CTF密码学(Crypto)方向入门指北

ver 1.2 *By 鲤唐可可*

什么是密码学(Cryptography)

~~我偷懒了，不如直接放个去年的指北吧（~~

> 与平时所说的“账号密码”不同，这里提到的密码学研究的是“加密”，而常说的“账号密码”对应的说法应该是“口令”（password）。一个明文经过密钥加密后，就会变为一段“让人看不懂”的密文，而经过密钥解密后，就会变回有意义的明文。然而，有些人没有密钥也想获取密文中的部分或者全部内容，甚至是直接利用密文去达到自己的要求，这就是CTFer在比赛中需要达成的目的。在比赛中，通常会把加密算法公开出来，然后再将密文等信息发送给参赛者，参赛者需要分析加密算法的缺点，以找到利用密文的方法。在密码学中还有许多有意思的问题，例如，如何在没有事先商定好密钥的情况下，在一个公开的网络中完成信息的秘密传输？如何确认这个信息是由朋友本人传输的而不是他人伪造？是否存在理论上不可攻破的密码体系……诸多问题，将在以后的学习中得到解答。

Crypto方向所需要的知识

编程方面

1. 一点点python基础
    1. 会基本的语法即可
        1. 真的只要一些基本的知识
        2. 就算不会也可以边写边学 ~~（事实上这是我最推荐的python学习方法）~~
    2. 会一些基本的python库
        1. pycryptodome（密码学常用库，提供了一些基本的数学工具和包括AES RSA之类的密码库）
        2. gmpy2（一个高精度的算法库）
        3. numpy（一个强大的python数学库）
    3. 小部分题目可能会有交互，可能要写一些脚本
        1. 常见的交互用的是socket,可以用pwntools来写脚本
        2. 有小部分可能会用到http，可以用requests库来写脚本 ~~（具体可以参考web方向的指北~~ ~~（主打的就是一个方向交叉~~

数学方面

可以参考oi-wiki

入门是不需要太多的，但是如果想要真正成为队里的crypto扛把子，你将会接触到包括但不限于以下数学

1. ~~初等数论~~
2. ~~线性代数~~
3. ~~抽象代数~~
4. ~~离散数学~~
5. ~~...（前面的区域之后再来探索吧~~

但是但是但是，千万不要被数学给劝退 但是但是但是，千万不要被数学给劝退 但是但是但是，千万不要被数学给劝退

包括写这个指北的人(鲤唐可可 ver2023.06.16)，现在的数学知识也极度贫瘠，千万不要想着学完再来做crypto，打ctf和做学术是不一样的，ctf更需要的是你**在面对具体题目的时候，即时学习并且运用的能力**，在做题的过程中学习，也是一种学习的好方法

算法知识

但是还是要指出，算法知识虽然重要，但并不是没有算法基础就做不了crypto了，更重要的是理解一些算法背后的思想，比如：

1. 时间复杂度 ~~（e.g. 有的时候需要小小的爆破一下，估算时间复杂度就有必要了）~~
2. 二分思想
3. 优化技巧（如快速幂等）（这些算法在上面提到的算法库里面大多都有实现，所以不一定要背下这些算法的具体实现，以了解为主）
4. bsgs（BABY-STEP GIANT-STEP）
5. ...（可以参考oi-wiki）

可能需要用到的软件&工具

1. python ~~（废话，不然手算吗（~~
    1. 建议使用3.10或者3.11，这两个版本不算太新，出问题容易查找解决方案，并且执行速度相对较快
2. 一个写python的ide ~~（这位同学，你也不想用记事本写代码吧~~
    1. 推荐vscode（
    2. pycharm也行
    3. 真别用记事本，折磨自己（
    4. ~~试试vim（？~~
3. Sagemath
    1. 这是一个基于python的强大的数学工具，并且可以使用python的语法，上手较快
    2. 缺点是好像现在没有适用于windows的版本，对于使用windows的同学，你可能需要（选一个）
        1. 在win10或者win11上启用WSL（Windows Subsystem for Linux）
        2. 装一个Linux的Virtual Machine（VMware或者VirtualBox，~~你要用Hyper-V的话，做好觉悟吧~~）
        3. Docker for Windows（但其实还是基于WSL2）
        4. 直接用sagemath的在线服务（很慢，不推荐
    3. ~~sage实在用不了的话，万一matlab和mathematica可以呢（我没试过，随口说的~~

比较推荐的学习方法

1. 学ctf前先学会如何使用搜索引擎
    1. 推荐使用的搜索引擎
        1. cn.bing.com（国内版和国际版都还行）
        2. www.bing.com（没错，这玩意和上面那个是不太一样的）
        3. www.google.com（别问我为什么timeout，我只能说dddd）
    2. 搜索引擎的使用是一门学问，要学会如何使用正确的关键词，以表达你想要的搜索结果，也可以使用一些小tricks

        1. Refine web searches
        2. How to Google like a Pro
2. 看书 ~~or paper~~
    1. 包括但不限于上文中提到的关于数学方面的书
    2. 也有一些关于密码学的书，也可以看看
    3. 偶尔读一些论文也有帮助你的学习
    4. ~~但是要做好心理准备：很多书和论文，是**全英文**的~~
3. 看相关wiki（~~不放链接了，找到下列网站就当作第一part的课后练习吧~~）
    1. ctf-wiki（主要是crypto分区）
    2. oi-wiki（主要是math分区）
    3. 实在找不到相关词条就去wikipedia吧（~~别问我怎么打不开，我不知道~~
4. 找dalao的blog或者writeup狠狠的学习 ~~（排名不分先后，都是L team的神~~ 怎么都不更新，都是鸽子（？
    1. dbt的blog
    2. shallow的blog

遇到困难不要慌

1. 先根据上面所列出的学习方法，进行搜索（可以找相关知识点，也可以找同类题）
2. 为什么不试试神奇的gpt和new bing呢 ~~（虽然在crypto大概率没什么用就是了~~
3. 在moectf和一些其他的新生赛中，出题人一般是乐意回答初学者关于题目的问题的，所以不妨去找找出题人，获取一点提示
    1. ~~但是在别的大比赛中，要提示或者flag还是算了~~
    2. 课后作业part2：找到《提问的智慧》，并**仔细阅读**，适当摘抄笔记（bushi

Finally

你已经学会crypto的学习方法了，接下来就试试moectf的crypto方向吧（

~~很简单的，做不出来欢迎来锤出题人（~~

Example

例题：ezRSA

```
from secret import flag # 这个secret不是一个库，flag是你要自己求出来的，所以不要问为什么

p,q = getPrime(512), getPrime(512) # 随机生成了两个质数
e = 65537 # 这是什么？
n = p*q
m = int.from_bytes(flag.encode(),"big") # 把flag编码成bytes，再转换成一个长整数
c = pow(m,e,n) # RSA加密！

print(p)
print(q)
print(c)
```

```
# p = 12159513633827954344017322980901502929962132417834495872720768386674355
# q = 91319784256573640966155404455643523892240984725763453191351029920254959
# c = 45004723652943919619793244265650220807087540804571893073483013489011463
```

标准答案

```
from Crypto.Util.number import * # 一个非常好用的crypto库

p = 12159513633827954344017322980901502929962132417834495872720768386674355634
q = 91319784256573640966155404455643523892240984725763453191351029920254959830
c = 45004723652943919619793244265650220807087540804571893073483013489011463394
e = 65537
n = p*q
phi = (p-1) * (q-1) # 你知道什么是 欧拉函数吗 [1]
d = pow(e, -1, phi) # 什么是乘法逆元？ [2]
m = pow(c,d,n)
print(long_to_bytes(m))
```

1. 欧拉函数（Euler's totient function）
2. 模意义下乘法运算的逆元（Modular Multiplicative Inverse）