Crypto 2020网鼎杯 you raise me up Writeup （离散对数）

原创

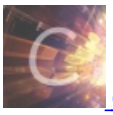
龙雪 于 2020-05-11 14:49:30 发布 1552 收藏 1

分类专栏： CTF 文章标签： 安全 经验分享

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/lostnerv/article/details/106052921

版权


CTF 专栏收录该内容

4 篇文章 0 订阅

订阅专栏

> 关键点：离散对数、同余运算
> 知识点：Antigonae整理的相关概念

看不懂-.-以后再研究，简单理解为：

- 普通对数为 a ** x=b （求$x=\log_a b$）
- 离散对数为 a ** x=b % p

先会用SageMath~

题目

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
from Crypto.Util.number import *
import random

n = 2 ** 512
m = random.randint(2, n-1) | 1
c = pow(m, bytes_to_long(flag), n)
print 'm = ' + str(m)
print 'c = ' + str(c)

# m = 391190709124527428959489662565274039318305952172936859403855079581402770986890308469084735451207885386318986881041563704825943945069343345307381099559075
# c = 666585139420321424585678945072365863252081679162179677590976689523300023402364287878602564495379799537321130848560539702412318008592411761080248597258449
```

解题

乍一看以为是RSA，但关系并不大，整理下：

```
已知 c,m,n, c=(m**d)%n
求  d

即 c 与 m ** d 对模n同余，或者说  (c-m**d) % n=0
```

写sage脚本：

```
d=discrete_log(c,mod(m,n))
print (d)
```

再就是flag了

```
print libnum.n2s(d)
```

以上~

SageMath相关

官网点我
学习SAGE(SAGEMATH)密码学基本使用方法

求离散对数

$$2^x \equiv 13 \mod 23$$

```
x=discrete_log(mod(13,23),mod(2,23))
#或discrete_log(13,mod(2,23))
print(x)
```

> 主要参考的文章：
> SAGE(SAGEMATH)密码学基本使用方法