A STUDY ON THE PERFØRMANCE ANALYSIS OF SECURE MULTI-PARTY ELECTRONIC PAYMENTS IN MOBILE COMPUTING USING ECC ALGORITHM

K.Ravikumar [1], *A.Udhayakumar* [2]

ABSTRACT

This paper describe the Elliptic Curve Cryptography algorithm and its performance and its suitability for Multi-party Electronic Payments in Mobile computing. We propose an efficient Multi-party electronic payments scheme for mobile networks where a given number of users can be combined to sign the paper, and finally it verified by any authorized public key. The proposed multi-party electronic payment scheme takes lower computation as well as in communication cost It also and flexible, efficient thereby providing strong security for low power devices such as mobile communication environment. It not only provides security but also it checks whether the data receive or not in the public key verification.

Key words - Elliptic Curve, Cryptography, Multi-party, Diffi-Hellman, AES, DES, RSA, MD5, SHA

I. INTRODUCTION

Elliptic Curve Cryptography has emerged as a viable and popular alternative to building public key cryptosystems, primarily because they give rise to algebraic structures that offer a number of advantages over other algebraic structures, smaller key sizes and higher strength-per-bit being some of them[2]. These characteristics make ECC particularly amenable to hardware implementations. Any two points lying on an elliptic curve can be added to obtain another point lying on the curve. A point can also be added to itself. This operation is the basis of elliptic curve cryptography.

Cryptographic schemes employing Elliptic Curve Cryptography (ECC) begin with the identification of domain parameters that allow for the level of security that is warranted by the application at hand. Current systems implementing ECC make use of standard curve parameters published by various organizations. These systems use standard cryptographic algorithms in encrypting and decrypting information using predefined curves. These predefined curves are well exposed to the outside world. So, it limits the security provided by these curves as pre-computation can be done in advance[1].

For current cryptographic purposes, an elliptic curve is a plane curve which consists of the points satisfying the equation $y^2 = x^3 + ax + b \pmod p$ along with a distinguished point at infinity, denoted ∞. The field is defined by p in the prime case and the constants a and b used in its defining equation.

NIST recommends fifteen elliptic curves. Particularly, FIPS 186-3 has ten recommended finite fields:

- Now, five prime fields Fp for certain prime's p of sizes 128,192, 224, 256, 384, and 521 bits. For every prime field, one elliptic curve is recommended.

[1] Tamil University / Dept. of Computer Science, Thanjavur - 613010, Tamil Nadu, India. E-mail : ravikasi2001@yahoo.com

[2] A.M.JAIN College, Chennai, Karpagam University / Dept. of Computer Science, Coimbatore, Tamil Nadu, India
E-mail : umaudhaya83@gmail.com

An efficient implementation of Multi-party electronic payments based on elliptic curves and a comparative study of the Elliptic curve with reference to other cryptographic algorithms. This paper is proposed to be an implementation of multiparty electronic payments using ECC algorithm. The proposed scheme has an added advantage of lower computational and communication cost. Moreover this system provides more security and integrity. It not only provides security, but also has less key storage and execution speed. In the proposed scheme, the multi-party transactions require more keys when compared to other schemes. Thic scheme can be implemented in low power devices such as smart cards and mobile phones very efficiently because of low computation and communication cost [3]. At the same time, it helps to maintain multiple secrets. Thus the comparative study helps in the analysis of the Multiparty Electronic Payments using Elliptic curves.

II. Elliptic Curve Cryptosystem

In resourcc constrained system, Elliptic Curve Cryptography is a promising alternative for public algorithms, because it provides similar level of security with proposed shorter keys than conventional integer based public key algorithm[4]. ECC over binary field is taken up with special interest because the operation in binary filed operation, are thought to be more in space and efficient in time. However, ECC's software implementations, on binary field are slow, especially on low end processors, which are used in small computing devices such as sensors node, mobile phone, etc. This proposed paper, studied the Cryptography algorithms and software implementation of ECC[8]. Firstly, while implementing ECC with software, for example byte size may affect the choice of algorithm some architectural parameters has been examined. In addition, the proposed paper has implemented ECC algorithm in Multiparty Electronic transaction[5].

ECC can be used with fewer keys to give more security, high speed in a less bandwidth. While these advantages make ECC propose for mobile devices, they can provide computational burden on secure web server[6]s. In resource constrained system, Elliptic Curve Cryptography is a promising alternative for public algorithms, because it provides similar level of security with proposed shorter keys than conventional integer based public key algorithm. ECC over binary field is taken up with special interest because the operation in binary filed operation, are thought to be more in space and efficient in time. However, the software implementation of ECC over binary field are still slow, especially on low end processors, which are used in small computing devices such as sensors node, mobile phone, etc. This proposed paper, studied the Cryptography algorithms and software implementation of EC[7]C. Firstly, while implementing ECC with software, the choice of some architectural parameters like word size may affect the choice of algorithms or not, has been examined. Also, identification of software for low-end processors has been done. In addition, this paper has examined several implements to the instruction that architecture of an 8 bit processor and studied their impact on the performance of ECC with other algorithms. ECC is well is well suited for high speeds, lower power consumption, bandwidth savings, storage efficiencies, smaller certificates and it reduces computational time and also the amount of data transmitted and stored, and strong security for low-power devices in wireless networks[9].

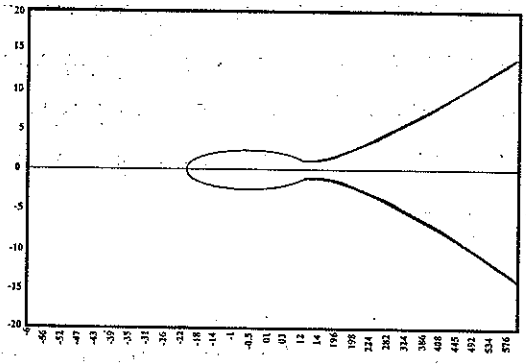
A. Elliptic Real curves

An elliptic curve is an algebraic curve defined by an equation of the form:

$$y^2 = x^3 + ax + b$$
$$4a^3 + 27b^2 \neq 0$$
$$x, y, a, b \in R$$

Where *a* and *b* are parameters of the curve, i.e. the nature of the curve can be varied by selecting different values of *a* and *b*. For the purpose of elliptic curve cryptography, elliptic curves may be defined over the field F_p, the set of integers from *0* to *p – 1*, where *p* is a prime number[10].

$$y^2 \, mod \, p = (x^3 + ax + b) \, mod \, p$$
$$4a^3 + 27b^2 \, mod \, p \neq 0$$
$$x, y, a, b \in F_P$$
$$F_P = \{0,1,2,\cdots,p-1\}$$



p is prime

Figure 1: Elliptic curve over R^2:y^2=x^3-3x+3 Whereas

[Figure 1] elliptic curves defined over the real domain have an infinite number of points on them, elliptic curves defined over prime fields have a finite number of points, the exact count being a function of *p*, *a*, and *b[11]*.

Performing scalar multiplication, *kP*, to obtain *Q*, is a computationally trivial task. Finding the value of *k*, given points *P* and *Q*, has been proven to be as computationally infeasible as the discrete logarithm problem [7]. Any elliptic curve cryptographic scheme involves representing the plaintext as a point *P*, selecting a key *k*, and producing cipher text *Q*.

The domain parameters of an Elliptic Curve Cryptographic Scheme dictate the nature of the curve that will be employed. For an elliptic curve defined over prime fields, the parameters *p, a, b* are of primary interest, *p* being the prime number that specifies the field *{0, 1, ... , p – 1}*, and *a* and *b* being parameters of the elliptic curve algebraic equation. The curve, having been defined over a finite field, contains a finite number of points, the exact count being a function of the aforementioned three parameters[12].

B. Point Multiplication

In point multiplication a point P on the elliptic curve is multiplied with a scalar k using elliptic curve equation to obtain another point Q on the same elliptic curve i.e. kP=Q Point multiplication is achieved by two basic elliptic curve operations:

1. Point addition, adding two points J and K to obtain another point L i.e., L = J + K.
2. Point doubling, adding a point J to itself to obtain another point L i.e. L = 2J.

Here is a simple example of point multiplication [13].

Let P be a point on an elliptic curve. Let k be a scalar that is multiplied with the point P to obtain another point Q on the curve i.e. to find Q = KP ; If k = 23 then kP = 23.P = 2(2(2(2P) + P) + P) + P. Thus point multiplication uses point addition and point doubling repeatedly to find the result. The above method is called 'double and add' method for point multiplication. There are other efficient methods for point multiplication such as NAF (Non – Adjacent Form) and wNAF (windowed NAF) method for point multiplication[14].

C. Point Addition

Point addition is the addition of two points J and K on an elliptic curve to obtain another point L on the same elliptic curve.

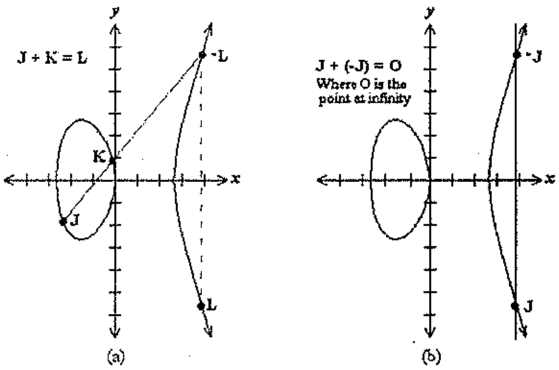


Figure 2: Point Addition

Consider [Figure 2] two points J and K on an elliptic curve as shown in figure (a). If K $\neq$ -J then a line drawn through the points J and K will intersect the elliptic curve at exactly one more point –L. The reflection of the point –L with respect to x-axis gives the point L, which is the result of addition of points J and K. Thus on an elliptic curve L = J + K[15].

If K = -J the line through this point intersect at a point at infinity O. Hence J + (-J) = O. This is shown in figure (b). O is the additive identity of the elliptic curve group. A negative of a point is the reflection of that point with respect to x-axis.

D. Point Doubling

Point doubling is the addition of a point J on the elliptic curve to itself to obtain another point L on the same elliptic curve.

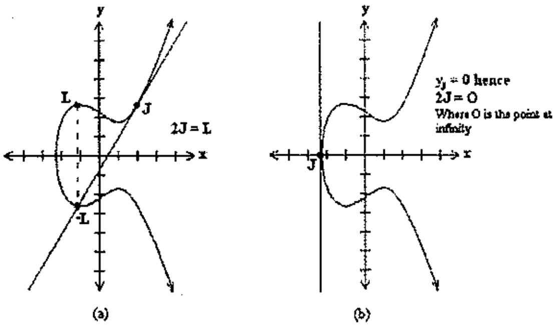


Figure 3 : Point Doubling

To double a point J to get L, i.e. to find L = 2J, consider a point J on an elliptic curve as shown in figure (a). If y coordinate of the point J is not zero then the tangent line at J will intersect the elliptic curve at exactly one more point –L. The reflection of the point –L with respect to x-axis gives the point L, which is the result of doubling the point J[Figure 3].

Thus L = 2J.

If y coordinate of the point J is zero then the tangent at this point intersects at a point at infinity O. Hence 2J = O when yJ = 0.[17] This is shown in figure (b).

E. Elliptic Curves over F_p

The Let Fp be a prime finite field so that p is an odd prime number, and let a, b belongs to Fp satisfy $4a^3 + 27b^2 \neq 0(\text{mod } p)$. Then an elliptic curve E(Fp) over Fp defined by the parameters a,b belongs to Fp consists of the set of solutions or points P=(x,y) for x,y belongs to Fp to the equation:

$$y^2 = x^3 + ax + b \ (\text{mod } p)$$

together with an extra point O called the point at infinity. The equation $y^2 = x^3 + ax + b$ (mod p) is called the defining equation of E(Fp). For a given point P=(xP,yP), xP is called the x-coordinate of P, and yP is called the y-coordinate of P.

The number of points on E(Fp) is denoted by #E(Fp). The **Hasse Theorem** states that:

p+1-2.sqrt(p) <= #E(Fp) <= p+1+2.sqrt(p).

Whereas elliptic curves defined over the real domain have an infinite number of points on them, elliptic curves defined over prime fields have a finite number of points, the exact count being a function of *p, a,* and *b*.

F. Elliptic Curve: Some definitions

- **The Scalar Multiplication:** Given an integer k and a point P on the elliptic curve, the elliptic scalar multiplication kP is the result of adding Point P to itself k times.
- **Order :** Order of a point P on the elliptic curve is the smallest integer r such that

  $rP = \mathcal{O}$. Further if c and d are integers, then cP = dP iff $c \equiv d \ (\text{mod } r)$.
- **Curve Order:** The number of points on the elliptic curve is called its curve order and is denoted #E.

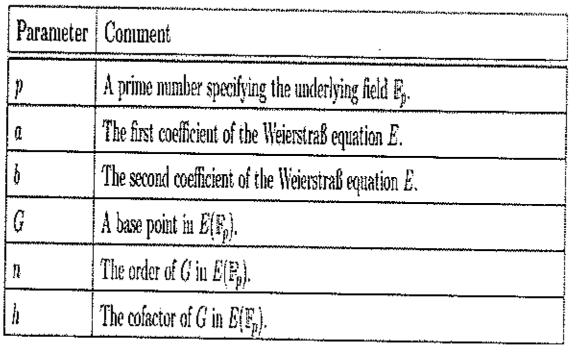
G. Elliptic Curve Domain Parameter over F_p

Elliptic curve domain parameters over Fp are a sextuple:

$$T = (p, a, b, G, n, h)$$

consisting of an integer p specifying the finite field Fp, two elements a, b belongs to Fp specifying an elliptic curve E(F p) defined by the equation: $y^2 = x^3 + ax + b$ (mod p); a base point G=(xG, yG) on E(Fp), a prime n which is the order of G, and an integer h which is the cofactor h= #E(Fp)=n. Elliptic curve domain parameters over Fp precisely specify an elliptic curve and base point. This is necessary to precisely define public-key cryptographic schemes based on ECC shown in (Table 1).

Table 1 : Elliptic curve domain parameters over Fp.



Parameter	Comment
p	A prime number specifying the underlying field $\mathbb{F}_p$.
a	The first coefficient of the Weierstraß equation E.
b	The second coefficient of the Weierstraß equation E.
G	A base point in $E(\mathbb{F}_p)$.
n	The order of G in $E(\mathbb{F}_p)$.
h	The cofactor of G in $E(\mathbb{F}_p)$.

H. Elliptic Curve Discrete Logarithm Problem

The strength of the Elliptic Curve Cryptography lies in the Elliptic Curve Discrete Log Problem (ECDLP). The statement of ECDLP is as follows.

Let E be an elliptic curve and $P \in E$ be a point of order n. Given a point $Q \in E$ with

$Q = mP$, for a certain $m \in \{2, 3, \ldots\ldots, m-2\}$.

Find the m for which the above equation holds.

When E and P are properly chosen, the ECDLP is thought to be infeasible. Note that m = 0, 1 and m – 1, Q takes the values *O*, P and – P. One of the conditions is that the order of P i.e. n be large so that it is infeasible to check all the possibilities of m.

The difference between ECDLP and the Discrete Logarithm Problem (DLP) is that, DLP though a hard problem is known to have a sub exponential time solution, and the solution of the DLP can be computed faster than that to the ECDLP. This property of Elliptic curves makes it favorable for its use in cryptography.

I. Application of Elliptic curve in Cryptography

Elliptic Curve Cryptography has emerged as a viable and popular alternative to building public key cryptosystems, primarily because they give rise to algebraic structures that offer a number of advantages over other algebraic structures, smaller key sizes and higher strength-per-bit being some of them. These characteristics make ECC particularly amenable to hardware implementations. An elliptic curve over Fq is defined in terms of the solutions to an equation in Fq. The form of the equation defining an elliptic curve over Fq differs depending on whether the field is a prime finite field or a characteristic 2 finite field .Any two points lying on an elliptic curve can be added to obtain another point lying on the curve. A point can also be added to itself. This operation is the basis of elliptic curve cryptography .

III. Objectives and Scope

The usage of predefined elliptic curves has its advantages and disadvantages. On one side there exist a set of curves recommended by NIST, which are publicly available in FIPS 186-3. Contained in the document are both curves over prime and binary fields with bit-sizes ranging from about 160 bits to about 500 bits. Those curves meet some security requirements and are especially suited for efficient implementations. When it comes to elliptic curve cryptography it is probably the straightforward way to use the well-known NIST-recommended curves. Test-benches are available and the additional task to generate a curve is obsolete.

Table 2 : Equivalent Key sizes in bits, recommended by NIST

Symmetric key size	$\mathbb{Z}_n^*$ public key size	Elliptic Curve public key size
80 bit	1024 bit	160 bit
112 bit	2048 bit	224 bit
128 bit	3072 bit	256 bit
192 bit	7680 bit	384 bit
256 bit	15360 bit	521 bit

On the other side, there are not many reasons why not to use a new random curve for each key-agreement, given that the curves meet the same security properties than predefined curves. The main advantage is obvious: when the curve is not known in advance, no pre-computation can be done on it that might compromise its security. As an illustrative result, claims the security when using a random curve is the same as using a fixed curve with an 11 bit longer key [Table 2]. The critical factor however is efficiency.

So, in this project we tend to generate a new set of new curves using the given prime p. Using this prime we select a particular (a,b) value which generates a new curve having highest order O(p). Base point of the curve is generated in a random way, thereby JCE is used for encryption and decryption with the help of DES and Diffie Hellman Key Exchange. For enhanced security Image Steganography is implemented for hiding the cipher text using LSB Technique Shown in (Table II).

IV. The ECC Algorithm

Majority of public key cryptography use either integer or polynomial arithmetic with very large number[26]. Imposes a significant load is storing and processing keys and messages instead of use elliptic curve. It is having an alternative is to use elliptic curve offers some security with smallest bit sizes newer but not as well analysed [1].

A. *Real Elliptic Curve*

An elliptic curve is defined by an equation in two variables x and y, with coefficients. Consider a cubic elliptic curve of form

$$Y^{2=}X^{3+}a*y\,b$$

Where x, y, a, b are all real numbers, defined at zero point. Consider set of points E (a, b) that have an additional operation for elliptic curves. The sum of point Q is

B. *Finite Elliptic Curve*

Elliptic curve cryptography uses curves whose variables and coefficients are finite and they have two families commonly used: Prime curves EP (a, b) defined over ZP. Use integers modulo, which are prime numbers, is best in software. Binary curves EZM (a, b) are defined over GF (Z^N).User polynomials with binary coefficients are best in hardware [1], [2].

C. *Elliptic Curve Cryptography*

ECC addition is an analog of modulo multiply. In ECC repeated addition is an analog of modulo exponentiation. "Hard problem" equivalent to discrete log Q, kp, where Q, p belongs to a prime curve is to carry and to compute Q given k ,p but "hard" to find k given Q ,p known as the elliptic curve logarithm prime.

Initialization Phase: ECC Diffie-Helmen can do key exchange analogers to Diffie-Helmen users select a suitable curve Eq (a, b):

Step 1: Select base point G=(X1, Y1) within large order n such that nG=0 A and B

Step 2: Select private keys nA<n, nB<n,

Step 3: Compute public keys: pA=nAG, pB=nBG

Step 4: Compute shared keys=k=nApB, k=nBpA

Step 5: Same since k=nAnBG

Attacker would need to find k

D. *ECC Encryption and Decryption*

Several alternatives, which are considered simpler must first encode any message M and a point on the elliptic curve prime pm. Select suitable curve and point h assign different numbers for each user chooser.

Step 1: private key nA<n

Step 2: And computes public key pA=nAG N

Step 3: Which finally encrypts as pm: Cm< {kG, pm+kpb}, k random

Step 4: And Decrypt as Cm compute: Pn+kpb-nb (kG) <p+K (nBG)-nB (KG) <pm

When compared to factory method "pollared rho method" is the fastest with, much smaller key sizes than with RSA etc., and for equivalent key length computation are roughly equivalent. Hence for similar security ECC refers to significant computational advantages [5], [3].

A typical ECC algorithm for Multi-party electronic payments is as follows:

Given: A composite integer's $n \in N$ (with no small prime factors).

Output: A non-trial divisor d of n

Procedure:

```
While (1) {
Select a random curve E : Y2=x3+aX+b modulo n.
Choose a point p ≠Q in E (Zn).
Try to compute mP./*where m is as defined in the text */
If (the computation of mP fails) {
/*we have found a divisor d>1 of n*/
If (d ≠n) {Return d} } }
```

V. Result Analysis

A. *Computational Key Storage*

In our proposed scheme, we try to reduce multi-party key storage. TABLE 3 shows the comparison of Key storage sender and recipient among our encryption/decryption scheme and others.

Table 3 : MAKE THE COMPARISON BASED ON KEY STORAGE MEMORY SIZE IN BITS

Algorithm	Aes	Des	RSA	Sha	MD5	ECC
Key Size in Bit	256	56	1024	512	124	112

Table 4 : MAKE THE COMPARISON BASED ON KEY STORAGE MEMORY SIZE IN BYTES

Algorithm	Aes	Des	Rsa	Sha	MD5	ECC
Memory Size in Bytes	37	31	28	31	23	17

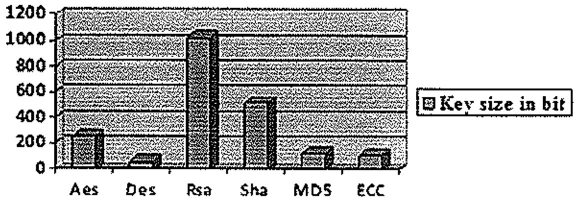


Figure 4 : Comparison of Key size in bits of Crptography algorithms

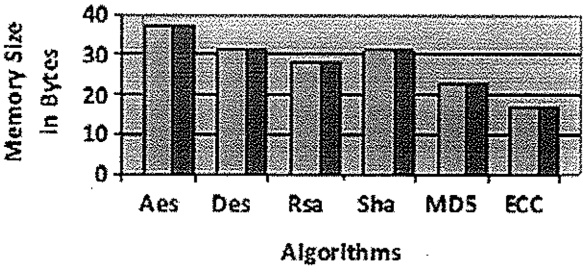


Figure 5 : Comparison of Key Memory size in bytes of Cryptography algorithms

From the above content (Figure 4) it is proved that the key size of the ECC is very less when compared with the other algorithms such as Aes, Des, Rsa, Sha and MD5.

B. *Computational Memory Size in Bytes*

We try to reduce multi-party key storage. TABLE 4 shows the comparison of Key storage Memory size in bytes sender and recipient among our encryption/decryption scheme and others.

From the display (Figure 5) it is evident that the memory size is reduced, when ECC algorithm is applied. The Aes, Des, Rsa, Sha and MD5 occupy large memory space when compared with ECC.

C. Computational Encryption/Decryption time of Cryptography algorithm

We try to reduce multi-party key storage. TABLE 5 shows the comparison of sender and recipient among our encryption/decryption time in sec and others.

Table 5 : MAKE THE COMPARISON BASED ON ENCRYPTION/DECRYPTION TIME IN SEC

Algorithm	Aes	Des	Rsa	Sha	MD5	ECC
Key Size	256	56	1024	512	128	112
Encryption	1.6	3	7.3	1.7	1	0.7
Decryption Time in sec	1.1	1	4.9	1	1.7	0.6

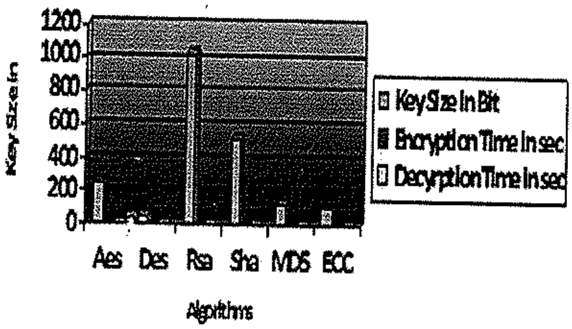


Figure 6: Comparison of Cryptography algorithm Encryption/Decryption in sec

From the above (Figure 6) analysis, it is clear that the encryption and decryption time of the ECC is lesser than other algorithms. While looking at the chart it is very much evident that the key size is very small than other algorithms such as Aes, Des, Rsa, Sha, and MD5, thereby memory size of the ECC is reduced.

Table 6 : MAKE THE COMPARISON BASED ON DECRYPTION IN SEC

Algorith	Ae	De	Rs	Sh	M	ECC
Key Size in	25	56	10	51	12	112
Decyrption Ti	1.	1	4.	1	1.7	0.6

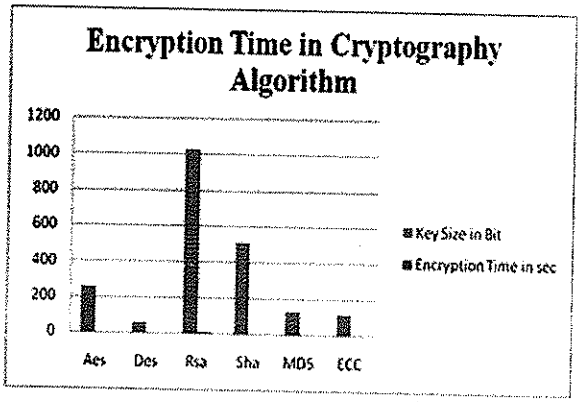


Figure 7 : Comparison of Cryptography algorithm Encryption in sec

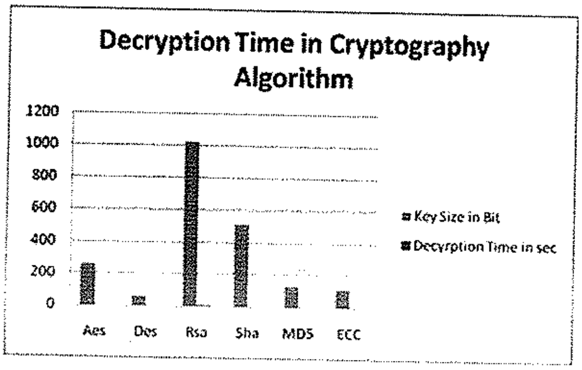


Figure 8 : Comparison of Cryptography algorithm Decryption in sec

From the above (Figure 6) data the encryption time of the ECC is reduced when compared to other algorithms such as Aes, Des, Rsa, Sha and MD5. The the size of key file also less than the other algorithms. The reduction of the key file is uses less memory space so memory size also consumed less than the other algorithms (TABLE VI).

From the above (Figure 7) data the decryption time of the ECC is reduced when compared to other algorithms such as Aes, Des, Rsa, Sha and MD5. The the size of key file also less than the other algorithms. The reduction of the key file is uses less memory space so memory size also consumed less than the other algorithms(TABLE VII).

Table 7 : MAKE THE COMPARISON BASED ON ENCRYPTION

Algorithm	Aes	Des	Rsa	Sha	MD5
Key Size in Bit	256	56	102	51	128
Encryption Time	1.6	3	7.3	1.	1

VI. Conclusion

The paper is a study of performance evaluation of ECC and it has been examined that how some architectural features, such as key size and ISA, which affect the performance of ECC. The algorithm, for ECC over binary field has been first examined, and after comparing algorithms for the major field operations that are required in ECC, the identification of set of efficient algorithms suitable for resource constrained systems has been done. Besides, the performance of these algorithms for different word sizes has been compared. As a result, the change of word sizes result in different choices of algorithms. The stimulation of the implementation is on an 8-bit micro controller. The proposed implementations are more than twice faster than previous results without instruction set architecture extensions or hardware accelerations.

In addition, this paper has evaluated three instructions for accelerating ECC: binary field multiplication, shift with an arbitrary shift amounts, and index of most significant combining all three instructions, one can achieve a speed up of 3.89. Important of all, the new instructions make the projective co-coordinators a better choice for point representations. The projective co-ordinates achieve a speed up of 8.23 over the base line architecture. It takes about 0.85 seconds to perform a 163-bit scalar point multiplication on 8-bit AVR processors at 16MHz.

This paper focuses mainly on performance analysis of ECC. Application specific hardware can be integrated into processors to accelerate the multiplication and inversion operations further. The performance of multiplication and inversion should be evaluated to choose the best point representation for better performances, when the new hardware is implemented. Finally, one can come to the conclusion that ECC is more secure than other Cryptographic algorithms and its executed speed and key storage is much better than others.

References

[1] N. gura, a. patel a.wanter *"comparing elliptic curve cryptography and RSA on 8-bit cpu, "proceedings of cryptographic hardware and embedded system* "2004.

[2] D. Hankerson and A. Menezes *"software implementation of elliptic curve cryptographic over binary fields,"* proceedings of workshop cryptography hardware embedded system "2000.

[3] D.J. Malan, m.welsh *"a public key infrastructure for key distribution in tinyOS based on elliptic curve cryptographic,"* 2004.

[4] Atmel Corporation, 8-bit microcontroller with 128K bytes in-system programmable flash: AT mega 128, 2004.

[5] N. koblitz, *"elliptic curve cryptosystem,"* mathematics of computation, vol. 48, pp. 203-209, 1987.

[6] I. Blake, g. seroussi, and n. smart, elliptic curves in cryptography, Cambridge University press, 1999.

[7] L.lopcz and r.dahab, *"high speed software multiplication in F (2m),"* proceedings of idocrypto '00, pp. 203-212,2000.

[8] V.miller, *"uses of elliptic curves in cryptography,"* advance in cryptology: proceedings of crypto '85, pp. 471-426,1986.

[9] National institute of standards and technology, digital signature standard, FIPS publication 186-2, Feb. 2000.

[10] J.solinas *"efficient arithmetic on koblitz curves, designs, codes and crypto graphy, vol. 19,"* pp.195-249, 2000.

[11] N. koblitz, *"elliptic curve cryptosystem,"* mathematics of computation, vol. 48, pp. 203-209, 1987.

[12] I. Blake, g. seroussi, and n. smart, elliptic curves in cryptography, Cambridge University press, 1999.

[13] L.lopcz and r.dahab, *"high speed software multiplication in F(2m),"* proceedings of idocrypto '00, pp. 203-212,2000.

[14] V.miller, *"uses of elliptic curves in cryptography,"* advance in cryptology: proceedings of crypto '85, pp. 471-426,1986.

[15] National institute of standards and technology, digital signature standard, FIPS publication 186-2, Feb. 2000.

[16] J.solinas *"Efficient arithmetic on koblitz curves, designs,"* codes and crypto graphy, vol. 19, pp.195-249, 2000.

[17] I. Blake, g. seroussi, and n. smart, elliptic curves in cryptography, Cambridge University press, 1999.

Author's Biography

Dr. K. Ravikumar, is working in Tamil University Thanjavur. He is received Tamil sudar award. He is presented paper on 20 international and National Conferences. He is completed UGC Research Project. His Research areas is Network security,Cryptography and Tamil Computing. He is written 16 Books for Computer Science in Tamil Medium.

A. Udhayakumar, his B.Sc and M.Sc degree in computer science from the Bharathidasan University in 2003 and 2005 respectively. He received his M.Phil degree in computer science from Alagappa University in the year 2009. He is pursuing his Ph.D research in Karpagam University, Tamilnadu, India,focusing on"Multiparty Mobile computing security". He joined as an Assistant Professor in the Department of computer science, A.M.JAIN college, Meenambakkam,Chennai-114 affiliated to the University of Madras Tamilnadu,India in the year 2005. He is the author/co-author for more than 10 National papers. He is the co-author of "Elements of Computer Network". His papers have received a wide range of awards in the National seminars and conferences. His area of interest is in the field of networking security.