

# **Survey on Asymmetric Key Cryptography Algorithms**

S. Nithya , Dr. E. George Dharma Prakash Raj  
Computer Science and Engineering, Bharathidasan University, Trichy, India.  
Computer Science and Engineering ,Bharathidasan University, Trichy, India.  
niyaa20@yahoo.co.in,georgeprakashraj@yahoo.com

## **ABSTRACT**

Asymmetric key encryption is also called public key encryption. It uses two keys known as a public and a private key. Data encrypted with one key can be decrypted only with the other key. This paper discusses some public key algorithms and its advantages and disadvantages such as Rivest Shamir Adleman, Diffi-Hellman, Elliptic Curve Cryptography, Elliptic Curve Diffi-Hellman, ElGamal Encryption Algorithm, Knapsack Algorithm, Digital Signature Algorithm and Short Range Natural Numbers.

## **Keywords**

*Cryptography, Rivest Shamir Adleman, Elliptic Curve Cryptography, Diffi - Hellman, Digital Signature.*

## **I. INTRODUCTION**

All public key cryptosystems are based on the notion of a one-way function. Which depending on the public key, converts plaintext into cipher text using are relatively small amount of computing power but whose inverse function is extremely expensive to compute, so that an attacker is not able to derive the original plaintext from the transmitted cipher text within a reasonable time frame. In public key cryptography, each user or the device taking part in the communication have a pair of keys, a public key and a private key, and a set of operations associated with the keys to do the cryptographic operations. Only the particular user/device knows the private key whereas the public key is distributed to all users/devices taking part in the communication. It increased security and convenience: private keys never need to be transmitted or revealed to anyone [1].

This paper work focuses on brief descriptions of various public key cryptography algorithms which are discussed on section II and in section III advantages and disadvantages of these algorithms were made in the form of comparative study.

## **II. PUBLIC KEY CRYPTOGRAPHY ALGORITHMS**

### **A. Rivest Adi Adleman (RSA)**

In 1978, Ron Rivest, Adi Shamir, and Leonard Adleman introduced a cryptographic algorithm, which was essentially to replace the less secure National Bureau of Standards (NBS) algorithm. In RSA, this asymmetry is based on the practical difficulty of factoring the product of two large prime numbers, the factoring problem [2].

In such a cryptosystem, the encryption key is public and differs from the decryption key which is kept secret. The RSA algorithm involves three steps: key generation, encryption and decryption.

#### **(i) Key generation:**

- R1. Users of RSA must determine two primes at random:  $p, q$
- R2. Select either  $e$  or  $d$  and compute the other
- Primes  $p, q$  must not be easily derived from modulus  $N=p*q$
- R3. Exponent  $e, d$  is inverses, so use inverse algorithm to compute the other.

#### **(ii) Encryption:**

- R4. To encrypt a message  $m$  the sender
- R5. Obtains public key of recipient  $Pu = \{e, N\}$
- R6. Computes:  $C = M^e \bmod N$ , where  $0 \leq M < N$

#### **(iii) Decryption:**

- R7. To decrypts the cipher text  $c$  the owner
- R8. Uses their private key  $PR = \{d, p, q\}$
- R9. Computes:  $M = M^d \bmod N$

### **B. Diffi-Hellman key exchange (D-H)**

The DH algorithm, introduced by Whitfield Diffie and Martin Hellman in 1976, was the first system to utilize "public-key" or "asymmetric" cryptographic keys. These systems overcome the difficulties of "private-key" or "symmetric" key systems because asymmetric key management is much easier [3]. DH is a mathematical algorithm that allows two computers to generate an identical shared secret on both systems, even though those systems may never have communicated with each

other before. That shared secret can then be used to securely exchange a cryptographic encryption key. That key then encrypts traffic between the two systems.

The Diffie-Hellman Key Exchange is one of the more popular and interesting methods of key distribution. It is a public-key cryptographic system whose sole purpose is for distributing keys.

(i) *Key algorithm:*

It is also possible to use Diffie-Hellman as part of a public key infrastructure.

D1. First, both Alice and Bob agree upon a prime number and another number that has no factor in common. Let's call the prime number as  $p$  and the other number as  $g$ . Note that  $g$  is also known as the generator and  $p$  is known as prime modulus.

D2. Now,  $g^a \bmod p$  and  $g^b \bmod p$  is the public key for Alice and Bob respectively.

D3. Alice and Bob exchanged their public keys.

D4. Alice computes  $(g^b \bmod p)^a \bmod p$  that is equal to  $g^{ba} \bmod p$ .

D5. Bob computes  $(g^a \bmod p)^b \bmod p$  that is equal to  $g^{ab} \bmod p$ .

D6.  $K = g^{ba} \bmod p = g^{ab} \bmod p$ , shared secret  $= k$ .

C. *Digital Signature Algorithm (DSA)*

The Digital Signature Algorithm (DSA) is a Federal Information Processing Standard for digital signatures. It was proposed by the National Institute of Standards and Technology (NIST) in August 1991 for use in their Digital Signature Standard (DSS) and adopted as FIPS 186 in 1993.

A digital signature scheme typically consists of three algorithms: A key generation algorithm that selects a private key uniformly at random from a set of possible private keys. The algorithm outputs the private key and a corresponding public key [4]. A signing algorithm that, given a message and a private key, produces a signature. A signature verifying algorithm that, given a message, public key and a signature, either accepts or rejects the message's claim to authenticity.

(i) *Key generation:*

S1. select a prime  $q$  of 160 bits

S2. Choose  $0 \leq t \leq 8$ , select  $2^{511+64t} < p < 2^{512+64t}$  with  $q|p-1$

S3. Select  $g$  in  $Z_p^*$ , and  $\alpha = g^{(p-1)/q} \bmod p$ ,  $\alpha \neq 1$

S4. Select  $1 \leq a \leq q-1$ , compute  $y = \alpha^a \bmod p$

S5. public key  $(p, q, \alpha, y)$ , private key  $a$

(ii) *Signing:*

S6. Select a random integer  $k$ ,  $0 < k < q$

S7. Compute  $r = (\alpha^k \bmod p) \bmod q$

S8. Compute  $k^{-1} \bmod q$

S9. Compute  $s = k^{-1} * (h(m) + ar) \bmod q$

S10. Signature  $= (r, s)$

(iii) *Verification:*

S11. Verify  $0 < r < q$  and  $0 < s < q$ , if not, invalid

S12. Compute  $w = s^{-1} \bmod q$  and  $h(m)$

S13. Compute  $u_1 = w * h(m) \bmod q$ ,  $u_2 = r * w \bmod q$

S14. Compute  $v = (\alpha^{u_1} \gamma^{u_2} \bmod p) \bmod q$

S15. Valid if  $v = r$

D. *Elliptic curve cryptography (ECC)*

Elliptic curve cryptography (ECC) is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields. The use of elliptic curves in cryptography was suggested independently by Neal Koblitz and Victor S. Miller in 1985. Elliptic curves are also used in several integer factorization algorithms that have applications in cryptography [5].

E. *Elliptic curve Diffie-Hellman (ECDH)*

Elliptic curve Diffie-Hellman is an anonymous key agreement protocol that allows two parties, each having an elliptic curve public-private key pair, to establish a shared secret over an insecure channel. This shared secret may be directly used as a key, or better yet, to derive another key which can then be used to encrypt subsequent communications using a symmetric key cipher. It is a variant of the Diffie-Hellman protocol using elliptic curve cryptography [6].

(i) *Key generation algorithm:*

E1. Alice wants to establish a shared key with Bob.

E2. The domain parameters must be agreed upon.

E3. Let Alice's key pair be  $(d_A, Q_A)$  and Bob's key pair be  $(d_B, Q_B)$ .

E4. Alice computes  $(x_K, y_K) = d_A Q_B$ . Bob computes  $(x_K, y_K) = d_B Q_A$ .

E5. The shared secret calculated by both parties is equal  $d_A Q_B = d_A d_B G = d_B d_A G = d_B Q_A$ .

F. *ElGamal Encryption Algorithm*

In cryptography, the ElGamal encryption system is an asymmetric key encryption algorithm for public-key cryptography which is based on the Diffie-Hellman key exchange. It was described by Taher Elgamal in 1984. ElGamal encryption

consists of three components: the key generator, the encryption algorithm, and the decryption algorithm [7].

(i) *Key generation:*

G1. Alice generates an efficient description of a multiplicative cyclic group  $G$  of order  $q$  with generator  $g$ . See below for a discussion on the required properties of this group.  
 G2. Alice chooses a random  $x$  from  $\{1 \dots q-1\}$ .  
 G3. Alice computes  $h = g^x$ .  
 G4. Alice publishes  $h$ , along with the description of  $g$ ,  $q$ ,  $g$ , as her public key. Alice retains  $x$  as her private key which must be kept secret.

(ii) *Encryption:*

The encryption algorithm works as follows: to encrypt a message  $m$  to Alice under her public key  $(g, q, g, h)$

G5. Bob chooses a random  $y$  from  $\{1 \dots, q-1\}$ , then calculates  $c_1 = g^y$ .  
 G6. Bob calculates the shared secret  $s = h^y$ .  
 G7. Bob converts his secret message  $m$  into an element  $m'$  of  $G$ .  
 G8. Bob calculates  $c_2 = m' \cdot s$ .  
 G9. Bob sends the ciphertext  $(c_1, c_2 = (g^y, m' \cdot h^y) = g^y, m' \cdot (g^{xy})$  to Alice.

(iii) *Decryption:*

The decryption algorithm works as follows: to decrypt a ciphertext  $(c_1, c_2)$  with her private key  $x$ ,  
 G10. Alice calculates the shared secret  $s = c_1^x$ .  
 G11. Then computes  $m' = c_2 \cdot s^{-1}$  which she then converts back into the plaintext message  $m$ , where  $s^{-1}$  is inverse of  $s$  in the group  $G$ . The decryption algorithm produces the intended message.

G. *Knapsack Algorithm*

The Merkle–Hellman knapsack cryptosystem was one of the earliest public key cryptosystems invented by Ralph Merkle and Martin Hellman in 1978.

(i) *Key generation:*

To encrypt  $n$ -bit messages, choose a super increasing sequence  $w = (w_1, w_2 \dots w_n)$  of  $n$  nonzero natural numbers. Pick a random integer  $q$ , such that  $q > \sum_{i=1}^n w_i$ ,

And a random integer,  $r$ , such that  $\gcd(r, q) = 1$  (i.e.  $r$  and  $q$  are co prime).

$q$  Is chosen this way to ensure the uniqueness of the cipher text. If it is any smaller, more than one plaintext may encrypt to the same cipher text. Since

$q$  is larger than the sum of every subset of  $w$ , no sums are congruent mod  $q$  and therefore none of the private key's sums will be equal.  $r$  must be co prime to  $q$  or else it will not have an inverse mod  $q$ . The existence of the inverse of  $r$  is necessary so that decryption is possible.

Now calculate the sequence

$\beta = (\beta_1, \beta_2 \dots \beta_n)$  where  $\beta_i = r w_i \text{ mod } q$ .  
 The public key is  $\beta$ , while the private key is  $(w, q, \text{ and } r)$ .

(ii) *Encryption:*

To encrypt an  $n$ -bit message

$\alpha = (\alpha_1, \alpha_2 \dots \alpha_n)$ ,

Where

$\alpha_i$  Is the  $i$ -th bit of the message and  $\alpha_i \in \{0, 1\}$ , calculate

$$c = \sum_{i=1}^n \alpha_i \beta_i.$$

The cryptogram then is  $c$ .

(iii) *Decryption:*

In order to decrypt a cipher text  $c$  a receiver has to find the message bits  $\alpha_i$  such that they satisfy

$$c = \sum_{i=1}^n \alpha_i \beta_i.$$

This would be a hard problem if the  $\beta_i$  were random values because the receiver would have to solve an instance of the subset sum problem, which is known to be NP-hard. However, the values  $\beta_i$  were chosen such that decryption is easy if the private key  $(w, q, \text{ and } r)$  is known [8].

H. *Short Range Natural Numbers Algorithm (SRNN)*

In asymmetric key cryptography, also called Public Key cryptography, two different keys are used. One key is used for encryption & only the other corresponding key must be used for decryption. No other key can decrypt the message - not even the original (i.e. the first) key used for encryption [9]. The beauty of this scheme is that every communicating party needs just a key pair for communicating with any number of other communicating parties. Once someone obtains a key pair, he /she can communicate with anyone else. The Short Range Natural Number (SRNN) algorithm is similar to RSA algorithm with some modification. This modification increases the security of the cryptosystem [10].

(i) *Key generation:*

N1. Generate two large random prime  $p, q$ .

N2. Compute  $n=p*q$   
 N3. Compute  $\phi=(p-1)(q-1)$   
 N4. Choose an integer  $e, 1<e<\phi$ , such that  $\gcd(e, \phi)=1$  compute the such that  $(e*d) \bmod \phi=1$   
 N5. Pick short range natural number  $u$  randomly such that  $u<\phi-1$   
 N6. Pick another Short range natural number  $a$  randomly such that  $\phi>a>u$  and compute  $ua$   
 N7. Find  $d$  such that,  $e*d \bmod ((p-1)(q-1))=1$   
 N8. Public key is  $(n, e, ua)$   
 N9. Private Key is  $(d, a, u)$   $P, q, \phi$  should also be kept secret.

*(ii) Encryption:*

N10. Obtains the recipient's public key  $(n, e, ua)$   
 N11. Represent the plaintext message as positive integer  $M$   
 N12. Computes the cipher text  $C=(m \text{ } ua) \bmod n$   
 Send the cipher text  $C$  to recipient.

*(iii) Decryption:*

N13. Use Recipient private key  $(d, a, u)$   
 N14. Compute  $M=(ve \text{ } c) \bmod n$  where  $v=\phi-a \bmod n$   
 N15. Extracts the plaintext from the integer representative  $M$

**III. Advantages and Disadvantages**

The various asymmetric key cryptography algorithms' advantage and disadvantage is discussed in the following table.

**Table 1. Advantage and disadvantage of various asymmetric key cryptography algorithms**

| S.NO | Algorithms     | Advantages                                                                    | Disadvantages                                                                                                       |
|------|----------------|-------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------|
| 1    | RSA            | Only intended user can read the message using their private key.              | Many secret-key encryption methods that is significantly faster than any currently available public-key encryption. |
| 2    | Diffie-Hellman | The shared key (i.e. the secret) is never itself transmitted over the channel | Lack of authentication                                                                                              |
| 3    | DSA            | It is used for authentication and Integrity                                   | The security of the private key depends entirely on the security of the computer.                                   |
| 4    | ECC            | Short key is faster and requires less computing power.                        | It is more expensive and it shortens the life time of batteries.                                                    |
| 5    | ECDH           | Very secure means of                                                          | Little difficulty in exchanging keys                                                                                |

|   |          |                                                                                                 |                                                                        |
|---|----------|-------------------------------------------------------------------------------------------------|------------------------------------------------------------------------|
|   |          | exchanging keys between two parties                                                             |                                                                        |
| 6 | ElGamal  | The same plaintext gives a different ciphertext (with near certainty) each time it is encrypted | The need for randomness, and its slower speed and has long ciphertext. |
| 7 | Knapsack | A perfect protocol for distribution of secret keys                                              | deciphering keys are easy sequences, they are breakable                |
| 8 | SRNN     | The attack can be avoided by increasing the key size.                                           | Increase the time-performance.                                         |

**IV. CONCLUSION**

Public-key authentication, on the other hand, prevents type of repudiation; each user has sole responsibility for protecting his or her private key. This property of public-key authentication is often called non-repudiation. Secret-key cryptography lags behind asymmetric cryptography. Combinations of the two can be implemented for improved security but secret-key cryptography by itself proves insecure against man in the middle attacks. Asymmetric cryptography has been the foundation for secure data exchange over networks and while it still has its shortcomings, new ideas still come forth as the field continues to evolve.

**REFERENCES**

- [1]. Caregia Mellon Software Engineering institute, "Public Key Cryptography".
- [2]. Yaun Xue, "Public key Cryptography and RSA Algorithm", Technical notes and papers.
- [3]. Alese, B. K. Philemon E. D., Falaki, S. O., "Comparative Analysis of Public-Key Encryption Schemes", International Journal of Engineering and Technology Volume 2 No. 9, September, 2012
- [4]. Aqeel Khalique, Kuldip Singh Sandeep Sood, "Implementation of Elliptic Curve Digital Signature Algorithm", International Journal of computer Application, May 2010.
- [5]. Kamlesh Gupta, Sanjay Silakari, "ECC over RSA for Asymmetric Encryption: A Review", IJCSI May 2011.
- [6]. Ruchika Markan, Gurvinder Kaur, "Literature Survey on Elliptic Curve Encryption Techniques", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 9, September 2013.
- [7]. Rashmi Singh, Shiv Kumar, "ElGamal Algorithm in Cryptography", International Journal of Scientific & Engineering Research Volume 3, Issue 12, December-2012.
- [8]. Preeti L. Darade, "Knapsack Based ECC with Encryption and Decryption", International Journal of Emerging Technology and Advanced Engineering, April 2012.
- [9]. Mr. Hemant Kumar, Dr. Ajith Singh, "An Efficient Implementation of digital signature Algorithm with SRNN Public Key Cryptography", IJRREST, June 2012.
- [10]. E. George Dharma Prakash Raj, k. Sheela, "Survey on public key cryptography algorithms", IJSRCSMS July 2013.