RSA Algorithm

RSA algorithm is asymmetric cryptography algorithm. Asymmetric actually means that it works on two different keys i.e. Public Key and Private Key. As the name describes that the Public Key is given to everyone and Private key is kept private.

An example of asymmetric cryptography :

A client (for example browser) sends its public key to the server and requests for some data.

The server encrypts the data using client's public key and sends the encrypted data.

Client receives this data and decrypts it.

Since this is asymmetric, nobody else except browser can decrypt the data even if a third party has public key of browser.

The idea! The idea of RSA is based on the fact that it is difficult to factorize a large integer. The public key consists of two numbers where one number is multiplication of two large prime numbers. And private key is also derived from the same two prime numbers. So if somebody can factorize the large number, the private key is compromised. Therefore encryption strength totally lies on the key size and if we double or triple the key size, the strength of encryption increases exponentially. RSA keys can be typically 1024 or 2048 bits long, but experts believe that 1024 bit keys could be broken in the near future. But till now it seems to be an infeasible task.

Let us learn the mechanism behind RSA algorithm :

>> Generating Public Key :

Select two prime no's. Suppose P = 53 and Q = 59.

Now First part of the Public key : n = P*Q = 3127.

We also need a small exponent say e :

But e Must be :

- An integer.
- Not be a factor of n.
- 1 < e < Φ(n) [Φ(n) is discussed below],

Let us now consider it to be equal to 3.

Our Public Key is made of n and e

>> Generating Private Key :

We need to calculate Φ(n) :

Such that Φ(n) = (P-1)(Q-1)

so, Φ(n) = 3016

Now calculate Private Key, d :

d = (k*Φ(n) + 1) / e for some integer k

For k = 2, value of d is 2011.

Now we are ready with our – Public Key (n = 3127 and e = 3) and Private Key(d = 2011)

Now we will encrypt “HI” :

Convert letters to numbers : H = 8 and I = 9

Thus Encrypted Data c = 89e mod n.

Thus our Encrypted Data comes out to be 1394

Now we will decrypt 1394 :

Decrypted Data = cd mod n.

Thus our Encrypted Data comes out to be 89

Aska – Finding The Article
Harry – Finding article and Finishing gitpages