2022红明谷-部分Crypto

原创

mxx307 于 2022-03-24 21:15:15 发布 74 收藏 2

分类专栏： CTF 文章标签： python

版权声明：本文为博主原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接和本声明。

本文链接：https://blog.csdn.net/qq_53283643/article/details/123721791

版权

CTF 专栏收录该内容

12 篇文章 0 订阅

订阅专栏

2022红明谷-部分Crypto

easy_ya

题目

```
from Crypto.Util.number import *
import os
from flag import flag
def gen():
    e = 3
    while True:
        try:
            p = getPrime(512)
            q = getPrime(512)
            n = p*q
            phi = (p-1)*(q-1)
            d = inverse(e,phi)
            return p,q,d,n,e
        except:
            continue
    return
p,q,d,n,e = gen()
r = getPrime(512)
m = bytes_to_long(flag+os.urandom(32))
M = m%r
c = pow(m,e,n)
print("r = %d"%r)
print("M = %d"%M)
print("n = %d"%n)
print("e = %d"%e)
print("c = %d"%c)
'''
r = 7996728164495259362822258548434922741290100998149465194487628664864256950051236186227986990712837371289585870678059397413537714250530572338774305952904473
M = 4159518144549137412048572485195536187606187833861349516326031843059872501654790226936115271091120509781872925030241137272462161485445491493686121954785558
n = 1315529642737317427440014393264700354142708643481395940041179596312865001989563029133779479206775253192602421215071960433232923747365959439429561949028148422062688709414854293391324216763671676218122604826247438216711832970237185732934523542849323488025488388479819167489518288262371121941420353805590205602870
e = 3
c = 4679466400670841713214794191871993836567148517629317201457539220316200581354444472018115104681864841734629228865674105641178081304474952072571892753526261831767984467150020472028621875453664388148374989220751675830569452999354229667028154811169244363966222057829371439622432559169783457220974604861614430728 2
'''
```

思路

已知$M = m \% r, c = m \% n$

则$c = (M + k$

因为$e = 3$。并不是很大，所以在模n下可以求出k

脚本

```
# SageMath
from Crypto.Util.number import long_to_bytes
r = 7996728164495259362822258548434922741290100998149465194487628664864256950051236186227986990712837371289585870678059397413537714250530572338774305952904473
M = 4159518144549137412048572485195536187606187833861349516326031843059872501654790226936115271091120509781872925030241137272462161485445491493686121954785558
n = 1315529642737317427440014393264700354142708643481395940041179596312865001989563029133779479206775253192602421215071960433232923747365959439429561949028148422062688709414854293391324216763671676218122604826247438216711832970237185732934523542849323488025488388479819167489518288262371121941420353805590205602870
e = 3
c = 467946640067084171321479419187199383656714851762931720145753922031620058135444447201811510468186484173462922886567410564117808130447495207257189275352626183176798446715002047202862187545366438814837498922075167583056945299935422966702815481116924436396622205782937143962243255916978345722097460486161443072820
R.<x> = Zmod(n)[]
f = (M+x*r) ^ 3 - c
f = f.monic()
k = f.small_roots()[0]
# k = 810968823598060539864535
m = M + k * r
print(long_to_bytes(int(m)))
# b'flag{53a2e494-964d-4506-a2c4-c34b9475dedd}W\xf1X6\xacP\x9bc~9\xfd\x0f\x96\xbf\x92\xb9+\xe5\xebPJ\x17\xc4\xb2\xe8\xad\x01\n\xf2j\xae\x15'
```

```
IPython: home/mxx307
sage: # SageMath
6978345722097460486161443072820
R.<x> = Zmod(n)[]
f = (M+x*r) ^ 3 - c
f = f.monic()
k = f.small_roots()[0]
# k = 810968823598060539864535
m = M + k * r
sage: from Crypto.Util.number import long_to_bytes
sage: r = 7996728164495259362822258548434922741290100998149465194487628664864256950051236186227986990712837371289585870
....: 6780593974135377142505305723387743059529044 73
sage: M = 4159518144549137412048572485195536187606187833861349516326031843059872501654790226936115271091120509781872925
....: 0302411372724621614854454914936861219547855 58
sage: n = 1315529642737317427440014393264700354142708643481395940041179596312865001989563029133779479206775253192602421
....: 1507196043323292374736595943942956194902814842206268870941485429339132421676367167621812260482624743821671183297
....: 0237185732934523542849323488025488388479819167489518288262371121941420353805590205602870287
sage: e = 3
sage: c = 4679466400670841713214794191871993836567148517629317201457539220316200581354444472018115104681864841734629228
....: 8656741056411780813044749520725718927535262618317679844671500204720286218754536643881483749892207516758305694529
....: 9935422966702815481116924436396622205782937143962243255916978345722097460486161443072820
sage: R.<x> = Zmod(n)[]
sage: f = (M+x*r) ^ 3 - c
sage: f = f.monic()
sage: k = f.small_roots()[0]
sage: # k = 810968823598060539864535
sage: m = M + k * r
sage: print(long_to_bytes(int(m)))
b'flag{53a2e494-964d-4506-a2c4-c34b9475dedd}W\xf1X6\xacP\x9bc~9\xfd\x0f\x96\xbf\x92\xb9+\xe5\xebPJ\x17\xc4\xb2\xe8\xad\x
01\n\xf2j\xae\x15'
sage:
```

sm2

待复现