13 RSA Encryption

The mathematics behind the very successful RSA encryption method is very simple and uses mainly Euler's congruence for any $N \geq 1$:

$$b^{\varphi(N)} \equiv 1 \pmod{N}$$

if $(b, N) = 1$. (When N is a prime, this is Fermat's little theorem.)

Imagine that a person X wants to send a carefully encrypted message to another person Y, say. X will look in a directory which publishes the *public key* of various people including Y. The public key of Y will be a pair (e, N) of positive integers, where N will be a large number which is a product of 2 distinct primes p and q. The point is that the directory will contain no information on the factorization of N. For large enough N it will become impossible (virtually) to factor N. The number e will be chosen mod N and it will be prime to $\varphi(N)$.

The person X will first represent his/her *plain text* message by a numeral a (which can be done in many ways). For simplicity, suppose that a is prime to N. X will then raise a to the power e mod N and send the message as b. So

$$b \equiv a^e \pmod{N}.$$

If someone intercepts the message, he or she will be unable to recover a from b without knowing the factorization of N. So it is secure. On the other hand, the recipient of the message, namely Y, will be able to decode (decrypt) the message as follows. He/she will pick a number d (*decryption constant*) such that

$$de \equiv 1 \pmod{(p-1)(q-1)}.$$

Y can do this because he/she knows the prime factors p, q and because e is prime to $\varphi(N)$; observe that since p and q are distinct primes and $N = pq$, one has

$$\varphi(N) = \varphi(p)\varphi(q) = (p-1)(q-1).$$

So by applying Euler's congruence mod N, we get

$$b^d \equiv a^{ed} \equiv a^{1+c(p-1)(q-1)} \equiv a \pmod{N}.$$

Thus Y recovers a.

Note that if someone does not have the factorizatino of N, he/she cannot decrypt the message.