The RSA Algorithm

The RSA scheme is a block cipher in which the plaintext and ciphertext are integers between 0 and *n*- 1 for some *n*. A typical size for *n* is 1024 bits, or 309 decimal digits. That is, *n* is less than 2^{1024}. We

examine RSA in this section in some detail, beginning with an explanation of the algorithm. Then we examine some of the computational and cryptanalytical implications of RSA.

Description of the Algorithm

The scheme developed by Rivest, Shamir, and Adleman makes use of an expression with exponentials.

Plaintext is encrypted in blocks, with each block having a binary value less than some number *n*. That is, the block size must be less than or equal to log2(*n*); in practice, the block size is *i* bits, where 2*i* < *n* <2*i*+1. Encryption and decryption are of the following form, for some plaintext block *M* and ciphertext block *C*:

$C = M^e \bmod n$

$M = C^d \bmod n = (M^e)^d \bmod n = M^{ed} \bmod n$

Both sender and receiver must know the value of *n*. The sender knows the value of *e*, and only the receiver knows the value of *d*. Thus, this is a public-key encryption algorithm with a public key of *PU* = {*e*, *n*} and a private key of *PR* = {*d, n*}. For this algorithm to be satisfactory for public-key encryption,

the following requirements must be met:

1.It is possible to find values of *e, d, n* such that $M^{ed} \bmod n = M$ for all *M* < *n*.

2.It is relatively easy to calculate mod *Me* mod *n* and *Cd* for all values of *M* < *n*.

3.It is infeasible to determine *d* given *e* and *n*.

For now, we focus on the first requirement and consider the other questions later. We need to find a relationship of the form

$M^{ed} \bmod n = M$

The preceding relationship holds if *e* and *d* are multiplicative inverses modulo f(*n*), where ϕ(*n*) is the Euler totient function.

$\phi(pq) = (p-1)(q-1)$ The relationship between *e* and *d* can be expressed as

$ed \bmod \phi(n)=1$

This is equivalent to saying

$ed \equiv 1 \bmod \phi(n)$

$d \equiv e^{-1} \bmod \phi(n)$

That is, *e* and *d* are multiplicative inverses mod $\phi(n)$. Note that, according to the rules of modular arithmetic, this is true only if *d* (and therefore *e*) is relatively prime to $\phi(n)$. Equivalently, $\gcd(\phi(n),d) = 1$.

We are now ready to state the RSA scheme. The ingredients are the following:

p,q, two prime numbers (private, chosen)

$n = pq$ (public, calculated)

e, with gcd($\phi(n),e$) = 1;1 < e < $\phi(n)$ (public, chosen)

$d \equiv e^{-1}(\text{mod } \phi(n))$ (private, calculated)

The private key consists of {d, n} and the public key consists of {e, n}. Suppose that user A has published its public key and that user B wishes to send the message M to A. Then B calculates $C = M^e \text{ mod } n$ and transmits C. On receipt of this ciphertext, user A decrypts by calculating $M = C^d \text{ mod } n$.

the keys were generated as follows:

1.Select two prime numbers, p = 17 and q = 11.

2.Calculate $n = pq$ = 17 x 11 = 187.

3.Calculate f(n) = (p 1)(q 1) = 16 x 10 = 160.

4.Select e such that e is relatively prime to $\phi(n)$ = 160 and less than $\phi(n)$ we choose e = 7.

5.Determine d such that $de \equiv 1$ (mod 160) and d < 160. The correct value is d = 23, because 23 * 7 = 161 = 10 x 16 + 1; d can be calculated using the extended Euclid's algorithm.

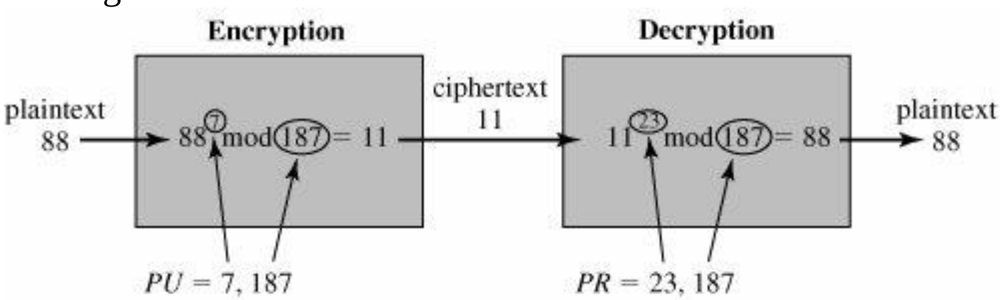

Figure (16) Example of RSA Algorithm

The resulting keys are public key PU = {7,187} and private key PR = {23,187}. The example shows the use of these keys for a plaintext input of M = 88. For encryption, we need to calculate $C = 88^7$ mod 187.

Exploiting the properties of modular arithmetic, we can do this as follows:

88^7 mod 187 = [(88^4 mod 187) x (88^2 mod 187) x (88^1 mod 187)] mod 187

88^1 mod 187 = 88

88^2 mod 187 = 7744 mod 187 = 77

88^4 mod 187 = 59,969,536 mod 187 = 132

88^7 mod 187 = (88 x 77 x 132) mod 187 = 894,432 mod 187 = 11

For decryption, we calculate $M = 11^{23}$ mod 187:

11^{23} mod 187 = [(11^1 mod 187) x (11^2 mod 187) x (11^4 mod 187) x (11^8 mod 187) x (11^8 mod 187)] mod 187

11^1 mod 187 = 11

11^2 mod 187 = 121

11^4 mod 187 = 14,641 mod 187 = 55

$11^8 \bmod 187 = 214{,}358{,}881 \bmod 187 = 33$

$11^{23} \bmod 187 = (11 \times 121 \times 55 \times 33 \times 33) \bmod 187 = 79{,}720{,}245 \bmod 187 = 88$

Key Generation

Before the application of the public-key cryptosystem, each participant must generate a pair of keys. This involves the following tasks:

- Determining two prime numbers, *p* and *q*
- Selecting either *e* or *d* and calculating the other

First, consider the selection of *p* and *q*. Because the value of $n = pq$ will be known to any potential adversary, to prevent the discovery of *p* and *q* by exhaustive methods, these primes must be chosen from a sufficiently large set (i.e., *p* and *q* must be large numbers). On the other hand, the method used for finding large primes must be reasonably efficient.