CTF-CRYPTO-OTHER-pure math

原创

大熊何在 于 2020-12-30 00:16:23 发布

分类专栏： CRYPTO

本文链接：https://blog.csdn.net/zippo1234/article/details/111939734

CTF-CRYPTO-OTHER-pure math

俺又回来了。很遗憾世事难料，CTF以后只能作为兴趣爱好了。希望能做到经常更新

pure math

题目分析

开始

1.题目

```
p, q为大素数，且gcd(p, q)=1。已知：
1) p ** p % q = 714524049641435943105999741526097792730322384404239011969937133613136793624750369055376989216550533939260872429494033538186859407304595735358326864493864
2) q ** q % p = 379392682639932084968374768374319186018419611812741665239505534654558686213097067725307803930577708474995405008609852193360348844700856483242727427987395 1
3) (p ** q + q ** p) % (p*q) = 190241889284698572647433454966193059288115762232511539430185821226168269536810527958308825673826632620791272011410981099714652796730946963754178909427854 48
4) (p+q) ** (p+q) % (p*q) = 345840992053358741951386876510206537553414121124891277780426651020469892358848423261187013787246330653600568706288841616868471729582627943963914317131708502791414052633853924016950933627371254710461842492656267594575580791342417892781476599869061445241298624030100733062528664864385664309896306615565302731 59
5) FLAG ** 65537 % (p*q) = 707801499132468328267664068977069574910564021015585655036800020332219341864169651949835230482918629500936184565876338040364832546090227843161484088479684219659044970002928787049336838627486778843803427592334859662472016037223309112880523492191333154349234330094861697407085268639646975865618946614216423375 15

求FLAG
```

2.分析

（直接照抄WP了，说实在话没看懂）
我们的目的基本上就是求得 FLAG，那么怎么做呢?这个题目需要我们具有较好的数论功底。

根据题目中这样的内容，我们可以假设 p，q 都是大素数，且gcd(p, q)=1，那么

$$p \equiv 1 \bmod$$

那么

$$p \equiv p \bmod$$

那么我们可以根据 3）知道

$$p + q \equiv (p + \quad) \quad 1$$

而 $p+$ 又显然小于 pq，所以我们就知道 $p+$ 的数值。

进一步，我们假设1），2），3），4），5）对应的值分别为 x，x，x，x，x 则

根据4），我们可以知道

$$(p+q) \equiv (p \quad + \quad) \quad 1$$

又因为1）和 2），则

$$p \quad p \equiv px \bmod$$

$$q \quad q \equiv qx \bmod$$

因此

$$px + qx - \quad \bmod 1$$

根据 x 和 x 的求得方式，我们可以知道这里也是等号，因此我们得到了一个二元一次方程组，直接求解即可。

3.上脚本

```python
#!python3
# -*- coding: utf-8 -*-
# @Time : 2020/12/12 19:17
# @Author : A.James
# @FileName: tt.py
import gmpy2
#p ** p % q
x1 = 7145240496414359431059997415260977927303223844042390119699371336131367936247503690553769892165505339392608724294940335381868594073045957353583268644938 64
#q ** q % p
x2 = 3793926826399320849683747683743191860184196118127416652395055346545586862130970677253078039305777084749954050086098521933603488447008564832427274279873951
#(p ** q + q ** p) % (p*q)
p_q = 1902418892846985726474334549661930592881157622325115394301858212261682695368105279583088256738266326207912720114109810997146527967309469637541789094278544 8
#(p+q) ** (p+q) % (p*q)
x4 = 3458409920533587419513868765102065375534141211248912777804266510204698923588484232611870137872463306536005687062888416168684717295826279439639143171317085027914140526338539240169509336273712547104618424926562675945755807913424178927814765998690614452412986240301007330625286648643856643098963066155653027315 9
if (x4 - x1 * p_q) % (x2 - x1) == 0:
    print 'True'
q = (x4 - x1 * p_q) / (x2 - x1)
print q
p = p_q - q
#FLAG ** 31337 % (p*q)
c = 7078014991324683282676640689770695749105640210155856550368000203322193418641696519498352304829186295009361845658763380403648325460902278431614840884796842196590449700029287870493368386274867788438034275923348596624720160372233091128805234921913331543492343300948616974070852686396469758656189466142164233751 5
phin = (p - 1) * (q - 1)
d = gmpy2.invert (65537, phin)
flag = gmpy2.powmod(c, d, p * q)
flag = hex(flag)[2:]
print flag.decode('hex')
```

4. get flag

```
True
6816535611857119526525425522812910890357875802390460142336620200481846488148196550247533615596759336304754896675434478233803011042738249509973769227246001
flag{puRe_Math_pr06Lem}
```

结语

纯粹就是数学呗