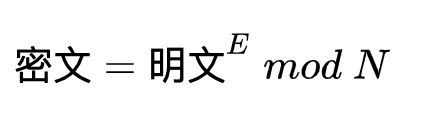
什么是RSA？

非对称加密算法也叫公钥密码算法，通过生成的公私钥来对明文密文进行加密解密。

RSA的加密

RSA的加密可以用下面的公式来表示：



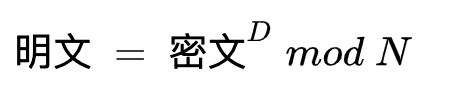
通过公式我们可以知道RSA的密文是通过明文的E次方再对N进行mod运算得到的。这个加密过程只用到了阶乘和取模运算，可以算是非常简单明了了。

简洁的才是最好的，这可能也是RSA算法这么通用的原因吧。

如果知道了E和N，那么就可以得到密文，所以我们把E和N的组合称为公钥，可以这样表示 公钥{E,N}。

RSA的解密

先看一下RSA解密的公式：



通过公式可以看到，明文是通过密文的D次方，再和N取模得到的。这里的N和加密的N是同一个数字。

D和N的组合表示为私钥{D,N}。

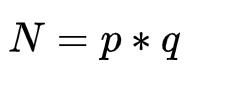
N,E,D的生成

知道了RSA的加密和解密原理之后，接下来我们就要探讨一下加密和解密过程中的N,E,D是怎么生成的。

生成过程如下：

1. 生成N

生成N的公式如下：

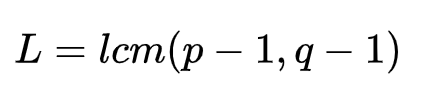


p和q是两个很大的质数，太小的话容易被破译，太大的话会影响计算速度。通常p和q的大小为1024比特。这两个数是通过伪随机数生成器生成的。伪随机数生成器不能直接生成质数，它是通过不断的重试得到的。

2. 求L

L是一个中间数，它和p，q一样，不会出现在RSA的加密和解密过程。

L的计算公式如下：



L是p-1和q-1的最小公倍数

3. 求E

E就是用来加密的公钥了，E是一个比1大，比L小的数。并且E和L必须互质。只有E和L互质才能计算出D值。

1< E < L

gcd(E, L) = 1

这里E也是通过伪随机数生成器来生成的。

找到了E和N，我们的公钥就生成了。

4. 求D

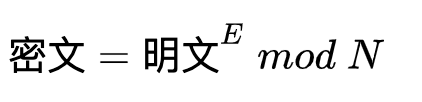
计算D的公式如下：

1<D<E

E\*D mod L = 1

破解RSA

如果想破解RSA， 对于密码破解者来说，他知道了公钥{E,N}, 知道了密文，根据公式：



有没有可能直接通过已知的三个变量，求出未知变量明文呢？

这个求解其实是一个离散对数的问题。目前还没有发现求离散对数的高效的方法。可以说是非常困难的。

那么有没有可能通够暴力破解来得出密钥中的D呢？

目前RSA算法中p和q的长度一般为1024比特以上，生成的N的长度为2048比特以上，E和D的长度和N差不多，如果要暴力破解2048比特的D是非常困难的。

由公式：



可知，如果破解者知道了L的值，那么就可以轻易的求出D。而L是通过p和q计算出来的，所以p和q一定要保密，否则跟密码泄露是一样的。

因为 N= p \* q ， 而p和q都是质数， N又是已知的，那么我们可不可以通过质因数分解来得到 p和q呢？

目前来说，还没有有效的对大整数进行质因素分解的高效算法，所以目前来说RSA算法还是很安全的，但是一旦有这样的算法出现，那么RSA将会很容易被攻破。

所以官方推荐：1024比特的RSA算法不应该被用于新的用途。2048比特的RSA算法可以用到2030年，4096比特的算法可以用到2031年。