VNU Journal of Science:
Computer Science and Communication Engineering

Journal homepage: http://www.jcsce.vnu.edu.vn/index.php/jcsce

Original Article

Building an Elliptic Curve Cryptography to Encode and Decode Vietnamese Texts

Mai Manh Trung[1,3,*], Do Trung Tuan[2], Le Phe Do[3]

[1]*University of Economics Technology for Industries, 456 Minh Khai, Hai Ba Trung, Hanoi, Vietnam*
[2]*VNU University of Science, 334 Nguyen Trai, Thanh Xuan, Hanoi, Vietnam*
[3]*VNU University of Engineering and Technology, 144 Xuan Thuy, Cau Giay, Hanoi, Vietnam*

Received 03 June 2020
Revised 30 September 2020; Accepted 15 November 2020

Abstract: This article presents building an Elliptic curve cryptography and using it to encode and decode Vietnamese text. Here we have illustrated the prime number p = 151 in the future, which will use a large prime number. We consider an elliptic curve with a total score of 172 points. Encode and decode with standard Vietnamese text and combine with the special characters in ASCII code. The program is designed and installed and on the C# environment to give the correct result of the encryption algorithm.

Keywords: Data sequence, Decryption, Discrete logarithm, Elliptic curve, Elliptic curve cryptosystem, Encryption, Public key.

1. Introduction[*]

The elliptic curve cipher systems (ECC) invented by Neal Koblitz [1] and Victor Miller [2] in 1985 can be considered as elliptic curves of discrete logarithmic cryptographic systems, in which the group ∗ is replaced by the group of points on an elliptic curve over a finite field. The mathematical basis for the security of elliptic curve cryptographic systems is the computational computation of the discrete elliptic logarithm problem (ECDLP).

The Elliptic curve cryptography system is used in dynamic secure routing link detection [3], in an effective and secure RFID authentication [4], as well as in wireless sensor networks using the number theoretic to 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 knapsack algorithm on ECC encrypted message over the finite field GF(p). Based on the total number of points of the elliptic curve found, we will convert them into matrices with size (n +1) * m. Among the matrices, n + 1 is the number of rows; m is the number of columns of the matrix. However, when we convert to get a sequence of numbers

[*] Corresponding author.
E-mail address: mmtrung@uneti.edu.vn

https://doi.org/10.25073/2588-1086/vnucsce.255

as the basis for encoding and decoding, n is the number of points in the curve; m is the number of digits in a row.

During working on encryption and decryption in terms of Vietnamese plaintext input, each character is identified as a point on the curve. The output of the ECC algorithm is provided with the string values generated by the provided algorithm. Therefore, this article proposes more precisely an elliptic curve cryptography system to create a data series along with its application to the output of a cryptographic system. We also illustrate the implementation of the cryptographic system based on a characteristic elliptic curve using the following equation:

$$y^2 = x^3 - 5x + 7 \pmod{151} \quad (1)$$

We want to optimize the encoding and transmission of Vietnamese textual information. It is proposed in the article [6], [7] to encode in English texts, but the encryption here is used in Vietnamese texts. The main difference is that Vietnamese texts have more syllables, tones, especially more characters than English texts. The alphabetical arrangement in Table 4 has been consulted by experts [8] from The Institute of Linguistics - Vietnam Academy of Social Sciences so far.

2. Overview of Elliptic curve Cryptosystem

RSA cryptography is a widely used public-key algorithm, but cryptography based on Elliptic Curve (ECC) can replace RSA for a higher level of security and processing speed. The advantage of ECC is that it uses a key of smaller length than RSA as mentioned in Table 1 that increases the processing speed significantly, because the number of operations used to encode and decode is less. and lower computational capabilities are required. Therefore, they increase speed but decrease energy used in encoding and decoding. The comparison of the key sizes between conventional and public-key encryption at the same level of security is displayed in Table 2.

Table 1. Key length for public-key and symmetric-key cryptography [11].

Symmetric-key	ECC	RSA/DLP
64 bit	128 bit	700 bit
80 bit	160 bit	1024 bit
128 bit	256 bit	2048-3072 bit

Table 2. Comparison between RSA and ECC key sizes at the same security level

Time it takes Key (unit: year)	Key size		Key size ratio RSA: ECC
	RSA/DSA	ECC	
10^4	512	106	5:1
10^8	768	132	6:1
10^{11}	1024	160	7:1
10^{20}	2048	210	10:1
10^{78}	21000	600	35:1

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

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

Hereby: a_1, a_2, a_3, a_4, a_6 are real number belonging to R, x and y take on values in the real numbers. If L is an extension field of real numbers, 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. In this way, the elliptic curve E is defined over the field of integers K, because a_1, a_2, a_3, a_4, a_6 are 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 [9, 10]. In the present paper for the purpose of the encryption and decryption using elliptic curves, it is sufficient to consider the equation of the form $y^2 = x^3 + ax + b$ (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.

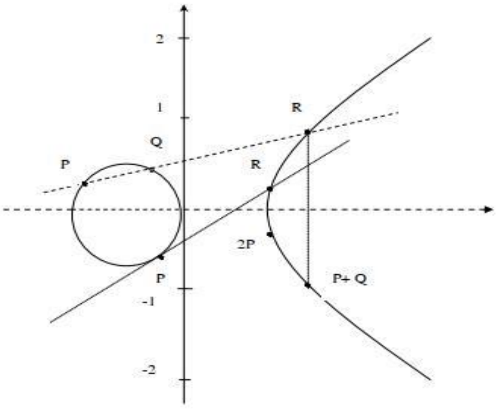

Figure 1. Summation of two points of an elliptic curve

To be safe, p must be large, due to the Demo program, we take a small number of p to illustrate. Reasons for choosing equation (1): Conditions for the equation to be an Elliptic curve satisfying $4a^3 + 27b^2 \neq 0$, with the parameters of the equation (1) we have $4. (- 5)^3 + 27. (7)^2 = 823 \neq 0$. This is the equation for an elliptic curve. Also, choose the parameter at random; find a total score of 172. Each point corresponds to a letter. Choosing the above parameter is a challenge for the researchers because the total number of points found, i.e. 172 is not a prime number. For a prime number, however, every point is an arising one. Thus, the researchers need one more step to find the arising that causes the increasing complexity of cryptanalysis.

2.1. Addition 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 this 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 $P + Q = \infty$. Otherwise, $P + Q = (x_3, y_3) \in E$ where:

$$\begin{cases} x_3 = \lambda^2 - x_1 - x_2 \\ y_3 = \lambda(x_1 - x_3) - y_1 \end{cases}$$

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 $P \neq Q$ means $X_1 \neq X_2$, we have:

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

If P = Q mean $X_1 = X_2$ we have:

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

Note that the points (x_3, y_3), $(x_3, -y_3)$ are also on the E curve and geometrically, the points (x_1, y_1), (x_2, y_2), $(x_3, -y_3)$ is also on a straight line. Besides, define an infinite plus point by itself. $P + \infty = \infty + P = P$.

2.2. Point Multiplication

Multiplying an integer k by a point P on an elliptic curve E is a Q point determined by adding k times the point P and of course $Q \in E$: k × P = P + P + P …… + P (k addition of point P). So if G is a point in an elliptic curve E, then it is easy to determine the point Q = k x G for every k positive integer.

When the sum of the points P and Q on the elliptic curve E is shown in Figure 1, the result is visibly shown that the point S is obtained by reversing the sign of the y coordinate of the point R. In fact, 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. Also, the sum of the points is at infinity and the point P is determined to be exactly the point P.

3. Description of the Algorithm

To implement the encoding Vietnamese text, it is necessary to use two algorithms, including the algorithm (1) that creates a data sequence and the algorithm (2) that uses an elliptic curve cryptosystem to encrypt and decode.

3.1. Algorithm (1) 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:

$$M = \begin{pmatrix} a_{0,0}\ a_{0,1} \cdots a_{0,m-1} \\ a_{1,0}\ a_{1,1} \cdots a_{1,m-1} \\ . \quad\quad . \\ . \quad\quad . \\ . \quad\quad . \\ a_{n,0}\ a_{n,1} \cdots a_{n,m-1} \end{pmatrix}$$

With n = 39 we have the size of the matrix M is 40 x 4

$$M = \begin{pmatrix} 0000 \\ 0001 \\ 0002 \\ 0010 \\ \dots\dots \\ 1110 \end{pmatrix}$$

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

$$\left[a_{i,0}\ a_{i,1}\ a_{i,2} \dots a_{i,m-1}\right]$$

$$\Rightarrow \left[a_{i,m-1}\ a_{i,0}\ a_{i,1}\ a_{i,2} \dots a_{i,m-2}\right]$$

Step 5: The sequence formed is:

$$S : \left[S_0 = \left[a_{0,m-1}\ a_{0,0}\, a_{0,1}\ a_{0,2} \dots a_{0,m-2}\right], S_1 \right.$$

$$= \left[a_{1,m-1}\ a_{1,0}\ a_{1,2} \dots a_{1,m-2}\right], \dots, S_1$$

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

3.2. Algorithm (2) ECC by Using a Sequence Generated

It is supposed that we have some elliptic curves E defined on a finite field GF(p). One point P ∈ E is known publicly, such as embedded systems m→ P_m; each character of Vietnamese text corresponds to 1 point on the elliptic curve. Afterwards, when the sender (A) wants to communicate secretly with the recipient (B), A sends B the ciphertext which regularly proceeds with the following steps:

Encryption:

Step 1: B chooses a random integer , and publishes the point P (while remains secret).
Step 2: A chooses her own random integer *l* and computes the pair of points:

$$P_1(x_1, y_1) = lP \quad (6)$$

$$P_2(x_2, y_2) = P_i + l(\alpha P) \quad (7)$$

Step 3: Read the sequence generated from an algorithm (1)

Step 4: Calculate $S(x_1, y_1)$ and $S(x_2, y_2)$ with S considered as a corresponding sequence value in step 3. Then, the ciphertext is as following:

$$Cm = (S(x_1, y_1), S(x_2, y_2))$$

Step 5: Based on the results of calculating S in the step 4, we know the ordinal number of the points. To read the data sequence in the step 5 algorithm (1), we have ciphertext.

Encryption:

To decrypt the message, B knows the sequence of Si, his own secret , the base point P, and a; b; p - values of the ECC. B receives the encrypted message Cm = $(S(x_1,y_1); S(x_2, y_2))$

Step 1: Transform Cm into groups of 2m

Step 2: Extract a group of m digits in sequence of step 1.

Step 3: Circularly shift this sequence of m digits by one element to the left.

Step 4: Convert a sequence to decimal form, and store a value in *k*.

For example: 00110 will be in the form: $0 * 3^4 + 0 * 3^3 + 1 * 3^2 + 1 * 3^1 + 0 * 3^0 = 12 \Rightarrow k = 12$

Step 5: Obtain (k+1) P from pre-computed and stored point (k+1)P = (x_1, y_1).

Step 6: The procedure is repeated for the next element of the sequence of step 2 for the recovery of $S(x_2, y_2)$.

Step 7: The procedure is repeated for the next groups of step 1, which is not run earlier. Recall that lP represented by (x_1, y_1) and P_i + l(αP) is represented by $S(x_2, y_2)$. In order to pull out P_i from P_i + l(αP), B applies his secret key and calculates (lP) from the first part of the pair, subtracts it from the second part to obtain:

P_i + l(αP) - α(lP) = P_i + l(αP) - lαP = P_i, and reverses the embedding to get back the Vietnamese text.

4. Implement Vietnamese Text Encryption of the Above Algorithm

The elliptic curve being used here is given by the following equation (1)

The base point P is selected as (3, 64). The table below represents a set of all points in the curve:

Table 3. A set of all point on EC

(3,64)	(56,118)	(134,39)	(25,135_	(120,32)	(78,123	(102,142
(86,136)	(130,106)	(106,3)	(29,98)	(26,115)	(39,40	(143,130
(41,13)	(47,3)	(103,6)	(64,65)	(70,76)	(115,103	(34,118
(27,128)	(61,33)	(20,31)	(21,42)	(149,148)	(94,136)	(140,40)
(51,91)	(46,110)	(49,101)	(104,55)	(18,39)	(116,74)	(122,15)
(150,112)	(142,94)	(50,129)	(63,49)	(19,91)	(123,111)	(13,146)
(15,49)	(144,150)	(84,89)	(31,97)	(40,38)	(85,63)	(80,64)
(68,87)	(105,30)	(127,78)	(118,18)	(6,58)	(146,71)	(24,11)
(129,103)	(113,17)	(2,55)	(76,34)	(65,98)	(91,29)	(89,56)
(88,123)	(145,127)	(52,128)	(135,81)	(87,35)	(81,60)	(136,28)
(16,61)	(57,53)	(100,76)	(35,27)	(131,84)	(62,7)	(58,48)
(111,69)	(54,114)	(132,76)	(45,55)	(73,102)	(114,44)	(72,128)
(97,2)	(139,0)	(97,149)	(72,23)	(114,107)	(73,49)	(45,96)
(132,75)	(54,37)	(111,82)	(58,103)	(62,144)	(131,67)	(35,124)
(100,75)	(57,98)	(16,90)	(136,123)	(81,91)	(87,116)	(135,70)
(52,23)	(145,24)	(88,28)	(89,95)	(91,122)	(65,53)	(76,117)
(2,96)	(113,134)	(129,48)	(24,140)	(146,80)	(6,93)	(118,133)
(127,73)	(105,121)	(68,64)	(80,87)	(85,88)	(40,113)	(31,54)
(84,62)	(144,1)	(15,102)	(13,5)	(123,40)	(19,60)	(63,102)
(50,22)	(142,57)	(150,39)	(122,136)	(116,77)	(18,112)	(104,96)
(49,50)	(46,41)	(51,60)	(140,111)	(94,15)	(149,3)	(21,109)
(20,120)	(61,118)	(27,23)	(34,33)	(115,48)	(70,75)	(64,86)
(103,145)	(47,148)	(41,138)	(143,21)	(39,111)	(26,36)	(29,53)
(106,148)	(130,45)	(86,15)	(102,9)	(78,28)	(120,119)	(25,16)
(134,112)	(56,33)	(3,87)	∞			

Curve (1) contains 172 points with the generator point P viewed as the base point as well, which produces points on an elliptic curve representing letters, numbers and other special characters. Therefore, from Table 4, we have the letter “a” corresponding to P, the character ‘à’ corresponds to 2P,..., 133P is the space.

Table 4. A set of alphanumeric characters and other characters

Stt	1	2	3	4	5	6	7	8	9	10
1	a	à	ã	ả	á	ạ	ă	ằ	ẵ	ẳ
2	ắ	ặ	â	ầ	ẫ	ẩ	ấ	ậ	b	c
3	d	đ	e	è	ẽ	ẻ	é	ẹ	ê	ề
4	ễ	ể	ế	ệ	f	g	h	i	ì	ĩ
5	ỉ	í	ị	k	l	m	n	o	ò	õ
6	ỏ	ó	ọ	ô	ồ	ỗ	ổ	ố	ộ	ơ
7	ờ	ỡ	ở	ớ	ợ	p	q	r	s	t
8	u	ù	ũ	ủ	ú	ụ	ư	ừ	ữ	ử
9	ứ	ự	v	x	y	ỳ	ỹ	ỷ	ý	ỵ
10	z	0	1	2	3	4	5	6	7	8
11	9	-	=	[	]	\	;	'	,	.
12	/	!	@	#	$	%	^	&	*	(
13	)	_	+	{	}	|	:	"	<	>
14	?	±								

As can be seen, there are 133 characters including spaces. The order of alphabetic characters is referenced in specialists [8], the phonetic Room-Vietnam Institute of Linguistics.

It is assumed that A wants to send B the password with the Vietnamese password (plaintext) as “Hòa Bình”. To ensure confidentiality during transmission, the plaintext above is encrypted before sending. The encryption process is carried out as follows:

Step 1: Generate the data sequence

P is a point generator with order n = 172. Then m = 5.

Convert a sequence 0 to 172 to the form:
00000, 00001, 00002, 00010, …, 20100, 20101

Represent the above form in (173*5) matrix:

$$\boldsymbol{M} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 1 & 0 \\ & \dots & \dots & \dots & \dots \\ 2 & 0 & 1 & 0 & 0 \\ 2 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Circularly shift each row of M by one element to the right. We have a new matrix:

$$\boldsymbol{M}^* = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ & \dots & \dots & \dots & \dots \\ 0 & 2 & 0 & 1 & 0 \\ 1 & 2 & 0 & 1 & 0 \end{bmatrix}$$

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], [21120], [01121], [11121], [21121], [01122], [11122], [21122], [01200], [11200], [21200], [01201], [11201], [21201], [01202], [11202], [21202], [01210], [11210], [21210], [01211], [11211], [21211], [01212], [11212], [21212], [01220], [11220], [21220], [01221], [11221], [21221], [01222], [11222], [21222], [02000], [12000], [22000], [02001], [12001], [22001], [02002], [12002], [22002], [02010], [12010].

Step 2: Encryption - Decryption

Encryption:

Hence we shall assume that l = 17 và = 28.

Plaintext is “H”, Therefore:

lP = 17(3, 64) = (103, 6).

P_B = P = 28(3, 64) = (140, 40)

P_i= (142, 94)

lP_B = 17(140, 40) = (19, 60)

P_i + lP_B = (134, 112)

Encrypted version of the message is: Cm = (S(103, 6), S(134, 112)), here x_1 = 103, y_1 = 6 và x_2 = 134, y_2 = 112.

Apply algorithm (1) for generating the sequence S: S (103, 6) = 10012 and S (134, 112) = 02002. Therefore, the character 'H' is converted to a ciphertext: 1001202002. Similarly, regarding the remaining characters, we find the ciphertext transmitted: 10012020021001220002100120112 2100121101010012012121001222002100120000 21001202002

Table 5. Encoding character representation

Character	PointP_i	**Encryption before data sequence applied $C_m=(lP, P_i+lP_b)$**	**Encryption After data sequence applied $Cm=(S(x_1,y_1), S(x_2,y_2))$**
H	(142, 94)	((103, 6),(134, 112))	1001202002
ò	(80, 64)	((103, 6),(130, 106))	1001220002
a	(3, 64)	((103, 6),(63, 102))	1001201122
	(19, 60)	((103, 6),(132, 75))	1001211010
B	(70, 76)	((103, 6),(34, 33))	1001201212
ì	(63, 49)	((103, 6),(3, 87))	1001222002
n	(40, 38)	((103, 6),(102, 142))	1001200002
h	(142, 94)	((103, 6),(134, 112))	1001202002

Encryption: The decryption is performed as follows:

- Extract five digits on the cipher text: 10012

- Circularly shift this sequence by one digit to the left: 00121

- Convert a sequence to decimal form, and store a value in k, it mean $00121(3) = 0*3^4 + 0*3^3 + 1*3^2 + 2 * 3^1 + 1* 3^0 = 16 \Rightarrow k = 16$

- Obtain (k + 1)P from pre-computed and stored point $(k + 1)P = (x_1, y_1)$.

Thus, $(x_1, y_1) = (103, 6)$ is 17P. Similarly, with sequence 02002, we compute $(x_2, y_2) = (134, 112)$, it is 169P. Therefore, we are able to recover the encrypted version: ((103, 6), (134, 112)). From this Pi should be retrieved using B private key: 28(103, 6) = (19, 60). And $P_i = (134, 112) - (19, 60) = (142, 94)$ is $P_i = 169P - 132P = 37P$. Compared to Table 4, we find the character "H". Particularly, when we repeat all the above steps, we find the remaining characters and find the original plaintext "Hòa Bình". Table 6 below shows the results of the plaintext after decoding.

Table 6. Encryption before
and after applying a data sequence

Sequence	**Reversal of sequence**	**Decryption**	**Plaintext**
1001202002	((103, 6), (134, 112))	(142, 94)	H
1001220002	((103, 6), (130, 106))	(80, 64)	ò
1001201122	((103, 6), (63, 102))	(3, 64)	a
1001211010	((103, 6), (132, 75))	(19, 60)	
1001201212	((103, 6), (34, 33))	(70, 76)	B
1001222002	((103, 6), (3, 87))	(63, 49)	ì
1001200002	((103, 6), (102, 142))	(40, 38)	n
1001202002	((103, 6), (134, 112))	(142, 94)	h

5. Conclusion

Based on the idea of an elliptic curve, we have built an elliptic curve cryptosystem which is applied to encode Vietnamese. The complexity of the algorithm depends on the parameter of the Elliptic curve and the random positive integer l, **α.** This cryptosystem has scientific and practical implications used frequently to encrypt big data.

References

[1] N. Koblitz, "Elliptic curve cryptosystems", Mathematics of Computation", 203 - 209, 1987.

[2] V. Miller, "Uses of elliptic curves in cryptography, Advances in Cryptology - Crypto", Lecture Notes in Computer Science, SpringerVerlag, 1986, pp. 417-426.

[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 68(30) (2020) 0474-9030.

[4] Negin Dinarvand, Hamid Barati, "An efficient and secure RFID authentication protocol using ellipticcurvecryptography",Springer Science+Business Media, LLC, 2017

[5] Utku Gulen, Selcuk Baktir, "Elliptic Curve Cryptography for Wireless Sensor Networks Using the Number Theoretic Transform", journal-sensors, Published: 9 March, 2020.

[6] D. Sravana Kumar, C.H. Suneetha, A.R. Chandrasekh, "Encryption of Data Using Elliptic

Curve Over Finite Fields", International Journal of Distributed and Parallel Systems (IJDPS). 3(1) (2012) 301-308.

[7] F. Amounas, E.H. El Kinani, ECC Encryption and Decryption with a Data Sequence, Applied Mathematical Sciences 6(101) (2012) 5039-5047.

[8] Vu Thi Hai Ha, Dinh Thi Hang, Bui Dang Binh, "The influence of volume on the formant of vowels and the identification of Vietnamese speakers", Vietnam Institute of Linguistics, 2015.

[9] A. Enge, "Elliptic curves and their applications to cryptography", Norwell, MA: Kulwer Academic publishers, 1999.

[10] 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 (1998) 327-337.

[11] S. Sandeep, Kumar, "Elliptic curve cryptography for constrained devices", PhD thesis, Ruhr-University Bochum, June, 2006.