[NCTF2020]babyRSA Writeup

原创

_bestkasscn 于 2021-11-29 21:23:20 发布 200 收藏

分类专栏： CTF 文章标签： 密码学

本文链接：https://blog.csdn.net/bestkasscn/article/details/121619974

[NCTF2020]babyRSA

题目描述

```
from Crypto.Util.number import *
from flag import flag

def nextPrime(n):
    n += 2 if n & 1 else 1
    while not isPrime(n):
        n += 2
    return n

p = getPrime(1024)
q = nextPrime(p)
n = p * q
e = 0x10001
d = inverse(e, (p-1) * (q-1))
c = pow(bytes_to_long(flag.encode()), e, n)

# d = 19275778946037899718035455438175509175723911466127462154506916564101519923603308900331427601983476886255849200332374081996442976307058597390881168155862238533018621944733299208108185814179466844504468163200369996564265921022888670062554504758512453217434777820468049494313818291727050400752551716550403647148197148884408264686846693842118387217753516963449753809860354047619256787869400297858568139700396567519469825398575103885487624463424429913017729585620877168171603444111464692841379661112075123399343270610272287865200880398193573260848268633461983435015031227070217852728240847398084414687146397303110709214913
# c = 53827231680738281106961685582942066817579911490227778211275633014134832238745272333007211808392986170767056850411742474158261570965830550693373939878922627642112252270358807544174570567239091355252449579359069026656797771011301113927802375029286562257052624314319530035200939329243759021112800772552051182174367441120640694296786329232598986279971458038927539892556152731403000210406545059014427878106536265243057063166631693417972057529387555900565689867382278034874672741143982571879621407965511362205328096876068673856393677437055275116807199553807463776311564686898441508783814605609907556528994493400453135218 04
```

这道题给出了c,d,e,没有给p,q,但是p,q是相邻的素数，所以只需要知道n或者phiN就能把p,q爆破出来。

根据扩展欧几里得算法:ax+by=gcd(a,b)可以得到一组整数解（d,k），题目d已给出，所以可以列如下方程：

e * d + phiN * y = gcd(e,phiN),phiN * y = 1 - e * d,根据以往经验可知y的值不会很大，所以这里用for循环爆破。

（写脚本的时候忘记加break导致在这道题上浪费了两个多小时。。）

exp如下：

```python
import gmpy2
from Crypto.Util.number import *


def nextPrime(n):
    n += 2 if n & 1 else 1
    while not isPrime(n):
        n += 2
    return n


def check(phiN):
    tmp = gmpy2.iroot(phiN, 2)[0]
    while True:
        if isPrime(tmp):
            break
        else:
            tmp = tmp - 1
    if (tmp - 1) * (nextPrime(tmp) - 1) == phiN:
        return tmp


e = 65537
d = 19275778946037899718035455438175509175723911466127462154506916564101519923603308900331427601983476886255849200332374081996442976307058597390881168155862238533018621944733299208108185814179466844504468163200369996564265921022888670062554504758512453217434777820468049494313818291727050400752551716550403647148197148884408264686846693842118387217753516963449753809860354047619256787869400297858568139700396567519469825398575103885487624463424429913017729585620877168171603444111464692841379661112075123399343270610272287865200880398193573260848268633461983435015031227070217852728240847398084414687146397303110709214913
c = 53827231680738281106961685582942066817579911490227778211275633014134832238745272333007211808392986170767056850411742474158261570965830550693373939878922627642112252270358807544174570567239091355252449579359069026656797771011301113927802375029286562257052624314319530035200939329243759021112800772552051182174367441120640694296786329232598986279971458038927539892556152731403000210406545059014427878106536265243057063166631693417972057529387559900565689867382278034874672741143982571879621407965511362205328096876068673856393677437055275116807199553807463776311564686898441508783814605609907556528994493400453135218043
phiNy = 12632767247864858338208896430517083448494179857555954872199197908615213112331900554010207707511911246945495890421828002118008833382356992972061791174307435267384414263919862302017861737038797185882893300116626484648322956660770547698896345783586306465110230340200145597088447093849157021141199818475638038231513965464374644427818717743309127430899122412336065154368180232188232321065968873207569801715448898455234939471464165833432024444594468632094429438528354269704623749167330615747454988503020673622227599259854149298216700986564122107962129814311980083805801015285008674142507204159282582853515134400539665498177532803
p = 0
q = 0
for i in range(2, 100000):
    if (gmpy2.gcd(e, phiNy // i) == 1) & (d == gmpy2.invert(e, phiNy // i)):
        phiN = phiNy // i
        p = check(phiN)
        if p is not None:
            q = nextPrime(p)
            break
c = pow(c, d, p * q)
print(long_to_bytes(c))
```

NCTF{70u2_nn47h_14_v3ry_gOO0000000d}