RSA是目前使用最为广泛的公钥加密算法，它的加解密形式简单，理论基础也比椭圆加密算法更易理解，这些或许是RSA流行的原因之一。曾经看到过一个比较夸张的说法描述RSA的广泛性：哪里有计算机网络，哪里就有RSA。用阮一峰的话来说，RSA是计算机通信安全的基石。

下面描述RSA的加密和解密式子：

加密：Me mod n = C (1)

解密：Cd mod n = M (2)

其中，M为明文，C为密文，e是公钥，d是私钥

将(1)式代入(2)式，可以得到Cd mod n = (Me mod n)d mod n = Med mod n = M，从而有Cd mod n = M。

要想加解密操作正确，从(1)和(2)分析可以看出，关键是要使Med mod n = M成立。那在e、d和n满足什么条件下，可以使Med mod n = M成立呢？为了找到答案，我们先引入欧拉函数（Euler’s totient function）和欧拉定理（Euler's Theorem）。

欧拉函数：φ(n), 统计从1~n中，与n构成互质关系的数的个数。

若两个数的公约数有且只有1，那么两个数就是互质的。

比如φ(8)=4，因为，1到8中与8构成互质的有：1，3，5，7.一共有4个。

关于欧拉函数，有几个重要的结论：

1. 若p是质数，则φ(p)=p-1.
2. 若p,q都为质数，n=p\*q,则φ(n)=φ(p\*q)=φ(p)\*φ(q)=(p-1)\*(q-1).

欧拉函数其实还可以用一个等式表示，在这里就不在扩展了。

欧拉定理：aφ(n) ≡ 1(mod n) (3)

其中，a,n为[正整数](https://baike.baidu.com/item/%E6%AD%A3%E6%95%B4%E6%95%B0" \t "https://baike.baidu.com/item/%E6%AC%A7%E6%8B%89%E5%AE%9A%E7%90%86/_blank)，且a,n[互质](https://baike.baidu.com/item/%E4%BA%92%E8%B4%A8" \t "https://baike.baidu.com/item/%E6%AC%A7%E6%8B%89%E5%AE%9A%E7%90%86/_blank)。

其实aφ(n) ≡ 1(mod n)是aφ(n) mod n = 1的简写。

关于欧拉定理，也有个重要的推论：

1. aφ(n)+1 ≡ a(mod n),其中，a与n不需要构成互质关系，这点与(3)式有点不同。

从欧拉定理的推论中可以看出，它的形式与Med mod n = M相同。于是，就有下面的推导过程：

ed =φ(n)+1

ed modφ(n) = (φ(n)+1) modφ(n)

ed modφ(n) = 1 (4)

于是，答案已经找到了，在e、d和n满足(4)式时，可以使Med mod n = M成立。

因此，使用RSA算法必须要找到其中一组e、d和n，满足ed modφ(n) = 1成立。

下面描述RSA是如何找到一组e、d和n的。

1. 选两个质数p，q
2. 令n=p\*q
3. 取e: e满足gcd(φ(n),e)=1,换句话说，e与φ(n)是互质的
4. 求d: 当确定了e和n，通过(4)式求出d，求解方法可用扩展欧几里得算法(extended Euclid’s algorithm)。

举例：

1. 选两个质数p = 13,q=19
2. n=p\*q=13\*19=247，φ(n)=φ(p)\*φ(q)=(p-1)\*(q-1)=12\*18=216
3. 取e: e与φ(n)要互质，且e<φ(n),则可以选e=7
4. 求d, 根据公式：ed modφ(n) = 1, 则7\*d mod 216 = 1, d = 31.(因为有7\*31 = 216 +1)

整理上述参数：p=13,q=19,n=247,e=7,d=31. 并且令明文M=34

则加密运算：

347mod247 = (34mod247)\*(342mod247)\*(344mod247)= ((34\*168\*66)mod247) = 70

解密运算：

7031mod247 = ((70mod247)\*(702mod247)\*(704mod247)\*(708mod247)\*(7016mod247))mod247 = (70 \* 207 \* 118 \* 92 \* 66)mod247 = 34

为了方便，将公钥取为{e,n},私钥为{d,n}.

上面是RSA主要加解密流程，接下了，你会问，RSA安全吗，因为公钥{e,n}是公开的，那已知公钥，能否反推出d ？第一种情况，如果告诉你φ(n)的值，根据ed modφ(n) = 1，可以求出私钥d。第二种情况，如果告诉你p和q的值，根据n=p\*q, φ(n)=φ(p)\*φ(q),可以求出φ(n),也就可以求出私钥d。第三种情况，如果告诉你n,可以求得私钥d吗？关键还是要得到φ(n),φ(n)=φ(p)\*φ(q),于是p和q是关键，那如果可以把n分解成两个质数p和q之积，就可以求出d；或者暴力统计求出n的欧拉函数φ(n)，之后也可以求出d。前者方法较后者容易一些，但难度还是很大，n越大(常见的有1024bit、2048bit)，求解越困难。大整数的质因子分解已是世界性难题，至今没有快速有效的方法解决它。所以，RSA的安全性就是基于这个问题求解的困难性。