ISSN 2229-5518

Review and Analysis of Cryptography Techniques

Nitin Jirwan, Ajay Singh, Dr. Sandip Vijay

Abstract- Today data communication mainly depends upon digital data communication, where prior requirement is data security, so that data should reach to the intended user. So for providing data security many cryptography techniques are employed, such as symmetric and asymmetric techniques. In this review paper different asymmetric cryptography techniques, such as RSA (Rivest Shamir and Adleman), Diffie-Hellman, DSA (Digital Signature Algorithm), ECC (Elliptic curve cryptography) are analyzed.

KEY WORDS: RSA (Rivest Shamir and Adleman), Diffie-Hellman, DSA (Digital Signature Algorithm), ECC (Elliptic curve cryptography).

1. INTRODUCTION

Every user while communicating wants a secure network so that data communication should secure and no intruder can read their data. For providing secure data communication cryptography is used in wireless and wired network, where cryptography converts to plain text into cipher text and cipher text into a plain text. At a sender side plain text is converted into a cipher text known as encryption and receiver side cipher text is converted into a plain text known as decryption. Cryptography classified as Symmetric cryptography and Asymmetric cryptography techniques. In symmetric-key cryptography, the same key is used by both parties. The sender uses this key and an encryption algorithm to encrypt data; the receiver uses the same key and the corresponding decryption algorithm to decrypt the data. In asymmetric or public-key cryptography, there are two keys: a private key and a public key are used. The private key is kept by the receiver and public key is announced to the public. Further some types of asymmetric cryptography are given by different researchers. Some commonly used asymmetric cryptography techniques are RSA (Rivest Shamir and Adleman), Diffie-Hellman, DSA (Digital Signature Algorithm), ECC (Elliptic curve cryptography). All these technique are discussed below in this paper.

- *Nitin Jirwan is currently pursuing masters degree program in Wireless and Mobile Communication in Uttarakhand Technical University, India. E-mail: neetu_jirwan@yahoo.co.in.*
- *Co-Author Ajay Singh is currently pursuing masters degree program in Wireless and Mobile Communication in Uttarakhand Technical University, India. E-mail:* arasyal@gmail.com.
- *Co-Author Dr. Sandip Vijay is currently posted as Head of Department of ECE and AEI at Dehradun Institute of Technology, Dehradun (INDIA). E-*

2. ANALYSES OF DIFFERENT TECHNIQUES

In this review paper above described techniques of cryptography are analyzed based on different research paper in respective journals.

2.1. Rivest Shamir and Adleman (RSA) algorithm

RSA is an algorithm for public-key cryptography that is based on the presumed difficulty of factoring large integers, the factoring problem. A user of RSA creates and then publishes the product of two large prime numbers, along with an auxiliary value, as their public key. The prime factors must be kept secret. Anyone can use the public key to encrypt a message, but with currently published methods, if the public key is large enough, only someone with knowledge of the prime factors can feasibly decode the message [1]. RSA algorithm can be used in Wireless Sensor Network (WSN), because WSN is insecure network and vulnerable to many attacks because of broadcast nature of transmission medium. The security requirements of wireless sensor networks are [2]:

a. Confidentiality
b. Integrity
c. Authentication

Algorithm of RSA is given below [3]:

To create public key K_p:

a. Select two different prime P and Q
b. Assign x=(P-1)(Q-1)
c. Choose E relative primes to x which must satisfy a condition for K_s
d. Assign N=P*Q
e. K_p is N concatenated with E

To create private key:

a. Choose D: D*E mod x=1
b. K_s is N concatenated with E.

To encode the plain text m by:

a. Assume m is a numeric
b. Calculate $c=m^E$ mod N

To decode c back to m:

a. Calculate m= c^D mod N.

RSA is not suitable for WSN because of high time complexity and consumption demand [4]. Any encryption algorithm such as RSA can be implemented in software. The disadvantages are in speed, cost, and ease of modification (or manipulation). The advantages are in flexibility and portability, ease of use, and ease of upgrade. The algorithms can be inexpensively copied and installed on many machines [5].

Another implementation of RSA algorithm is given by Chandra M. Kota et al. [6]. In this paper, the secret key consists of two large prime numbers p and q, and a part of the public key is their product, n = p*q. The RSA cryptosystem security is investigated. It is shown that if the secret exponent length is p proximately one quarter as many bits as the modulus (n), there is a way to attack the RSA algorithm. The cryptanalysis of the RSA algorithm is also discussed. It is shown that it is possible to find the key if it is less than $n^{0.25}$. It is worthwhile to mention that if e is chosen to be greater than $(p*q)^{1.5}$ then this attack algorithm is not guaranteed to work [6].

In [7] one example for selection of large prime number (p, q), Selection of Encryption Key (E), Selection of Decryption Key (D)

a. Selection of large prime number (p, q): The main feature of RSA algorithm is the selection of large prime number (p, q) because it is logical that fraction of large number is always typical and any users or force attackers could not be able to find the capable numbers, timely to force attack is shortly non-feasible.
Example: p = 5, q = 3 N= p*q = 5*3 = 15 = 1*15 = 15*1 = 3*5 = 5*3.

b. Selection of Encryption Key(E): Selection of large of large prime fraction always create impact during the selection of Encryption key, if the factor is high then the estimation of Encryption is infeasible. Example: If p=7, q=17 must not be a factor of (p-1)*(q-1)
i.e. (7-1)*(17-1) = 6*16 = 96 = 2*2*2*2*2*3 So, E can be 5, 7, 11…

c. Selection of Decryption Key (D): Selection of large factors always creates an effect on the Decryption key, there may be an inversely relation:

(E*D) mod (p-1)*(q-1) =1 D α 1/ [(E) (p) (q)] [7].

Note: Some important points as given below:
According to Euler's Totient Function:
1. Ø (1) = 0
2. Ø (p) = p – 1 (if p is prime number)
3. Ø (m*n) = Ø (m)* Ø (n) (if m and n is relative prime number)
4. Ø (p^e) = p^e – p^{e-1} (if p is a prime number)

a. There is no need for a user to know his secret parameters p, q and Ø (n).
b. The plain text or message (M) has the form of one or more positive integer M<N.
c. Any user can use his private key to authenticate the communication.
d. RSA cryptosystem provides the facility of digital signature scheme.
e. The message consists of letters, numbers and special characters (i.e. stop, colon, space etc.). Each character is represented by its own arrangement of eight bits (0 & 1).
f. The most of the hardware & software products and standards that use public key technique for Encryption, Decryption etc. are based on RSA cryptosystem.

2.2. Diffie-Hellman Algorithm

This algorithm is used for exchanging cryptography keys between two users. Here user doesn't have any knowledge about the keys used by each other and they use a shared secret key over an insecure communication channel, then this key is used to encrypt subsequent communications using a symmetric key cipher [8].

New protocol proposed for two goals: authenticated key agreement and authenticated key agreement with key confirmation in the asymmetric (public-key) setting is given by Simon Blake-Wilson et al. [9]. Here they have proposed formal definitions of secure AK (Authenticated Key Agreement) and AKC (Authenticated Key agreement with key confirmation) protocols within a formal model of distributed computing and a unified model of key agreement is proposed with several variants of this model are demonstrated to provide secure AK and AKC protocol in the random oracle model. Here AK and AKC are made secure by providing clear, formal definitions of the goals of AK and AKC protocols, and secondly by furnishing practical, provably secure solutions in the random oracle model [9]. Briefly speaking; the process of providing security can be explained in five steps [9]:

a. Specification of model
b. Definition of goals within this model
c. Statement of assumptions
d. Description of protocol
e. Proof that the protocol meets its goals within the model.

2.2.1. Properties of Key agreement algorithm

a. Known Session Key: This protocol has stored some previous session key.

b. (Perfect) Forward Secrecy: This protocol can be compromised in long term secrets of one or more entities then secrecy of previous key is not affected.

c. Unknown Key Share: Suppose there are two users *i* and *j* then *i* cannot share the key with *j* without *i*'s knowledge.

d. Key-Compromise Impersonation: If the value of *i* is disclosed, and can be copied by intruders. But the nature of *i* should be like that the other properties of *i* can't be copied and affected.

e. Loss of Information: Compromise of other information that would not ordinarily be available to an adversary does not affect the security of the protocol.

f. Message Independence: This protocol run between two users are unrelated.

This algorithm can be practically implemented with increased security as compared to the currently used protocol [9].

Another way of implementation of Diffie-Hellman algorithm in internet is given [10]. It can be used nearly in every encryption technology used in the Internet today, including SSL, SSH, IPSec, PKI, and everything else that depends on these protocols [10]. In SSl (Secure Sockets Layer), Today in communication process client and server exchanges unencrypted messages. The asymmetric key is used in exchange process and compression option they each accept and prefer [10]. In SSH (Secure Shell), client and server start their process by negotiating parameters (e.g., preferred encryption and compression algorithms, and certain random numbers) [10]. In IPsec (Internet Protocol Security), some preliminary information exchange is necessary for starting encrypting the data stream [10]. In PKI (Public Key Infrastructure), two complementary uses can be made of public key cryptography. If one encrypts a message with the public key of another person, only that person can decrypt it because only that person knows his private key [10].

2.3. Digital Signature Algorithm (DSA)

It is used by receiver of a message to verify that the message has not been altered during transit as well as certain the sender's identity. A digital signature is an electronic version of a written signature in that the digital signature can be used in proving to the recipient or a third party that the message was, in fact, signed by the sender. Digital signatures may also be generated for stored data and programs so that the integrity of the data and programs may be verified at any later time [11]. One method for sending low size and capacity data by using DSA is proposed by Erfaneh Noorouzil et al. "Hash function" is used

in this method and it generates dynamic and smaller size of bits which depends on each byte of data. The main function which is used for hashing is bitwise or and multiply functions. If hashed file sized is 4% of the original file in the messages with size lower than 1600 bytes. This algorithm can be used in several applications which have low file size for sending and want simple and fast algorithms for generating digital signature [12].

Hash function follows some properties, which are given below.

a. Hash function should destroy all homomorphism structures in the underlying public key cryptosystem (be unable to compute hash value of 2 messages combined given their individual hash values) [13].

b. Hash function should be computed on the entire message [13].

c. Hash function should be a one-way function so that messages are not disclosed by their signatures [13].

d. Hash function should be computationally infeasible given a message and its hash value to compute another message with the same hash value [13].

e. Hash function should resist birthday attacks (finding any 2 messages with the same hash value, perhaps by iterating through minor permutations of two messages) [13].

This algorithm works on ".doc, .pdf, .txt" and other types of files, and hash function can be used for dynamic size of data. The term dynamic means, results of hash function depends on size of the data [13].

2.4. Elliptic Curve Cryptography (ECC)

Elliptic curve cryptography is a relatively new family of public-key algorithms that can provide shorter key lengths and, depending upon the environment and application in which it is used, improved performance over system based on integer factorization and discrete logarithms [14]. Here its security, advantages and performance [15] are discussed. ECC has its security problems based on some difficult mathematical. Elliptic curve is based on a mathematical structure in which certain operation can be defined. These operations provide a one way function that can be used to produce efficient cryptographic systems. ECC uses this one way function is called Elliptic Curve Discrete logarithm Problem (ECDLP). The ECDLP is similar to the one way function on which DSA and Diffie-Hellman are based, and hence, elliptic curve analogs of each of these algorithms have been defined [14].

Security and advantages of using elliptic curve based cryptographic systems instead of integer factorization and discrete logarithm based methods is that they provide similar security levels using smaller key lengths. Most people consider the integer factorization and discrete logarithm problems to have approximately equivalent security [14]. Performance of ECC with other algorithms is it is 5 to 15, 20 and 60, and sometimes 400 times faster than others depend on ECC bit [15].

Elliptic Curve Cryptography algorithm is also suitable for smart card application, as it is faster and occupies less memory than RSA [16].

3. ANALYTICAL TABLE

S. No.	Cryptography Technique	ANALYSIS
1.	Rivest Shamir and Adleman (RSA) algorithm	RSA can be used in Mobile nodes; because they are vulnerable to many attacks due to their broadcast nature [2].

		RSA is not suitable for WSN because of high time complexity and consumption demand [4].
2.	Diffie-Hellman Algorithm	Here keys are exchanged between two users; unknown to each other [8].
		A proposed for two goals: authenticated key agreement and authenticated key agreement with key confirmation in the asymmetric (public-key) setting [9].
		It can be used in Internet and nearly in every encryption technology used in the Internet today, including SSL, SSH, IPSec, PKI [10].
3.	Digital Signature Algorithm	Used by the receiver to verify that the message received is unaltered; a digital signature is used for performing this task [11].
		Hash function is used to generate dynamic and smaller size of bits which depends on each byte of data [12].
		Result of Hash function depends on size of data [13].
4.	Elliptic Curve Cryptography	Public-key algorithms that can provide shorter key lengths and, depending upon the environment and application in which it is used, improved performance over system based on integer factorization and discrete logarithms [14].
		Performance of ECC with other algorithms is, it is 5 to 15, 20 and 60, and sometimes 400 times faster than others depend on ECC bit [15].

4. CONLUSION

After reviewing all the above defined cryptography techniques it can be concluded that ECC is faster than RSA, because it uses small key. But its mathematically operation is complex as compare to RSA. In Diffie-Hellman cryptography algorithm secret keys are exchanged between two users. Whereas a digital signature is used by receiver in DSA to confirm that the signal received is unaltered.

ACKNOWLEDGMENT

The authors wish to thank Ajay Singh and Dr. Sandip Vijay for their support in this work.

REFRENCES:

[1]. Wikipedia, "http://en.wikipedia.org/wiki/RSA_(algorithm)," Dated: 12-dec-2012 at 19:40.

[2]. A. Perrig, J. Stankovic, and D. Wagner, "Security In Wireless Sensor Networks," ACM, Vol. 47, No.653.2004.

[3]. H. Anderson. Introduction to Computer Security, Prentice Hall, 2004, pp: 85-86.

[4]. F. Amin, A. H. Jahangir and H. Rasifard, "Analysis Of Publickey Cryptography For Wireless Sensor Networks Security," In Proceedings of World Academy of Science, Engineering and Technology, ISSN 1 307-6884, 2008.

[5]. Abdullah Said Alkalbani et al., "Comparison between RSA Hardware and Software Implementation for WSNs Security Schemes," In proceeding 3rd International Conference on ICT4M 2010.

[6]. Chandra M. Kota et al., "Implementation of the RSA algorithm and its cryptanalysis," In proceedings of the 2002 ASEE Gulf-Southwest Annual Conference, March 20 – 22, 2002

[7]. Prasant Singh Yadav et al., "Implementation of RSA algorithm using Elliptic Curve Algorithm for security and performance enhancement," International Journal of Scientific & Technology Research Volume 1, Issue 4, May 2012.

[8]. Wikipedia, "http://en.wikipedia.org/wiki/Diffie%E2%80%93 Hellman_key_exchange," Dated: 13-dec-2012 at 10:33.

[9]. Simon Blake Wilson et al., "Key agreement protocols and their security analysis," 9-sep-1997.

[10]. David A. Carts, "A Review of the Diffie-Hellman Algorithm and its Use in Secure Internet Protocols," SANS institute, 5-nov-2001.

[11]. Vocal, "http://www.vocal.com/cryptography/dsa-digital-signature-algorithm/," Dated: 13-dec-2012 at 13:18.

[12]. Erfaneh Noorouzil et al, "A New Digital Signature Algorithm", International Conference on Machine Learning and Computing, IPCSIT vol.3, 2011.

[13]. William-Stallings, http://williamstallings.com/Extras/Security Notes/lectures/authent.html, Dated: 13-dec-2012 at 14:05.

[14]. Robert Zuccherato, "Elliptic Curve Cryptography Support in Entrust," Entrust ltd. in Canada, Dated : 9-may-2000.

[15]. Kristin Lauter, "The Advantages of Elliptic Curve Cryptography for Wireless security," IEEE Wireless Communication, Feb 2004.

[16]. Vivek Kapoor et al., "Elliptic Curve Cryptography," ACM Ubiquity, Volume 9, Issue 20, (20-26)-may-2008.