	192.168.25.128	5 HITP	01010101010101010101010101010101010101
1/8]			01010101010101010101010101000110001390122090122090211010311001001040010901000101010101010000
			0122334914169302970805964097440031000220103030201030302040600001700122330041301012213141001 143381914169833/3819115520115221320002201030302011417181914255527283930333353637383930334445464745
, toxt/html			657585954636465666768696A737475767778797A838485868788898A9293949596978899AA2A344A546A7A84
Lex C/ II Clini			B6B7B8B9BAC2C3C4C5C6C7C8C9CAD2D3D4D5D6D7D8D9DAE1E2E3E4E5E6E7E8E9EAF1F2F3F4F5F6F7F8F9FAFFC4
			10101010101010100000000000000000000000
50 2f 35 2e 36 2e	32 35 0d 0a	By: PHP/ 5.6.	1104052131061241510761711322328108144291A1B1C109233352F0156272D10A162434E125F11718191A2627
74 2d 4c 65 6e 67	74 68 3a 20	Content- Leng	738393A434445464748494A535455565758595A636465666768696A737475767778797A82838485868788898A5
65 65 70 2d 41 6c	69 76 65 3a	221Kee p-A1	98999AA2A3A4A5A6A7A8A9AAB2B3B4B5B6B7B8B9BAC2C3C4C5C6C7C8C9CAD2D3D4D5D6D7D8D9DAE2E3E4E5E6E7
75 74 3d 35 2c 20	6d 61 78 3d	timeout =5,	4F5F6F7F8F9FAFFDA000C03010002110311003F00FC18823DB907E62481211D6493F86143D914E012BCF5E3005
6e 6e 65 63 74 69	6f 6e 3a 20	99Conn ecti	D0CC40EFFC30478E3B0DFF00FD8F352DA3DBB0AF0769F2C1FF00964839699CF3866C9C11CF719E33AD6F1B7C84
6c 69 76 65 0d 0a	43 6f 6e 74	Keep-Ali ve	6720732780C1D99EC0632179FF49A8C75F376FF002F89DFAD9BE65066EDF556D79EFADBB3D9AB5BE4AC95FB69L
70 65 3a 20 74 65	78 74 2f 68	ent-Type : te	4C9C703CD238D89FC30A1F523AE3D4F6539D88632E4E6013080CA57FE58C5FC31A7FD34933F377E7DCD54B78F
68 61 /2 /3 65 /4	3d 55 54 46	tml; cha rset	038F9UC70F9UFF095E703FA6EA09810108A83702FF20FF9U5779187FCF34391CF5F339AF568356EF0F78A036F
20 3e /c 2e 2f 09	32 30 31 37	-8> ./.	0UE3U/9F4UUAFDAFAZF33U1EFAZD33D403/D90A042093031HF/C30L1F90/LC/10UE7/33U001L00UDA0U0DD/0AF 77019EF0A7EA7E3033D89/E33A389AD8336E51D34E091EF3326791D4131A12739C73DE6AEDDF0/A6AB93EE033AF
	30 31 37 34	-12-00 1 1:42	7/316FF06CF47L3035B06C0324A0AAB0220331024F301EF735076104131A13C-3067000000004004502FF0630E
31 31 39 33 39 39	31 30 09 34	12-08 11 .39.	318941415872375847324545547584758478757572314153455455555757628033865555535555555555555555555555555555
37 37 0a 31 2e 70	68 70 09 32	096.0777 .1.p	32 孝戶灣 分超, 0 服务器 分超, 0 turn(s).
2d 30 38 20 31 31	3a 33 33 3a	017-12-0 8 11	Entire conversation (407 kB)
30 36 36 36 0a 36	36 36 36 2e	16.33.06 66.6	****
31 37 2d 31 32 2d	30 38 20 31	jpg.2017 -12-	E1%-
31 09 31 30 32 32	32 36 09 30	1:42:11. 1022	滤掉此流 打印 Save as*** Back Cl.
61 67 2e 74 78 74	09 32 30 31	666.flag .txt	
38 20 31 31 3a 33	35 3a 32 39	7-12-08 11:35	:29
36 36 0a 68 65 6c	6c 6f 2e 7a	.17.0666 .hell	0.Z
37 2d 31 32 2d 30	38 20 30 39	ip.2017- 12-08	89 https://blog.opdp.pat/CliffordP
09 32 32 34 09 30	36 36 36 0a	:32:36.2 24.06	bb.

c.把FFD8和FFD9之间的数据复制写到winhex中,注意以十六进制数据写入,然后改为png打开



12.stego

a.在google上安装插件PDF Viewer

b.控制台输入document.documentElement.textContent,得到一串AB编码而成的字符串,将A变为.,B变为-,摩斯解密为 CONGRATULATIONSnullFLAGnull1NV151BL3M3554G3 c.即flag为1NV151BL3M3554G3 本文优点仁者见仁智者见智

本文缺点:

本文缺点之一就是单纯的就题论题,并没有做一些拓展和归纳总结。

希望在以后自己题量上去以后好好地分类总结一下,更加深入的理解每一种解题方法和思路。