RESEARCH PAPER OPEN ACCESS

A Survey of Different Public-Key Cryptosystems

Anita Ganpati [1], Narender Tyagi [2]
Department of Computer Science
Himachal Pradesh University
Shimla - India

ABSTRACT

Cryptography is the study of techniques which is used to communicate, store information or data securely, without being intercepted by third parties. In the real world, there are so many organizations working on large databases over a public network. The security of data is the primary concern in the public network. Encryption is the most commonly used technique where transactions continuously take place between the users. Other cryptography techniques like digital time-stamping, digital signature, digital certificates etc., are also used for security purpose. In this paper, we have compared four Public-Key cryptosystem i.e. RSA, Diffie-Hellman Key Exchange, Elgamal Cryptographic System, and Elliptical Curve Cryptography. This paper performs security analysis of the above public key cryptosystem and concluded that ECC is the most efficient public-key cryptosystem. It provides high security solutions that do not impact performance even on constrained devices where storage, computing power and bandwidth are limited such as PDAs and cell phones.

Keywords:- Cryptography, Encryption, Decryption, Asymmetric Encryption , Symmetric Encryption.

I. INTRODUCTION

Cryptography is the process of encoding messages to make them non-readable for achieving security. In modern times, cryptography is considered to be a branch of both mathematics and computer science and is closely associated with information theory, computer security and engineering. Cryptography is used for the security of ATM cards, computer passwords and electronic commerce [14].

A cryptosystem is a five-tuple (P, C, K, E, D), satisfying the following conditions: (1) P is a finite set of possible plain text (2) C is a finite set of possible ciphertexts (3) K, the keyspace, is a finite set of possible keys (4) For each K ε k, there is an encryption rule eK ε E. and a corresponding decryption rule dK ε D. Each eK: P →C and dK: C → P are functions such that dK(eK(x)) = x for every plaintext x ε P. The property 4 says that if a plaintext x is encrypted using eK, and the resulting cipher text is subsequently decrypted using dK, then the original plaintext x results[15][19].

The concept of Public Key Cryptography (PKC) was introduced by Whitfield Diffie and Martin Hellman in 1976. After that many implementations of it have been proposed, and many of these cryptographic applications base their security on the intractability of hard mathematical problems, namely the finite field Discrete Logarithm Problem (DLP) and Integer Factorization Problem (IFP). To solve these problems, sub-exponential time algorithms have been developed over the years. As a result, key sizes grew to more than 1000 bits, so as to attain a reasonable level of security. In the environments where bandwidth, computing power and storage are limited, carrying out thousand-bit operations becomes an unrealistic approach for providing ample security. This is most evident in hand-held devices such as the mobile phones, PDAs and pagers that have very limited processing power and battery life[20][15].

The concept of PKC evolved from an attempt to attack two of the most difficult problem associated with symmetric encryption. The first problem is that of key distribution under symmetric encryption requires either: (1) that two communicants already share a key, which has been distributed somehow to them; or (2) the use of key distribution center. The public key cryptography process is described in Figure 1. From Figure 1, it is evident that asymmetric algorithms rely on one key for encryption and a different but mathematically related key for decryption. These algorithms have the following important characteristics. (a) It is computationally infeasible to determine the decryption key on the basis of knowledge of the encryption key and cryptographic algorithm. (b) Either of the two related keys can be used for encryption with the other used for decryption [3][14].

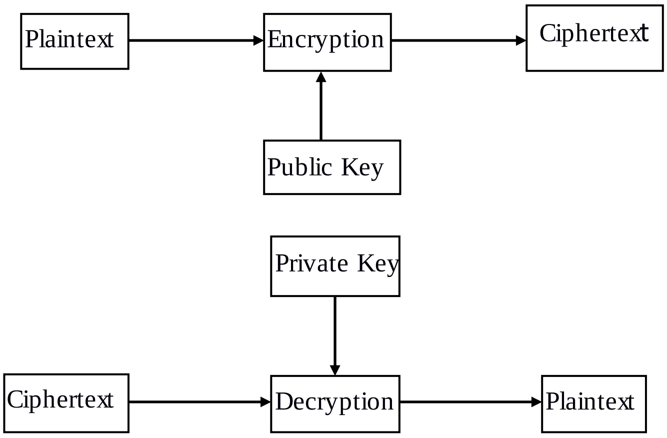

Figure 1: Public Key Cryptography Process

A secure public-key-cryptosystems depends on a cryptographic algorithm based on two related keys. The prime requirement of public-key-cryptosystem is the trap-door one-way function. A trap-door one-way function is easy to calculate in one direction and infeasible to calculate in other direction unless the certain additional information is known. With the additional information, the inverse can be calculated in polynomial time. A trap-door one-way function is a family of invertible function fk, such that [14]:

Y = fk(X) easy, if k and X are known
X = fk-1(Y) easy, if k and Y are known
X = fk-1(Y) infeasible, if Y is known, but k is not known

Thus to give the practical dimension to public-key cryptosystem depends on discovery of suitable trap-door one-way function. Public-key systems are characterized by the use of the cryptographic algorithm with two keys, one private and one public. Depending on the application, the sender's private key or the receiver's public key, or both are used by the sender, to perform some type of cryptographic function. The use of public-key cryptosystems can be classified into three categories[14]:

- Encryption/Decryption: The sender encrypts a message with the recipient's public key
- Digital signature: The sender "signs" a message with its private key. Signing is achieved by a cryptographic algorithm applied to the message or to a small block of data that is a function of the message
- Key exchange: Two sides cooperate to exchange a session key. Several different approaches are possible, involving the private key(s) of one or both parties

II. ASYMMETRIC KEY ALGORITHMS

A. RSA

It was developed in 1977 by Ron Rivest, Adi Shamir, and Adleman at MIT and first published in 1978. 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}. RSA makes use of an expression with exponentials. Plaintext is encrypted in blocks, with each block having binary value less than some number n. Encryption and decryption are of the following form, for some plaintext block M and ciphertext block C[14][18].

For Encryption

$C = M^e \bmod n$

For Decryption

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

Algorithm

1. Choose two large prime numbers p and q
2. Compute n = p * q
3. Choose the public key e such that gcd(ϕ(n),e) = 1; 1<e< ϕ(n)
4. Select the private key d such that d*e mod ϕ(n) = 1
5. Public key is (n, e) and Private key is (n, d)

B. Diffie-Hellman Key Exchange

The purpose of the algorithm is to enable two users to securely exchange a key that can then be used for subsequent encryption of messages. The Diffie-Hellman algorithm depends for its effectiveness on the difficulty of computing discrete logarithms. For this scheme, there are two publically known numbers: a prime number q and an integer α that is a primitive root of q. suppose the users A and B wish to exchange a key. User A selects a random integer $X_A < q$ and computes $Y_A = \alpha^{X_A} \bmod q$. Similarly, user B independently selects a random integer $X_B < q$ and computes $Y_B = \alpha^{X_B} \bmod q$. Each side keeps the X value private and makes the Y value available publicly to the other side. User A computes the key as $K = (Y_B)^{X_A} \bmod q$ and user B computes the key as $K = (Y_A)^{X_B} \bmod q$. By the rules of modular arithmetic these two calculations produce the same result [14].

$K = (Y_B)^{X_A} \bmod q$
$= (\alpha^{X_B} \bmod q)^{X_A} \bmod q$

$= (\alpha^{X_B})^{X_A} \bmod q$
$= \alpha^{X_B X_A} \bmod q$
$= (\alpha^{X_A})^{X_B} \bmod q$
$= (\alpha^{X_A} \bmod q)^{X_B} \bmod q$
$= (Y_A)^{X_B} \bmod q$

The result is that the two sides have exchanged a secret value. As X_A and X_B are private, an adversary only has the following ingredients to work with: q, α, Y_A, and Y_B. Thus, the adversary is forced to take a discrete logarithm to determine the key.

C. Elgamal Cryptographic System

In 1984, T. Elgamal announced a public key scheme based on discrete logarithms. The Elgamal cryptosystem is used in Digital Signature Standard (DSS). The global elements of Elgamal are a prime number q and α, which is a primitive root of q. User A generates a private/public key pair as follows[14]:

1. Generate a random integer $X_{A,}$ such that $1< X_A < q – 1$
2. Compute $Y^A = \alpha^{X_A} \bmod q$
3. A's private key is X_A: A's public key is {q, α, Y_A }

Any user B that has access to A's public key can encrypt a message as follows:

1. Represent the message as an integer M in the range 0 <= M <= q -1.
2. Choose a random integer k such that 1 <= k <= q – 1.
3. Compute a one-time key $K = (Y_A)^K \bmod q$.
4. Encrypt M as the pair of integers (C_1, C_2) where
$C_1 = \alpha^k \bmod q$; $C_2 = KM \bmod q$

User A recovers the plaintext as follows:

1. Recover the key by computing $K = (C_1)^{X_A} \bmod q$
2. Compute $M = (C_2 K^{-1}) \bmod q$

D. Elliptic Curve Cryptography (ECC)

The idea of using Elliptic curves in cryptography was introduced by Victor Miller and N. Koblitz as an alternative to established public-key systems such as DSA and RSA. An elliptic curve E(F p) over a finite field F p is defined by the parameters a, b ∈ Fp (a, b satisfy the relation $4a^3 + 27b^2 \neq 0$), consists of the set of points (x, y) ∈ Fp, satisfying the equation $y^2 = x^3 + a\ x + b$. The set of points on E(F p) also include point O, which is the point at infinity and which is the identity element under addition . Elliptic Curve Encryption/Decryption algorithm can be described as[14][19][17]:

1. Consider a message 'P_m' sent from A to B. 'A' chooses a random positive integer 'k' , a private key 'n_A' and generates the public key $P_A = n_A \times G$ and produces the ciphertext 'C_m' consisting of pair of points C_m={ kG , $P_m + kP_B$ } where G is the base point selected on the Elliptic Curve, $P_B = n_B \times G$ is the public key of B with private key 'n_B' .
2. To decrypt the ciphertext, B multiplies the 1st point in the pair by B's secret & subtracts the result from the 2nd point : $P_m + kP_B - n_B(kG) = P_m + k(n_B\ G) – n_B(kG)=P_m$

III. LITERATURE REVIEW

Kute et al. [7] discussed the performance attribute of public key cryptosystems. The algorithms studied and compared are RSA, ECC. Algorithms were implemented in Java in order to perform software tests to gain insight into the relative performance of each algorithm. Each algorithm is tested for key generation and encryption/decryption of ordinary but large files. From the implementation of RSA and ECC algorithms, it was concluded that operations in ECC are comparatively slower than RSA. Key generation and encryption are faster in RSA, whereas decryption is slower. On the other hand, key generation and encryption are slower in ECC, whereas the decryption is faster. It was concluded that RSA is faster, but it is said that security wise ECC is stronger than RSA.

Kumar et al. [6] proposed a new encryption algorithm using Elliptic Curve over finite fields. In the proposed encryption algorithm, the communicating parties agree upon to use an elliptic curve and a point C on the elliptic curve. The security of the Elliptic Curve Cryptography depends on the difficulty of finding the value of k, given kP where k is a large number and P is a random point on the elliptic curve. This is the Elliptic Curve Discrete Logarithmic Problem. The elliptic curve parameters for cryptographic schemes should be carefully chosen in order to resist all known attacks of Elliptic Curve Discrete Logarithmic Problem (ECDLP). Hence, the method of encryption proposed here provides sufficient security against cryptanalysis at relatively low computational overhead.

Shankar et al. [12] discussed the basic idea of ECC and its implementation through co-ordinate geometry for data encryption. An overview of ECC implementation on two-dimensional representations of plaintext coordinate systems and data encryption through Elgamal Encryption technique has been discussed. In this study brief overview of elliptic curve cryptography was provided and developed an alphabetical table for ECC data encryption and decryption in a suitable manner. The strength of encryption depends on its key and if the alphabetical table is used then there will be no impact on strength and runtime performance. Runtime will be faster by this process, i.e. the use of alphabetical table will provide better performance in this regard. Moreover, Public key is used for message encryption in the case of socket layer security. It is clearly evident from the above that the alphabetical table described here can be used as a reference to build elliptic curve cryptography software for providing socket layer security.

Kumar et al. [5] gave an introduction to the public key cryptography and its use in applications such as Key Agreement, Data Encryption and Digital Signature. The main emphasize is on some public key algorithms such as RSA and ECC along with the idea how ECC is better and more secure method of encryption in comparison to RSA and other asymmetric cryptosystems. It was concluded that ECC is a stronger option than the RSA and discrete logarithm systems in the future. Thus, it can be said that ECC is an excellent choice for doing asymmetric cryptography in portable, necessarily constrained devices right now. As an example: a popular, recommended RSA key size for most applications is 2,048 bits. For equivalent security using ECC, a key of only 224 bits is needed. In short, to make a device with a smaller band, make them run longer on the same battery, and produce less heat, the most elegant and most efficient asymmetric cryptosystem that scales for the future is ECC.

Pallipamu et al. [10] performed a security analysis of Digital Signature schemes; RSA, DSA and ElGamal. In this paper the concept of Cryptography including the various digital signature schemes of a system based on the kind of key and a few algorithms such as RSA, DSA and ECDSA. The mathematical foundations of various algorithms for generation of keys and verification of digital signatures and also their security strengths were analyzed.

Yadav et al. [16] discussed the concept and implementation of RSA algorithm for security purpose and to enhance the performance of software system using this algorithm. This study includes what is RSA algorithm and why they are used in the field of Cryptography & Network Security. It was concluded that RSA algorithm is important to Network Security because they are the components (i.e. Encryption & Decryption key) which interact with the Security system. Without them the system will be useless as RSA are used to fire a particular Encryption & Decryption keys process because of which Security system is build. The attacks made against the underlying structure of the RSA algorithm, which exploit weaknesses in the choice of values for the encryption and decryption keys, and their relation to the RSA modulus N were described.

Shanmugalakshmi et al. [13] made a comparison between ECC and other cryptography algorithms and explained their role in the network security. ECC's uses with smaller keys to provide high security, high speed in a low bandwidth. In this paper a comparative study between ECC and RSA, ECC's advantages and some application of ECC like ECDSA was made. A detailed study of ECDSA was done for our verification. The security, performance and future enhancement of ECC were discussed.

Alese et al. [1] made a comparative analysis of ElGamal Elliptic Curve Encryption algorithm, RSA Encryption algorithm, and Menezes-Vanstone Elliptic Curve Encryption algorithm. These elliptic curve which are analogues of ElGamal Encryption scheme were implemented in Java, using classes from the Flexiprovider library of ECC. The RSA algorithm used in the comparison is the Flexiprovider implementation. The performance evaluation of the three algorithms based on the encryption and decryption algorithms, time lapse for their Key generation and encrypted data size was compared. The results showed that elliptic curve-based implementations are more superior to the RSA algorithm on all parameters which are used for evaluation. The future of ECC looks brighter than that of RSA as today's applications (smart cards, pagers, and cellular telephones etc) cannot afford the overheads introduced by RSA. Finally, both systems can be considered as good given the low success rate associated with attacking them.

Giripunje et al. [4] provides effective security solution using Public key cryptography. Secure Access Authentication in mobile communication is very crucial to protect information of the subscribers and avoid fraud. This paper studied security by means of elliptic curve cryptographic technique. The actual implementation of ECC on GF (P) shows that security of the proposed system is very hard. It has been mentioned by various researchers that a considerably smaller key size can be used for ECC compared to RSA. Also, ECC requires a low calculation power as the mathematical calculations in ECC are easier. Therefore, ECC is a more appropriate cryptosystem to be used on small devices like mobile phones.

Markan et al. [9] described that ECC schemes are public-key based mechanisms that provide digital signatures, encryption, and key exchange algorithms. The best-known encryption scheme is the Elliptic Curve Integrated Encryption Scheme (ECIES), which is included in IEEE and also in SECG SEC1 standards. Wireless devices are rapidly becoming more dependent on security features such as the ability to do secure email, secure Web browsing, and virtual private networking to corporate networks, and ECC allows more efficient implementation of all of these features. ECC provides greater security and more efficient performance than the first generation public key techniques. Even though ECIES provides some valuable advantages over other cryptosystems as RSA, the number of slightly different versions of ECIES included in the standards may obstruct the adoption of ECIES.

Chandel et al. [2] in his research paper concentrates on the different kinds of encryption techniques that are existing. It is a literature survey of some modern cryptography techniques. The advantages and disadvantages of the methods are also discussed in brief. It also aims image encryption techniques, double encryption, information encryption techniques and Chaos-based encryption techniques. In this paper, the existing encryption techniques are studied and analyzed. All the techniques are useful for real-time encryption. There advantages and disadvantages are discussed. It is found that a faster version of RSA may be proposed. Also, there is a need to remove wieners and some other similar attacks.

Lamba et al. [8] studied and analyzed various enhancement schemes in the basic Diffie- Hellman algorithm. In order to provide more security to Diffie-Hellman algorithm, different approaches have been followed till date. One is the mechanism of group keys in which only the group members know the secret key. Another approach is the one in which the key size has been increased. The comparison between the two DH algorithms was done according to the key size generated, which shows that the generated key size from the modified method is greater than the classical one. Attacking the methods shows that the time needed to compute the private key for the modified algorithm is greater than the classical one. Therefore, the modified method is more secure as more time is needed to crack the key. The more recent approach used is the inclusion of a mathematical function to make the key harder and the time required to crack the key here is even more than previous approaches, therefore, is more secure.

IV. NEED OF STUDY

Encryption of data is mostly done by Private Key Cryptography because of its speed. Advanced Encryption Standard (AES) which has tremendously fast encryption speed and efficiency is the most widely used today. But AES is unsuitable for the mobile commerce environment applications due to certain shortcomings [20]:

Key Management Problem: The wireless user must be capable of doing business with a large number of different enterprises, instead of only one. Therefore, communication on a public network is done by many users interacting with each other, rather than confining it to one-on-one. If n is small, n(n-1)/2 private keys are to be generated for a network of n-users and if n is large, the number of private keys become unmanageable [Koblitz 2008]. When one has to generate such a large number of keys, the task of generating the keys and finding a secured distribution channel becomes a difficult task on networks.

No Digital Signatures Possible: An electric analogue for a hand written signature is digital signature. If Bob got an encrypted message from Alice, then Bob must be able to make sure that the received message is from Alice, with the help of Alice's signature. This capability is not a feature of Private Key Cryptography.

In direct contrast, two keys are used by Public Key Cryptography (PKC). On a network, every user puts out a public encryption key that anyone can use to send messages; however, the private key is kept secret for decryption. PKC requires n private and n public keys on a network of n-users. This decreases the number of keys required from O(n2) to O(n). Moreover, PKC admits the use of digital signatures, which guarantees non-reunification. Yet Public Key Cryptography is much slower compared to Private Key Cryptography. DES requires just 64-bits while RSA, the most widely user public key algorithm that supports encryption and digital signatures, requires at least 1024-bit keys [11].

V. OBJECTIVES OF THE STUDY

The objective of the study is to analyze the various Asymmetric encryption algorithms: RSA, Elliptic Curve, Diffie-Hellman, and DSA. However the specific objectives are:

1. To have a deeper understanding of cryptography.
2. To perform a comparative analysis of asymmetric encryption algorithms of cryptography.

VI. ANALYSIS

Cryptography provides the information security services of confidentiality, authentication, integrity, and no-repudiation. Confidentiality is provided by private key cryptography (Symmetric Key Cryptography or SKC) by the encryption and decryption operations. The four security services can be provided by Public Key Cryptography (PKC), but this kind of cryptography is mainly used to provide the authentication and no-repudiation services by implementing the concept of digital signatures. The hash functions are cryptographic primitives often used along with public or private key algorithms to provide the integrity service. Examples of hash functions, public key and private key algorithms are given in Table 1:

Table 1: Different Kind of Cryptographic Algorithms

Kind of Cryptography	Examples
Hash Functions	MD4-5, SHA-0-1-2, RIPDEM
Symmetric Key Cryptography	DES, AES, 3DES, RC4
Public Key Cryptography	ECC, RSA, DSA, ElGammal

SKC and PKC algorithms rely on the use of a key or a pair of keys. A key is a n-bit string that is used to transform data. The size in bits of the key is an important security parameter in the cryptographic algorithms. Table 2 shows the key sizes for different SKC and PKC cryptographic algorithms with equivalent security level.

Table 2: Key Sizes for Cryptographic Algorithms [21]

Private key size (bits)	Public Key Size (bits)		MIPS To attack	Protection Lifetime
	ECC	RSA/DH/DSA		
80	160	1024	10^{12}	Until 2010
112	224	2048	10^{24}	Until 2030
128	256	3072	10^{28}	Beyond 2031
192	384	7680	10^{47}	--
256	512	15360	10^{66}	--

In PKC, private and public keys have a mathematical relation f, but the private key cannot be obtained from the public one. In order to recover the private key to decrypt data or to sign documents, a mathematical problem P related to f must be solved. The security of public key cryptosystems depends on the difficulty to solve P. In practice, three problems have been considered to be difficult to solve and are used for cryptographic applications. Table 3 lists these problems and the cryptosystems that rely their security on such problems.

Table 3: Public Key Cryptosystems and their Underlying Mathematical Problems

Cryptosystems	Mathematical Problem	Description
RSA, Rabin-Williams	Integer factorization	Given a number n, find its prime factors
ElGammal, DSA, Hellman-Diffie	Discrete logarithm	Given a prime n, and numbers g and h, find x such that h = g^x mod n
ECDSA, EC-Diffie-Hellman	Elliptic curve discrete logarithm	Given an elliptic curve E and points P and Q on E, find x such that Q = x.P

Computational complexities for each of these mathematical problems is described in Table 4, where n is the size of the keys used. The sub-exponential complexity of the problem on which RSA and other public key methods base their security means that the problems can be considered hard to solve but not as hard as problems that only allow fully exponential solutions, as elliptic curve cryptography. Because of this, ECC can offer a similar security level than other public key cryptosystems but using shorter length keys, which implies less space for key storage, time saving when keys are transmitted and less costly modular computations. These characteristics make ECC the best choice for securing devices with constrained resources, like the mobile ones.

Table 4: Complexity of Mathematical Problems in Public Key Cryptography

Mathematical Problem	Best known methods for solving mathematical problem	Running times
Integer factorization	Number field sieve: $e^{1.923}(\log n)^{1/3} (\log \log n)^{2/3}$	Sub-exponential
Discrete logarithm	Number field sieve: $e^{1.923}(\log n)^{1/3} (\log \log n)^{2/3}$	Sub-exponential
Elliptic curve discrete logarithm	Pollard-rho algorithm: $\sqrt{n}$	Fully exponential

Another important point of analysis is whether public-key encryption is more secure from cryptanalysis than symmetric encryption. In fact, the security of any encryption scheme depends on the length of key and the computational work involved in breaking a cipher. There is nothing in principle about either symmetric or public-key encryption that makes one superior to another from the point of view of resisting cryptanalysis. The following table compare between the Elliptic Curve Cryptography and RSA, A 160 bit key in ECC is considered to be secured as 1024 bit key in RSA. The key size relationship between the ECC and the RSA, and the appropriate choice of the AES key size is as given in Table 5:

Table 5: ECC and RSA Key Comparison [17]

ECC Key Size(bits)	RCC Key Size(bits)	Key Size Ratio (ECC/RCC)	AES Key Size(bits)
163	1024	1:6	NA
256	3072	1:12	128
384	7680	1:20	192
512	15360	1:30	256

Some public-key algorithms are suitable for all the three applications (i.e. Encryption/Decryption, Digital Signature, Key Exchange) whereas others can be used only for one or two of these applications. Table 6 describes the applications supported by the algorithms discussed in this paper.

Table 6: Applications for Public-Key Cryptosystem

Algorithm	RSA	Elliptic Curve	Diffie-Hellman	DSS
Encryption/Decryption	Yes	Yes	No	No
Digital Signature	Yes	Yes	No	Yes
Key Exchange	Yes	Yes	Yes	No

From Table 6 it can be analyzed that RSA and Elliptic Curve algorithm are suitable for all the three applications. It was also analyzed that the Elliptic Curve Cryptography takes more computation but reduced key size in the term of encryption/decryption of messages.

VII. CONCLUSIONS

It is concluded from above study that, Asymmetric key cryptosystem is more secure than Symmetric key cryptosystem. Public-key encryption scheme is also vulnerable to a brute-force attack. Large key sizes that have been proposed do make brute-force attack impractical, but result in encryption/decryption speeds that are too slow for general–purpose use. Also public-key system depends on some sort of invertible mathematical function. The complexity of calculating these functions may not scale linearly with the number of bits in the key but grow more rapidly than that. Due to above facts, public-key encryption is currently confined to key management and signature applications. Only a few algorithms (RSA, Elliptic curve, Diffie-Hellman, DSS) have received widespread acceptance in the several decades since the concept of public-key cryptography was proposed. In truth, Public and Private Key Cryptography work best together. Private Key Cryptography is best suited for ensuring confidentiality like encrypting data and communication channels whereas Public Key Cryptography is most suited for ensuring data integrity, key distribution and management, providing authentication and non-repudiations which are the most important objectives that play a vital role in any cryptographic application.

It is also found that, the key length for secure RSA use has increased over recent years, and thus put a heavier processing load on applications using RSA. This burden has ramifications, especially for electronic commerce sites that conduct large number of secure transactions. A competing system challenges RSA: elliptic curve cryptography (ECC). The principal attraction of ECC compared to RSA, is that it appears to offer equal security for a smaller key size, thereby reducing processing overhead. Thus there is a computational advantage to using ECC with a shorter key length than a comparably secure RSA. Although the theory of ECC has been around for some time, it is only recently that products have begun to appear and that there has been sustained cryptanalytic interest in probing for weaknesses. However, the confidence level in ECC is not yet as high as that in RSA. ECC is fundamentally more difficult to explain than either RSA or other public-key cryptosystem.

Most of the studies currently made have shown that ECC is the most convenient cryptosystem for the smartcards. Saving time, cost and area are the reasons behind this for smart cards. On the other hand, the fact that the elliptic curve cryptosystem implementation is much more complicated and requires deeper mathematical understanding than the other cryptography implementation (for example RSA), makes it more susceptible to errors. Certainly, ECC systems solved some major problems which exist in others. The future scope of our study is to study the mathematical theory of elliptic curves in detail, and make it more practicable to be used in designing secure Public-Key Cryptosystem.

REFERENCES

[1]. 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.

[2]. Chandel Gagendra Singh, Singh Prabhat Kumar , “A Literature Review of Various Variants of RSA Cryptosystem”, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 2, February 2014.

[3]. Forouzan Behrouz A., “Data Communications & Networking”, Fourth Edition, 2008, New York: Tata McGraw- Hill.

[4]. Giripunje Lokesh, Nimbhorkar Sonali, “Comprehensive Security System for Mobile Network Using Elliptic Curve Cryptography over GF (p)”, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 5, May 2013.

[5]. Kumar Arun, Tyagi Dr. S.S., Rana Manisha, Aggarwal Neha, Bhadana Pawan, “A Comparative Study of Public Key Cryptosystem based on ECC and RSA”, International Journal on Computer Science and Engineering (IJCSE), Volume 3, No. 5, May 2011.

[6]. Kumar D.Sravana , Suneetha CH., Chandrasekhar A., “Encryption Of Data Using Elliptic Curve Over Finite Fields”, International Journal of Distributed and Parallel Systems (IJDPS), Volume 3, No.1, January 2012.

[7]. Kute vivek B., Paradhi P.R., Bamnote G.R., “ A Software Comparison of RSA and ECC” ,

International Journal of Computer Science And Applications, Volume 2, No. 1, April/May 2009.

[8]. Lamba Ekta, Garg Lalit, "Review on Diffie Hellman Algorithm", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 3, March 2014.

[9]. Markan Ruchika , Kaur Gurvinder, "Literature Survey on Elliptic Curve Encryption Techniques", International Journal of Advanced Research in Computer and Communication Engineering, Volume 3, Issue 9, September 2013.

[10]. Pallipamu Venkateswara Rao, K Thammi Reddy, P Suresh Varma , "A Survey On Digital Signatures", International Journal of Advanced Research in Computer and Communication Engineering, Volume 3, Issue 6, June 2014.

[11]. Schneier B., "Applied Cryptography", John Wiley & Sons Publication, New York, 1994.

[12]. Shankar Tarun Narayan, sahoo G., "Cryptography with Elliptic Curves", International Journal of Computer Science And Applications, Volume 2, No. 1, April/May 2009.

[13]. Shanmugalakshmi Dr.R., Prabu M., "Research Issues on Elliptic Curve Cryptography and Its applications", International Journal of Computer Science and Network Security, Volume 9, No.6, June 2009.

[14]. Stallings William, "Cryptography and Network Security Principles and Practice", Fifth Edition, Pearson Education, Prentice Hall, 2011.

[15]. Tanenbaum Andrew S., "Computer Networks", Third Edition, Prentice Hall India, 2000.

[16]. Yadav Prasant Singh, Sharma Pankaj, Yadav Dr K. P, "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.

WEB REFERENCES

[1]. http://www.certicom.com/index.php?action=ecc, Accessed on 12/09/2015 at 11:00 A.M.

[2]. http://en.wikipedia.org/wiki/RSA, Aaccessed on 18/12/2014 at 08:00 P.M.

[3]. http://en.wikipedia.org/wiki/Elliptic_curve_cryptography, Accessed on 15/06/2015 at 9:00 P.M.

[4]. http://eprint.iacr.org/2008/390.pdf, Accessed on 25/08/2015 at 7:30 A.M.

[5]. http://www.nist.gov/information-technology-portal.cfm, Accessed on 28/03/2015 at 4:30 A.M.