Reverse

FAKE

先上ida 好家伙反编译出来的什么鬼东西

最后找了一找 发现一个类似于检查flag的函数 但是好家伙这个函数真的长 一大堆goto

品一品 好像要我解一个36元方程一次 ~~az 离谱~~

算了先解吧 写了一个python算矩阵逆 但是 解出来的东西……

`hgame{@_FAKE_flag!-do_YOu_know_SMC?}`

好家伙假flag ~~淦~~

去百度一下SMC

`SMC(self-Modifying Code)，就是在真正执行某一段代码时，程序会对自身的该段代码进行自修改，只有在修改后的代码才是可汇编，可执行的。在程序未对该段代码进行修改之前，在静态分析状态下，均是不可读的字节码，IDA之类的反汇编器无法识别程序的正常逻辑。是一种反调试代码技术。`

然后就该找是哪个函数对代码执行了自加密~~大概先找找函数指针吧 可能还有骚操作~~

```
__int64 __usercall sub_40699B@<rax>(__int64 a1@<rbp>)
{
  __int64 result; // rax
  unsigned int i; // [rsp-Ch] [rbp-Ch]
  __asm { endbr64 }
  sub_4010E0();
  for ( i = 0; ; ++i )
  {
    result = i;
    if ( i > 0x43E )
      break;
    *((_BYTE *)sub_401216 + (signed int)i) ^= byte_409080[i];
  }
  return result;
}
```

那就是按照教程先给ida打个补丁吧

```
#include <idc.idc>
static main()
{
    auto addr = 0x401216;
    for (auto i = 0; i <= 0x43E; i++)
    {
    PatchByte(addr + i, Byte(addr + i) ^ Byte(0x409080 + i));
    }
}
```

这样sub_401216就被解密了 好家伙又是一个矩阵运算~~至少没用36行方阵子~~

~~md sb ida 竟然认不出一个数组~~

Crypto

LikiPrime

先看描述 `Wow! RSA! az` 估计是RSA加密了 打开一看 果真是RSA~~这不是废话吗~~

```
def get_prime(secret):
    prime = 1
    for _ in range(secret):
        prime = prime << 1
    return prime - 1
```

这个是获取 `p` `q` 的操作 会发现他们都是2的n次方减一的亚子~~估计是想让我们用这个特性爆破出 `p-q` 吧 但是我的操作好像有点骚~~

众所周知 找出 `p` `q` 这道题就完活了

怎么找呢 分解质因数

怎么分解呢 http://www.factordb.com/ 或者 ~~yafu~~好家伙 丢上去直接完活了

~~辜负了出题人的一片苦心~~

HappyNewYear!!

还是看描述 `群发` ~~这么真实的吗~~

打开代码康康 发现还是RSA ~~所以我们就可以一个个破解了 但是我试的时候 发现一分解一个 `n` 就碰到一个质数~~

这里就要科普一个东西 叫做 `广播明文攻击` 就是当明文相同的时候 就可以不用密钥来解出密文~~中国同余 我看不懂 所以就不写数学原理了~~

上网找了一个代码 直接套

```
from struct import pack,unpack
import zlib
import gmpy
def my_parse_number(number):
    string = "%x" % number
    #if len(string) != 64:
    #    return ""
    erg = []
    while string != '':
        erg = erg + [chr(int(string[:2], 16))]
        string = string[2:]
    return ''.join(erg)
def extended_gcd(a, b):
    x,y = 0, 1
    lastx, lasty = 1, 0
    while b:
        a, (q, b) = b, divmod(a,b)
        x, lastx = lastx-q*x, x
        y, lasty = lasty-q*y, y
    return (lastx, lasty, a)
def chinese_remainder_theorem(items):
  N = 1
```

```
  for a, n in items:
    N *= n
  result = 0
  for a, n in items:
    m = N//n
    r, s, d = extended_gcd(n, m)
    if d != 1:
      N=N/n
      continue
      #raise "Input not pairwise co-prime"
    result += a*s*m
  return result % N, N
sessions=[]#数据略
data = []
for session in sessions:
    e=session['e']
    n=session['n']
    msg=session['c']
    data = data + [(msg, n)]
print ("Please wait, performing CRT")
x, n = chinese_remainder_theorem(data)
e=session['e']
realnum = gmpy.mpz(x).root(e)[0].digits()
print(my_parse_number(int(realnum)))
```

然后就是非酋时间 ~~我试了3次才试出一组一样的 有这么非酋的吗~~

~~比较坑的一点是 这个题的flag是分两段的 然后我在解出下半段的时候 让我蒙蔽了好久 还以为是我代码出问题了 flag只有一半 然后脑瘫了半小时后 明白可能flag是分两段的~~

MISC

A R K

下载到一个流量包 然后翻一下就找到了DNS的网站 ~~在盯着年和夕的涩图并对涩图毛手毛脚操作一整天后~~ 发现网站的内容没啥子用

然后看到出题人的hint 仅用流量分析 emmmmmmmmm

看到流量包有两部分组成 前边是加密流量 后边是FTP数据

看FTP的流量 发现是出题人正在试图下载一个`ssl.log`这个估计和前边的加密流量有关 然后就把这个下下来 导到wireshark里

前边的加密流量就明文了 然后就结合hint 战斗回放

然后就是找啊找啊找 发现一个叫`getBattleReplay` 然后里边是有一段base64加密的数据

文件头是50 4B 05 06 emmmmm 这不是zip的文件尾吗

好家伙改一改 换成03 04 zip get daze

这个压缩包里藏着一段很长很长的字符串 品一品 这应该是自律数据了

再结合描述`没用 没用 出题人用可部署单位画了个东西 背景是白色的`好家伙 别告诉我这个屑博士让夕小姐站出一个二维码吗

然后用正则表达式把pos数据爬出来 然后py用PIL写个脚本把图片涂黑

emmmm 还真的是一个二维码

扫一下就得到flag了 ~~这涩图真的好看~~

A R C

压缩包里有三个文件 一个字体 一个压缩包 一个图片`8558`

图片上问你知不知道bv号 下边附带一个很长很长的字符串

先把字体装上然后把那个字符串一个个的抄下来 ~~这个ij ll真的难分~~

再结合hint~~和我不断的骚扰出题人~~ BV号是一个base58的变种 所以这串字符应该是base85编码的~~记得找几个靠谱的在线工具 不然玩死~~

然后我们得到一个长度为58的字符串 结合那个神奇的hint `这个字体表示的东西也是一个table` 恰巧bv号的生成过程里也是需要一个table

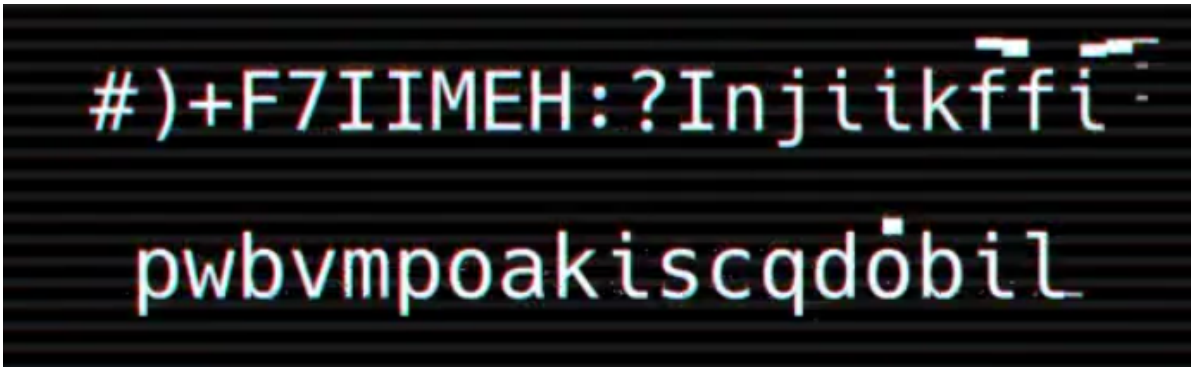
再康康压缩包的名字`BVenc(10001540)` 就是把这个av号编码成bv 然后我们就有压缩包的密码了

然而压缩包有一个视频和一段加密通话 再再结合hint `用了某种ROT的范围，但是位移不一样` 看里边出现的符号 估计是ROT47

然后就是挨个试位移 发现位移是9 然后就得到一段文本~~并用认知滤网滤掉了一些话~~

```
Flag is not here, but I write it because you may need more words to analysis what encoding the line1 is.
For line2, Liki has told you what it is, and Akira is necessary to do it.
```

然后康康视频 一帧一帧的康 发现里边有两行奇妙的文本



第一行就用和之前一样的ROT解码就好了 得到 `MSUpasswordis:6557225`

MSU则是一个AVI视频的隐写 ~~然后我就直接把这个MKV转成AVI然后解码 还真的解出来了~~ 得到一个网站和两行不知道啥意思的文字~~和一大堆让我误以为是snow隐写的空格~~

先登录那个网站 那两行就是账号密码 发现网站上啥都没有 就是一个`/` 再再再结合hint 看来要找到一个指定的URL~~然后我就开了一个全站扫描 发现没有用~~

然后就是解码第二行申必文本 week1密码学第一题 ~~淦~~是缝合怪 ~~然后我试了各种栅栏和凯撒 发现解不出来~~

再再去骚扰一下出题人 ~~并同时对自己的语文理解能力产生了怀疑~~

第二行提示的意思是 上次要用到Liki的地方 现在是Akira 好家伙维吉尼亚 最后又得到一段申必文本 估计是url了

登上去一看 真的是flag