RSA ALGORITHM - AN INTRODUCTION

Rivert, Shamir, and Aldeman developed the RSA public-key encryption and signature scheme. This was the first practical public-key encryption algorithm. RSA is based on the intractability of factoring large integers. Assume that a plaintext m must be encrypted to a ciphertext c. The RSA algorithm has three phases for this: key generation, encryption, and decryption.

Key Generation

In the RSA scheme, the key length is typically 512 bits, which requires an enormous computational power. A plaintext is encrypted in blocks, with each block having a binary value less than some number n. Encryption and decryption are done as follows, beginning with the generation of a public key and a private key.

Begin Key Generation Algorithm

1. Choose two roughly 256-bit prime numbers, a and b, and derive n = ab. (A number is prime if it has factors of 1 and itself.)
2. Find x. Select encryption key x such that x and (a - 1)(b - 1) are relatively prime. (Two numbers are relatively prime if they have no common factor greater than 1.)
3. Find y. Calculate decryption key y:

   Equation 10.5

   $$xy \bmod (a-1)(b-1) = 1.$$

4. At this point, a and b can be discarded.

5. The public key = {x, n}.
6. The private key = {y, n}.

In this algorithm, x and n are known to both sender and receiver, but only the receiver must know y. Also, a and b must be large and about the same size and both greater than 1,024 bits. The larger these two values, the more secure the encryption.

Encryption

Both sender and receiver must know the value of n. The sender knows the value of x, and only the receiver knows the value of y. Thus, this is a public-key encryption, with the public key {x, n} and the private key {y, n}. Given m<n, ciphertext c is constructed by

Equation 5.6

$$c = m^x \bmod n.$$

Note here that if a and b are chosen to be on the order of 1,024 bits, n $\approx$2, 048. Thus, we are not able to encrypt a message longer than 256 characters.

Decryption

Given the ciphertext, c, the plaintext, m, is extracted by

Equation 5.7

$$m = c^y \bmod n.$$

In reality, the calculations require a math library, as numbers are typically huge. One can see easily how Equations (5.6) and (5.7) work.

Example. For an RSA encryption of a 4-bit message of 1,000, or m = 9, we choose a = 3 and b = 11. Find the public and the private keys for this security action, and show the ciphertext.

Solution. Clearly, n = ab = 33. We select x = 3, which is relatively prime to (a - 1)(b - 1) = 20. Then, from xy mod (a - 1)(b - 1) = 3y mod 20 = 1, we can get y = 7. Consequently, the public key and the private key should be {3, 33} and {7, 33}, respectively. If we encrypt the message, we get $c = m^x \bmod n = 9^3 \bmod 33 = 3$. The decryption process is the reverse of this action, as $m = c^y \bmod n = 3^7 \bmod 33 = 9$.

Source : http://elearningatria.files.wordpress.com/2013/10/cse-vi-computer-networks-ii-10cs64-notes.pdf