2017湖湘杯网络安全大赛Writeup

转载

普通网友 于 2017-12-15 09:36:14 发布 4226 收藏 4
分类专栏： CTF


CTF 专栏收录该内容
10 篇文章 0 订阅
订阅专栏

大赛分为初赛、复赛、决赛。 初赛150道理论题，前300名进入复赛，复赛为夺旗赛（CTF）竞赛模式。

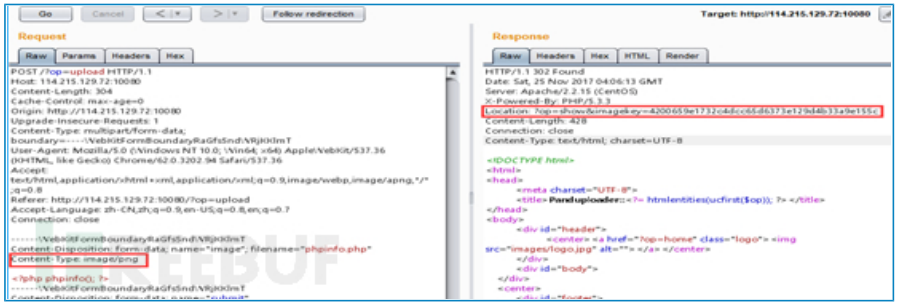
湖湘杯网络安全大赛相关链接：【传送门】

复赛为web渗透，反编译，破解，加密&解密 一共15道题。（以下是我们队伍答题的writeup）

Web200

题目：简简单单的上传，没有套路。
答题地址：

> http://118.190.87.135:10080/
>
> http://114.215.129.72:10080/
>
> http://118.190.86.101:10080/



首先是在网站进行文件上传。修改Content-Type可绕过上传 ， 这个题真的很想吐槽

最后经过各种尝试，get一枚文件包含。想到php://filter

http://114.215.129.72:10080/?op=php://filter/read=convert.base64-encode/resource=flag

get flag

简单的android

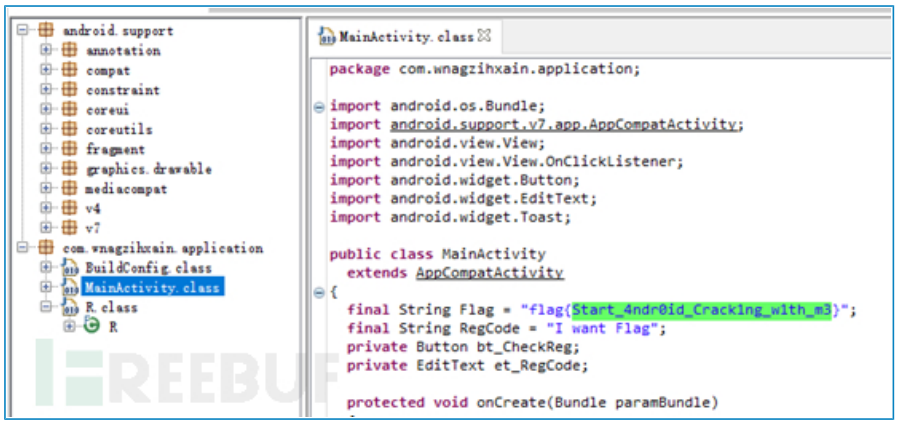
题目：关于Android Crack的基本操作，对你来说SoEasy

过程：

将apk格式改成zip解压出来

使用d2x2jar反汇编拿到源码

使用工具jd-gui查看源码



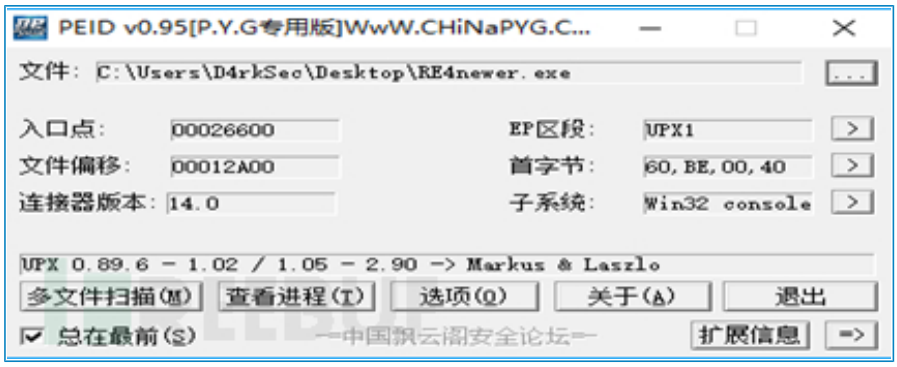
直接拿到flag。

Re4newer

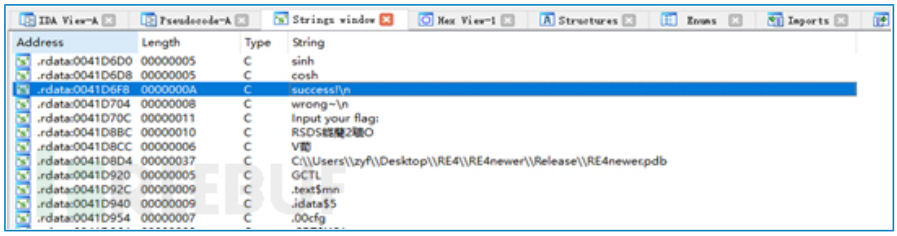
题目：逆向基本操作。

过程：

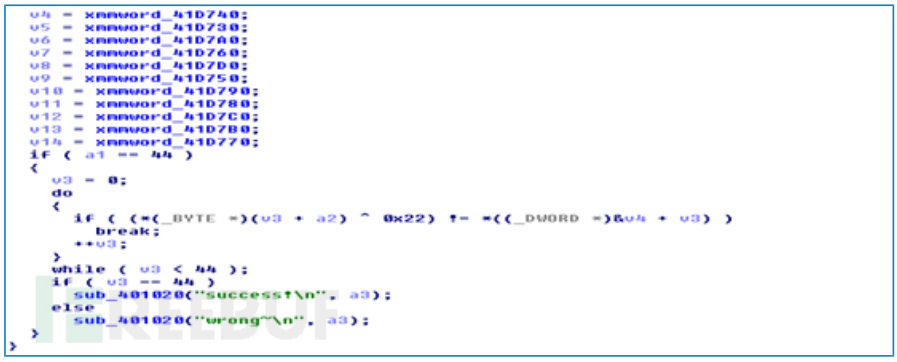
首先查壳，UPX，脱壳工具可以直接脱掉.



载入ida分析，shift+F12搜索字符串，发现字符串（success）



双击进去后，F5定位到关键代码



```
v4 = xmmword_41D740;
v5 = xmmword_41D730;
v6 = xmmword_41D7A0;
v7 = xmmword_41D760;
v8 = xmmword_41D7D0;
v9 = xmmword_41D750;
v10 = xmmword_41D790;
v11 = xmmword_41D780;
v12 = xmmword_41D7C0;
v13 = xmmword_41D7B0;
v14 = xmmword_41D770;
if ( a1 == 44 )
{
  v3 = 0;
  do
  {
    if ( (*(_BYTE *)(v3 + a2) ^ 0x22) != *((_DWORD *)&v4 + v3) )
      break;
    ++v3;
  }
  while ( v3 < 44 );
  if ( v3 == 44 )
    sub_401020("success!\n", a3);
  else
    sub_401020("wrong~\n", a3);
}
```

分析上述代码，取出41D740 等地址中的_DWORD与0×22异或即可得到flag

脚本及执行结果如下

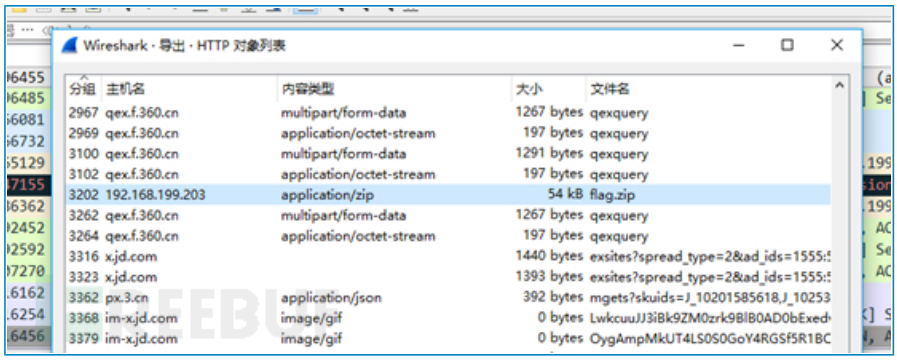
```
a=
[0x44,0x4E,0x43,0x45,0x59,0x76,0x4A,0x13,0x51,0x70,0x48,0x71,0x70,0x63,0x70,0x54,0x11,0x58,0x58,0x70,0x51,0x48,0x4F,0x52,0x4E,0x47,0x70,0x78,0x78,0x78,0x67,0x67,0x67,0x70,0x57,0x70,0x52,0x63,0x51, 0x71,0x70,0x13, 0x56,0x5F];

var flag='';
for(var i=0;i < a.length;i++){
    flag += String.fromCharCode(a[i]^0x22);
}
"flag{This_iS_A_v3ry_simple_RRREEE_u_pAsS_it}"
```

流量分析

题目：

过程：

使用wireshark打开下载的流量包，直接导出HTTP对象，发现flag.zip文件



打开flag.zip 发现内容为rgb值的ce.txt文件（rgb值还原图像相关代码）

经过调试图片高宽和方向，代码如下：

```
from PIL import Image

f = open('/Users/SPY/Desktop/ce.txt','r')

length = 111
width = 887
pic = Image.new("RGB",(length, width))

for y in range (0,width):
    for x in range (0,length):
        l = f.next().split(',')
        pic.putpixel([x,y],(int(l[0]),int(l[1]),int(l[2])))

pic.transpose(Image.FLIP_LEFT_RIGHT).transpose(Image.ROTATE_90).show()
```

转换成图片后，get flag

Web300

题目：拿个shell就给flag。

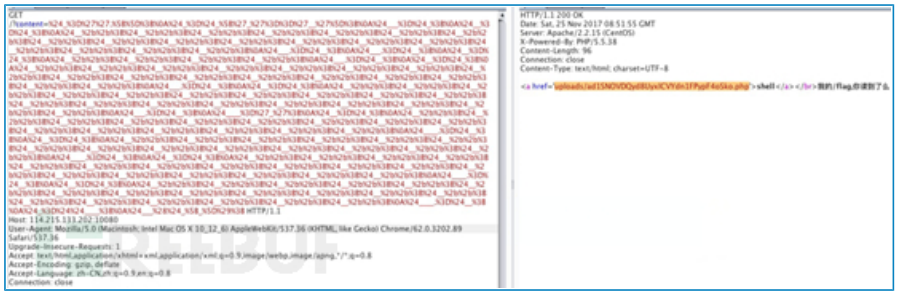
答题地址：

> http://114.215.71.135:10080/
>
> http://114.215.133.202:10080/
>
> http://118.190.77.141:10080

访问地址，直接给出源代码，过滤做的很全，并且只允许GET方式提交

参考文章：https://www.leavesongs.com/PENETRATION/webshell-without-alphanum.html

最后构造出不被拦截的payload，进行url编码提交



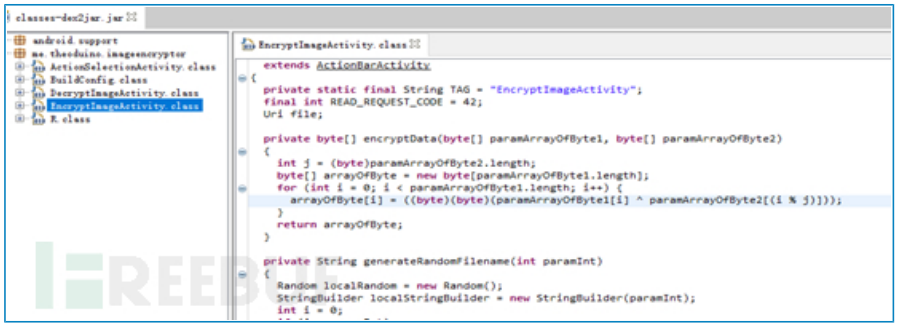
如图上，成功getshell （cat../flag.php）

Encryptor.apk

题目：

过程：

通过分析源码，发现程序需要输入一个密码和一张图片，程序将图片与密码的MD5值逐位循环异或



直接给出解题代码

```
png_file = open("/Users/SPY/Desktop/own.png","rb")
png_h = png_file.read(16)
png_file.close()

flag_file = open("/Users/SPY/Desktop/flag.encrypted","rb")
flag_h = flag_file.read(16)
flag_file.close()

secret = []
for i in xrange(16):
    secret.append(ord(png_h[i])^ord(flag_h[i]))

decryp_file = open("/Users/SPY/Desktop/flag.png","wb")
with open("/Users/SPY/Desktop/flag.encrypted","rb") as f:
    while True:
        data = f.read(16)
        if not data:
            break
        for i in xrange(len(data)):
            decryp_file.write(chr(ord(data[i])^secret[i]))
decryp_file.close()
```

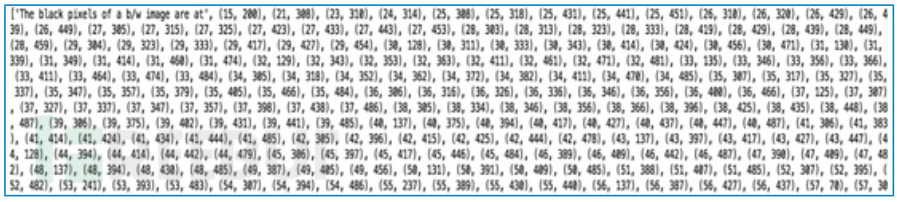
```
Flag{all_encryption_is_equal_but_some_are_More_equal_than_others}
```

MISC 300

题目：

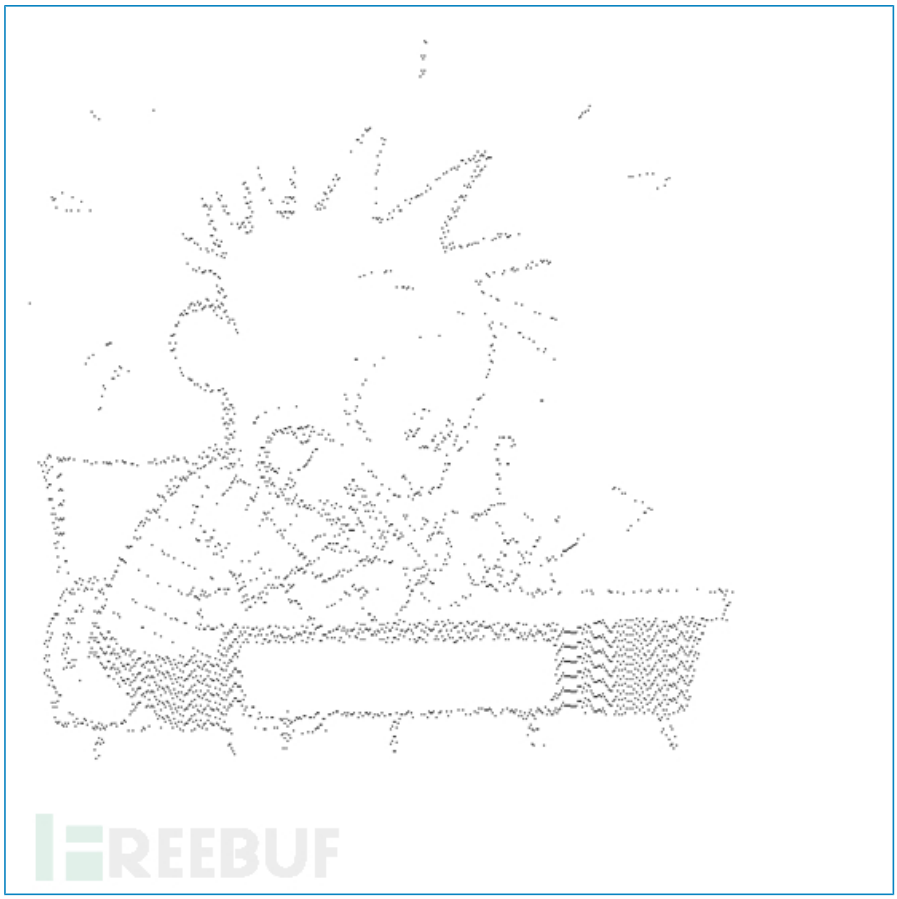
过程：

下载后查看是一个pkl文件，用pythonpickle模块打开，发现如下



由第一句话得知 ()里面的数字是黑色像素的坐标，然后通过脚本得到图片，脚本以及图片如下

```
import pickle
from PIL import Image


f = open('pixels.jpg.pkl')
locates = pickle.load(f)

pic = Image.new('RGB', (600,600), "white")
pixels_out = pic.load()

for i in locates[1:]:
    pixels_out[i[0],i[1]]=(0,0,0)

pic.save("misc300.png","png")
```



查看该图片发现图片为美国经典漫画《卡尔文与霍布斯虎》中的卡通人物，其中作者是

比尔·沃特森（BillWatterson）。根据ctf中的套路，将其改成小写的billwatterson提交flag。

热身运动（题目已更新）

过程：

图片背景是一个8X8的宫格，左上角第一个编号为0，横向编号，右下角最后一个编号为63，按照GIF中，老虎出现的顺序，转换为数字是：

25,38,49,33,25,55,44,49,29,5,60,49,13,21,61,38,29,22,57,46,30,23,52

8X8=64，联系到Base64,查看base64表，将数字转换为对应的字母或数字

Base64索引表：

Value	Char	Value	Char	Value	Char	Value	Char
0	A	16	Q	32	g	48	w
1	B	17	R	33	h	49	x
2	C	18	S	34	i	50	y
3	D	19	T	35	j	51	z
4	E	20	U	36	k	52	0
5	F	21	V	37	l	53	1
6	G	22	W	38	m	54	2
7	H	23	X	39	n	55	3
8	I	24	Y	40	o	56	4
9	J	25	Z	41	p	57	5
10	K	26	a	42	q	58	6
11	L	27	b	43	r	59	7
12	M	28	c	44	s	60	8
13	N	29	d	45	t	61	9
14	O	30	e	46	u	62	+
15	P	31	f	47	v	63	/

转换之后是ZmxhZ3sxdF8xNV9mdW5ueX0，对该字符串使用base64解码，即得到flag

pyc分析

题目：pyc逆向分析，解密enc

过程：

将pyc用工具还原成py,发现是一行python代码

使用lamda和for var in value进行混淆，分析源码可知，该代码会丢失原始数据的最高两位，只有 6位有效，但是通过转换的table的 长 度只有94，所以高8位是0，高7位可能为0或1，根据算法写出逆算法，分 别 求高7位为1和0 的情况，代 码如下图：

```
table='0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ!"#$%&\'()*+,-./:;<=>?@[\\]^_`{|}~'
result1 = ''
result2 = ''
def decode(s):
    global result1,result2
    a = s>>16
    b = (s>>8) & 0xFF
    c = s & 0xff
    s1 = (8-len(bin(a)[2:]))*'0'+bin(a)[2:]
    s2 = (8-len(bin(b)[2:]))*'0'+bin(b)[2:]
    s3 = (8-len(bin(c)[2:]))*'0'+bin(c)[2:]
    result1+=table[(int('1'+s1[2:],2)-1)%94]+table[(int('1'+s2[4:]+s1[:2],2)-1)%94]+table[(int('1'+s3[6:]+s2[:4],2)-1)%94]+table[(int('1'+s3[:6],2)-1)%94]
    result2+=table[(int(s1[2:],2)-1)%94]+table[(int(s2[4:]+s1[:2],2)-1)%94]+table[(int(s3[6:]+s2[:4],2)-1)%94]+table[(int(s3[:6],2)-1)%94]
    # print int(s1[2:],2)-1,int('1'+s1[2:],2)-1
    # print int(s2[4:]+s1[:2],2)-1,int('1'+s2[4:]+s1[:2],2)-1
    # print int(s3[6:]+s2[:4],2)-1,int('1'+s3[6:]+s2[:4],2)-1
    # print int(s3[:6],2)-1,int('1'+s3[:6],2)-1

a=open('key.enc','rb')
a=a.read()
for i in xrange(0, len(a), 3):
    decode(int(a[i:i+3].encode('hex'), 16))

print result1
print result2
```

执行结果：

```
V1ANking:PYC分析 SPY$ python decode.py
>>>>>{{{{g;4:,*&&/#*#.%$&+;(<}}}<<<<<<<"""""""""""""""""""""
hhhhhqqqqKeyd9733c070b2138e5fsssfffffff~~~~~~~~~~~~~~
```

其中“hhhhhqqqqKeyd9733c070b2138e5fsssfffffff”“”“”“”“”“”“”“”是高7位为0的情况。

“>>>>>{{{{g;4:,*&&/#*#.%$&+;(<}}}<<<<<<<xxxxxxxxxxxxx”是高7位为1的情况。

根据题目，只取{}中的内容，为g;4:,*&&/#*#.%$&+;(<或Keyd9733c070b2138e5f，根据Key分析，可能的原字符串为Key:9733c070b2138e5f,提交Key后面的内容，正确，所以9733c070b2138e5f即为flag

最后这次比赛还是学到了很多东西，欢迎各路师傅交流。

*本文作者：寻梦小生，转载请注明来自 FreeBuf.COM

寻梦小生 *1* 篇文章 等级： *1*级

- 上一篇：OpenATS续篇：搭建自己的卫星地球站
- 下一篇： 认证云安全专家（CCSP）考试攻略

这些评论亮了

- 
  D14tr0y (3级) 这家伙太懒了，还未填写个人描述！ 回复
  freebuf文章图片这边看起来特别不舒服，图片是多大，打开就是多大图片都不能放大的。看都看不清
) 15 (亮了

发表评论

已有 3 条评论

- 

  常运 (5级) c4td0g, 信安从业者，信安爱好者。（各位爷，轻点喷） 2017-12-10 回复 1楼

  咋writeup是抄来的，图片都看不清额？

  亮了(2)

- 

  D14tr0y (3级) 这家伙太懒了，还未填写个人描述！ 2017-12-10 回复 2楼

  freebuf文章图片这边看起来特别不舒服，图片是多大，打开就是多大图片都不能放大的。看都看不清

  亮了(15)

- 

  142 2017-12-12 回复 3楼

  基本上都是网上搬的原题，原创的基本上没有