破解加密数据

发表于 2021-01-25 更新于 2021-02-18 分类于 Challenge ， 2019 ， 工业信息安全技能大赛 ， 山东站
Challenge | 2019 | 工业信息安全技能大赛 | 山东站 | 破解加密数据

点击此处获得更好的阅读体验

WriteUp来源

https://www.dazhuanlan.com/2019/12/30/5e09dcd470284/

题目描述

> *某工控厂商自行研发了一套加密系统，这样的话只要不是系统内部人员，即使数据被窃听没关系了。你截获了一段密文：109930883401687215730636522935643539707，请进行解密，flag形式为flag{}*

题目考点

- *RSA*

解题思路

Rabin密码体制是RSA密码体制的一种，假定模数n=pqn=pq不能被分解，该类体制对于选择明文攻击是计算安全的。因此，Rabin密码体制提供了一个可证明安全的密码体制的例子：假定分解整数问题是整数上不可行的，那么Rabin密码体制是安全的。

Thm1 (Rabin密码体制) 设n=pqn=pq，其中pp和qq是素数，且p,q≡3(mod4)p,q≡3(mod4)，

设P=C=Z⋆nP=C=Zn⋆，且定义 对K=(n,p,q)K=(n,p,q)，定义：$$ eK(x)=x2(modn) 和 dK=y√(modn) $$

n为公钥，p和q为私钥。

注：条件p,q≡3(mod4)可以省去，条件P=C=Z⋆也可以弱化为P=C=Zn，只是在使用更多的限制性描述的时候，简化了许多方面的计算和密码体制分析。

下载附件得到

已知：e=2

通过分解,得到p=13691382465774455051，q=1084126018911527523

编写脚本解密

```
import binascii
import gmpy2
from libnum import *

c = 109930883401687215730636522935643539707
e = 2
p = 13691382465774455051
q = 1084126018911527523
n = p * q

mp = pow(c, (p + 1) / 4, p)
mq = pow(c, (q + 1) / 4, q)

inv_p = gmpy2.invert(p, q)
inv_q = gmpy2.invert(q, p)

a = (inv_p * p * mq + inv_q * q * mp) % n
b = n - int(a)
c = (inv_p * p * mq - inv_q * q * mp) % n
d = n - int(c)

print(a)
print(b)
print(c)
print(d)
```

Flag

```
flag{flag_EnCryp1}
```

- ***本文作者：*** *CTFHub*
- ***本文链接：*** *https://writeup.ctfhub.com/Challenge/2019/工业信息安全技能大赛/山东站/it2BmKggi3QFR2mdJgcD1M.html*
- ***版权声明：*** *本博客所有文章除特别声明外，均采用 BY-NC-SA 许可协议。转载请注明出处！*

 # Challenge # 2019 # 工业信息安全技能大赛 # 山东站
WeakRSA
特殊的工控流量