RSA Public Key Cryptosystem

pohanmal Zabihullah Zahir teacher of

Mathematics Department

ABSTRACT

Public Key Cryptoghraphy or asymmetric cryptography, is a cryptographic system that uses pairs of keys: public keys, which are disseminated widely, and private keys, which are known only to the owner. The generation of such keys depends on cryptographic algorithms based on mathematical problems to product one way functions. Effective security only requires keeping the private key private; the public key can be openly distributed without compromising security.

Keywords: *Asymmetric ciphers; creation; encryption; Decryption*

Introduction

RSA (Rivest–Shamir–Adleman) is one of the first public-key cryptosystems and is widely used for secure data transmission. In such a cryptosystem, the encryption key is public and distinct from the decryption key which is kept secret (private)[3]. In RSA, this asymmetry is based on the practical difficulty of factoring the product of two large prime numbers, the "factoring problem"[1]. The acronym RSA is the initial letters of the surnames of Ron Rivest, Adi Shamir, and Leonard Adleman, who publicly described the algorithm in 1977. Clifford Cocks, an English mathematician working for the British intelligence agency Government Communications Headquarters (GCHQ), had developed an equivalent system in 1973, which was not declassified until 1997[2].

A user of RSA creates and then publishes a public key based on two large prime numbers, along with an auxiliary value. The prime numbers must be kept secret. Anyone can use the public key to encrypt a message, but only someone with knowledge of the prime numbers can decode the message. Breaking RSA encryption is known as the RSA problem. Whether it is as difficult as the factoring problem is an open question. There are no published methods to defeat the system if a large enough key is used.

RSA is a relatively slow algorithm, and because of this, it is less commonly used to directly encrypt user data. More often, RSA passes encrypted shared keys for symmetric key cryptography which in turn can perform bulk encryption-decryption operations at much higher speed [9].

Symmetric and asymmetric ciphers

We have now seen several different examples of ciphers, all of which have a number of features in common. Bob wants to send a secret message to Alice. He uses a secret key k to scramble his plaintext message m and turn it into a ciphertext c. Alice, upon receiving c, uses the secret key k to unscramble c and reconstitute m. If this procedure is to work properly, then both Alice and Bob must possess copies of the secret key k, and if the system is to provide security, then their adversary Eve must not know k, must not be able to guess k, and must not be able to recover m from c without knowing k [5].

- Modern cryptographic methods use a key to control encryption and decryption
- Two classes of key-based encryption algorithms
  - symmetric (secret-key)
  - asymmetric (public-key)

Symmetric ciphers

Same key used for encryption and decryption.

Suppose that a message m is encrypted as c using a key k and is sent to B by A both A and B possess k, then it is called a symmetric cipher.

Let k be the space of all keys, M the space of all plain text messages and c the space of all cipher text messages [10].

Then encryption e can be thought of as a function $e: K \times M \to C$ and decryption $d: K \times C \to M$ for $k \in K$, $e_k: M \to C$

$$d_k: C \to M$$

Are such that

$$e_k(d_k(c)) = c\ , d_k(e_k(m)) = m,\ \ where\ e_k(m) = c,\ \ d_k(c) = m.$$

(k, M, c, e, d), to be successful method of encryption, it must have the following properties:

1. $e_k(m)$ should be easy to compute
2. $d_k(c)$ should also be easy to compute.
3. Given $c_1, c_2, \dots, c_n \in C$ encrypted using a key k, it should be difficult to compute $d_k(c_1), d_k(c_2), \dots, d_k(c_n)\ with\ out\ using\ k$.
4. Given $\{(m_1, c_1), (m_2, c_2), \dots, (m_n, c_n)\}$ it must be difficult to decrypt $c(\neq c_i)$ Without knowing k.
   This is called "SECURITY AGAINST CHOSEN PLAIN TEXT ATTACK."

There are some kinds of symmetric cipher:

a) Affine cipher
b) Hill cipher
c) XOR cipher …

- Main problem: key distribution
- Symmetric ciphers can be divided into stream ciphers and block ciphers
- **Stream ciphers**
  - can encrypt a single bit of plaintext at a time
- **Block ciphers**
  - take a number of bits and encrypt them as a single unit

Asymmetric ciphers

For encryption a different key is used for encryption and decryption.

- Said to be the most significant new development in cryptography in the last 300-400 years
  - first described publicly by Hellman and Diffie in 1976
- The encryption key is public, decryption key secret
  - anyone can encrypt a message but only the one who knows the corresponding private key can decrypt it
- In practise asymmetric and symmetric algorithms are often used together, called: hybrid encryption
- There are some kinds of asymmetric cipher:
  a) Diffie – Hellman
  b) RSA
  c) ElGamal

In my seminar I want to discuss about RSA public key cryptosystem [4,7].

RSA public key cryptosystem

RSA (RIVEST, ADI SHAMIR, LEONARD ADLEMAN)

- One of the more commonly used public key algorithms
- The RSA algorithm is user to do public key encryption and digital signatures based on factoring. The formula is simple, but takes a long time to calculate.
- RSA is used in most web-browsers as part of SSL.

The following figure illustrates the processes of key creation, Encryption and decryption in RSA PKC.

Bob	Alice
Key creation	
(i) Chooses two, secret primes p and q	

(ii) Chooses encryption exponent e satisfying $\gcd(e,(p-1)(q-1))=1.$ (iii) publishes $N(=pq)$ *and e.* *i.e,* (N,e) will be made public.	

Encryption

	(i) chooses plain text m (ii) Uses Bob's public key (N,e) to Computes cipher text c $c \equiv m^e \pmod N$ (iii) Sends c to Bob.

Decryption

(i) Computes and satisfying $ed = 1(mod(p-1)(q-1))$ (ii) Computes $m' \equiv c^d \pmod N$ (iii) Realizes that m' equals the plaintext m	

We shall illustrate the whole process involved in this RSA PKC with a numerical example [6].

Example 1.

RSA key creation

Bob (i) chooses two secret primes $p=1223$ and $q=1987$

(ii) Computes modulus $N = pq = 1223 \times 1987 = 2430101.$

(iii) Chooses encryption exponent $e=948047$ such that

$\gcd(e,(p-1)(q-1)) = \gcd(948047, 2426892)$ which also be made public. (N,e) is the public key [8].

RSA encryption

Alice (i) considers plaintext m $\ni 1 \le m < N$.

(ii) uses public key (N,e) to compute cipher text c as follows:

$$c \equiv m^e \pmod N$$

$$i.e,\ c \equiv 1070777^{948047} \pmod{2430101}$$

$$\equiv 1473513 \pmod{2430101}$$

And (iii) sends c to Bob.

RSA Decryption

Bob (i) knows $(p-1)(q-1) = 1222 \times 1986 = 2426892$

(ii) solve $ed \equiv 1(\mathrm{mod}(p-1)(q-1))$ to get d

$$i.e,\ 948047d \equiv 1 \pmod{2426892} \Rightarrow d = 1051235$$

(iii) takes cipher text c and computes $c^d \pmod N.$

$$i.e,\ 1473513^{1051235} \equiv 1070777 \pmod{2430101}$$ this is the message sent by Alice to Bob.

Here N is called the modulus, e is called the encryption on exponent, and d is called the decryption exponent of the RSA PKC [9].

Example 2. Alice publishes her RSA public key:

Modulus $N = 2038667$ and exponent $e = 103(encryption\ \mathrm{exp}onent)$

(a) Bob wants to send Alice the message $m = 892383$ what cipher text does Bob send to Alice?

(b) Alice knows that her modulus factors into a product of two primes one of which is $p = 1301$. Find the decryption exponent d for Alice.

(c) Alice receives the cipher text $c = 317730$ from Bob. Decrypt the message.

Solution

(a) Bob sends $c = m^e = 892383^{103} \pmod{2038667} \equiv 45293 \pmod{2038667}$

(b) Alice knows that $N = 2038667 = pq$, where $p = 1301.\ \therefore q = 1567.$

$$(p-1)(q-1) = 1300 \times 1566 = 2035800$$

Let us compute d using $ed \equiv 1(\mathrm{mod}(p-1)(q-1)).$

$$i.e, 103d \equiv 1 \bmod (2035800)$$
$$\Rightarrow d \equiv 810367 (\bmod 2035800)$$

(c) Alice has to solve

$m^{103} \equiv c(mod 2038667)$, given $c = 317730$

$$\Rightarrow m \equiv c^d \ (\bmod 2038667)$$
$$\Rightarrow m \equiv 317730^{810367} \ (\bmod 2038667)$$

$i.e, m \equiv 514407 (\bmod 2038667)$ is the decrypted message.

Now I want to solve some question about the factorization of p and q.

Example 3. For each of the given values of $N = pq$ and $(p-1)(q-1)$ find the primes p and q in the following:

a) $N = pq = 352717, (p-1)(q-1) = 351520$
b) $N = pq = 77083921, (p-1)(q-1) = 77066212$

Solution: if $N = pq$ then

$$(p-1)(q-1) = pq - (p+q) + 1 = N - (p+q) + 1.$$
$$\therefore p + q = N - (P-1)(Q-1) + 1$$

Consider the equation $x^2 - (p+q)x + N = 0.....(1)$ and its roots are p, q.

$$(a) \ \ p + q = N - (p-1)(q-1) + 1 = 352717 - 351520 + 1 = 1198$$

$$\therefore x^2 - 1198x + 352717 = 0$$
$$(x - 667)(x - 521) = 0$$

Has roots $677, 521$ which are p, q .

(b) $x^2 - (p+q)x + N = 0.....(1)$

$$(p-1)(q-1) = pq - (p+q) + 1 = N - (p+q) + 1.$$
$$\therefore p + q = N - (P-1)(Q-1) + 1$$

$p + q = pq - (p - 1)(q - 1) + 1 = 77083921 - (77066212) + 1 = 17710$

$x^2 - 17710x + 77083921 = 0$,

$(x - 10007)(x - 7703) = 0$

Has roots 10007 , 7703 which are p, q .

Conclusion

RSA (Rivest–Shamir–Adleman) is an algorithm used by modern computers to encrypt and decrypt messages. It is an asymmetric cryptographic algorithm. Asymmetric means that there are two different keys. This is also called public key cryptography, because one of the keys can be given to anyone. The other key must be kept private. The algorithm is based on the fact that finding the factors of a large composite number is difficult. When the factors are prime numbers, the problem is called prime factorization. It is also a key pair (public and private key) generator.

References

[1] J.M. Ortega, W.C. Rheinbolt. Iterative solution of nonlinear equations in several variables. Academic Press, New York, 1970.

[2] F.A. Potra, V. Ptak. Nondiscrete induction and iterative processes. Research Notes in Mathematics, Vol 203, Pitman, Boston, 1984.

[3] R.F. King. A family of fourth order methods for nonlinear equations. Society for Industrial and Applied Mathematics, 1973, 5(10):876-879.

[4] C. Chun. Some third-order families of iterative methods for solving nonlinear equations[J]. Applied Mathematics and Computation, 2007, 188(1):924-933.

[5] J. Kou, Y. Li, X. Wang. A family of fifth-order iterations composed of Newton and third-order methods[J]. Applied Mathematics and Computation, 2007, 186 (2):1258-1262.

[6] Y Ham, C Chun. A fifth-order iterative method for solving nonlinear equations[J]. Applied Mathematics and Computation, 2007, 194(1):287-290.

[7] J.R. Sharma, R.K. Guha. A family of modified Ostrowski methods with accelerated sixth-order convergence[J]. Applied Mathematics and Computation, 2007,190 (1):111-115.

[8] C Chun, Y Ham. Some sixth-order variants of Ostrowski root-finding methods[J]. Applied Mathematics and Computation, 2007, 193(2):389-394.

[9] L. Fang, G.P. He. Some modifications of Newton's method with higher-order convergence for solving nonlinear equations[J]. Journal of computational and Applied Mathematics, 2009, 228(1):296-303.

[10] L.P Liu, X Wang. Eighth-order methods with high efficiency index for solving nonlinear equations[J]. Applied Mathematics and Computation, 2010, 215(9):3449-3454.