

An overview of classical encryption:

There is a key function denoted $K()$ and for any plain message M and encrypted message E $K(M)=E$ and $K(E)=M$
assumptions:

Only people who should read the message M can get $K()$

problems:

In information networks like the internet, messages must be passed through untrusted networks before they reach their final destination, thus anyone in the middle can see $K()$

Public key encryption

In public key encryption schemes have two key functions, a public key denoted as $P()$ and a private key function denoted as $R()$. The properties of $P()$ and $R()$ are $P(M)=E$ and $R(E)=M$. With this scheme you can send your public key to anyone, and they will be able to send you messages that only you can decrypt (with your private key).

advantages:

You no longer need to worry about sending your key through an untrusted medium.

assumptions:

You must have a fast way to generate $R()$ and $P()$ for convenience.

No one will ever see your private key $R()$.

It must be impossible for someone with $P()$ to derive $R()$

problems:

Assumption three is so far a mathematical impossibility, but the job of the cryptanalyst is to make this as hard to do as possible (this itself is a VERY hard problem).

If any of the three assumptions are not met, the cryptosystem is worthless.

The RSA cryptosystem

The only public key cryptosystem in common use today.

Based on the fact that multiplication is done in linear time, while factorization is very hard

General generation method:

- 1 First, generate two primes A, B .
- 2 Then calculate $(A-1)(B-1)=C$.
- 3 Now you must find two numbers x and y such that $\gcd(x,y)=1$ and $x*y \bmod C=1$.
- 4 $A*B=N$
- 5 forget about all variables except for N , x , and y

Specific example:

- For this example we pick 7 and 11.
 $(7-1)(11-1)=60$
 $x=7$ and $y=43$ $\gcd(7,43)=1$ and
 $7*43=301$ and $301 \bmod 60=1$
77

We can now define $R()$ as $M^x \bmod N = E$ and $P()$ as $E^y \bmod N = M$

You can verify that $M^7 \bmod 77 = E$ and $E^{43} \bmod 77 = M$ for any M and E from 0-76

The process can be reversed by factoring N , but as you increase the sizes of A and B , the rate of key generation goes up linearly, while the time needed for factorization goes up at an exponential rate. The use of an algorithm that can factor numbers in linear time (such as Peter Shor's quantum factorization algorithm) will eliminate any cryptographic advantage that RSA provides.