WP

1.签到

扫码关注公众号发送消息得到flag

2.ezRSA

分析题目，并没有直接给出p、q的值，但给出了leak1和leak2。解出题目即求解出私钥d。联想到使用扩展的欧几里得算法进行模反演来求解。

```
from Crypto.Util.number import *
leak1 =
14912717007361127196818257675129033155901844180572531042609541283758922767075754
07439298658536503998391028384315072007447249396594632001580124696769799876964190
50900842798225665861812331113632892438742724202916416060266581590169063867688299
28898573410412763223217565735269789838344132347745065817972772890866 9
leak2 =
11612299271467091538130991696749043648902000117288064416717991546702179489292797
72720805966417855691191342590375223883351980431522061502591034855745588164247402
04736215551933482583941959994625356581201054534529395781744338631021423703171146
45666343295584359854812259330878224522079201871650853849740257670946 1
e = 0x10001
c =
10529481867532520034258056773864074017027019578041866245400647840230251661652999
70971591962081093343719166118000329592327365567572958855889959252423562272881606
55019180761208122365803449911409809915323479912527052886330149134799706100568455
43523591324177567061948922552275235486615514913932125436543991642607028689762693
61730524671649278311681307035551260697162664559496185056758634038970582131484209
64656318868122812898431322581318097737977770493587891822125706062525097908309942
63132020094153646296793522975632191912463919898988349282284972919932761952603379
73323457535162403916244002194059255276857963997771309997 1
def extended_gcd(a, b):
    if b == 0:
        return (a, 1, 0)
    else:
        d, x, y = extended_gcd(b, a % b)
        return (d, y, x - (a // b) * y)
def mod_inverse(a, m):
    d, x, y = extended_gcd(a, m)
    if d != 1:
        raise ValueError("Inverse does not exist")
    else:
        return x % m
phi_n = (leak1 - 1) * (leak2 - 1)
d = mod_inverse(e, phi_n)
flag=long_to_bytes(pow(c, d, leak1 * leak2))
print(f"Private key d: {d}")
print(f"Decrypted message: {flag}")
```

得到flag后发现预期解法是费马小定理。具体来说，如果我们知道e、n、leak1和leak2，可以使用费马小定理来计算私钥d。首先，我们计算欧拉函数φ(n)，其中φ(n) = (p-1)(q-1)。然后，使用费马小定理的推论欧拉定理，得到以下同余关系：这就是说，e和d是在模φ(n)下互为模反演的。所以，我们只需找到满足上述同余关系的d即可。