Basic idea: prime multiplication is very easy, integer factorization seems to be unfeasible.

* It is a public encryption method that relies on a public encryption algorithm, a public decryption algorithm, and a public encryption key.
* Using the public key and encryption algorithm, everyone can encrypt a message.
* The decryption key is known only to authorized parties.
* p and q are two prime numbers.
* n = pq
* m = (p-1)(q-1)
* a is such that 1 < a < m and gcd(m,a) = 1.
* b is such that (ab) mod m = 1.
* a is computed by generating random positive integers and testing gcd(m,a) = 1 using the extended Euclid’s gcd algorithm.
* The extended Euclid’s gcd algorithm also computes b when gcd(m,a) = 1.

**RSA Encryption And Decryption**

* Message M < n.
* Encryption key = (a,n).
* Decryption key = (b,n).
* Encrypt => E = Ma mod n.
* Decrypt => M = Eb mod n.

**Breaking RSA**

* Factor n and determine p and q, n = pq.
* Now determine m = (p-1)(q-1).
* Now use Euclid’s extended gcd algorithm to compute gcd(m,a). b is obtained as a byproduct.
* The decryption key (b,n) has been determined!

**Security Of RSA**

* Relies on the fact that prime factorization is computationally very hard.
* Let k be the number of bits in the binary representation of n.
* No algorithm, polynomial in k, is known to find the prime factors of n.
* Try to find the factors of a 100 bit number.