Implementation of Text based Cryptosystem using Elliptic Curve Cryptography

S. Maria Celestin Vigila[1], K. Muneeswaran[2]

[1] *Asst. prof. , Department of Information Technology, Noorul Islam College of Engg., Kumaracoil*

[2] *Prof., Department of Computer Science and Engg., MEPCO Schlenk Engg. College, Sivakasi.*

celesleon@yahoo.com, kmuni@mepcoeng.ac.in

***ABSTRACT:*-Data encryption is widely used to ensure security in open networks such as the internet. With the fast development of cryptography research and computer technology, the capabilities of cryptosystems such as of RSA and Diffie-Hellman are inadequate due the requirement of large number of bits. The cryptosystem based on Elliptic Curve Cryptography (ECC) is becoming the recent trend of public key cryptography. This paper presents the implementation of ECC by first transforming the message into an affine point on the Elliptic Curve (EC), over the finite field GF(p). In ECC we normally start with an affine point called $P_m(x,y)$ which lies on the elliptic curve. In this paper we illustrate the process of encryption/decryption of a text message. It is almost infeasible to attempt a brute force attack to break the cryptosystem using ECC.**

***Index Terms:*-**Elliptic Curve Cryptography (ECC), discrete logarithm, Elliptic Curve (EC), public key cryptography.

I. INTRODUCTION

The use of elliptic curves in public key cryptography was independently proposed by Koblitz and Miller in 1985 [1] and since then, an enormous amount of work has been done on elliptic curve cryptography. The attractiveness of using elliptic curves arises from the fact that similar lever of security can be achieved with considerably shorter keys than in methods based on the difficulties of solving discrete logarithms over integers or integer factorizations.

Elliptic curve cryptography makes use of elliptic curves in which the variables and coefficients are all restricted to elements of a finite field. For the interest of the readers, it is said that Elliptic curves are not ellipses. In ECC we normally start with an affine point called $P_m(x,y)$. These points may be the Base point (G) itself or some other point closer to the Base point. Base point implies it has the smallest (x,y) co-ordinates, which satisfy the EC.

A character in a message is first transformed into an affine point of the elliptic curve by using it as a multiplier of P_m. That is, if 'a' is a scalar value representing the ASCII value of a text character, then we determine $P_{ml} = a * P_m$. The newly evaluated P_{ml} is a point on the EC, determined by applying the scalar multiplication on the EC points. The multiplication of the points are implemented by the repeated addition and doubling strategy of ECC technique. Then as per ECC algorithm, P_{ml} is added with kP_B, where 'k' is randomly chosen large secret integer and P_B is the public key of user B, which is also a point EC field to yield $(P_{ml}+kP_B)$. This now constitutes second part of the encrypted version of the message. The other part, namely, kG, which is the product of the secret integer and the Base point, constitutes the first part. Thus the encrypted message is now made up of two sets of coordinates, namely $(kG, P_{ml} + kP_B)$.

Now to recover the information from the encrypted version, first we apply the decryption process of ECC, by applying the private key of recipient (n_B) on the first element (kG). This is subtracted from the second element to recover P_{ml}. Lastly by using the discrete logarithm concept, it is possible to evaluate the ASCII value and thereby recover the plaintext. Hence the keys are transformed over the EC field for both encryption and decryption. This promises to afford maximum security from intruders and hackers.

II. RELATED WORKS

In the literature, many authors have tried to exploit the features of EC field to deploy for security applications. We have outlined some of the highlights of the relevant work in this section. M.Aydos et.al [2] has presented an implementation of ECC over the field GF(p) on an 80 MHz, 32 bit RAM microprocessor along with the results. Kristin Lauter has provided an overview of ECC for wireless security [3]. It focuses on the performance advantages in the wireless environment by using ECC instead of the traditional RSA cryptosystem. Ray C., [4] in his work has explained the design of a generator, which automatically produces a customized ECC hardware that meets user-defined requirements. Alessandro Cilardo et al explains the engineering of ECC as a complex interdisciplinary research field encompassing such fields as mathematics, computer science and electrical engineering[5]. C. J. McIvor et.al [6] introduces a novel hardware architecture for ECC over GF(p). The work presented by Gang Chen presents a high performance EC cryptographic process for general curves over GF(p) [7]. The standard standard specifications for public key cryptography is defined in[8].

A simple tutorial of ECC concept is very well documented and illustrated in the text authored by Williams Stallings et.al [9]. The paper presented by Kevin M. Finnigin

et al outlines a brute-force attack on ECC implemented on UC Berkley' s Tiny OS operating system for wireless sensor networks [10]. The attack exploits the short period of the pseudorandom number generators used by cryptosystem to generate private keys. An efficient and novel approach of a scalar point multiplication method than existing double and add by applying redundant recoding, which originates from radix-4 Booths algorithm was proposed by Sangook Moon[11]. In the paper as proposed by Jaewon Lee [12] presents 3 algorithms to perform scalar multiplication on EC defined over higher characteristic finite fields such as OEA (Optimal Extension Field). Liu Yongliang [13] showed that Aydos et al.' s protocol is vulnerable to man-in the-middle attack from any attacker but not restricted on the inside attacker. They proposed a novel ECC based wireless authentication protocol. A comprehensive coverage of EC field with the in-depth mathematical treatment is given in [14]. Owing to these existing works on ECC and its popularity, it is proposed to implement the crypto system based on ECC for text based application. The proposed work can be extended to XML based application since the future middleware technologies are in the control of XML based documents which is purely based on text.

III. PROPOSED METHOD DESCRIPTION

A general elliptic curve takes the general form as

$$E : y^2 = x^3 + ax + b \quad (1)$$

where x, y are elements of GF(p), and a, b are integer modulo p, satisfying

$$4a^3+27b^2 \neq 0 \pmod p \quad (2)$$

Here 'p' is known as modular prime integer making the EC finite field. An elliptic curve E over GF(p) consist of the solutions(x, y) defined by (1) and (2), along with an additional element called O, which is the point of EC at infinity. The set of points (x, y) are said to be affine coordinate point representation.

The basic EC operations are point addition and point doubling. Elliptic curve cryptographic primitives [8] require scalar point multiplication. Say, given a point P(x, y) on an EC, one needs to compute kP, where k is a positive integer. This is achieved by a series of doubling and addition of P. Say, given k =13, entails the following sequence of operations, by which the efficiency of the scalar multiplication of the points is improved.

P	2P	3P	6P	12P	13P
	Doubling	Addition	Doubling	Doubling	Addition

Let us start with $P(x_P, y_P)$. To determine 2P, P is doubled. This should be an affine point on EC. Use the following equation, which is a tangent to the curve at point P.

$$S = [(3x_P^2 + a)/2y_P] \bmod p \quad (3)$$

Then 2P has affine coordinates (x_R, y_R) given by:

$$x_R = (S^2 - 2\,x_P) \bmod p$$
$$y_R = [S\,(x_P - x_R) - y_P] \bmod p \quad (4)$$

Now to determine 3P, we use addition of points P and 2P, treating 2P=Q. Here P has coordinates (x_P, y_P). Q=2P has coordinates (x_Q, y_Q). Now the slope is:

$$S = [(y_Q - y_P) / (x_Q - x_P)] \bmod p$$
$$P+Q = -R$$
$$x_R = (S^2 - x_P - x_Q) \bmod p$$
$$y_R = (S\,(x_P - x_R) - y_P] \bmod p \quad (5)$$

Therefore we apply doubling and addition depending on a sequence of operations determined for 'k'. Every point (x_R, y_R) evaluated by doubling or addition is an affine point (points on the Elliptic Curve).

IV. PROPOSED ALGORITHM

To do operations with EC points in order to encrypt and decrypt the points are to be generated first. The algorithm '*genPoints*' describes the process of generating the points for the given parameters 'a', 'b', and 'p'. Also the algorithm *'ECC'* describes the process of encryption and decryption on EC field.

Algorithm genPoints (a, b, p)

```
{
  x=0;
  While(x < p)
     y^2=(x^3 + ax + b) mod p;
     if (y^2 is a perfect square in GF(p))
        output(x, sqrt(y)) (x, -sqrt(y) );
     x=x+1;
}
```

Algorithm ECC

```
{
 //Key Distribution
   //Let U_A and U_B be legitimate users
        U_A = {P_A, n_A} //Key pair for U_A
        U_B = {P_B, n_B} //Key pair for U_B
   //Send the Public key of U_B to U_A
        Send(P_B,U_A);
  //Send the Public key of U_A to U_B
        Send (P_A, U_B);

 //Encryption at A
    P_ml = a P_m
    //a: Ascii value of text
    //P_m: random point on EC
    P_B = n_B * G
```

//G is the base point of EC
// n_B is the private key
CipherText={kG,P_{ml}+k*P_B}

//Decryption at B

Let kG be the first point and
P_{ml} + k*P_B be the second point
n_B k G = n_B * first point;
Calculate $P_{ml} = P_{ml} + kP_B - n_BkG$;
Calculate the P_m value from P_{ml}
using discrete logarithm

}

V. IMPLEMENTATION OF THE PROPOSED ALGORITHM

The typical Elliptic Curve is represented by:

$y^2 \bmod 37 = x^3+x+1 \bmod 37$

Points on the curve can be found as shown in Table I.

Table I : Set of Sample Points on EC

(0,1)	(0,36)	(21,25)	(21,12)
(1,15)	(1,22)	(24,14)	(24,23)
(2,14)	(2,23)	(25,0)	(25,0)
(6,36)	(6,1)	(26,18)	(26,19)
(8,15)	(8,22)	(27,8)	(27,29)
(9,31)	(9,6)	(28,15)	(28,22)
(10,7)	(10,30)	(29,31)	(29,6)
(11,14)	(11,23)	(30,24)	(30,13)
(13,18)	(13,19)	(31,36)	(31,1)
(14,24)	(14,13)	(33,9)	(33,28)
(17,11)	(17,26)	(35,18)	(35,19)
(19,16)	(19,21)	(36,6)	(36,31)

The base point G is selected as (0, 1). Base point implies that it has the smallest (x, y) co-ordinates which satisfy the EC. P_m is another affine point, which is picked out of a series of affine points evaluated for the given EC. We could have retained G itself for P_m. However for the purpose of individual identity, we choose P_m to be different from G. Let P_m = (1, 15). The choice of P_m is itself an exercise involving meticulous application of the ECC process on the given EC.

The ECC method requires that we select a random integer k (k<p), which needs to be kept secret. Then kG is evaluated, by a series of additions and doublings, as discussed above. For purpose of this discussion we shall call the source as host A, and the destination as host B. We select the private key of the host B, called n_B. k and n_B can be generated by random number generators to give credibility. For simplicity we shall assume that k = 13, and n_B = 17. The public key of B is evaluated by

$$P_B = n_B G \quad (6)$$

Suppose A wants to encrypt and transmit a character to B, he does the following. Assume that host A wants to transmit the character '#' . Then the ASCII value of the character '#' is 35.

Therefore,

$P_B = n_B G = 17(0,1) = (21,12)$
$P_{ml} = 35(1,15) = (2,14)$

The coordinates of the P_{ml} should fit into the EC. This transformation is done for two purposes. First the single valued ASCII is transformed into a (x, y) co-ordinate of the EC. Second it is completely camouflaged from the would-be hacker. This is actually intended to introduce some level of complexity even before the message is encrypted according to ECC.

As the next step of ECC, we need to evaluate k P_B, here P_B is a public key of user B. Determining this product involves a series of doubling and additions, depending on the value of k. For a quick convergence of the result, we should plan for optimal number of doublings and additions.

$kP_B = 13(21,12) = (21,12)$
$P_{ml}+kP_B = (2,14) + (21,12) = (30,24)$
$kG = 13(0,1) = (0,1)$

The encrypted message is derived by adding P_{ml} with kP_B, that is, $P_{ml} + kP_B$. This yields a set of (x2, y2) coordinates. Then kG is included as the first element (x1, y1) of the encrypted version. Hence the entire encrypted version for purposes of storing or transmission consists of two sets of coordinates as follows:

$Cm = (kG, P_{ml} + kP_B)$
$kG = x1, y1$
$P_{ml} + k P_B = x2, y2$

Encrypted version of the message is: ((0, 1), (30,24), where x1 = 0, y1 = 1, x2 = 30, y2 = 24. Thus far the modified plaintext has been encrypted by application of the ECC method. The modification of the plaintext in conjunction with P_m is a novel idea of this paper.

Recall that kG is represented by (x1, y1) and P_{ml} + kP_B is represented by (x2, y2). In order to pull out P_{ml} from $P_{ml} + kP_B$, B applies his secret key n_B and multiplies kG so that, $n_BkG = kP_B$. Subtract this from $P_{ml} + kP_B$, to get P_{ml} that is, $P_{ml} = P_{ml} + kP_B - n_BkG$.

$n_BkG = 17(0,1) = (21,12)$
$P_{ml} = (30,24) - (21,12) = (2,14)$

This subtraction is another ECC procedure involving doubling and addition. But the only difference is that the negative term will have its y co-ordinate preceded by a minus sign. With this subtle change in mind, the expression of determining the slope, new values of x_R, y_R are the same. Wherever y figures, it is substituted as –y. This will yield P_{ml} . Now apply discrete logarithm concept to get the ASCII value of "#".

#(1,15) = (2,14)

Therefore, # =35. Thus we retrieve the character "#".

VI. DISCUSSIONS & CONCLUSION

In this paper, a text based Elliptic Curve Cryptosystem is implemented. Each character in the message is represented by its ASCII value. Each of these ASCII value is transformed into an affine point on the EC, by using a starting point called P_m. Transformation of the plaintext ASCII value by using an affine point is one of the contributions of this work. The purpose of this transformation is two fold. Firstly a single digit ASCII integer of the character is converted into a set of co-ordinates to fit the EC. Secondly the transformation introduces non-linearity in the character thereby completely camouflaging its identity. This transformed character of the message is encrypted by the ECC technique. Decryption of ECC encrypted message is itself quite a formidable task, unless we have knowledge about the private key 'n_B',the secret integer 'k' and the affine point P_{ml}.

The attractiveness of ECC, compared to RSA, is that it appears to offer better security for a smaller key size, thereby reducing processing overhead. The benefits of this higher-strength per-bit include higher speeds, lower power consumption, bandwidth savings, storage efficiencies, and smaller certificates. These advantages are particularly beneficial in applications where bandwidths, processing capacity, power availability or storage are constrained. Such applications include chip cards, electronic commerce, web servers and cellular telephones. One of the applications that the ECC can be used for is in encryption of large image files. The selection of the primes and the faster multiplication and doubling algorithms are the focus of the research, the image encryption using ECC is a completely new domain and has tremendous scope of research. The work proposed for text encryption process can be easily extended to XML document which is a pure text document after performing the canonicalization process.

ACKNOWLEDGMENT

The authors are grateful to the principal and management of Noorul Islam College of Engineering and MEPCO Schlenk Engineering College for extending all the facilities and constant encouragement for carryingout this research work.

REFERENCES

[1] N.Koblitz, Elliptic Curve Cryptosystems, *Mathematics of Computation*, vol.48, 1987, pp.203-209.

[2] M.Aydos, T.Yanik and C.K.Kog, "High-speed implementation of an ECC based wireless authentication protocol on an ARM microprocessor," *IEE Proc Commun.*, Vol. 148, No. 5, pp. 273 – 279, October 2001.

[3] Kristin Lauter, "The Advantages of Elliptic Cryptography for Wireless Security", *IEEE Wireless Communications*, pp. 62 – 67, Feb. 2006.

[4] Ray C. C. Cheng, Nicolas Jean-baptiste, Wayne Luk, and Peter Y. K Cheung,"Customizable Elliptic Curve Cryptosystems" , *IEEE Trans. On VLSI Systems*, vol.13, no. 9, pp. 1048 – 1059, Sep. 2005.

[5] Alessandro Cilardo, Luigi Coppolino, Nicola Mazzocca, and Luigi Romano,"Elliptic Curve Cryptography Engineering", *Proceedings of the IEEE*, Vol. 94, no. 2, pp. 395 – 406, Feb. 2006.

[6] C. J. McIvor, M. McLoone, and J. V. McCanny, "Hardware elliptic curve cryptographic processor over GF(p)," *IEEE Trans. Circuits Syst.I:* Reg. Papers, vol. 53, no. 9, pp. 1946–1957, Sep. 2006.

[7] Gang Chen, Guoqiang Bai, and Hongyi Chen, " A High-Performance Elliptic Curve Cryptographic Processor for General Curves Over GF(p) Based on a Systolic Arithmetic Unit" , *IEEE Trans. Circuits Syst. - II*: Express Briefs, vol. 54, no. 5, pp. 412 – 416, May. 2007.

[8] Standard specifications for public key cryptography, *IEEE standard*, p1363, 2000.

[9] Williams Stallings, Cryptography and Network Security, *Prentice Hall*, 4th Edition, 2006.

[10] Kevin M. Finnigin, Barry E. Mullins, Richard A. Raines, Henry B.Potoczny, "Cryptanalysis of an elliptic curve cryptosystem for wireless sensor networks," *International journal of security and networks*, Vol. 2, No. 3/4, pp. 260 – 271, 2006.

[11] Sangook Moon, "A Binary Redundant Scalar Point Multiplication In Secure Elliptic Curve Cryptosystems," *International journal of network security*, Vol.3, No.2, PP.132–137, Sept. 2006.

[12] Jaewon Lee, Heeyoul Kim, Younho Lee, Seong-Min Hong, and Hyunsoo Yoon, "Parallelized Scalar Multiplication on Elliptic Curves Defined over Optimal Extension Field," *International journal of network security*, Vol.4, No.1, PP.99–106, Jan. 2007.

[13] Liu Yongliang, Wen Gao, Hongxun Yao, and Xinghua Yu , "Elliptic Curve Cryptography Based Wireless Authentication Protocol ," *International journal ofnetwork security*, Vol.4, No.1, PP.99–106, Jan. 2007.

[14] R.V.Kurja, Kirti Joshi, N.Mohan Kumar, Kapil H Raranape, A.Ramanathan, T.N.Shorey, R.R.Simha, and V.Srinivas, Elliptic Curves, *International Distribution by American Mathematical Society*, 2006.