CRYPTO

midRSA

题目存在非预期，m0即可解出结果

```
from Crypto.Util.number import *
m0=13292147408567087351580732082961640130543313742210409432471625281702327748963274496942276607
print(long_to_bytes(m0))
```

```
hgame{0ther_cas3s_0f_c0ppr3smith}
```

midRSA revenge

Coppersmith 攻击,已知明文前几位,可用sagemath中的small_roots进行求解。

```
from Crypto.Util.number import *

def copper(high_m, n, c):
    R.<x> = PolynomialRing(Zmod(n), implementation='NTL')
    m = high_m + x
    M = m((m^5 - c).small_roots()[0])
    print((int(M)))

n = 27814334728135671995890378154778822687713875269624843122353458059697288888640572922486287555
c = 45622131411586708863820720303449463624470661111162172357784872909606923006795813266301862560
m0= 99999002810033577734203106811693308232665325338039056370
high_m = m0 << 128
m=copper(high_m, n, c)
print(long_to_bytes(m))
```

```
hgame{c0ppr3smith_St3re0typed_m3ssag3s}
```

backpack

题目存在非预期，由enc即可解出

```
from Crypto.Util.number import long_to_bytes
enc=871114172567853490297478570113449366988793760172844644007566824913350088148162949968812541
print(long_to_bytes(enc))
```

```
hgame{M@ster_0f ba3kpack_m4nag3ment!}
```

backpack revenge

背包问题，分析代码，发现p的每一位即是否要将对应的a位放入bag，运用LLL算法攻击还原p,sagemath中有LLL算法实现。

```
import hashlib

a = [747630795102616991263455259 79, 517250494700689508104784875 07, 471903092695146090050453306 7
bag = 120254819682601389900652731 4947
n = len(a)
L = matrix.zero(n + 1)

for row, x in enumerate(a):
    L[row, row] = 2
    L[row, -1] = x

L[-1, :] = 1
L[-1, -1] = bag
res = L.LLL()
print(res.row(0).list()[:-1])
res=res.row(0).list()[:-1]
result = ''
for i in res:
    if i == 1:
        result = '1'+result
    else:
        result = '0'+result
print(int(result,2))
p=int(result,2)
flag='hgame{'+hashlib.sha256(str(p).encode()).hexdigest()+'}'
print(flag)
```

其中将矩阵第一行的1，-1置换时，-1对应0还是1都有可能。

```
hgame{04b1d0b0fb805a70cda94348ec5a33f900d4fd5e9c45e765161c434fa0a49991}
```

babyRSA

由gift可以解出e，e=73561，在求e对于phi的逆元时，发现报错。e与phi不互素，并且GCD(e,phi)=e,GCD(e,p-1)=e,GCD(e,q-1)=1。用sagemath中的nthroot解出。

```
from Crypto.Util.number import *

p=14213355454944773291
q=6184356205162070038634855117537193048606497844115920076561833974376400103329 7
c=10500213872246694649593663865603821400004347575163902508525511396508874927246190689258661625 0
gift=9751789326354522940

d=inverse(0x10001,p-1)
res=pow(gift,d,p)
e=res-114514
print(e)#e=73561
n=p**4*q
phi=(p**4-p**3)*(q-1)
print(GCD(e,phi))#73561
print(GCD(e,p-1))#73561
print(GCD(e,q-1))#1

e=73561
result = Zmod(n)(c).nth_root(e, all=True)
for i in result:
    plain=long_to_bytes(int(i))
    if b"hgame{" in plain:
        print(plain)
        break
```

```
hgame{Ad1eman_Mand3r_Mi11er_M3th0d}
```