Available online www.ejaet.com

European Journal of Advances in Engineering and Technology, 2014, 1(2): 93-101

Research Article

ISSN: 2394 - 658X

Elliptic Curve Cryptography (ECC) for Security in Mobile Communication

M Shanmugasundaram[1] R Shanmugasundaram[2]

[1]Department of MCA, Siddaganga Institute of Technology, Tumkur, India
[2]Department of Computer Science, Erode Arts and Science College, India
shanmuga_maran@yahoo.com

ABSTRACT

Mobile phones are most common way of communication and accessing Internet based services. Sending and receiving sensitive data are not only used for proper communication at mobile phones. However, the security of mobile communication has topped the list of concerns for mobile phone users. So Public key cryptography is effective security solution to provide secure the mobile communications. In this paper provides a ECC module to secure data encryption and decryption using public key cryptography. The form of key exchange, communication privacy through encryption, authentication of sender and digital signature to ensure message integrity are implementation of ECC module.

Key words: Public key cryptography, ECC (Elliptic curve cryptography), key agreement, confidentiality, authentication, integrity, security services, digital signature

---

INTRODUCTION

Elliptic curve cryptography was introduced by Victor Miller and Neal Koblitz in 1985. The popularity of elliptic curve cryptography is due to the determination that is based in a harder mathematical problem than other cryptosystem. It is an alternative to the conventional public key cryptosystem such as RSA, DSA. ECC offers the same level of security with smaller key size and it leads to the better performance in limited environments like mobile phones, sensor networking, and smart cards. Ex: ECC with a key size of 160 bits provides the same level of security as RSA with a key size of 1024 bits. The key agreement, signature generation, signing and verification involve scalar multiplication are main operations of elliptic curve. The scalar multiplication plays an important role in the efficiency of whole system. The Fast multiplication is very essential in some environments such as constrained devices, central servers, where large number of key agreements, signature generations and verification occurs. The mathematical operations of ECC is defined over the elliptic curve $y^2 = x^3 + ax + b$, where $4a^3+27b^2=0$. Each value of the 'a' and 'b' gives a different elliptic curve. The private key is a random number and the public key is a point in the curve. By multiplying the private key with the generator point G in the curve of public key is obtained. The generator point G, the curve parameters 'a' and 'b' together with few more constants constitute the domain parameter of ECC. Key exchange in elliptic curve cryptography based on the diffie-hellman key exchange. The encryption and decryption techniques, digital signature generated are based on ECC algorithm.

CRYPTOGRAPHIC TERMINOLOGY

William Stallings [9] provide a detailed description of commonly employed security concepts and terminology. The required security objectives practice is addressed by choosing a security protocol. Security protocols were realizing the security objectives through the use of appropriate cryptographic algorithms. Basic Security Terminologies used in cryptography are:

A message present in a clear form of plaintext. The encryption process converts the plaintext to a form that hides the meaning of the message, and the result is known as the cipher text. Decryption is the opposite of encryption. Valid users can be processes of encryption and decryption are controlled on a quantity known as the key. Strength of a security scheme depends on the secrecy of the keys used [9]. A security protocol formally specifies a number of steps to be followed by communicating practices, so that the mutually desired security objectives are satisfied. The four main security objectives include:

Confidentiality
The secrecy of the data being exchanged by the communicating parties is maintained, i.e., no one other than the valid parties should know the content of the data being exchanged.

Authentication
It should be feasible for the receiver to ensure that the sender of the message is who he claims to be, and the message was sent by him.

Integrity
It provides a means for the receiver of a message to prove that the message was not changed in transit. It checks originality of message.

Non-repudiation
The sender of a message should unable to falsely deny later that he send the message, and this fact should be verifiable independently by an independent third-party without knowing too much about the content of the distributed message(s). Security objectives thus provide hope on the Web. They are realized through the use of cryptographic algorithms which are divided into two categories depending on their characteristics. These categories are:

a) **Symmetric Algorithms**
These algorithms use the same key for encryption and decryption. They rely on the concepts of "confusion and diffusion" to realize their cryptographic properties and are used mainly for confidentiality purposes also known as secret key cryptosystems.

b) Asymmetric algorithms
These algorithms use different keys, known as the public key and the private key, for encryption and decryption, respectively. They are constructed from the mathematical abstractions which are based on computationally intractable number-theoretic problems like integer factorization, discrete logarithm, etc. They are primarily used for authentication and non-repudiation [9].Also known as public key Cryptosystems (PKC).

PRINCIPLES OF PUBLIC KEY CRYPTOSYSTEMS

Pair of keys for an encryption and decryption is the basic reason that differentiates public key Cryptosystems (PKC) from secret key cryptosystems. The Public key Cryptosystems (PKC), public key is known to all, but private key is kept confidential, and because it is without a solution to calculate the public key even if private key is known but same value is possible. Public key algorithms use different keys for signing and decryption, and for encryption and signature verification. Private Key may only be known to its holder and should be kept in secret. It may be used for generation of digital signatures or for decrypting private information encrypted with the public key. The public key may be used for verifying digital signatures or for encrypting information. Public key algorithms have a big advantage when used for ensuring privacy of communication. The sender A want to send plaintext message m to designation B, he calculate cipher text c by performing encryption on message (m) to obtain cipher text (c), then transmits cipher text (c) to designation B. When designation B gets message (c), he calculates original plaintext message m by performing decryption on message (c) [3] [9].

Table -1 Applications of Public Key Cryptosystem

Algorithm	Encryption/ decryption	Digital Signature	Key Exchange
RSA	Yes	Yes	Yes
Elliptic curve	Yes	Yes	Yes
Diffie- Hellman	No	No	Yes
DSS	No	Yes	No

Public key algorithms are ECC, RSA, DSA and Diffie- Hellman key exchange. In this paper mainly focus on Elliptic Curve Cryptography (ECC).

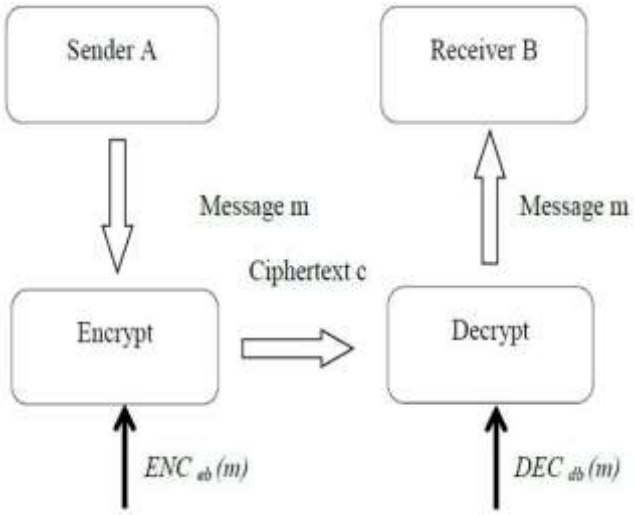

Fig. 1 Encryption/Decryption with public key Cryptosystems (PKC)

ELLIPTIC CURVES IN CRYPTOGRAPHY

History of ECC

Elliptic curves were proposed for use as the basis for discrete logarithm-based cryptosystems almost 20 years ago, independently by Victor Miller of IBM and Neal Koblitz of the University of Washington [2, 3]. At that time, elliptic curves were already being used in various cryptographic contexts, such as integer factorization and primarily proving.

Elliptic Curve Groups

For the reason of cryptography, an elliptic curve can be thought of as being given by, an equation form

$$y^2 = x^3 + ax + b \tag{1}$$

Where *a* and *b* are elements of a set field with *pn* elements, where *p* is a prime larger than 3. (The equation over binary and ternary fields looks slightly different.) The set of points on the curve is the collection of ordered pairs (x, y) with coordinates in the field and such that x and y satisfy the relation given by the equation defining` the curve, plus an extra point that is said to be *at infinity*. The set of points on an elliptic curve with coordinates in a finite field also form a group, and the operation is as follow: To add two points on the curve Q_1 and Q_2 together, pass a straight line through them and look for the third point of intersection with the curve, R_1. Then reflect the point R_1 over the x-axis to get –R1, the sum of Q_1 and Q_2.

Thus, $$Q_1 + Q_2 = -R_1 \tag{2}$$

The idea behind this group operation is that the three points Q_1, Q_2, and R_1 lie on a common straight line, and the points that form the intersection of a function with the curve are considered to add up to be zero.

Fig. 2 shows the addition of two points on an elliptic curve. The elliptic curves have the interesting property that adding two points on the elliptic curve results a third point on the curve. Hence, adding two points, P_1 and P_2, gets us to point P_3, as well on the curve. The small changes in P_1 or P_2 can cause a large change in the position of P_3.The point addition is the addition of two points P_1 and P_2 on an elliptic curve to obtain another point p3 on the same elliptic curve as shown in fig. 4 and point doubling Point doubling is the addition of a point J on the elliptic curve to itself to obtain another point L as shown in fig. 2.

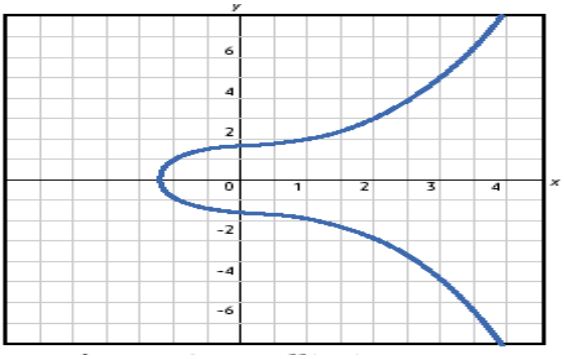

Fig. 2 basic elliptic curve

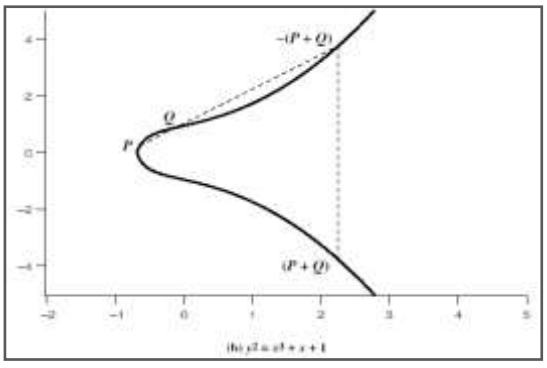

Fig. 3 Group laws of Elliptic Curve (point addition)

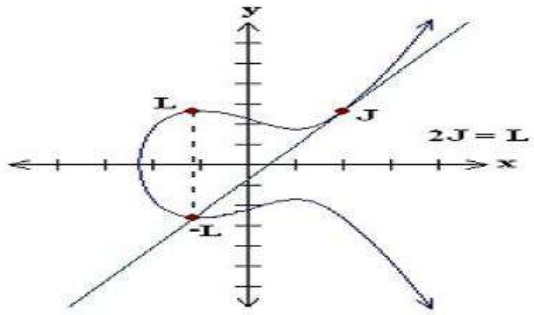

Fig. 4 Group laws of Elliptic Curve (point doubling)

GROUP LAW ON ELLIPTIC CURVES

In this part we recall the additive operation on points belonging to an elliptic curve (E). Suppose that the equation of (1) is converted into

$$y^2 = x^3 + ax + b \quad P \tag{3}$$

where p is a prime integer and a,b, £{1,2,….p-1}.

Let $p(x_1, y_1)$ and $Q(x_1, y_1)$ are two points on the curve (E) and an imaginary point at infinity.

if $x_1 \neq x_2$, then

$$x_3 \equiv m^2 - x_1 - x_2 \pmod{p},$$
$$m \equiv y_2 - y_1 / x_2 - x_1$$
$$y_3 \equiv m(x_1 - x_3) - y_1 \pmod{p} \tag{4}$$

if $x_1 = x_2$ and $y_1 = y_2$ then R = 0. (5)

if $p = Q$ and $y_1 = 0$ then R = 0. (6)

if $p = Q$ and $y_1 = 0$ then

$$x_3 \equiv m^2 - 2x_1,$$
$$m = 3x_1^2 + a / 2y_1$$
$$y_3 \equiv m(x_1 - x_3) - y_1 \tag{7}$$

With this stabilizer law, the elliptic curve becomes a finite field Abelian group.

ELLIPTIC CURVE CRYPTOGRAPHY SIMULATING ELGAMAL

A number of methods have been used to encrypt and decrypt using elliptic curves. The common one is to simulate the ElGamal cryptosystem using an elliptic curve over $GF(p)$ or $GF(n)$. Operations such as addition and multiplication are more an elliptic curve group.

Generating Public and Private Keys

1. Bob chooses $E(a,b)$ with an elliptic curve over $GF(p)$.
2. Bob chooses a generator point, $e_1(x_1, y_1)$ on the curve.
3. Bob chooses an integer d.
4. Bob calculates $e_2(x_2, y_2) = d \times e_1(x_1, y_1)$. Multiplication here means multiple additions of points.
5. Bob announces $E(a,b)$, $e_1(x_1, y_1)$ and $e_1(x_2, y_2)$ as his public key; he keeps d as his private key.
6. Similar process is carried out for User B.
7. Finally the session/secret key is generated with the help of Diffie- Hellman key exchange.

Point Addition

Consider two distinct points P1 and p2 such that

$$p_1 = (x_1, y_1) \text{ and } p_2 = (x_2, y_2)$$

(8)

Let $p_3 = p_1 + p_2$ 9)

Where $p_3 = (x_3, y_3)$ (10)

then

$$x_3 = m^2 - x_1 - x_2 \bmod p,$$
$$y_3 = m(x_1 - x_3) - y_1 \bmod p,$$
$$m = (y_2 - y_1)/(x_2 - x_1) \bmod p \quad (11)$$

m is the slope of the line through P_1 and P_2.

Point Doubling

Consider a point P such that

$$p = (x_1, y_1) \quad (12)$$

Where $y_1 \neq 0$

Let $p_3 = 2p_1$ (13)

where $p_3 = (x_3, y_3)$ (14)

Then

$$x_3 = m^2 - 2x_1 \bmod p$$
$$y_3 = -y_1 + m(x_1 - x_3) \bmod p$$
$$m = (3x_1^2 + b)/(2y_1) \bmod p \quad (15)$$

PLAINTEXT ENCRYPTION

From the above basic theory on elliptic curve cryptography, in this section we describe the concept of plaintext encryption by defining a two-dimensional alphabetic table. It is worth noting that in the case of elliptic curve cryptography there is no specified rule and/ or algorithm to specify the letters of the English alphabet as well as special symbols. For this , a 6x5 table(Table 1) has been formed here for both the upper case and lower case letters of the English alphabet along with some of the other symbols like **,** , **.** , **?** and space for illustration purpose only. Other symbol of punctuation marks and special characters can also be considered in a similar way. Note that such tables play some important role in ECC as two-dimensional plaintext co-ordinate representation requires adding with any point on the elliptic curve.

Now, for any plaintext to be encrypted we add or multiply coordinates of a given character with selected points on the elliptic curve. For this point we consider the respective co-ordinates of the respective character. All the match points should be on the surface of the elliptic curve**.** We show the process with the following examples.

Example: For the encryption plain text ‘India’, the two dimensional co-ordinate representations is

$$p_1, p_2, p_3, p_4, p_5 = (1,3), (2,3), (0,3), (1,3), (0,0) \quad (16)$$

Table - 2 Two-Dimensional Alphabetical Table

	0	1	2	3	4
0	A a	B b	C c	D d	E e
1	F f	G g	H h	I i	J j
2	K k	L l	M m	N n	O o
3	P p	Q q	R r	S s	T t
4	U u	V v	W w	X x	Y y
5	Z z	,	.	?	

It can be thought here that for more security point of view the above alphabetic table may also be formed randomly, that is, by assigning any position in the table chosen randomly to any character or symbol.

Following defining an appropriate table, respective coordinates can be assigned as described below.

Algorithm1 (Alphabetic Table_Value_Assign)

Step 0: Generate appropriate alphabetic table

Step 1: Use an appropriate data structure to store the text to be encrypted.
Step 2: Read the table in row-major form and find the character stored in that position.
Step 3: Note the row and column values.
Step 4: Assign these values to the same character in all positions it appears.
Now, we define an analogous algorithm due to ElGamal[10] for encprypting the required text as follows:

Algorithm for Key Generation

Step 0: Select $E(a,b)$ with an elliptic curve over $GF(p)$ or $GF(2m)$.

Step 1: Select a point on the curve $e_1(x_1, y_1)$.

Step 2: Select d

Step 3: Calculate $e_2 = (x_2, y_2) = d * e_1$

Step 4: Announce e_1, e_1 as public key and keep *"d"* as a private key.

Algorithm for ECC Encryption/Decryption

1. Encryption

Step 1: User A selects p, a point on the curve, as a plaintext,

Step 2: Then calculates a pair of points on the text as cipher texts: $c_1 = r \times e_1$ and $c_2 = p + r \times e_2$

2. Decryption

Step 1: User B, after receiving c_1 and c_2, calculates p, the plaintext using the following formula,

$$p = c_2 - (d \times c_1)$$

The Minus sign here means adding the inverse.

Step 2: We can prove that the P calculated by Bob is the same as that sent by Alice, as shown below:

$$p + r \times e_2 - (d \times r \times e_1) = p + (d \times d \times e_1) - (d \times d \times e_1) = p + o = p$$

p, c_1, c_2, e_1, e_2 are all points on the curve. Note the Result of adding two inverse points on the curve is the zero point.

Signature Generation

For signing a message m by sender is Alice, using Alice's private key r
Step 1: First, calculate e = hash(m) using by hash function. It will give message digest.
Step 2: Sign the message digest with his private key by using Alice's software. It is called digital signature.
Step 3: Encrypt digitally signed signature with Bob's public key using ECC algorithm.
Step 4: Encrypted cipher message will be send to Bob.

Signature Verification

For Bob to authenticate Alice's signature, Bob must have Alice's public key
Step 1: Calculate e = HASH (m), where HASH is the same function used in the signature generation
Step 2: Bob authenticate Alice's signature using the public key of Alice for ensuring that whether the incoming message is from Alice or from attacker.

SECURITY OF ELLIPTIC CURVE CRYPTOGRAPHY

As RSA depends on the difficulty of large-number factorization for its security, ECC depends on the large number discrete logarithm calculation. This is referred as the Elliptic Curve Discrete Logarithm Problem (ECDLP). Elliptic curves for which the total number of points on the curve equals the number of essentials in the primary finite field are also considered cryptographically pathetic. Once more the security of ECC depends upon how to calculate k when point is given in scalar multiplication.

Table - 3 Key Sizes for Equivalent Security Levels (In Bits)

Symmetric	ECC	DH/DSA/RSA
80	163	1024
128	283	3072
192	409	7680
256	571	15,360

The security levels which is given by RSA can be provided by smaller keys of elliptic curve Cryptosystem As compared to RSA, which offer 1024 bit security strength, ECC offer the same in 160 bit key length. The efficiency of ECC is depends upon factors such as computational outlay, band width, ECC provides higher-strength per- bit which include higher speeds, key size, smaller power consumption, bandwidth reserves, storage efficiencies and smaller certificates. For providing security mechanism will require fundamental basic security services such as confidentiality, authentication, non-repudiation and message integrity. The implementation ECC shows that it offers complete security solution.

EXPERIMENTAL RESULTS: KEY GENERATION

Select $e_{11}(1,1), e_1(1,1), y_1 = (1,3), d = 2$ (17)

Calculate $e_2 = d * e_1$ (18)

$= 2 * (1,3)$

$= (1,3) + (1,3)$

$m = (3 \times 1^2 + a) / 2y_1 \bmod p$ (19)

$= (1,3) + (1,3)$

$m = (3 \times 1^2 + a) / 2y_1 \bmod p$ (20)

$= (3,1)^2 + 1 / 2(3) \bmod 11 = 8$

$x_3 = (m^2 - x_1 - x_2) \bmod p$ (21)

$= 8^2 - 1 - 1 \bmod 11 = 7$

$y_3 = m(x_1 - x_3) - y_1$ (22)

$= 8(1 - 7) - 3 \bmod 11 = 4$

$(x_3, y_3) = e_2 = (7,4)$ (23)

1. Encryption

Select $r = 1$, Plaintext $p = (1,3)$ (24)

Calculate $c_1 = r * e_1$ (25)

$= 1 * (1,3)$

Cipher text $c_1 = (1,3)$

Calculate $c_2 = p + r * e_2$

First, $r * e_2 = 1 * (7,4) = (7,4)$

Second, $c_2 = (1,3) + (7,4)$

$m = y_2 - y_1 / x_2 - x_1 \bmod p$ (26)

$4 - 3/7 - 2 \bmod 11 = 9$

$x_3 = (m^2 - x_1 - x_2) \bmod p$ (27)

$= 9^2 - 2 - 7 \bmod 11 = 6$

$y_3 = m(x_1 - x_3) - y_1 \bmod p$ (28)

$= 9(1 - 6) - 3 \bmod 11 = 5$

Cipher text $c_2 = (6,5)$

2. Decryption

$p = c_2 - (d * c_1)$ (29)

First calculate $d * c_1 = 2 * (1,3) = (7,4)$

$p = (6,5) + (7,4)$

$$m = y_2 - y_1 / x_2 - x_1 \bmod p \quad (30)$$

$$4 - 5/7 - 6 \bmod 11 = 2$$

$$x_3 = m^2 - x_1 - x_2 \bmod p \quad (31)$$

$$= 2^2 - 6 - 7 \bmod 11 = 2$$

$$y_3 = m x_1 - x_3 - y_1 \bmod p \quad (32)$$

$$= 2 \cdot 6 - 2 - 5 \bmod 11 = 3 \quad (33)$$

Plaintext $p = 2,3$

PERFORMANCE PARAMETERS FOR ELLIPTIC CURVE CRYPTOGRAPHY IMPLEMENTATION

Although RSA, El-GAMAL and Diffie –Hellman are secure asymmetric key cryptosystem, security comes with a price, their large keys. So researchers have look for providing substitute that provides the same level of security with smaller keys. The Elliptic Curve Cryptography implementation should meet following consideration such as:

• Suitability of methods accessible for optimizing set field arithmetic like addition, multiplication, squaring, and inversion.
• Suitability of methods accessible for optimizing elliptic curve arithmetic like point addition, point doubling, and scalar multiplication.
• Applications platform like software, hardware or firmware.
• Constraints of a particular computing environment. For example: processor speed, storage, code size, gate count and power consumption.
• Constraints of a particular communication environment.

For example: bandwidth, response time.
Efficiency of ECC depends upon the factors such as computational overheads key size, bandwidth, ECC provides higher-strength per- bit which includes higher speeds, lower power consumption, bandwidth savings, storage efficiencies and smaller certificates.

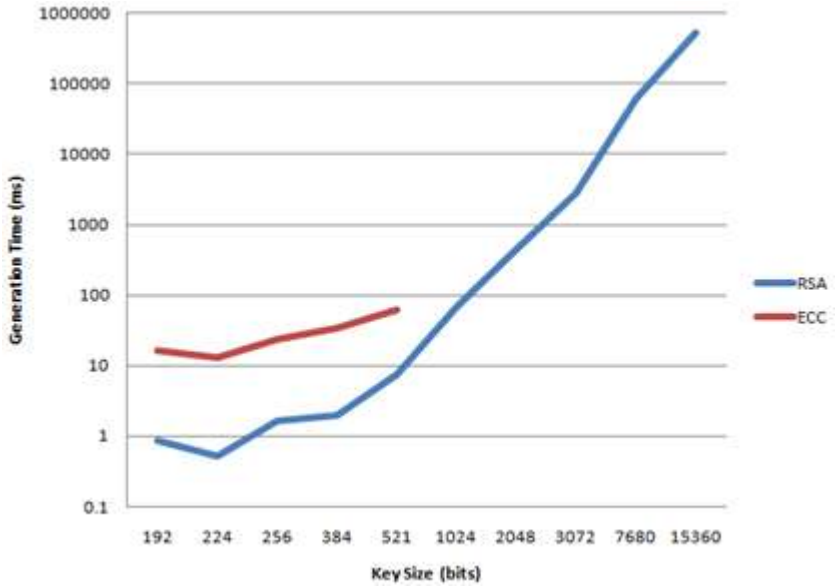

Fig. 5 Key Generation by Key Size

APPLICATIONS OF ELLIPTIC CURVE CRYPTOGRAPHY

An application of ECC includes security of SMS for mobile banking by means of data encryption. Also the implementation of ECC for web's security infrastructure, integration is open SSL and for the implementation of cryptographic algorithms and protocols. Many devices are constrained devices that have small and restricted storage and computational power for constrained devices ECC can be applied. ECC can be functional for wireless communication devices like PDA's multimedia cellular phones. It can be used for security of Smart cards, wireless sensor networks and wireless mesh networks. Web servers that need to handle many encryption sessions. Any kind

of application for security is needed for our current cryptosystems. For secret key sharing offers Diffie-Hellman protocol, in order to implement the Diffie-Hellman protocol scalar multiplication is used in public-key cryptosystem.

CONCLUSION

Encryption in mobile communication is very crucial to protect information of the subscribers and avoiding the fraud. This paper studied security by means of elliptic curve cryptographic technique. The actual implementation of encryption/decryption using elliptic curve cryptography on GF (P) shows that a security that security of the proposed system is very hard. It has been mentioned in many literatures that a considerably smaller key size can be used for ECC compared to RSA. Also mathematical calculations required by elliptic curve cryptosystem are easier, hence, require a low calculation power. Therefore ECC is a more appropriate cryptosystem to be used on small devices like mobile phones. The ECC has received considerable attention from mathematicians throughout the world, and no significant breakthroughs have been made in weaknesses in the algorithm. Hence, the future some efficient methods for point multiplication can be used to speed up the computation.

REFERENCES

[1] William Stallings,*Cryptography and Network Security Principles and Practices*, Prentice Hall, 4th edition, **2006**.
[2] Andrej Dujella, Applications of Elliptic Curves in Public Key Cryptography, *Conference on Applied Mathematics*, Bilbao, **2011**.
[3] Pardeep Malik, Elliptic Curve Cryptography For Security Inwireless Networks, *5th Canadian Conference in Applied Statistics*, Canada, **2011**.
[4] Dujella, Applications of Elliptic Curves in Public Key Cryptography, *Conference on Applied Mathematics*, Bilbao, **2011**.
[5] Moncef Amara, Amar Siad, Elliptic Curve Cryptography and its Applications, *7th International Workshop on Systems, Signal Processing and their Applications (WOSSPA)*, **2011**.
[6] R Rivest, A Shamir and L Adeleman, A method for obtaining digital signatures and public key cryptosystems, *Communication of the ACM*, **1978**, 21, 120-126.
[7] D. R. Stinson, *Cryptography Theory and Practice*, Chapman & Hall/CRC, 3rd edition, **2006**.
[8] V J Vazram, V V Kumari and J V R Murthy, Privacy in Mobile Ad Hoc Networks, *Advances in Digital Image Processing and Information Technology, CCIS,* **2011**, 205, 336-345.
[9] J Lopez and R Dahab, An Overview of Elliptic Curve Cryptography, *Technical Report, IC-00-10*, web. : http:// www.dcc.unicamp.br/ic-main/public-cation - e.html.
[10] M Aydos, B Sunar, and C K Ko, Securing Mobile Communication using Elliptic Curve Cryptography Over GF(P*), International Journal of Advanced Computational Engineering and Networking,* **2013**, 1, 28-33.
[11] Sawlikar, Point Multiplication Methods for Elliptic curve Cryptogrphy, *International Journal of Engineering and Innovative Technology,* **2012**,1, 67-71.
[12] Hilyati Hanina Zazali and Wan Ainun Mior Othman, Key Exchange in Elliptic Curve Cryptography Based on the Decomposition Problem, *Sains Malaysiana* , **2012**, 41, 907–910.
[13] Ounasser Abid, , Jaouad Ettanfouhi and Omar Khadir, New Digital Signature Protocol Based On Elliptic Curves, *International Journal on Cryptography and Information Security*, **2012**, 2, 210-216.
[14] Asha Rani Mishra, Mahesh Singh, Elliptic Curve Cryptography(ECC) for Security in Wireless Sensor Network, *International Journal of Engineering Research & Technology,* **2012**, 1, 34-38.