公钥密码及其应用

191850128 数学系 陆青阳

谈到密码，相信大多数人首先想到的都是登陆账号时输入的一串字符。然而，不知你有没有想过，密码真的只有你和厂商两方知道吗？如果有人窃听了你的网络通信，是否可以轻易获取你的密码呢？事实上，如果你尝试使用 telnet 传输协议访问网络，并用 Wireshark 软件获取数据包，你就会发现此时你的密码是一位一位不加修改地向外传输的。毫无疑问，这样访问网络是十分危险的。而正是密码学的发展，我们传输类似的数据时才能够得到保护，而我们平时输入的密码，不过是被加密的内容中的一小部分罢了。

从对称加密到非对称加密

密码的应用已经有了长达数千年的历史，其中最著名的莫过于古罗马帝国皇帝恺撒所发明的凯撒密码了。在使用恺撒密码时，加密者把所有的字母向后移动三位，例如 A 变为 D，B 变为 E，Z 变为 C 等等，例如 ”This is a book” 经加密后就变为 ”Wklv lv d errn”，而解密者在收到密文后只要将所有字母向前移动三位即可得到明文。对称加密，指的就是这种加密和解密使用同一密钥的算法，例如凯撒密码的密钥就是字母移动的位数。通过分组加密，多次迭代加密等技术，对称加密的强度得以在恺撒密码的基础上大大增强，凭借其计算量相对较小，速度较快的优势，对称加密目前仍然被广泛使用。

然而，对称加密有一个致命的缺陷，那就是加解密双方必须持有相同的密钥。换言之，如果两个陌生人想进行加密通信，他们必须先在没有保护的情况下商定一个密钥，而如果在这一过程中密钥被他人窃取，之后的通信就将不再安全。正是由于这个问题始终得不到解决，公钥密码应运而生。

公钥密码的原理

公钥密码使用非对称加密技术，也即加密和解密时使用不同的密钥，且从其中一个很难推出另一个。与对称密码所采用的“不破解密钥就不可能破译密码”的思路不同，公钥密码采用另一种保证安全的方式——不会有人愿意花费比加密信息的价值更高的代价来破解密码。因此，非对称加密算法一般都以难以求解的数学问题为基础，例如 RSA 密码求解大整数的质因数分解问题，离散对数密码求解离散对数问题等等。要生成这类问题并不困难，然而在没有密钥的情况下可能需要数年乃至数百年才能破解，因此便达到了保密的效果。

下面以 RSA 密码为例分析公钥密码的实现流程。

RSA 加密算法

- 消息接收方 A 生成公钥-私钥密钥对：

  任取两个大质数p,q,计算$\mathrm{N}=\mathrm{pq}$,计算欧拉函数$\varphi(\mathrm{N})=(\mathrm{p}-1)(\mathrm{q}-1)$,销毁$\mathrm{p},\mathrm{q}$

  任取$1<\mathrm{E}<\varphi(\mathrm{N}),(\varphi(\mathrm{N}),\mathrm{E})=1$

  由 Bezout 定理,$\exists \mathrm{x},\mathrm{y}\in \mathrm{Z},\mathrm{x}\varphi(\mathrm{N})+\mathrm{yE}=1$,取$\mathrm{D}=\mathrm{y}$,则$\mathrm{DE}\equiv 1(\mathrm{mod}\ \varphi(\mathrm{N}))$

  在 D,E 中任取一个(例如 D)作为私钥保存,另一个(例如 E)与 N 共同作为公钥对外广播

- 消息发送方 B 对消息$\mathrm{t}<\mathrm{N}$进行加密（如果消息过长则分段加密）

  密文$\mathrm{s}=\mathrm{t}^{\mathrm{E}}\ mod\ N$

- B 将 s 发送给 A

- A 对 s 进行解密

  明文$\mathrm{t}'=\mathrm{s}^{\mathrm{D}}\ mod\ N=(t^{E}\ mod\ N)^{D}\ mod\ N=t^{ED}\ mod\ N=t^{k\varphi(N)+1} mod\ N=t$

  这样便完成了一次加密通信

对于攻击者而言，由于 D 始终在 A 自己手中，不存在泄露的可能，因此只能用s, E, N进行破译。如果尝试通过 A 的方法获得明文，必须要知道 D，而 D 由$\varphi(N)$确定，$\varphi(N)$又由 p 和 q 确定，因此破译者必须将 N 分解为两个素数的乘积。在实际应用中，p, q一般选用超过 1024bit，约 300 十进制位的大素数，在这样的数量级上分解 N 对于目前的计算机而言仍然是几乎不可能完成的。当然，攻击者也可以尝试通过 B 加密的逆过程得到明文，即求解方程$s = t^E \ mod \ N$，而这是离散对数问题，在 N 很大的时候也没有得到解决。因此，目前而言 RSA 算法可以被认为是安全的。

公钥密码的发展(以 RSA 为例)

公钥密码并非十全十美。开创了公钥密码学时代的 RSA 算法问世至今四十余年来，其缺陷大都得到较好的解决，从而可以应用于更广泛的场景中。

RSA 最大的问题是计算速度过慢。即使是在知道密钥的情况下，要计算几百位数的乘方、取模运算也绝非易事，尤其是要加密大量文本时，往往要耗费大量时间。将 RSA 加密和对称加密结合起来就可以很好地解决问题——使用 RSA 加密对称加密过程的密钥，再用密钥加密文本，这样一来，既保证了加密的速度，也解决了对称密码需要共享相同密钥的困难。

RSA 的另一个问题是所有人都可以向消息接收方 A 发送密钥，而 A 很难确认对方的身份，也即 C 也可以像 B 一样向 A 发送信息，并谎称自己是 B。根据 RSA 的密钥生成过程，我们不难看出，D 和 E 的地位其实是完全相同的，用 E 加密的消息可以用 D 解密，用 D 加密的消息也可以用 E 解密。这样一来，在通信协议中要求发送方首先用自己私钥加密即可识别发送方的真伪。当 B 向 A 发送信息时，先用自己的私钥加密，再用 A 的公钥加密，A 只需用自己的私钥和 B 的公钥依次解密，仍然能收到正确的消息，但 C 由于不知道 B 的私钥，便无法进行类

似的攻击了。

上面的方案看似完美，但仍然存在“第一次”的问题。当 B 想要向陌生人 A 发送消息，他如何确定在网络中扫描到的 A 的公钥真的属于 A 呢？如果 C 把自己的公钥冒充成 A 的在网络中发布，那么 B 发送给 A 的消息将被 C 全部获取，而真正的 A 却反而无法解密。为了解决这一问题，同时考虑到生成密钥过程较为困难，便出现了专门的第三方机构 CA 中心，对网络账号进行认证并发放与之一一对应的公钥-私钥对，形成数字证书，从而杜绝了上述冒充现象的发生。

CA 中心虽然一般由政府部门控制，但也不可避免地存在被攻击导致密钥泄露的可能性。因此，近年来诸如比特币一类采用区块链技术的新型数字系统在公钥密码的基础上采用了分布式网络结构，所有节点自由接入网络，不受任何限制。通过将所有交易记录记录在所有节点组成的区块链上，使得交易信息无法被个人伪造或篡改，同样成功地达到了避免冒充的作用。

公钥密码的应用

由于公钥密码具备较强的保密属性，因而可以在零知识证明过程中防止信息的泄露。

问题 1 （美女过河）A,B,C 各有一个数 a,b,c，在保密的情况下求出总和

解：A 取随机数 d，用 B 的公钥加密 a+d 发送给 B

B 用自己的私钥解密，用 C 的公钥加密 a+b+d 发送给 C

C 用自己的私钥解密，用 A 的公钥加密 a+b+c+d 发送给 A

A 用自己的私钥解密，得到 a+b+c

由于 B 和 C 都不知道 d 的值，因此无法从收到的数字中得到任何信息，而 A 得到的仅仅是 b+c 的和，也无法知道具体的值，因此达到了保密的效果。

问题 2 （核对投诉者）A,B 各有一个 1-n 之间的整数 a,b，在保密的情况下判断 a,b 是否相同

解：A 先将 1-n 之间所有的整数和任意 n 个不同的大整数$\mathrm{p}_1, p_2, \ldots, p_n$建立一一映射，并保密

然后 A 将p_a用 B 的公钥加密，将$\mathrm{p}_1, \ldots, p_{a-1}, p_{a+1}, \ldots, p_n$用自己的公钥加密，并全部发给 B

B 收到消息后，用自己的私钥解密所有的数，然后将第 b 个的解密结果发送给 A

A 核对收到的结果是否是p_a，是则相等，不是则不相等。

如果 a,b 相等，p_a先经过 B 的公钥加密，又经过 B 的私钥解密，A 应该收到p_a，如果 a,b 不相等，p_a先经过 A 的公钥加密，又经过 B 的私钥解密，仍然变为p_a的概率几乎可以忽略不计，因此这一过程确实可以对两个数是否相等进行有效的判断。

当 B 按照$s^{\mathrm{D}}\ mod\ N$解密时，即使密钥不配对也能得到一个正整数，而由于 A 加密的信息本来就是随机的大整数，因此 B 无法知道自己解密的结果到底是解密错误而产生的数还是恰好就是 A 选定的数，也即 B 在这一过程中无法得知关于 a 的任何信息。

当 A 收到 B 的解密结果时，由于 A 不知道 B 的私钥，因此 A 无法模拟 B 的解密过程，从而也就不知道 B 究竟是通过解密哪一个数得到的这一结果，因此除非相等，A 也无法得知关于 b 的任何信息。因此这一算法能够满足保密的要求。

当然，A 可能在得到结果后对 B 进行欺骗，只需要交换双方角色，在同一时间进行上述操作即可解决这一问题。

基于难以求解的数学问题，公钥密码学为我们保护信息安全提供了全新的思路。如今，随着计算机性能的逐步提高，诸如 RSA-512 之类的公钥加密算法已不再安全。而量子计算机的发展也使得大数分解等数学难题可能在短时间内得以解决，公钥密码到时候恐怕也将寿终正寝。尽管如此，这一划时代的发明必将永远在密码学的历史长河中熠熠生辉。