2025黄鹤杯Crypto 栅栏里的保险箱

by zijeff

这道题比较考验脑洞啊，有点大开脑洞的意思了。

首先，打开附件对我们有用的数据只有两个长长的整数。然后一看，第二个整数被一群n给围住了。

看到这里，我们不妨大胆猜测，这里有个**RSA**加密。被n包裹的数就是**RSA**加密过程中使用的模数，第一个数字就是加密后的密文。接下来的工作就很简单了，这个模数并不是很大，所以我们尝试直接分解找到p和q。

尝试后，我们发现在线分解无法得到结果，所以我们使用**yafu**工具在本地端分解（时间略微有点长），这次可以得到结果了。

```
p = 475983529392229563986300263627
q = 509948513243927226081750482929803134872729362481766202468213697366084 73
```

但是接下来的问题就是缺少**加密指数e**了，我们可以尝试爆破，爆破范围从3到常见的65537。具体代码如下：

```
for e in range(3, 65537):
    if e % 2 == 0:
        continue
    if GCD(e, phi_n) != 1:
        continue
    d = inverse(e, phi_n)
    m = pow(c, d, n)
```

利用e在RSA加密里的一些特殊性质来加快遍历速度，再结合题目中的**栅栏**，我们再次猜测**RSA**解出的明文应该是真正的flag过了一遍栅栏加密，利用在线工具即可求解。

所以判定代码如下：

```
    if b'{'in long_to_bytes(m) and b'}'in long_to_bytes(m) and b'f'in
long_to_bytes(m):
        if b'l'in long_to_bytes(m) and b'a'in long_to_bytes(m) and b'g'in
long_to_bytes(m):
            print(e)
            print(long_to_bytes(m))

# m = b'f513lb9f22}a1a161gfe1f9{74ac82cc27cf23'
```

经在线栅栏解密后得到flag：***flag{2fc7f1b59ae4c22a11f126fc73c89123}***