cry1

flag{e6e5722e-4b9a-11ec-b784-00155d9a1603}



判断是base64编码

解码后得

from secret import flag

from Crypto.Util.number import \*

m = bytes\_to\_long(flag)

e1 = 667430104865289

e2 = 537409930523421

p = getPrime(512)

q = getPrime(512)

n = p\*q

c1 = pow(m, e1, n)

c2 = pow(m, e2, n)

print(f'c1 = {c1}')

print(f'c2 = {c2}')

print(f'n = {n}')

"""

c1 = 65902678572727724179176496573968997182712063317082289120453094068199325419989688382177808529042322217887334005084504796397220804856167255176415690217348252126097809130195208020694026250194047460581165024178358434305495364983830756552379335985399876528922076030595232679046941310786637260764992499375421464529

c2 = 85809403678250150153291471185999805870858123001273034212582847731825296891016810871397546134117012197599651729401590980020028382884068513201758926416192211821922593686232475967808964006786076460160428639353153658323208119453055070199243295330522804974849330926501091430419775155670264306222962413289616957519

n = 93012379949596679874010836520972463438155175961283277743514203871114329008044735500726440012464029144204813413909322389585966313426611488927292874319628063526009405144436605996389985977340280983469803412119458185047475253059636126555451557348169514975249710901899526974246139559730461540660990375034669042959

已知n,c1,c2,e1,e2.，m和n一个，e两个，所以共模攻击。

需要注意的是求e1,e2公因数，是3，最后结果是要开三次方。。

from Crypto.Util.number import \*  
from gmpy2 import \*  
import gmpy2  
  
  
c1 = 65902678572727724179176496573968997182712063317082289120453094068199325419989688382177808529042322217887334005084504796397220804856167255176415690217348252126097809130195208020694026250194047460581165024178358434305495364983830756552379335985399876528922076030595232679046941310786637260764992499375421464529  
c2 = 85809403678250150153291471185999805870858123001273034212582847731825296891016810871397546134117012197599651729401590980020028382884068513201758926416192211821922593686232475967808964006786076460160428639353153658323208119453055070199243295330522804974849330926501091430419775155670264306222962413289616957519  
  
n = 93012379949596679874010836520972463438155175961283277743514203871114329008044735500726440012464029144204813413909322389585966313426611488927292874319628063526009405144436605996389985977340280983469803412119458185047475253059636126555451557348169514975249710901899526974246139559730461540660990375034669042959  
e1 = 667430104865289  
e2 = 537409930523421  
def modulus(n, e1, e2, c1, c2):  
 \_, s, t = gcdext(e1, e2)  
 m = (pow(c1, s, n) \* pow(c2, t, n)) % n  
 print(long\_to\_bytes((iroot(m, 3)[0])))  
  
print gmpy2.gcd(e1,e2)  
print(gcdext(e1, e2))  
modulus(n, e1, e2, c1, c2)