ISSN 2278-3091
Volume 9, No.3, May - June 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcs246932020.pdf
https://doi.org/10.30534/ijatcse/2020/246932020

Proposing an Elliptic Curve Cryptosystem with the Symmetric Key for Vietnamese Text Encryption and Decryption

Mai Manh Trung [1*], Le Phe Do[2], Le Trung Thuc[3], Dao Thi Phuong Anh[1]
[1]Faculty of Information Technology- the University of Economics Technology for Industries
[2]University of Engineering and Technology, Vietnam National University, Hanoi, Vietnam
[3]East Asia University of Technology

ABSTRACT

The article describes the basic idea of Elliptic curve cryptography (ECC). Elliptic curve arithmetic can be used to develop Elliptic curve coding schemes, including key exchange, encryption, and digital signature. The main attraction of Elliptic curve cryptography compared to RSA is that it provides equivalent security for a smaller key size, which reduces processing costs. To encode the Vietnamese text, we are based on the sound of Vietnamese characters to make a table of these characters' order. We are also based on the algorithm to create the data sequence as the basis of building an encryption algorithm by using Elliptic curves on finite fields with symmetric keys to encrypt this Vietnamese text.

Key words: Algorithm sequence, Decryption, Elliptic curve, Encryption, Symmetric.

1. INTRODUCTION

The study of the Elliptic curves of algebraists and number theorists dates back to the mid-nineteenth century. The Elliptic curve code (ECC) was discovered in 1985 by Neil Koblitz and Victor Miller [1], [2]. Elliptic Curve Cryptography (ECC) can be considered as Elliptic curves of discrete logarithmic cryptosystem. In which the group $\boldsymbol{Z}_{\boldsymbol{P}}^{*}$Is replaced by the group of points on an elliptic curve over a finite field. The mathematical basis for the security of Elliptic curve cipher systems is the computational computation of the Elliptic Curve Discrete Logarithm Problem(ECDLP).

Elliptic curve cryptography system is used in dynamic secure routing link detection [3], in an effective and secure RFID authentication [4], in wireless sensor networks using the number theoreticto transform [5]. In the paper [6], the authors presented the implementation of ECC by first converting the message into an affine point on the Elliptic curve, then applying the data sequence reads algorithm on the plaintext. In encryption and decryption working, from our viewpoint, the input is plaintext text, of which each character is defined as a point on the Elliptic curve. Using symmetric key plays as a random value to encrypt and decode. Applying the idea of data sequence created, we apply reading the sequence of points on the curve. The output is a ciphertext of a sequence of points on an Elliptic curve. We also illustrate the implementation of a cryptographic system based on an Elliptic curve with a symmetric key that corresponds to the chosen Elliptic curve equation:

$$y^2 = x^3 + 19x + 29 \pmod{139} \qquad (1)$$

2.OVER VIEW OF ELLIPTIC CURVER CRYPTOSYTEM

An elliptic curve E over a field R of real numbers is defined by an equation.

$$E: y^2+a_1xy+a_3y=x^3+a_2x^2+a_4x+a_6 \qquad (2)$$

Here a_1, a_2, a_3, a_4, $a_6$are real number belong toR, x and y take on values in the real numbers. If L is an extension field of real numbers, then the set of L-rational points on the elliptic curve E is E(L) $=\{(x, y) \in L \times L: y^2+a_1xy+a_3y - x^3-a_2x^2-a_4x-a_6= 0\} \cup \{\infty\}$ here ∞ the point is at infinitys. Equation (2) is called Weierstrass equation. Here is the elliptic curve E defined over the field of integers K, because a_1, a_2, a_3, a_4, a_6 as integers. If E is defined over the field of integers K, E is also defined over any extension field of K. The condition $4a^3 + 27b^2 \neq 0$ ensures that the elliptic curve is "smooth". The point ∞ is the only point on the line at infinity that satisfies the projective form of the Weierstrass equation [7, 8, 9]. For the purpose of the encryption and decryption using elliptic curves in this paper, it is sufficient to consider the equation of the form

$$y^2 = x^3 + ax + b \qquad (3)$$

For the given values of a and b the plot consists of positive and negative values of y for each value of x. Thus, this curve is symmetric about the x-axis.

2.1Addition formula

There is a rule, called the chord – and - tangent rule, for adding two points on an elliptic curve E(Fp) to give a third elliptic curve point. Together with this addition operation, the set of points E(Fp) forms a group with ∞ serving as its identity. It is the group that is used in the construction of elliptic curve cryptosystems. The addition rule is best explained geometrically. Let P = (x_1, y_1) và Q (x_2, y_2) be two distinct points on an elliptic curve E. If $x_1 = x_2$ and $y_1 = -y_2$ then we define P + Q = ∞. Otherwise, P + Q = $(x_3, y_3) \in E$ where $x_3 = \lambda^2 - x_1 - x_2$, $y_3 = \lambda(x_1 - x_3) - y1$, with:

$$\lambda = \begin{cases} (y_2 - y_1)/(x_2 - x_1), \text{khi } P \neq Q \\ (3{x_1}^2 + a)/(2y_1), \text{khi } P = Q \end{cases}$$

So if P ≠ Q means $x_1 \neq x_2$, we have:

$$\begin{cases} \mathbf{x_3} = \left(\dfrac{\mathbf{y_2} - \mathbf{y_1}}{\mathbf{x_2} - \mathbf{x_1}}\right)^{\mathbf{2}} - \mathbf{x_1} - \mathbf{x_2} \\ \mathbf{y_3} = \left(\dfrac{\mathbf{y_2} - \mathbf{y_1}}{\mathbf{x_2} - \mathbf{x_1}}\right)(\mathbf{x_1} - \mathbf{x_3}) - \mathbf{y_1} \end{cases} \qquad (\mathbf{4})$$

If P = Q means $x_1 = x_2$, we have:

$$\begin{cases} \mathbf{x_3} = \left(\dfrac{\mathbf{3x_1}^{\mathbf{2}} + \mathbf{a}}{\mathbf{2y_1}}\right)^{\mathbf{2}} - \mathbf{2x_1} \\ \mathbf{y_3} = \left(\dfrac{\mathbf{3x_1}^{\mathbf{2}} + \mathbf{a}}{\mathbf{2y_1}}\right)(\mathbf{x_1} - \mathbf{x_3}) - \mathbf{y_1} \end{cases} \qquad (5)$$

Note that the points (x3, y3), (x3, -y3) are also on the E curve and geometrically, the points (x1, y1), (x2, y2), (x3, -y3) is also in a straight line. Besides, define an infinite plus point by itself. P + ∞ = ∞ + P = P.

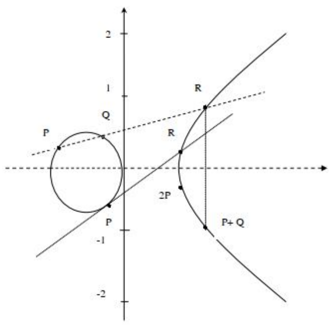

Figure 1: Sum of two points of an elliptic curve

2.2 Point Multiplication

If P is a point on an elliptic curve and k is a positive integer, kP denotes P + P + · · · + P (with k summands). If k < 0, then kP = (-P) + (-P) + · · · +(-P), with |k| summands. To compute kP for a large integer k, it is inefficient to add P to it repeatedly. It is much faster to use successive doubling. For example, to compute 19P, we compute 2P, 4P = 2P+2P, 8P = 4P+4P, 16P = 8P+8P, 19P = 16P+2P+P. This method allows us to compute kP for very large k, say of several hundred digits, very quickly. The only difficulty is that the size of the coordinates of the points increases rapidly if we are working over the rational numbers. However, when we are working over a finite field, for example Fp, this is not a problem because we can continually reduce mod p and keep the numbers involved relatively small. It should be noted that the associative law allows us to make these computations without worrying about what order we use to combine the summands [11, 12].

When the sum of the points P and Q on the elliptic curve E is shown in Figure 1. The result is determined that the point S is obtained by reversing the sign of the y coordinate of the point R, where R is the intersection point of E and the line through P and Q. If P and Q are in the same position, the line is the tangent of E at P. In addition, the sum of the points at infinity and the point P is determined to be exactly the point P.

3. ALGORITHM PROPOSAL

Cryptographic composition: (P, C, E, D, K)

P: Plaintext

C: Ciphertext

E: Encode function

D: Decode funtion

K: Key

Algorithm for generating the sequence

Step 1:Determine the total number of points on an elliptic curve, find P as a point generator.

Step 2:Convert the total number of points (n) in base 3. Therefore, m which is the number of digits of the sequence of numbers converted is found. For example,when n = 37, we get sequence number 1101. We have m = 4. And convert each element from 0 to n in base 3.

Step 3: Set the matrix M with dimensions (n + 1) * m. Where n + 1 is the number of rows, n is the total number of points in curve E, m is the number of columns (m is the number of digits in a row). We have the matrix.

$$M = \begin{pmatrix} a_{0,0}a_{0,1} \dots \dots a_{0,m} \\ a_{1,0}a_{1,1} \dots \dots a_{1,m} \\ a_{2,0}a_{2,1} \dots \dots a_{2,m} \\ \dots \dots \\ a_{n,0}a_{n,1} \dots \dots a_{n,m} \end{pmatrix}$$

With n = 37 we have the size of the matrix M is 38 x 4

$$M = \begin{pmatrix} 0000 \\ 0001 \\ 0002 \\ 0010 \\ \dots \dots \\ 1101 \end{pmatrix}$$

Step 4: Circularly shifteach row of M by one element to the right

$[a_{i,0}a_{i,1}\ a_{i,2}\dots.a_{i,m-1}] = [a_{i,m-1}a_{i,0}a_{i,1}\ a_{i,2}\ \dots.a_{i,m-2}]$

Step 5: The sequence formed is :

$S:[S_0 = [a_{0,m-1}\ a_{0,0}a_{0,1}a_{0,2}\dots a_{0,m-2}],\ S_1 = [a_{1,m-1}\ \ a_{1,0}\ \ a_{1,1}\ \ a_{1,2}\dots a_{2,m-2}],\dots,\ S_n = [a_{n,m-1}\ \ a_{n,0}\ \ a_{n,1}\ a_{n,2}\dots a_{n,m-2}]]$

Encryption:

Step 6: Choose random key value K.

Step 7: Encode function

$$C = E(P) = [(P_i + K) \bmod (n)]P \qquad (6)$$

Step 8: Read the sequence generated from step 5.

Decryption:

Step 9: Consider the m-digit segment of the coded sequence then circularly shift this sequence of m digits by one element to the left and convert the sequence in base 3 to the decimal.We find the coordinates of the point.

Step 10: Decode function

$$P = D(C) = [(C_i - K) \bmod (n)]P \qquad (7)$$

Step 11: Repeat step 9 and 10 until the digits sequence is ended.

In which parameters in (6), (7):

P_i: The position of the plaintext character

Ci : The position of the ciphertext character

E:Encode function

D: Decode funtion

K: Key K is a random positive integer.

n: The total number of points on the elliptic curve.

P: a point generator of an elliptic curve.

4.IMPLEMENTATION OF THE PROPOSED ALGORITHM

Party A sends Party B a plaintext (input document) as "VIỆT NAM". To ensure the confidentiality of the transmission process, Party A will encrypt the plaintext before sending it on the channel. The encoding process is presented as follows:

Step 1: Determine the total number of points on an elliptic curve, find P as a point generator.

For curve E at (1) we have 127 points on the curve including the infinity. We find the point P = (1, 7). Using the formula (4) and formula (5) calculates points on the curve as shown in Table 1.

Table 1:A set of all points on ECC

∞	(1, 7)	(26, 53)	(50, 103)
(133, 106)	(49, 96)	(16, 26)	(47, 83)
(89, 41)	(34, 15)	(71, 35)	(6, 130)
(120, 68)	(118, 51)	(24, 39)	(4, 13)
(138, 136)	(25, 12)	(10, 78)	(101, 69)
43, 17)	(125, 76)	(112, 31)	(109, 26)
(80, 19)	(8, 50)	(111, 52)	(78, 31)
(48, 65)	(14, 113)	(63, 129)	(124, 20)
(0, 53)	(30, 76)	(113, 86)	(90, 135)
(57, 103)	(135, 81)	(33, 133)	(32, 36)
(117, 19)	(62, 49)	(73, 128)	(39, 72)
(82, 41)	(137, 20)	(97, 131)	(96, 73)
(79, 110)	(88, 108)	(31, 78)	(17, 20)
(107, 98)	(94, 22)	(123, 63)	(60, 32)
(20, 25)	(3, 35)	(53, 99)	(98, 78)
(65, 35)	(115, 30)	(77, 24)	(81, 19)
(81, 120)	(77, 115)	(115, 109)	(65, 104)
(98, 61)	(53, 40)	(3, 104)	(20, 114)
(60, 107)	(123, 76)	(94, 117)	(107, 41)
(17, 119)	(31, 61)	(88, 31)	(79, 29)
(96, 66)	(97, 8)	(137, 119)	(82, 98)
(39, 67)	(73, 11)	(62, 90)	(117, 120)
(32, 103)	(33, 6)	(135, 58)	(57, 36)
(90, 4)	(113, 53)	(30, 63)	(0, 86)
(124, 119)	(63, 10)	(14, 26)	(48, 74)
(78, 108)	(111, 87)	(8, 89)	(80, 120)
(109, 113)	(112, 108)	(125, 63)	(43, 122)
(101, 70)	(10, 61)	(25, 127)	(138, 3)
(4, 126)	(24, 100)	(118, 88)	(120, 71)
(6, 9)	(71, 104)	(34, 124)	(89, 98)
(47, 56)	(16, 113)	(49, 43)	(133, 33)
(50, 36)	(26, 86)	(1, 132)	

Step 2: Convert the total number of points (n) in base 3. Therefore, m which is the number of digits of the sequence of numbers converted is found.

Determine the total of the curve is 127 points, that is, n = 127. A Converts n to base 3. We get the sequence number 11201. The result is m = 5.

Step 3: Set the matrix M with dimensions 38 x 6

$$M = \begin{pmatrix} 00000 \\ 00001 \\ 00002 \\ 00010 \\ \ldots\ldots \\ 11201 \end{pmatrix}$$

Step 4: Circularly shift each row of M by one element to the right. We have the new matrix M^*.

$$M^* = \begin{pmatrix} 00000 \\ 10000 \\ 20000 \\ 00001 \\ \ldots\ldots \\ 11201 \end{pmatrix}$$

Step 5: The sequence formed is:

[00000], [10000], [20000], [00001], [10001], [20001], [00002], [10002], [20002], [00010], [10010], [20010], [00011], [10011], [20011], [00012], [10012], [20012], [00020], [10020], [20020], [00021], [10021], [20021], [00022], [10022], [20022], [00100], [10100], [20100], [00101],[10101],[20101],[00102],[10102],[20102], [00110],[10110],[20110],[00111],[10111],[20111], [00112],[10112],[20112],[00120],[10120],[20120], [00121],[10121],[20121],[00122],[10122],[20122], [00200],[10200],[20200],[00201],[10201],[20201], [00202],[10202],[20202],[00210],[10210],[20210], [00211],[10211],[20211],[00212],[10212],[20212], [00220],[10220],[20220],[00221],[10221],[20221], [00222],[10222],[20222],[01000],[11000],[21000], [01001],[11001],[21001],[01002],[11002],[21002], [01010],[11010],[21010],[01011],[11011],[21011], [01012],[11012],[21012],[01020],[11020],[21020], [01021],[11021],[21021],[01022],[11022],[21022], [01100],[11100],[21100],[01101],[11101],[21101], [01102],[11102],[21102],[01110],[11110],[21110], [01111],[11111],[21111],[01112],[11112],[21112], [01120],[11120]

Encryption:

Step 6: Choose random key value, K. = 6.

Step 7, 8: Encode function, read digit sequence

Table 2: Characters corresponding to points on the curve considered from point P

∞ *	(1, 7) a	(26, 53) à	(50, 103) ã
(133, 106) ả	(49, 96) á	(16, 26) ạ	(47, 83) ă
(89, 41) ằ	(34, 15) ẵ	(71, 35) ẳ	(6, 130) ắ
(120, 68) ặ	(118, 51) â	(24, 39) ầ	(4, 13) ẫ
(138, 136) ẩ	(25, 12) ấ	(10, 78) ậ	(101, 69) b
(43, 17) c	(125, 76) d	(112, 31) đ	(109, 26) e
(80, 19) è	(8, 50) ẽ	(111, 52) ẻ	(78, 31) é
(48, 65) ẹ	(14, 113) ê	(63, 129) ề	(124, 20) ễ
(0, 53) ể	(30, 76) ế	(113, 86) ệ	(90, 135) f
(57, 103) g	(135, 81) h	(33, 133) i	(32, 36) ì
(117, 19) ĩ	(62, 49) ỉ	(73, 128) í	(39, 72) ị
(82, 41) k	(137, 20) l	(97, 131) m	(96, 73) n
(79, 110) o	(88, 108) ò	(31, 78) õ	(17, 20) ỏ
(107, 98) ó	(94, 22) ọ	(123, 63) ô	(60, 32) ồ
(20, 25) ỗ	(3, 35) ổ	(53, 99) ố	(98, 78) ộ
(65, 35) ơ	(115, 30) ờ	(77, 24) ỡ	(81, 19) ở
(81, 120) ớ	(77, 115) ợ	(115, 109) p	(65, 104) q
(98, 61) r	(53, 40) s	(3, 104) t	(20, 114) u
(60, 107) ù	(123, 76) ũ	(94, 117) ủ	(107, 41) ú
(17, 119) ụ	(31, 61) ư	(88, 31) ừ	(79, 29) ữ
(96, 66) ử	(97, 8) ứ	(137, 119) ự	(82, 98) v
(39, 67) x	(73, 11) y	(62, 90) ỳ	(117, 120) ỹ
(32, 103) ỷ	(33, 6) ý	(135, 58) ỵ	(57, 36) z

(90, 4) 0	(113, 53) 1	(30, 63) 2	(0, 86) 3
(124, 119) 4	(63, 10) 5	(14, 26) 6	(48, 74) 7
(78, 108) 8	(111, 87) 9	(8, 89) dấu cách	(80, 120) .
(109, 113) ,	(112, 108) ;	(125, 63) ?	(43, 122) !
(101, 70) @	(10, 61) $	(25, 127) %	(138, 3) ^
(4, 126) &	(24, 100) -	(118, 88) +	(120, 71) (
(6, 9))	(71, 104) [	(34, 124)]	(89, 98) {
(47, 56) }	(16, 113) =	(49, 43) |	(133, 33) <
(50, 36) >	(26, 86) ‘	(1, 132) :	

- Clear points: According to Table 2, we get plaintext characters corresponding to the number of points for the results in Table 3.

Table 3: Characters corresponding to points on curves

V	I	Ệ	T		N	A	M
(82, 98)	(33, 133)	(113, 86)	(3, 104)	(8, 89)	(96, 73)	(1, 7)	(97, 131)

- Apply: $C = E(P) = [(P_i + K) \bmod (n)]P$

Consider the character ‘V’: We get Pi of ‘V’ to 83P for the point (82, 98)

We have $C = [(83 + 6) \bmod 127]P = 89P = 89(1, 7) = (33, 6)$. For x = 33 and y = 6, read the sequence of numbers in the matrix M in step 5. We have00102, 00002.

Similarly consider the character ‘I’: We get Pi of ‘I’ is 38P corresponding to the point (33, 133)

We have $C = [(38+6) \bmod 127]P = 44P = 44(1, 7) = (82, 41)$. For x = 82 and y = 41,read the sequence of numbers in the matrix M in step 5. We have11000, 20111.

Similar to the remaining characters, we get the result as in Table 4.

Table 4: Table of symbols after encryption

Character	Clear points	Point encryption	Encryption sequence
V	(82, 98)	(33, 6)	00102 00002
I	(33, 133)	(82, 41)	11000 20111
Ệ	(113, 86)	(117, 19)	01110 10020
T	(3, 104)	(17, 119)	20012 21110
	(8, 89)	(101, 70)	21020 10212
N	96, 73)	(94, 22)	11011 10021
A	(1, 7)	(47, 83)	20120 21000
M	(97, 131)	(107, 98)	21022 21012

So the ciphertext after encoding is 00102 00002 11000 20111 01110 10020 20012 21110 21020 10212 11011 10021 20120 21000 21022 21012.

This ciphertext is sent on the channel to party B.

Decryption:

When Party B receives the ciphertext and decrypts it as follows:

Step 9: Convert to decimal

With m = 5, considering the string 00102 shifts 1 bit to the left, we get 01020 and then convert it to decimal.

$01020_{(3)} = 0*3^4 + 1*3^3 + 0*3^2 + 2*3^1 + 0*3^0 = 33$.

Similarly, considering the string 00002 shifts 1 bit to the left, we get 00020 and then we convert it to $00020_{(3)} = 6$ so we get a point (33, 6).

We compute with the remaining sequence, we determine (82, 41); (117, 19); (17, 119); (101, 70); (94, 22); (47, 83); (107, 98).

Step 10: Decode function

- The key to decrypt is 6 (K = 6)
- Apply:$P = D(C) = [(C_i - K) \bmod (n)]P$

Considering the point (33, 6) with position 89P on the curve, we have:

$P = [(89 - 6) \bmod 127]P = 83P = 83(1, 7) = (82, 89)$ This point corresponds to the letter 'V'

Similarly, considering the point (82, 41) with position 44P on the curve, we have:

$P = [(44 - 6) \bmod 127]P = 38P = 38(1, 7) = (33, 133)$ This point corresponds to the letter ‘I’.

Similarly to the remaining points, we get the decoded results as in Table 5:

Table 5: Table of decryption results

Sequence	Reversal of sequence	Decryption	Character
00102 00002	(33, 6)	(82, 98)	V
11000 20111	(82, 41)	(33, 133)	I
01110 10020	(117, 19)	(113, 86)	Ệ
20012 21110	(17, 119)	(3, 104)	T
21020 10212	(101, 70)	(8, 89)	
11011 10021	(94, 22)	96, 73)	N
20120 21000	(47, 83)	(1, 7)	A
21022 21012	(107, 98)	(97, 131)	M

Therefore, we find the original plaintext is:VIỆT NAM

5. THE PROGRAM INSTALLATION

The algorithm installed on the device with hardware configuration is the Intel CPU (R) Core (TM) i5, 2.5 GHz; RAM: 4GB; HDD: 500 GB; And software with Windows 10 Operating System, Visual Studio .NET - 2019 programming environment.

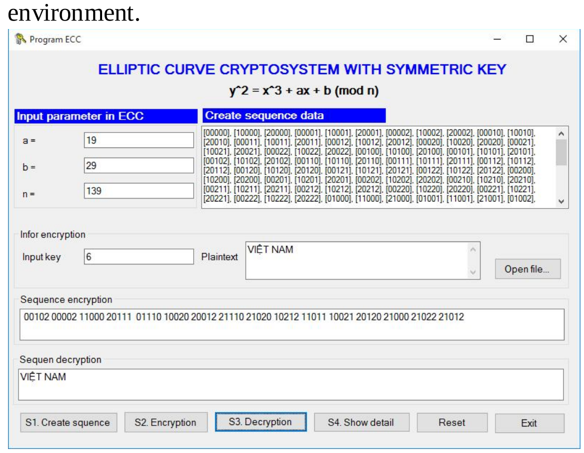

Figure 2: Application programming interface.

The program that implements the encryption and decryption algorithm on the Elliptic curve is implemented on the programming language C # in Visual Studio .NET -2019 with the interface in Figure 2. The program runs and brings correct results with the algorithm presented above.

6. CONCLUSION

The communicating parties agree upon to use an elliptic curve and a point generator P on the elliptic curve in the encryption algorithm proposed. The security of the Elliptic Curve Cryptography depends on the difficulty of finding the value of k, actually kP consists of the fact that k is a large number and P is a random point on the elliptic curve. This is the Elliptic Curve Discrete Logarithmic Problem. The security depends on m which is the number of digits in a group of numbers. Besides, m is long or short depending on the total number of points (n) on the Elliptic curve. Nevertheless, n depends on the parameter of the curve. The elliptic curve parameters for cryptographic schemes should be carefully chosen in order to resist all known attacks of Elliptic Curve Discrete Logarithmic Problem. Therefore, the encryption method proposed here provides an adequate security against breaking code with relatively low computing costs. The algorithm is installed and tested on the C # programming language to give the correct results according to the proposed algorithm.

REFERENCES

1. Darrel Hankerson, Alfered Menezes, Scott Vanstone. **A Gide to elliptic curve Cryptography**.*Springer*, 2004.
2. V.Miller. **Uses of Elliptic curves in Cryptography**.*In advances in Cryptography (CRYPTO 1985), Springer* LNCS 218,417-4 26, 1985.
3. S.Sugantha Priya, Dr.M.Mohanraj. **A Review on Secure Elliptic Curve Cryptography (ECC) and Dynamic Secure Routing Link Path Detection Algorithm (DSRLP) Under Jamming Attack**. ISSN: 0474-9030, Vol-68-Issue-30, February, 2020.
4. Negin Dinarvand, Hamid Barati. **An efficient and secure RFID authentication protocol using elliptic curve cryptography**. *Springer Science+Business Media*, LLC, 2017
https://doi.org/10.1007/s11276-017-1565-3
5. Utku Gulen, Selcuk Baktir. **Elliptic Curve Cryptography for Wireless Sensor Networks Using the Number Theoretic Transform**. *Journal-sensors*, Published: 9 March, 2020.
6. F. Amounas and E. H. El Kinani. **ECC Encryption and Decryption with a Data Sequence**. *Applied Mathematical Sciences*, Vol. 6, no. 101, 5039 – 5047, 2012.
7. Alfred J. Menezes and Scott A. Vanstone. **Elliptic Curve Cryptosystems and their implementations**. *Journal of Cryptology*, Volume-6, Number-4, pages 209-224, 1993.
https://doi.org/10.1007/BF00203817
8. Enge A. **Elliptic curves and their applications to cryptography**. *Norwell*, MA: Kulwer Academic publishers, 1999.
9. Neil Koblitz. **An Elliptic Curve implementation of the finite field digital signature algorithm**. in *Advances in cryptology*,(CRYPTO 1998), *Springer Lecture Notes in Computer Science*, 1462, 327-337,1998.
https://doi.org/10.1007/BFb0055739
10. Irma T. Plata, Edward B. Panganiban, Bryan B. Bartolome, **ASecurity Approach for File Management System using Data Encryption Standard (DES) algorithm,** *International Journal of Advanced Trends in Computer Science and Engineering, Volume 8, No.5, September - October 2019*.
https://doi.org/10.30534/ijatcse/2019/30852019
11. Abdul Male Ssentumbwe1, ByeongMan2Kim and HyunAh Lee, **English to Luganda SMT: Ganda Noun Class Prefix Segmentation for Enriched Machine Translation**, *International Journal of Advanced Trends in Computer Science and Engineering, Volume 8, No.5, September - October 2019*.
https://doi.org/10.30534/ijatcse/2019/08852019