实验十三-十四. 综合

【实验目标】综合使用各中第三方模块。

【实验要求】根据所给条件，编写代码，破解RSA。（1）借助sympy实现；（2）除了我们们介绍的比如numpy、密码模块之外，仅借助gmpy2模块完成。（3）尝试不要借助这些比较完善的第三方模块，完成破解，报告尝试过程以及尝试结果。

【实验内容】已知有人写了如下的代码，并将生成的(n,e,c)以及(n2,e2,c2,(p2+1)\*(q2+1))输出。

from Crypto.Util.number import \*

def ef():

p=getPrime(512)

q=getPrime(512)

flag=open("flag","rb").read()

m=bytes\_to\_long(flag)

e=65537

n=p\*q

c=pow(m,e,n)

print(n,e,c) #

return p,q

def epq(p,q):

p2=getPrime(1024)

q2=getPrime(1024)

e2=65537

m2=p+q

n2=p2\*q2

c2=pow(m2,e2,n2)

print(n2,e2,c2,(p2+1)\*(q2+1)) #

p,q=ef()

epq(p,q)

输出(n,e,c)以及(n2,e2,c2,(p2+1)\*(q2+1))如下：

(86902041785171683231305301154659574124170175717542478743469937094129588097899526472297338165763793395403138052107436307148655516237914989260641272558370267587581870375426597336633068656306063961150434589742092844368242428011721324737315708847899038491875643600715933822290329122093126146305056053635875427149L, 65537, 15833685544860151615528563318175057334854555807563811248911092184962045410721043935895279999715546874344448072757248024629816018687138160349497178202841685747385455984597261139196670787723191129190281464722284764818569442853999503305719960273532581788705319675785365330814771611391805600982635624166616994909L)

(18591274039015146371355461783271676729436253989312203484465267365829253730222536917260636899143825565341750487371058728119621305795504534209590790135844600724302582139316981344931079461283380928221091934542959757790517904211220170283310587231962911318192160954877212603169707693638590690557518269614342828433745751900636977114625384201443358493405946175157229330447527380345317933922535718700042422977893867943656666930789561406630888405895439054817890293649823406983304364531032464229623788486039136673884818993067818472490283937605836580590843487927803181532325018537076264511576862025094690559279649143498659102849L, 65537, 7392251042794448813724942265095985628606500546061544026052413736090062985580390082556940969813307099293888028876436850564855740297176392396726998769680589219160575972891500463789856128741737806143813567755324525667951325489807002065570746854557547784519649141813876707420713784143969266692484619922991587339418980913573323306950701125825931923149350170489304772562424051072424741158162269063304635895188390600892069271483630208193319815348033784877031408074730563809389105637498825225265591866481815230255385046830196748192233124781928655809322769875744414705189416861877961926362356094198048709542228727393696757158L, 18591274039015146371355461783271676729436253989312203484465267365829253730222536917260636899143825565341750487371058728119621305795504534209590790135844600724302582139316981344931079461283380928221091934542959757790517904211220170283310587231962911318192160954877212603169707693638590690557518269614342828434020648157035533967661393926860975983001244775722594600595802713222548261077015534360904269663877978505992018966625264663179236777522008665977063509158288897299644184441313228368572324485490486359818154230942787018240908451883098571389138398322839231059197174670650732257722735400365023996558029601185819826736L)

请编写代码，求出flag。