web

本来打算往web方向做，但是，这周的web题我一题都没做出来。。。

pwn

太菜了

re

babypyc

这里的pyc不能直接用uncompyle6等工具反编译，但可以用marshal读取字节码。一开始不会用marshal，感谢Lurkrul大佬。
根据字节码写出来源码大概长这样

```
O0o = '/KDq6pvN/LLq6tzM/KXq59Oh/MTqxtOTxdrqs8OoR3V1X09J'
O0o = b'QreZoJSeWr2ioJN4cXhvvNO8f4mfqKDtrrftpI/JZ1JvRV9Q'
flag = getflag()

raw_flag = flag[6:-1]
if len(flag) -7 != 36:
    print('Wrong length')
else:
    raw_flag = raw_flag[::-1]
    ciphers = [[raw_flag[6*row+col] for row in range(6)] for col in
range(6)]#第6，12。。    7，14。。。。 8，16。。。
    for row in range(5):
        for col in range(6):
                ciphers[row][col] += ciphers[row+1][col]
                ciphers[row][col] %= 256
    cipher = ''
    for row in range(6):
        col = 0
        while col < 6:
```

```
            cipher += bytes([ciphers[row][col]])
            col += 1

    b64encode(cipher)
```

exp:

```
import base64

O0o = b'QreZoJSeWr2ioJN4cXhvvNO8f4mfqKDtrrftpI/JZ1JvRV9Q'
a = base64.b64decode(O0o)
x = [[a[i + 6 *j] for i in range(6)] for j in range(6)]

for i in range(1, 6):
    for j in range(6):
        x[5 - i][j] = x[5 - i][j] - x[6 - i][j]
        if x[5 - i][j] < 0:
            x[5 - i][j] += 256
s = []
for i in range(6):
    for j in range(6):
        s.append(x[j][i])

s = bytes(s).decode()
print('hgame{' + s[::-1] + '}')
```

> hgame{PytH0n_0pc0dE_Is-so~!NTERe$TiNgG89!!}

crackme

好像是用c#写的东西，用jetbrains的dotpeek反编译。发现是关于aes加密的
主要就是CBC模式的特点。

```
import base64
from Crypto.Cipher import AES

r = AES.new(base64.b64decode('SGc0bTNfMm8yMF9XZWVLMg=='), AES.MODE_CBC,
base64.b64decode('MFB1T2g5SWxYMDU0SWN0cw=='))
t = AES.new(base64.b64decode('SGc0bTNfMm8yMF9XZWVLMg=='), AES.MODE_ECB)
a1 = base64.b64decode('mjdRqH4d1O8nbUYJk+wVu3AeE7ZtE9rtT/8BA8J897I=')
```

```
a2 = 'MFB1T2g5SWxYMDU0SWN0cw=='
a3 = 'dJntSWSPWbWocAq4yjBP5Q=='#密文分组2
def xor(s1, s2):
    #assert len(s1)==len(s2)
    return bytes( map( (lambda x: x[0]^x[1]), zip(s1, s2) ) )

s1 = a1[:16]
s2 = a1[16:]
s2 = t.decrypt(s2)
s1 = t.decrypt(s1)

s3 = b'Same_ciphertext_'
text1 = base64.b64encode(xor(s1, s3)).decode()

s4 = r.encrypt(s3)#密文分组1
s5 = t.decrypt(base64.b64decode(a3))
text2 = xor(s5, s4).decode()

print('hgame{' + text1+text2 + '}')
```

> hgame{L1R5WFl6UG5ZOyQpXHdlXw==DiFfer3Nt_w0r1d}

babyPy

python字节码
翻译过来大概长这样

```
import dis

def foo(flag):
    000 = flag[::-1]
    00o = list(000)
    for 00 in range(1, len(00o)):
        0o = 00o[00 - 1] ^ 00o[00]
        00o[00] = 0o
    0 = bytes(00o)
    0.hex()
dis.dis(foo)
```

```
s = '7d037d045717722d62114e6a5b044f2c184c3f44214c2d4a22'
s = [0x7d, 0x03, 0x7d, 0x04, 0x57, 0x17, 0x72, 0x2d, 0x62, 0x11, 0x4e]
s += [0x6a, 0x5b, 0x04, 0x4f, 0x2c, 0x18, 0x4c, 0x3f, 0x44, 0x21, 0x4c]
s += [0x2d, 0x4a, 0x22]
b = '}'
for i in range(1, len(s)):
    a = s[i - 1] ^ s[i]
    b += chr(a)

print(b[::-1])
```

> hgame{sT4cK_1$_sO_e@Sy~~}

unpack

按照群里的资料脱壳
再用ida打开

```
s = [0x68 ,0x68, 0x63, 0x70, 0x69, 0x80, 0x5b, 0x75, 0x78, 0x49, 0x6d]
s += [0x76, 0x75, 0x7b, 0x75, 0x6e, 0x41, 0x84, 0x71, 0x65, 0x44]
s += [0x82, 0x4a, 0x85, 0x8c, 0x82, 0x7d, 0x7a, 0x82, 0x4d, 0x90]
s += [0x7e, 0x92, 0x54, 0x98, 0x88, 0x96, 0x98, 0x57, 0x95, 0x8f, 0xa6]
for i in range(42):
    print(chr(s[i] - i), end='')
```

> hgame{Unp@cking_1s_R0m4ntic_f0r_r3vers1ng}

crypto

Remainder

中国剩余定理在rsa方面的应用 (啥也不懂，套公式公式就完事儿了)

```
from Crypto.Util import number
import gmpy2
```

```
x =
784307860116505212245619248148436142948069749885995910589155203975185262964
227910896921074885341575898566112299780686599709763749716589099872997597195
335193582321807214807196356025155259426789888967271288848036382572278481762
981728961554638132642069825057976130672151828495593563360156345431818062963
55552543
y =
495763564234742221882051873068841676207464796775901212137910939089772958034
762035100010601809591909172768175411424115238675551472019924802205314310196
276815723351032005863885196959313483049706518755824130524112248188441609454
108841305757716179191496193417623256333013137329472641255768660339340184628
43559419
z =
481310779626494978331892926378614427675621474470401344110788844855138405531
881859543833302361902533889377855306582797686202130622440531516149628936289
463435956425138707668778105344805367372003026995393968105454200210542252046
834285228203503564708835744638491464221502443041476181956137963990104921253
83322922
e = 65537
p =
945982963057133766525404116319494343013962351116733727382767546541882670108
055225420680044531376785988913354081702776013819445842793393620565792623084
275446716886149238397945226713785592767847347587272130704038386322862804734
500867622867068639229687232028303982662205338851291755021425336005592923880
05914561
q =
150088216417404963893679242888992998793257903343994792697939121738029477790
454833496600101388493792476973514786401036309378542808470513073408894727406
158296404360452232777491992630316999043165374635001806841520490997788796152
678742544032835808854339130676283497122770901196468323977265095016407164510
827505883
r =
145897736096689096151704740327665176308625097484116713780050311198775607465
862066406830851710261868913835866335107146242979359964945125214420821146670
919741118254402096944139483988745450480989706524191669371208210272907563936
516990473246615375022630708213486725809819360033470468293100926616729742277
729705727

M = p*q*r
Mp = q*r
Mq = p*r
Mr = p*q
tp = gmpy2.invert(Mp, p)
tq = gmpy2.invert(Mq, q)
tr = gmpy2.invert(Mr, r)
```

```
c = x*tp*Mp + y*tq*Mq + z*tr*Mr #c=pow(m,e,M)

phin = (p-1)*(q-1)*(r-1)
d = gmpy2.invert(e, phin)
print(number.long_to_bytes(pow(c,d,M)).decode())
```

运行结果为

> 1hAyuFoOUCamGW9BP7pGKCG81iSEnwAOM8x
> ** DO NOT GUESS ME ****
> hg In number theory,
> am the Chinese
> e{ remainder theorem
> Cr states that if one
> T_ knows the
> w0 remainders of the
> Nt Euclidean division
> +6 of an integer n
> Ot by several
> h3 integers, then
> R_ YOU CAN FIND THE
> mE FLAG, ;D
> !!
> !}
> ** USE YOUR BRAIN ****
> cbl8KukOPUvpoe1LCpBchXHJTgmDknbFE2z

> hgame{CrT_w0Nt+6Oth3R_mE!!!}

Verification_code

sha256
签到题，暴力穷举即可

```
import os, sys, signal
import string, random
from hashlib import sha256

a = 'H3VPl1mQutHB3NZG'
res = 'e9872a762c9c35c32efc4a1bc935fa8cd0e48e7b70ca27ea50392a546e892f59'
```

```
    s = string.ascii_letters + string.digits
    for i in s:
        for j in s:
            for k in s:
                for l in s:
                    cmp = i + j + k + l + a
                    if sha256(cmp.encode()).hexdigest() == res:
                        print(cmp)
```

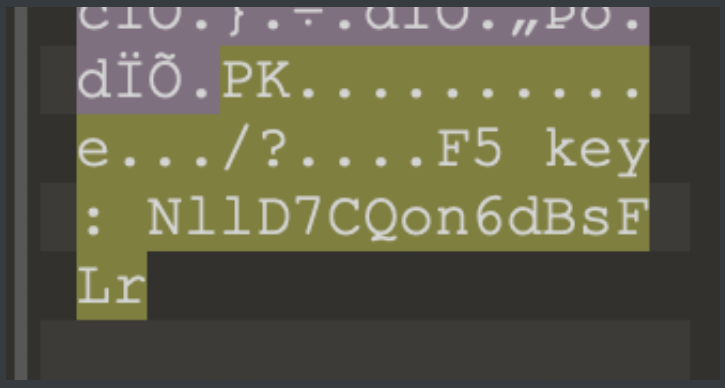
```
wctpd@WCTPDdeMBP F5-steganography % nc 47.98.192.231 25678
sha256(XXXX+H3VPl1mQutHB3NZG) == e9872a762c9c35c32efc4a1bc935fa8cd0e48e7b70ca27ea50392a546e892f59
Give me XXXX: NAuh
The secret code?
> I like playing Hgame
Ok, you find me.
Here is the flag: hgame{It3Rt0O|S+I5_u$3fu1~Fo2_6rUtE-f0Rc3}
Bye~
wctpd@WCTPDdeMBP F5-steganography %
```

misc

所见即为假

下载得到一个压缩文件，双击就解压了（mac上直接就解压了，后来在Windows上打开发现需要密码，是伪加密），解压后是一张图片，名字是flag in picture

用010editor打开压缩包



发现这么一串东西

百度一通，发现有个叫f5隐写的东西

下载使用工具

```
[wctpd@WCTPDdeMBP F5-steganography % java Extract /Users/wctpd/Downloads/FLAG_IN_
PICTURE.jpg -p N1lD7CQon6dBsFLr
Huffman decoding starts
Permutation starts
10911744 indices shuffled
Extraction starts
Length of embedded file: 222 bytes
(1, 127, 7) code used
```

```
526172211A0701003392B5E50A010506000501018080008B9527AEA2402030BA70004A70020CB5BDC2D80000008666C61672E7478
```

得到一串字符，看起来像是16进制

Edit As: Hex Run Script Run Template

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789ABCDEF
0000h:	52	61	72	21	1A	07	01	00	33	92	B5	E5	0A	01	05	06	Rar!....3’µå....
0010h:	00	05	01	01	80	80	00	B9	52	7A	EA	24	02	03	0B	A7	....€€.¹Rzê$...§
0020h:	00	04	A7	00	20	CB	5B	DC	2D	80	00	00	08	66	6C	61	..§. Ë[Ü-€...fla
0030h:	67	2E	74	78	74	0A	03	02	9A	9D	6C	65	DF	CE	D5	01	g.txt...š.leßÎÕ.
0040h:	68	67	61	6D	65	7B	34	30	38	37	5E	7A	23	6D	73	77	hgame{4087^z#msw
0050h:	33	34	45	52	74	6E	46	55	79	71	70	4B	55	6B	32	64	34ERtnFUyqpKUk2d
0060h:	6D	4C	50	57	36	30	7D	1D	77	56	51	03	05	04	00		mLPW60}.wVQ....

地球上最后的夜晚

打开压缩包，里面是一个加密的压缩包和一个pdf，pdf名字是no password

这里是pdf隐写，用wbs43open提取内容，no password的意思是提取的时候不用密码

得到压缩文件的密码

解压后是一个word

这里可以改成zip格式，打开看到很多文件，在其中一个文件里找到flag

```
<?xml version="1.0" encoding="UTF-8" standalone="true"?>
<flag>hgame{mkLbn8hP2g!p9ezPHqHuBu66SeDA13u1}</flag>
```

Cosmos的午餐

又是wireshark，这次又多了个log文件
打开发现都是tcp，只有一个http并且里面只能看到一个crt文件
百度一番发现要导入log文件
多出了很多http，导出
找到那个最大的文件，改格式解压，根据题目提示找到详细信息

> Key: gUNrbbdR9XhRBDGpzz

然后看到图片名字，使用outguess解密得到一个网址
打开后下载了一个文件，是个二维码，扫码即得flag

> hgame{ls#z^$7j%yL9wmObZ#MKZKM7!nGnDvTC}

玩玩条码

视频隐写，用MSU StegoVideo提取到7z密码
打开是一个条形码，扫码得flag

> hgame{9h7epp1flwIL3fOtsOAenDiPDzp7aH!7}

- 那么那个JPNPostCode是干嘛的？？？