2021 陇原抗疫 WriteUp

原创

是Mumuzi 于 2021-11-08 09:57:03 发布 5334 收藏 13

分类专栏： ctf buuctf 文章标签： 网络安全

本文链接：https://blog.csdn.net/qq_42880719/article/details/121202118

ctf 同时被 2 个专栏收录

75 篇文章 28 订阅

订阅专栏

buuctf

15 篇文章 2 订阅

订阅专栏

				CRYPTO				MISC					PWN				REVERSE						WEB				
				mostlycommon	easytask	Civet cat tor Prince	打疫苗	soEasyCheckin	SOS	EasySteg	ez_misc	checkin	bbbaby	Magic	h3apclass	httpd	EasyRe	fmdme	power	EasyRE_Reveng e	Eat_something	o	CheckIN	easasyphp	EasyJaba	rsablog	MagicMail
Place	Team	Score	Solves	200	908	938	200	200	859	1000	1000	1000	232	908	975	1000	200	200	859	908	992	995	514	800	985	1000	1000
1	VN战队	9886	16																								
2	n03tAck战疫	8412	14																								
3	D1no	8067	14																								
4	Light1ng	8060	14																								
5	Torchwood	8050	14																								
6	AAA	7146	12																								
7	USTC-NEBULA	6432	10																								
8	广外女生	6073	11																								
9	我真的好饿!	5347	11																								

希望疫情早日结束

注：打*的是赛后出的

总WP：https://wp.n03tack.top/posts/14620/

Misc

soEasyCheckin

base32，但有问题，倒数出现了0$，0–>O,$—>S，得到一串hex。
结果hex那里又有问题，具体是出在`83¥6988ee`这里
根据规律,每6个字节的第1个字节为e，然后把`¥`替换成`e`
得到社会主义核心价值观编码，但是还是有个地方是错误的，中间有一段为：和谐赮明平
然后把他那个赮随便改一下，我改的“富强”
解码得到的`SET{Qi2Xin1Xie2Li4-Long3Yuan0Zhan4Yi4}`
根据拼音，可以知道是Yuan2
所以最终flag为：

```
SET{Qi2Xin1Xie2Li4-Long3Yuan2Zhan4Yi4}
```

打败病毒

游戏打开之后发现在末地，打完末影龙后没有反应，flag藏在终末之诗里，于是直接去找文本
在.minecraft/version/陇原战“疫”.jar下
将其改为zip，找到assets/minecraft/texts/end.txt，得到11F9sACbBBBWKTiCIYDtNF2ylEfThXdfIGPxF
base62解码即可

```
SETCTF{Fi9ht1ng_3ItH_V1rUs}
```

SOS

拨号音，踩正确的来组合出flag
用手机录音，然后m4a格式转wav格式，之后DTMF，因为录的总有问题，所以一共录了三次

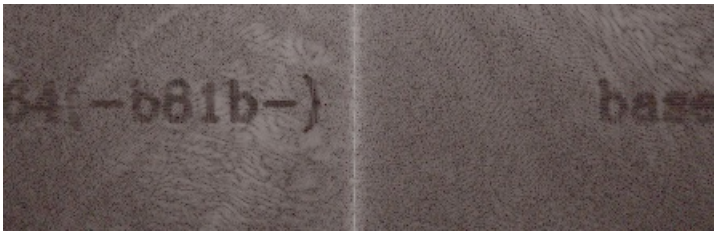
```
- DTMF numbers:  630AB1C75
PS F:\aaaCTF\aaa工具\DTMF拨号音识别手机键盘密码> .\dtmf2num.exe .\4skj2-ydens.wav

DTMF2NUM 0.1c
by Luigi Auriemma
e-mail: aluigi@autistici.org
web:    aluigi.org

- open .\4skj2-ydens.wav
  wave size      815104
  format tag     1
  channels:      2
  samples/sec:   48000
  avg/bytes/sec: 192000
  block align:   4
  bits:          16
  samples:       407552
  bias adjust:   -162
  volume peaks:  -13878 13878
  normalize:     18889
  resampling to: 8000hz

- MF numbers:    7

- DTMF numbers:  663300AB1C755
PS F:\aaaCTF\aaa工具\DTMF拨号音识别手机键盘密码> .\dtmf2num.exe .\aq75x-ar9ae.wav

DTMF2NUM 0.1c
by Luigi Auriemma
e-mail: aluigi@autistici.org
web:    aluigi.org

- open .\aq75x-ar9ae.wav
  wave size      3616768
  format tag     1
  channels:      2
  samples/sec:   48000
  avg/bytes/sec: 192000
  block align:   4
  bits:          16
  samples:       1808384
  bias adjust:   -58
  volume peaks:  -14155 14155
  normalize:     18612
  resampling to: 8000hz

- MF numbers:    77

- DTMF numbers:  6688330AB1C775
PS F:\aaaCTF\aaa工具\DTMF拨号音识别手机键盘密码>
```

前面都没出现8，这次终于出现了8，所以应该是6830AB1C75，得到flag

EasySteg

哥哥球球了别套了再套下去套神都哭了呜呜呜

JK为单图盲水印(java)，用imagein也行的



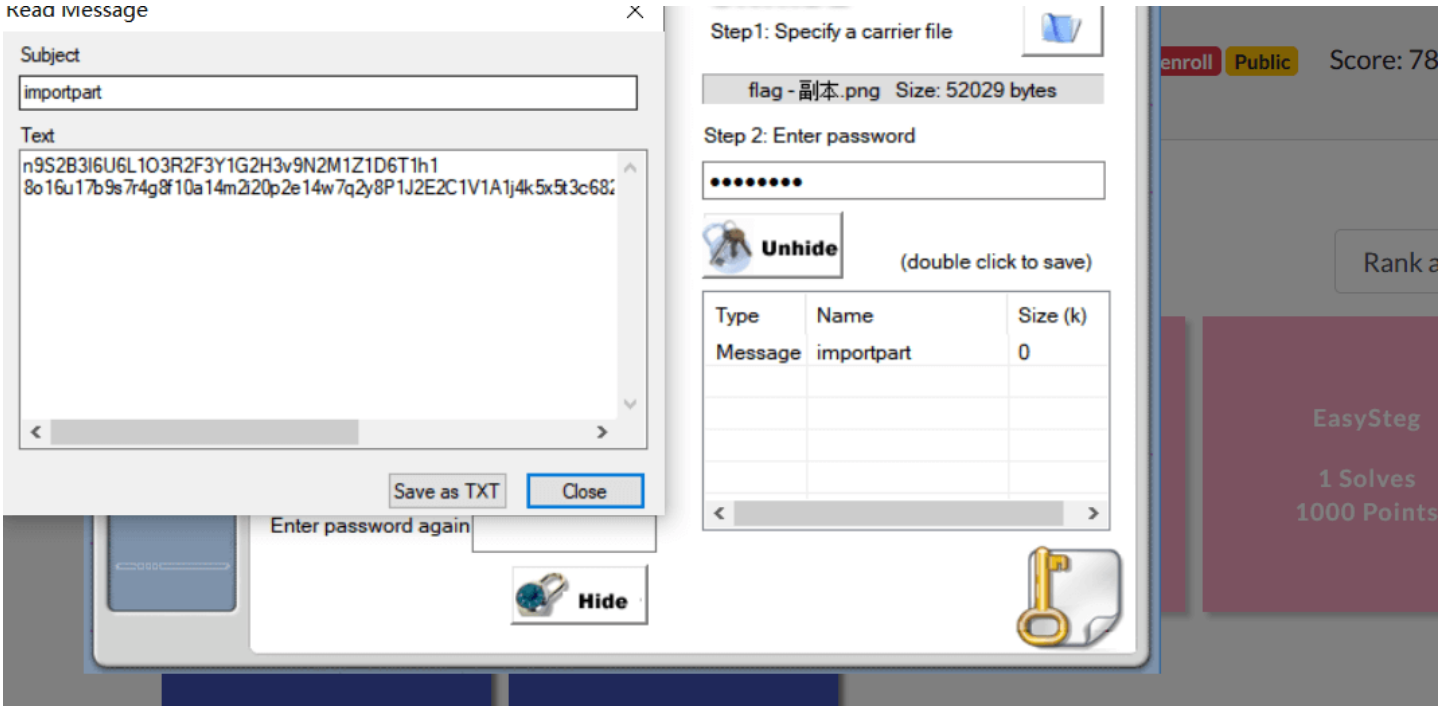
然后在flag.rar的注释里面有一串tab和space组成的密文，转space转0，tab转1

然后解压，在flag.png的末尾有另一个png，那个png明显有flag字样，用stegsolve反一下色即可看清

flag{156cca8e

然后还在这个图片后面发现了明显的oursecret的特征，结合上面的base64(-b81b-)

得到oursecret的密码为LWI4MWIt



除了数字以外，没有重复字符，里面还有{}，结合01串，得出为哈夫曼编码，利用红明谷的脚本即可解码

对应关系看脚本即可理解

```
import copy
import re

def dfs(c, d):
    if len(c.keys()) == 1:
        # g = {'j':29,'z':31,'7':25,'e':31,'l':23,'6':37,'4':32,'p':38,'h':27,'g':26,'x':28,'i':25,'u':27,'n':25
```

```
,'8':36,'0':24,'o':23,'c':28,'y':24,'1':29,'b':26,'m':27,'2':28,'v':25,'d':33,'f':28,'9':33,'t':21,'w':22,'a':31
,'r':24,'s':16,'k':32,'5':25,'q':23,'3':32,'{':1,'-':4,'}':1,}
        # num = 0
        # for k in g.keys():
        #     num += g[k] * len(d[k])
        # print(num)
        # print(c, d)
        g = {}
        for k in d.keys():
            g[d[k]] = k
        a = '11101111110001000011110000110100011011111001101000111101111000101001111101110011011111000110001111
11110101110001111110011100001101000111101001101111100111011110001000011100111001111011100011111110011101111011
01110101111011111010111011011010001101010111010111111110111101111011011101011110100110101101000111001000110101011
11010111110110110111100011010111010111110110111001001011011110100111010111111010111111011110111101001111010111
11011111010111001001111001010110111011101111110010101101001001101111100111110011010001111101110010110011100001111
00001111000011011111001100001110000110110011010001110000111111001111000011011001101000111001110110000111011000102
01111111110010110011010001111011110110010011011111000001111100010010001001111101110110111111101101001001001011011
11011111011011011110011010010110111111001111101101101101101111011101011011101111001111101101101000011011111100
00111111011110011101011111001101001101111010100110111011110011011011110100110101110101111100110111101110110101010
11010111111001111110111010110110101001101011111101001101000111010010111000010011001110111111011110110111110110
11101110101101001011011011010110101111111011110011101111111100110101101001011011110111011101100111001000001111001
10010010101100000011110010101100100010110010000001100101100100100000111100110010010101100000011110010101100101110
11111001010001010001010010000011110011001001010110000001111001010001010001101111100100000110010101001000001111001100100
11000001101111100100000111100110010010101100000011110010100010100011011111001000000110010101001000001111001100100
01010010101100000011110010101100100010001001000011111110001010010000011110011001001010110000001111001010110010111
11111100101000101100010001001000001111001100100101011000000111100101000101000110111110010000001100101010010000011
11100110010010101100000011110010101100100110010010101100000011011111001000001111001100100101011000000111100101011
00100010110010000001100101100100100000111100110010010101100000011110010100010100010100101011000001101111100100000
01111001100100101011000000111100101001010110010001011001000000110010110010010000011110011001001010110000000111100
10101100101111111100101000101000101001000001111001100100101011000000111100101011001000100010010000111111100010100
01000001111001100100101011000000111100101011001000101100100000011001011001001000001111001100100101011000000111100
01010110010101001000000110010111111111001000001111001100100101011000000111100101011001000101100100000011001011001
00100000111100110010010101100000011110010101100100010001001010010000011111110001010010000011110011001001010110000
00111100101011001010100100000011001011111111100100000111100110010010101100000011110010100010100010100101011000001
10111110010000011110011001001010110000001111001010001010001101111100100000011000000111100100000111100110010010101
11000000111100101000101000000100101000101000011111101101001111001110101111101110000101101011010111101000111110011
10111110101111111011010100111011111011001000001011110110101111011011011100110011100011000111110011100100000101100
11000101111011111111011011011100001110100001011111100011000001100011110010100100000110000101110000100010110111100
00010111001111111000011101101011110100110100111011110100010010111110110101111110011111100111000110000100001101110
11000001111100010011001110111101111111011111001000111000011101101110000111011011110011101111111011011010110100011
11110001001011111110111000001000001011100101111000110101101000111111011110011101011110101110110111001001100111011
11110011110111110000000100100111010011011111111011010111101110111100111010011011110101001101110111010111100111000
00011010010111111000100110011101111101011110111110
```

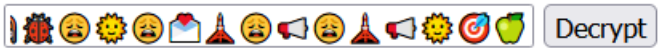
```
        k1 = list(c.keys())[1]
        if c[k0] > c[k1]:
            k0, k1 = k1, k0
        for k in list(c.keys())[2:]:
            if c[k] < c[k1]:
                if c[k] < c[k0]:
                    k0, k1 = k, k0
                else:
                    k1 = k
        for a in k0:
            d[a] = '0' + d[a]
        for a in k1:
            d[a] = '1' + d[a]
        c[k0+k1] = c[k0]+c[k1]
        del c[k0]
        del c[k1]
        dfs(copy.deepcopy(c), copy.deepcopy(d))

c = {'n':9,'S':2,'B':3,'I':6,'U':6,'L':1,'O':3,'R':2,'F':3,'Y':1,'G':2,'H':3,'v':9,'N':2,'M':1,'Z':1,'D':6,'T':1
,'h':18,'o':16,'u':17,'b':9,'s':7,'r':4,'g':8,'f':10,'a':14,'m':2,'i':20,'p':2,'e':14,'w':7,'q':2,'y':8,'P':1,'J
':2,'E':2,'C':1,'V':1,'A':1,'j':4,'k':5,'x':5,'t':3,'c':6,'8':24,'9':56,'{':1,'3':217,'d':97,'4':83,'6':54,'0':2
5,'1':12,'2':10,'5':8,'7':10,'}':1,'l':4,'Q':1,'W':1,'z':1}
d = {}
for k in c.keys():
    d[k] = ''
dfs(copy.deepcopy(c), copy.deepcopy(d))

#nSBIULORFYGHvNMIOUZSDTNhnoubuosrgfbouvasmruiohauiopewgvfbwuivpqynqwUPIFJDDBUEDUIDHBUIDCVHJIOAejikxneiwkyiohwehi
ooiuyhiosehfhuiaetyhovauieyrghfuotgvac89xcboiyuweagihniaweo{3d46303d39463d39383d41393d46303d39463d39323d38433d46
303d39463d39383d38463d46303d39463d39333d39333d46303d39463d39303d39453d46303d39463d38453d41333d46303d3d39463d3937
3d42433d46303d39463d39323d38373d46303d39463d38453d41333d46303d39463d39303d39453d46303d39463d39383d41393d46303d39
463d38433d39453d46303d39463d3d39383d41393d46303d39463d39323d38433d46303d39463d39373d42433d46303d39463d39383d4139
3d46303d39463d39333d41323d46303d39463d39383d41393d46303d39463d39373d3d42433d46303d39463d39333d41323d46303d39463d
38433d39453d46303d39463d38453d41463d46303d39463d38443d3846}huilagsieufrcb78QWEGF678Rniolsdf149687189735489246ava
eukighf6497ejixcnbmlolohnbasik2647893hasfhuvzxchbjkaefgyhuetyuhjadfxcvbn
```

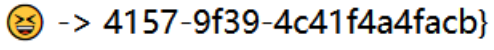
{}里面的内容拿去hex，发现是Quoted Printable，但是有些地方出现了两次==，于是将其重复的删掉，得到

```
▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯
```

不是base100，所以拿去试试国外的一种，以前做la佬的ctfshow月饼杯_共婵娟用到的

https://github.com/pavelvodrazka/ctf-writeups/tree/master/hackyeaster2018/challenges/egg17





拼接起来，得到flag

```
flag{156cca8e-b81b-4157-9f39-4c41f4a4facb}
```

*ez_misc

hint是零宽，说的是和piet很像的一种esolang，然后结合ctfshow6月赛八神的babyLSBwithHelicopter

可以知道是brainloller

然后010打开，提示解出来是后面steghide的密码，并且CRC报错，用爆CRC的脚本一爆就发现正确宽度为14，高为12

用bftools解，bftools.exe decode brainloller bf.png，得到的bf再去解码得到密码Hello Worl

但是解不出来，于是用我6月赛写的脚本去解https://blog.csdn.net/qq_42880719/article/details/117479024

解出来是Hello Worle!，我猜是Hello World!

结合题目新上的hint，得到的密码我试过有

```
Hello Worl
Hello_Worl
Hello Worle!
Hello_Worle!
Hello World!
Hello_World!
Hello World
Hello_World
```

可惜都不对，通过出题人的朋友问了下出题人，他朋友也表示解不出来，但是出题人是能解出来的(好像用的是本地的附件)
所以我总感觉是比赛题目附件的问题？

```
┌──(root💀kali)-[~/Desktop]
└─# steghide extract -sf 2.jpg -p Hello_Worl
wrote extracted data to "f1ag.txt".
```

```
PS F:\aaaCTF\aaa工具\00工具中的好工具（有重复）\aCTFtools\隐写\图像隐写\steghide> .\steghide.exe extract -sf .\00000000.
jpg -p Hello_Worl
steghide: could not extract any data with that passphrase!
PS F:\aaaCTF\aaa工具\00工具中的好工具（有重复）\aCTFtools\隐写\图像隐写\steghide>
```

—17:02—:经过一个半小时的积极反馈

```
┌──(root💀kali)-[~/桌面]
└─# md5sum brainlol.png
5c56bca27854e873234fc6f4a2248ba1  brainlol.png

┌──(root💀kali)-[~/桌面]
└─#

┌──(root💀kali)-[~/桌面]
└─# md5sum brainlol.png
39e733c83a3b34437db734d75fd7c04f  brainlol.png

┌──(root💀kali)-[~/桌面]
└─#
```

就bftools.exe解出来的把空格替换为下划线Hello_Worl解steghide，得到个文本
后面是一个熊曰，然后就完事

Re

EasyRe

flag就在常量里面。。

```
flag{fc5e038d38a57032085441e7fe7010b0}
```