Cryptographic Algorithms for Secure Data Communication

Zirra Peter Buba zirrapeter@yahoo.com
Department of Mathematical Sciences
Adamawa State University
Mubi, 650221, Nigeria.

Gregory Maksha Wajiga gwajiga@gmail.com
Department of Mathematics and Computer Science
Federal University of Technology
Yola, 640284, Nigeria.

Abstract

Personal privacy is of utmost importance in the global networked world. One of the best tools to help people safeguard their personal information is the use of cryptography. In this paper we present new cryptographic algorithms that employ the use of asymmetric keys. The proposed algorithms encipher message into nonlinear equations using public key and decipher by the intended party using private key. If a third party intercepted the message, it will be difficult to decipher it due to the multilevel ciphers of the proposed application.

Keywords: Cryptographic Algorithm, Asymmetric key, Communication, Nonlinear System

1. INTRODUCTION

Some vital information that are disseminated within an office, across offices, between branches, of an organization and other external bodies and establishments at times get into the hands of the unauthorized persons who may tamper with the contents of the information. And if no security measures are taken, there is no doubt that such data and other sensitive information will be exposed to threats such as impersonation, insecrecy, corruption, repudiation, break-in or denial of services [1,2] that may cause serious danger on the individual or organization.

A secure system should maintain the integrity, availability, and privacy of data [3]. Data integrity usually means protection from unauthorized modification, resistance to penetration and protection from undetected modification.

Therefore, algorithms which help prevent interception, modification, penetration, disclosure and enhance data/information security are now of primary importance. This paper suggests new methods for secured means of communication over unsecure channel. This is to ensuring that the intruders do not have access to the plaintext without a secret key.

2. PRELIMINARIES

2.1 Cryptography

One way to strengthen security [4,5] in computer systems is to encrypt sensitive records and messages in transit and in storage. The basic model of a cryptographic system is illustrated in Figure 1. The original unenciphered text is called the plaintext. The act of converting a plain text message to its ciphertext form is called enciphering [6]. In its cipher form, a message cannot be read by anyone but the intended receiver. Reversing that act (i.e., ciphertext form to plain text message) is deciphering. Enciphering and deciphering are more commonly referred to as encryption and decryption, respectively.

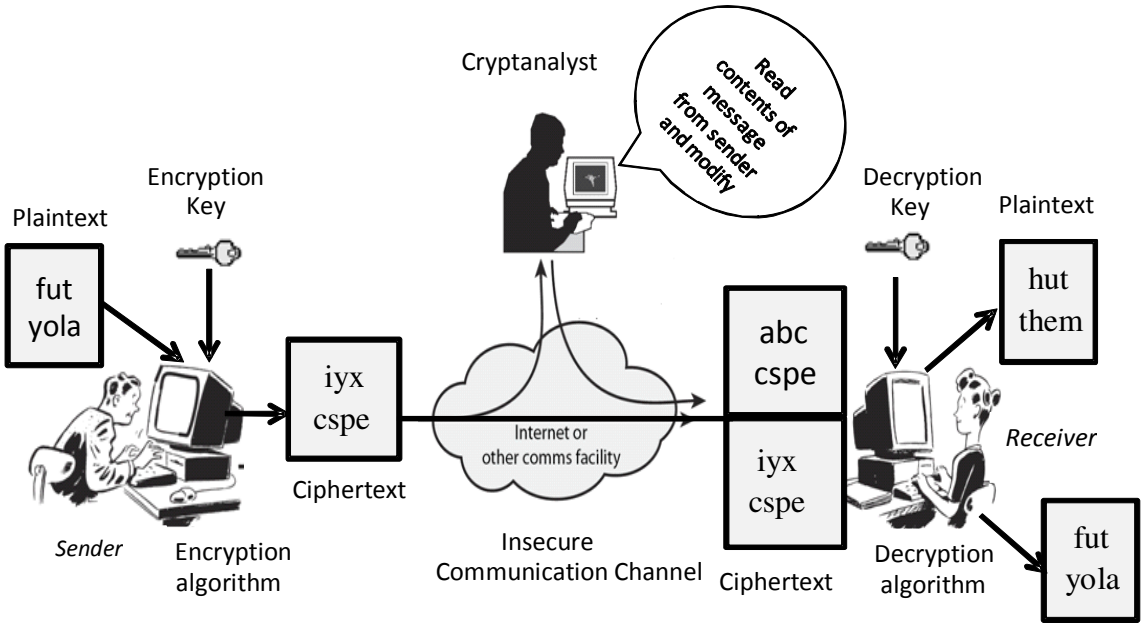


FIGURE 1: Encryption and Decryption Process

2.2 Modern Key-Based Cryptographic Techniques

There are several modern key-based cryptographic techniques. The two common key based encryption techniques are symmetric and asymmetric key cryptography [7].

In symmetric key cryptography, the same key is used for both encryption and decryption. In asymmetric schemes, one key is used for encryption and another is used for decryption [8]. The increased confidence in the integrity of systems that use encryption is based on the notion that ciphertext should be very difficult to decipher without knowledge of the key [3].

2.3 Types of Attacks

There are several types of code-breaking attacks. The first, known as the ciphertext attack, occurs when an adversary comes into possession of only the ciphertext [3]. The known plaintext problem occurs when the intruder has some matched portions of the ciphertext and the plaintext [9]. The most dangerous is the chosen plaintext problem, in which the attacker has the ability to encrypt pieces of plaintext at will. Brute-force is the ultimate attack on a cipher, by which all possible keys are successively tested until the correct one is encountered [9]. Codebook attacks are attacks that take advantage of the property by which a given block of plaintext is always encrypted to the same block of ciphertext as long as the same key is used. A "man-in-the-middle" attack is an attack that is placed by an active attacker who can listen to the communication between two entities and can also change the contents of this communication. While performing this attack, the attacker pretends to be one of the parties in front of the other party.

2.4. Types of Encryption Algorithms

Currently, there are several kinds of key based encryption software in the market categorized by their function and target groups. The most common key based encryption techniques are given [7,10,11] as follows:

i. The Caesar Cipher- one of the earliest cryptographic algorithms linked and attributed to Julius Caesar in the Gallic war for its usage. Julius Caesar used cipher to protect the messages to his troops by replacing each letter in a message by the third letter further along in the

alphabet. '***abc***' becomes '***def***' Obviously, this is extremely weak cryptographic algorithm in today's.

ii. Data Encryption Standard (DES)- was the first encryption standard to be recommended by National Institute of Standards and Technology (NIST). DES is (64 bits key size with 64 bits block size). Since that time, many attacks and methods recorded the weaknesses of DES, which made it an insecure block cipher.

iii. International Data Encryption Algorithm (IDEA) is a cryptosystem developed by X. Lai and J. Massey in 1991 to replace the DES standard. It is a symmetric block cipher, operating on 8 bytes at a time, just like DES, but with a key of 128 bits.

iv. Rivest Cipher 4 (RC4) - a cipher invented by Ron Rivest, a proprietary system by RSADSI, is used in a number of commercial systems like Lotus Notes and secure Netscape.

v. *Blowfish* is block cipher 64 bits.It takes a variable-length key, ranging from 32 to 448 bits; default 128 bits. Blowfish is unpatented, license-free, and is available free for all uses. Blowfish has variants of 14 rounds or less.

vi. Unix Crypt - Many Unix systems come supplied with an encryption system called crypt. This routine should never be used for encrypting anything because there exist programs on the net for producing the decrypted text and the key.

vii. Ron Rivest, Adi Shamir, and Leonard Adleman Algorithm (RSA) - a cipher algorithm based on the concept of a trapdoor function, which is easily calculated, but whose inverse is extremely difficult to calculate. The RSA algorithm is named after Ron Rivest, Adi Shamir and Len Adleman, who invented it in 1977. The RSA algorithm can be used for both public key encryption and digital signatures. Its security is based on the difficulty of factoring large integers.

viii. Pretty Good Privacy (PGP) - a public key system for encrypting electronic mail using the RSA public key cipher. It encrypts the message using the IDEA cipher with a randomly generated key. It then encrypts the key using the recipient's public key. When the recipient receives the message, PGP uses his private RSA key to decrypt the IDEA key and then uses that IDEA key to decrypt the message.

ix. Diffie-Hellman (DH)- is the first published public key cryptographic algorithm which allows two users to exchange a secret key over an insecure medium without any prior secrets. The original protocol had two system parameters, p and g. They are both public and may be used by all the users in a system. The Diffie-Hellman key exchange was vulnerable to a man-in-the-middle attack, as Diffie-Hellman key exchange does not authenticate the participants. Parameter *p* is a prime number and parameter *g* is an integer less than *p*, with the following property: for every number *n* between 1 and *p*-1 inclusive, there is a power *k* of *g* such that n = g^k mod p, where k is kept secret.

3. PROPOSED CRYPTOGRAPHIC ALGORITHM

The proposed encryption algorithm consists of a three level cipher attempt to keep your personal data secure. The first level is achieved through the words compression flowchart in Figure2, the second level is realized by transforming the compressed words from Figure 2 into systems of nonlinear equations and the third level is achieved by the applying the delta (δ_n) encoding principles.

3.1 Encryption and Decryption Keys

Asymmetric encryption key is used which means two keys are shared: a public key to encrypt the message and a private key to decrypt it.

3.2 Encryption and Decryption Rules

a) The encryption process requires that the sender must have the message and number of words that are left from Figure 2.

b) To decipher the message in Equation (1b), the intended receiver must be given the following key which contains the associated terms and corresponding code.

i. Solutions obtained from the system of nonlinear equations.

ii. The δ_p values associated with the variable index in *Table 2.*

iii. The formula $s_k = \sum_{p=1}^{k} \delta_p$.

iv. A lookup character position in *Table 1.*

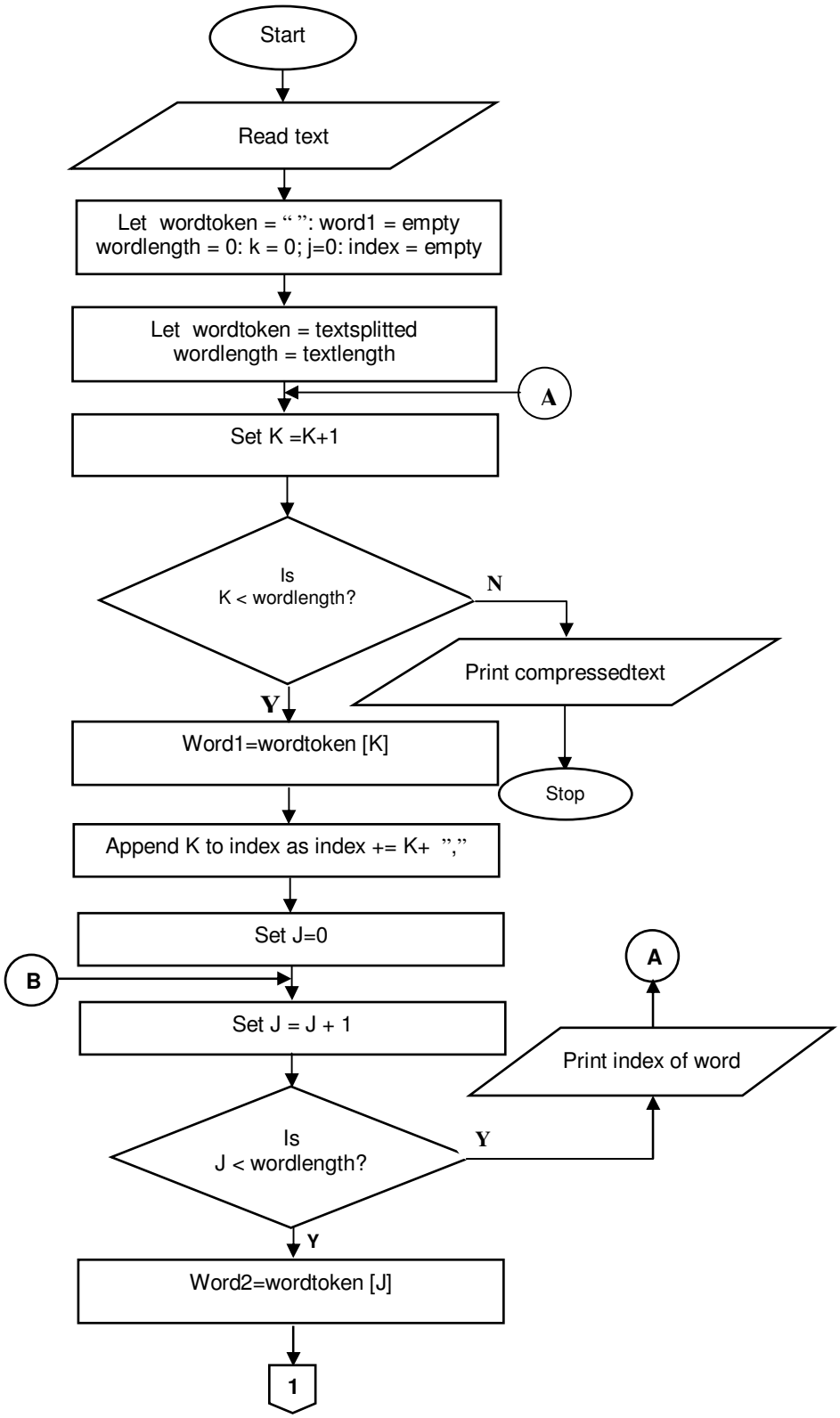
	2	3	4	5	6	7
0	space	0	@	P	`	P
1	!	1	A	Q	a	q
2	‘’	2	B	R	b	r
3	#	3	C	S	c	s
4	$	4	D	T	d	t
5	%	5	E	U	e	u
6	&	6	F	V	f	v
7	‘	7	G	W	g	w
8	(	8	H	X	h	x
9	)	9	I	Y	i	y
a	*	:	J	Z	j	z
b	+	;	K	[	k	{
c	,	<	L	\	l	}
d	-	=	M	]	m	|
e	.	>	N	^	n	□
f	/	?	O	_	o	₦

TABLE 1: A Lookup Character Position

c) The key is typically shared by trusted entities and be kept secret from the unauthorized users.

d) The variable indexes that represent the compressed characters in Equation (1b) are further hid in a file using delta encoding principle before transmission to the intended receiver to further create a state of confusion to the intruders.

e) A copy of the generated decryption key is saved in a file and sends to recipient email or via any secure communication medium such as telephones Short Message Service (SMS) on or before the message reaches the intended recipient.


Start
Read text
Let wordtoken = " ": word1 = empty
wordlength = 0: k = 0; j=0: index = empty
Let wordtoken = textsplitted
wordlength = textlength
A
Set K =K+1
Is
K < wordlength?
N
Print compressedtext
Y
Word1=wordtoken [K]
Stop
Append K to index as index += K+ ","
Set J=0
B
A
Set J = J + 1
Print index of word
Is
J < wordlength?
Y
Y
Word2=wordtoken [J]
1

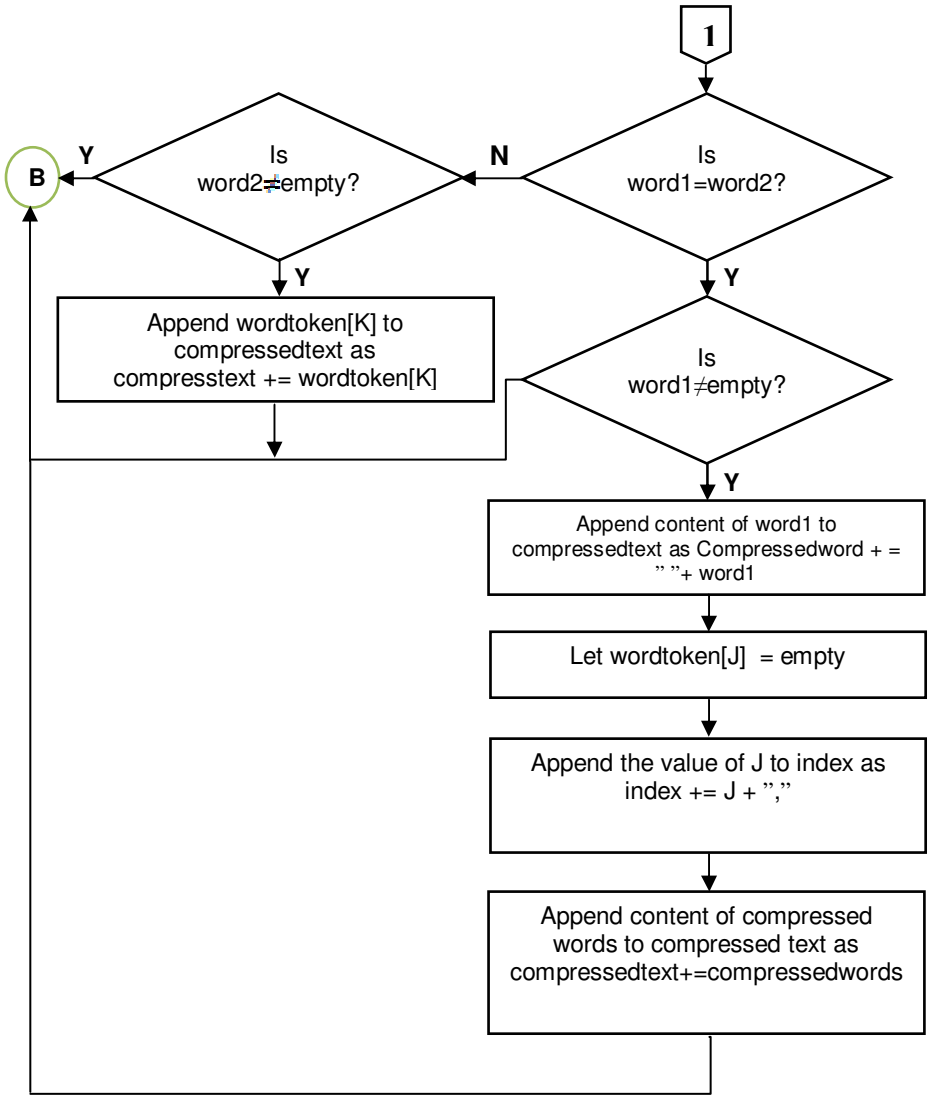
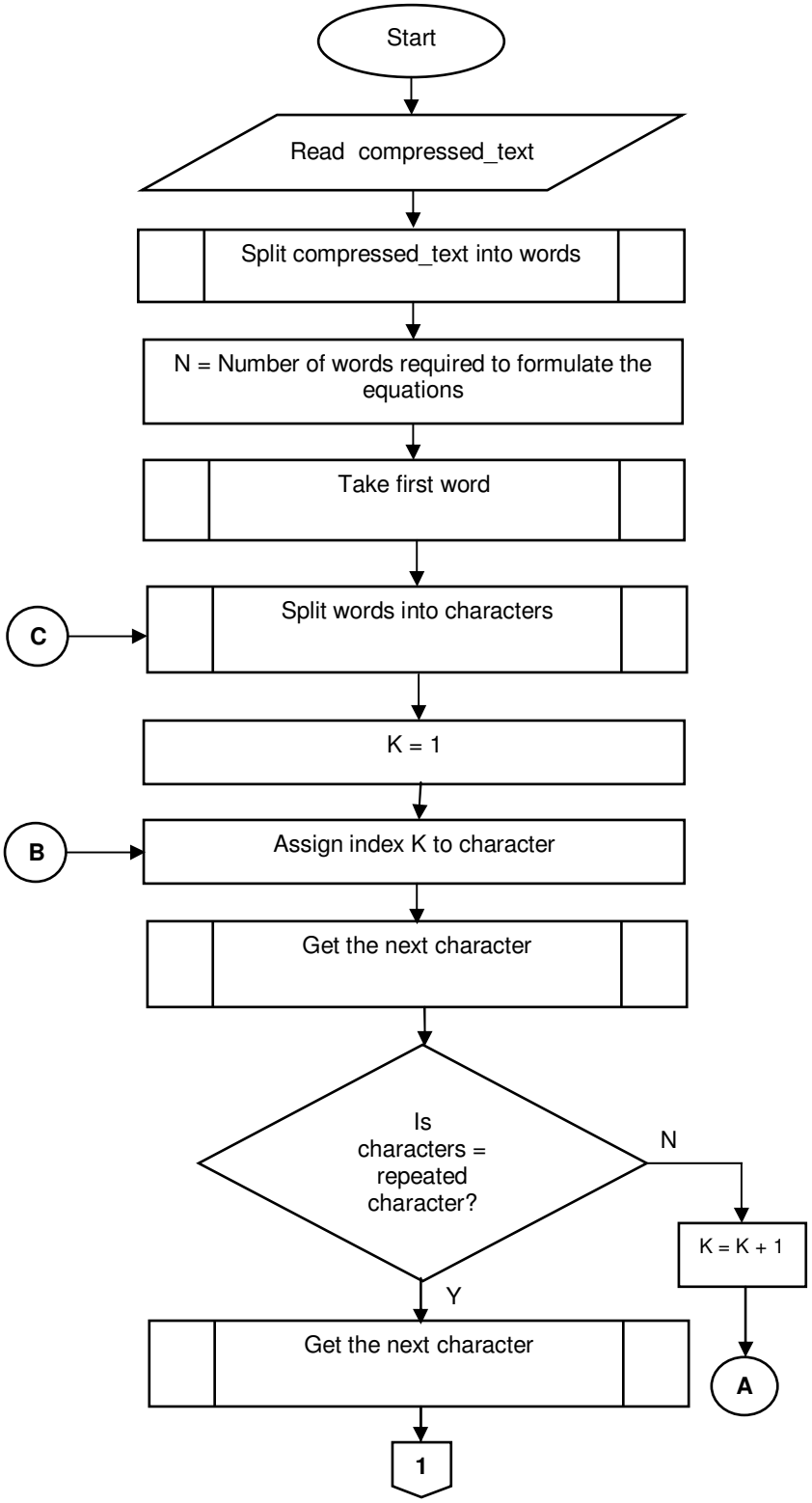


FIGURE 2: Word Compression Flowchart


Start
Read compressed_text
Split compressed_text into words
N = Number of words required to formulate the equations
Take first word
C
Split words into characters
K = 1
B
Assign index K to character
Get the next character
Is characters = repeated character?
N
K = K + 1
Y
Get the next character
A
1

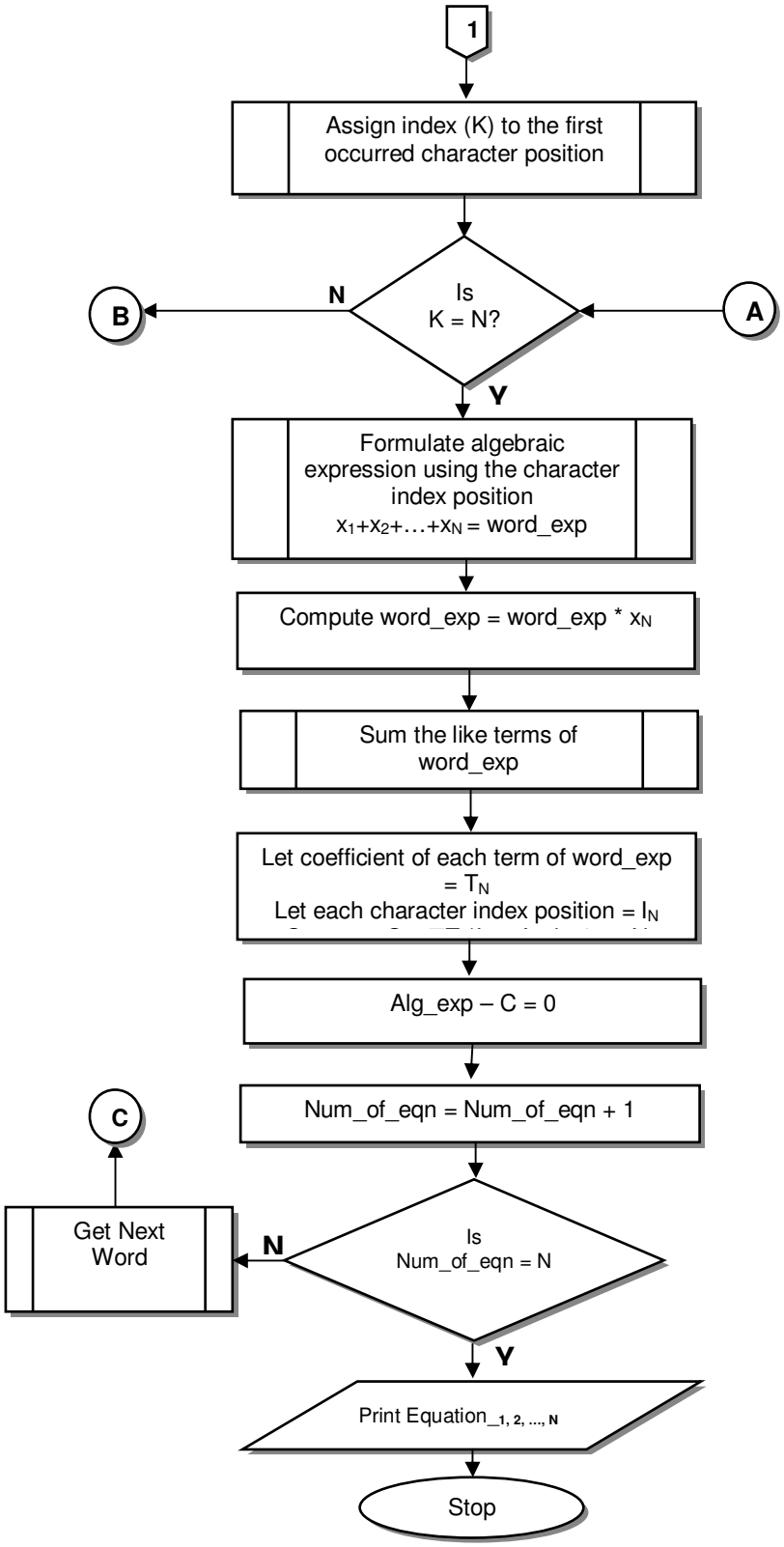


FIGURE 3: Flowchart to Formulate System of Nonlinear Equations

Illustrative example 1: To encrypt the message "*who is promising who*" we apply the procedure in Figure 2 and 3. The result yields the ciphertext in Equation (1b).

$$\begin{aligned} \text{who:} \quad & x_1^2 + x_1x_2 + x_1x_3 = 6 \\ \text{is:} \quad & x_1^2 + x_1x_2 = 3 \\ \text{promising:} \quad & 4x_1^2 + 2x_1x_2 + 3x_1x_3 = 17 \end{aligned} \tag{1a}$$

This above representation can be written as:

$$f(x_1,x_2,x_3) = \left.\begin{aligned} 2x_1^2 + x_1x_2 + x_1x_3 &= 6 \\ x_1^2 + x_1x_2 &= 3 \\ 4x_1^2 + 2x_1x_2 + 3x_1x_3 &= 17 \end{aligned}\right\} \tag{1b}$$

Equation (1b) becomes systems of nonlinear equations to be transmitted to the recipient in place of the plaintext "who is promising who"

3.3 Decryption Process

The decryption is performed simply by solving the systems of nonlinear equations in Equation (1b) using the below Algorithm 1.

The Newton's_ *Algorithm 1*

1: guess an approximation solution x_0

2: calculate $J(x_p)$ and $f(x_p)$, where $J(x_p)_{i,j} = \partial f_i(x)/\partial x_j$, for $1 \leq i, j \leq p$

3: solve the linear system $J(x_p)\delta_p = -f(x_p)$

4: set $x_{p+1} = x_p + \delta_p$

Algorithm 1: Newton's Algorithm

For the purpose of this paper,

a. Explicit computation of the inversion of Jacobi (i.e. $J(x_p)^{-1}$) is avoided as this will involve additional iteration for determining $J(x_p)^{-1}$

b. Instead we employ the linear system $J(x_p)\delta_p = -f(x_p)$ at the next iterate, thus $x_{p+1} = x_p + \delta_p$.

Illustrative example 2: To decipher the message in Equation (1b):

i. The receiver must solve and obtain the solutions of the system of nonlinear equations in equation (1) as follows:

ii.

$$J(x_n)\delta_x = -F(x_n) \Rightarrow$$

$$\begin{bmatrix} 2x_1 + x_2 + x_3, & x_1 & x_1 \\ 2x_1 + x_2, & x_1 & \\ 8x_1 + 2x_2 + 3x_3, & 2x_1 & 3x_1 \end{bmatrix} \delta x = - \begin{bmatrix} x_1^2 + x_1x_2 + x_1x_3 - 6 \\ x_1^2 + x_1x_2 - 3 \\ 4x_1^2 + 2x_1x_2 + 3x_1x_3 - 17 \end{bmatrix} \tag{2}$$

We take an initial guess of $x_0 = (1,1,1)^T$ and substitute in equation (2), to obtain $J(x_0)$ and $f(x_0)$ as in equation (3) in matrix notation.

Hence, from equation (2) we obtain

$$J(x_0)(x_1 - x_0) = -F(x_0) \Rightarrow \begin{bmatrix} 4 & 1 & 1 \\ 3 & 1 & 0 \\ 13 & 2 & 3 \end{bmatrix} \begin{bmatrix} x_1 - 1 \\ x_2 - 1 \\ x_3 - 1 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \\ 8 \end{bmatrix} \tag{3}$$

Solving equations (3) by Gussaian elimination or any other methods convenient yield the approximate solutions $x_1 = 1, x_2 = 2$ and $x_3 = 3$.

Equation (2) yields equation (4) on input of $x_1 = 1, x_2 = 2$ and $x_3 = 3$. This is a test of convergence of the approximate solutions.

$$J(x_0)(x_2 - x_1) = -F(x_1) \Rightarrow \begin{bmatrix} 7 & 1 & 1 \\ 4 & 1 & 0 \\ 20 & 2 & 3 \end{bmatrix} \begin{bmatrix} x_1 - 1 \\ x_2 - 2 \\ x_3 - 3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (4)$$

Therefore, the approximate solutions are $x_1 = 1$, $x_2 = 2$ and $x_3 = 3$.

iii. Get the lookup character position in Table 1 above.

iv. Use the values of the variables (solutions) in conjunction with the delta encoded in Table 2, and the formula $s_k = v_3 + \sum_{n=1}^{k} \delta_x$ to finally to obtain the original plaintext in Table 5.

Virtual position of characters (x_n)	*76*	*66*	*6c*		*68*	*71*		*6d*	*71*	*6c*	*6b*	*68*	*71*	*68*	*6b*	*6b*
Solution of Equations (v_n)	*1*	*2*	*3*		*1*	*2*		*3*	*1*	*3*	*2*	*1*	*2*	*1*	*3*	*1*
δ_n	*76*	*-10*	*6*		*68*	*9*		*6d*	*04*	*-05*	*-01*	*-03*	*09*	*-09*	*03*	*-05*

TABLE 2: Delta Encoding for "who is promising"

Illustrative example 3: To encrypt the message "Kill all Hippopotamus in the river Mississippi" we apply the procedure in Figure 2 and 3. The result yields the ciphertext in Equation (5b).

Kill $x_1^2 + x_1x_2 + 2x_1x_3 = 9$

all $2x_1x_3 + x_1x_4 = 10$

Hippoppotamus $x_1^2 + 2x_1x_2 + x_1x_3 + 2x_1x_4 + x_1x_5 + 3x_1x_6 + 2x_1x_7 = 53$

in $x_1x_2 + x_1x_5 = 7$

(5a)

the $x_1^2 + x_1x_6 + x_1x_7 = 14$

river $2x_1^2 + 2x_1x_2 + x_1x_7 = 13$

Mississippi $4x_1x_2 + x_1x_3 + 4x_1x_4 + 2x_1x_6 = 39$

Equation (5a) can be written as shown in equation (5b)

$x_1^2 + x_1x_2 + 2x_1x_3 - 9 = 0$

$2x_1x_3 + x_1x_4 - 10 = 0$

$x_1^2 + 2x_1x_2 + x_1x_3 + 2x_1x_4 + x_1x_5 + 3x_1x_6 + 2x_1x_7 - 53 = 0$

$x_1x_2 + x_1x_5 - 7 = 0$

(5b)

$x_1^2 + x_1x_6 + x_1x_7 - 14 = 0$

$2x_1^2 + 2x_1x_2 + x_1x_7 - 13 = 0$

$4x_1x_2 + x_1x_3 + 4x_1x_4 + 2x_1x_6 - 39 = 0$

Equation (5a) becomes the ciphertext to be transmitted to the recipient in place of the plaintext "Kill all Hippopotamus in the river Mississippi". The variable solutions of Equation (5b) are further concealed in a delta encoding file and then send to the intended receiver as shown in Table 3 to further create confusion to the intruder.

x_n	4a 69	67 73	69 62	69 5e	5d 71	69 67	69 74	43 5e	67 71	6a 4a	6a 67	68 6f	6a 6f	68 67	73 6f	5d 6f	6b 67	72 6a	6f 6a	67 67
v_s	1 5	2 1	3 6	3 7	4 1	3 2	3 2	5 7	2 1	6 3	6 2	7 4	6 4	7 2	1 4	4 4	2 2	3 6	4 6	2 2
δ_x	4a 02	1d 73	02 - 11	00 - 04	5d 71	0c - 0a	00 0d	43 - 16	24 13	03 4a	00 1d	- 02 08	02 00	- 02 - 08	0b 08	- 16 00	0e - 08	07 03	- 03 00	67 - 03

TABLE 3: Delta Encoding File For "Kill all Hippopotamus in the river Mississippi"

As usual, to decipher the message, the receiver, apply procedure in section 3.3 on the ciphertext in equation (5b) to obtain equation (6).

$$J(x_n)\delta_x = -F(x_n) \Rightarrow$$

$$\begin{bmatrix} 2x_1 + x_2 + 2x_3, & x_1, & 2x_1, & & & & \\ 2x_2 + x_4 & & 2x_1, & x_1, & & & \\ 2x_1 + 2x_2 + x_3 + 2x_4 + x_5 + 3x_6 + 2x_7, & 2x_1, & x_1, & 2x_1, & x_1, & 3x_1, & 2x_1 \\ & x_1, & & & x_1, & & \\ 2x_1 + x_6 + x_7, & & & & & x_1, & x_1 \\ 4x_1 + 2x_2 + x_7, & 2x_1, & & & & & x_1 \\ 4x_2 + x_3 + 4x_4 + 2x_6, & 4x_2, & x_1, & 4x_1, & & 2x_1, & \end{bmatrix} (x_n - x_{n-1}) =$$

$$- \begin{bmatrix} x_1^2 + x_1x_2 + 2x_1x_3 - 9 \\ 2x_1x_3 + x_1x_4 - 10 \\ x_1^2 + 2x_1x_2 + x_1x_3 + 2x_1x_4 + x_1x_5 + 3x_1x_6 + 2x_1x_7 - 53 \\ x_1x_2 + x_1x_5 - 7 \\ x_1^2 + x_1x_6 + x_1x_7 - 14 \\ 2x_1^2 + 2x_1x_2 + x_1x_7 - 13 \\ 4x_1x_2 + x_1x_3 + 4x_1x_4 + 2x_1x_6 - 39 \end{bmatrix}$$

(6)

Taking an initial guess of $x_0 = (1, 1, 1, 1, 1, 1, 1)^t$ and substituting in equation (6) produces:

$$\begin{bmatrix} 5 & 1 & 2 & 0 & 0 & 0 & 0 \\ 3 & 0 & 2 & 1 & 0 & 0 & 0 \\ 13 & 2 & 1 & 2 & 1 & 3 & 2 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 & 1 & 1 \\ 7 & 2 & 0 & 0 & 0 & 0 & 1 \\ 11 & 4 & 1 & 4 & 0 & 2 & 0 \end{bmatrix} \begin{bmatrix} x_1 - 1 \\ x_2 - 1 \\ x_3 - 1 \\ x_4 - 1 \\ x_5 - 1 \\ x_6 - 1 \\ x_7 - 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 7 \\ 41 \\ 5 \\ 11 \\ 8 \\ 28 \end{bmatrix}$$

(7)

The solution below is obtained as a result of the application of the procedure in section 3.3 on equation (7),

$$x_1 = 1,\ x_2 = 2,\ x_3 = 3,\ x_4 = 4,\ x_5 = 5,\ x_6 = 6,\ x_7 = 7 \quad .$$

Finally, to decipher the message in equation (5b), the receiver would further need other secret keys as stated in section 3.2. Using these keys, the intended receiver can now recover the encrypted text as shown in the result Table 6.

Illustrative Example 4

If we input the text "Credit A/c No: 6711645138110 with my VISA debit card No: 123456789101112", into the Figure 2 we obtain n=10, (the number of expected equations to be

formulated whose variables must not exceed the number of the words) and one indexed word that previously occurred.

Using Figure 3 gives the systems of nonlinear equations of the text "Credit A/c No: 6711645138110 with my VISA debit card 123456789101112" as shown in equation (8)

$$
\begin{array}{ll}
\text{Credit} & x_1^2 + x_1x_2 + x_1x_3 + x_1x_4 + x_1x_5 + x_1x_6 = 21 \\
\text{A/c} & x_1x_7 + x_1x_8 + x_1x_9 = 24 \\
\text{No:} & x_1^2 + x_1x_2 + x_1x_{10} = 13 \\
6711645138110 & 2x_1x_3 + x_1x_4 + 5x_1x_5 + x_1x_6 + x_1x_7 + x_1x_8 + x_1x_9 + x_1x_{10} = 75 \\
\text{with} & x_1^2 + x_1x_2 + x_1x_3 + x_1x_6 = 14 \\
\text{my} & x_1x_3 + x_1x_4 = 7 \\
\text{VISA} & x_1x_5 + x_1x_6 + 2x_1x_7 = 25 \\
\text{debit} & x_1x_3 + x_1x_4 + x_1x_5 + x_1x_6 + x_1x_8 = 26 \\
\text{card} & x_1x_2 + x_1x_4 + 2x_1x_9 = 24 \\
123456789101112 & x_1^2 + x_1x_3 + x_1x_4 + 5x_1x_5 + x_1x_6 + x_1x_7 + x_1x_8 + x_1x_9 + 3x_1x_{10} = 93
\end{array}
\qquad (8)
$$

Equation (8) becomes the enciphertext to be transmitted to the recipient without the words appearing against the Equation (8) in place of the plaintext "Credit A/c No: 6711645138110 with my VISA debit card No: 123456789101112". As usual, the variable solutions of Equation (8) are further conceal in a file called delta encoding file before transmitting it to the intended receiver as shown in Table 4 to enforce protection of the message.

x_n	42	70	62	60	64	6e	3a	27	5a	44	6e	38	33	33	2c	2c	33	2e	2e	2c	
	2b	2f	2c	2c	26	76	64	6e	66	6a	75	51	43	4c	3a	60	62	5a	64	6e	
	5a	58	70	60	2c	28	2b	2e	2e	33	33	2f	38	2c	26	2c	2c	2c	28		
v_s	1	2	3	4	5	6	7	8	9	a	1	2	3	4	5	5	3	6	7	5	
	8	9	5	5	a	1	5	6	2	3	4	5	6	7	7	4	3	8	5	6	
	9	9	2	4	5	a	8	6	7	3	4	9	1	5	a	5	5	5	a		
δ_x	42	2e	-0e	-02	04	0a	3a	-13	33	44	2a	-36	33	00	-07	00	07	-05	00	-02	
	-01	04	-03	00	-06	76	-12	0a	-08	6a	0b	51	-0e	09	-12	60	02	-08	0a	0a	
	5a	-02	18	-10	2c	-04	03	03	00	05	00	-04	09	-0c	-06	06	00	00	-04		

TABLE 4: Delta Encoding: Credit A/c No: 6711645138110 with my VISA debitcard 123456789101112

As usual, we compute $J(x_0)$ and $F(x_0)$ of equation (8). Then take initial guess of $x_0 = (1,1,1,1,1,1,1,1,1,1)^t$ and substitute in the equation, this will give the equation (9) in matrix form.

$$
\begin{bmatrix}
7 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
3 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\
4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
13 & 0 & 2 & 1 & 5 & 1 & 1 & 1 & 1 & 1 \\
5 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
2 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
4 & 0 & 0 & 0 & 1 & 1 & 2 & 0 & 0 & 0 \\
5 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\
4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\
16 & 0 & 1 & 1 & 5 & 1 & 1 & 1 & 1 & 3
\end{bmatrix}
\begin{bmatrix}
x_1 - 1 \\ x_2 - 1 \\ x_3 - 1 \\ x_4 - 1 \\ x_5 - 1 \\ x_6 - 1 \\ x_7 - 1 \\ x_8 - 1 \\ x_9 - 1 \\ x_{10} - 1
\end{bmatrix}
=
\begin{bmatrix}
15 \\ 21 \\ 10 \\ 62 \\ 10 \\ 5 \\ 21 \\ 21 \\ 20 \\ 78
\end{bmatrix}
\qquad (9)
$$

Similarly, equations (9), can be solve by applying the procedure in section 3.3 to obtain the following approximate solutions:

$$x_1 = 1,\ x_2 = 2,\ x_3 = 3,\ x_4 = 4,\ x_5 = 5,\ x_6 = 6,\ x_7 = 7, x_8 = 8,\ x_9 = 9,\ x_{10} = 10$$

To decrypt the message in Equation (8), the receiver would further need other secret keys as stated in section 3.2. Using these keys, the intended receiver can now recover the encrypted text as shown in the result Table 7.

4. RESULTS

Table 5 shows the result of the deciphered text from the enciphered text (Equation (1b)) obtained from the proposed algorithms.

Position of Word s_k	$s_k = v_s + \sum_{n=1}^{K} \delta_x$, where v_s = variable solution	*From Table I (R.C)*
1	$x_1 = 1 + 76 = 77$	*77 = w*
	$x_2 = 2 + 76 - 10 = 68$	*68 = h*
	$x_3 = 3 + 76 - 10 + 6 = 6f$	*6f = o*
2	$x_1 = 1 + 68 = 69$	*69 = i*
	$x_2 = 2 + 68 + 9 = 73$	*73 = s*
3	$x_3 = 3 + 6d = 70$	*70 = p*
	$x_1 = 1 + 6d + 4 = 72$	*72 = r*
	$x_3 = 3 + 6d + 4 - 5 = 6f$	*6f = o*
	$x_2 = 2 + 6d + 4 - 5 - 1 = 6d$	*6d = m*
	$x_1 = 1 + 6d + 4 - 5 - 1 - 3 = 69$	*69 = i*
	$x_7 = 2 + 6d + 4 - 5 - 1 - 3 + 9 = 73$	*73 = s*
	$x_4 = 1 + 6d + 4 - 5 - 1 - 3 + 9 - 9 = 69$	*69 = i*
	$x_8 = 3 + 6d + 4 - 5 - 1 - 3 + 9 - 9 + 3 = 6e$	*6e = n*
	$x_1 = 1 + 6d + 4 - 5 - 1 - 3 + 9 - 9 + 3 - 5 = 67$	*67 = g*
4	$x_1 = 1 + 76 = 77$	*77 = w*
	$x_2 = 2 + 76 - 10 = 68$	*68 = h*
	$x_3 = 3 + 76 - 10 + 6 = 6f$	*6f = o*

TABLE 5: Plaintext Recovery

Table 6 shows the result of the deciphered text from the enciphered text (Equation (5b))

Position of Word s_k	$s_k = v_s + \sum_{n=1}^{K} \delta_n$, where v_s = variable solution	*From Table 1 (R.C)*
1	$x_1 = 1 + 4a$	*4b= K*
	$x_2 = 2 + 4a + 1d$	*69 = i*
	$x_3 = 3 + 4a + 1d + 02$	*6c = l*
	$x_3 = 3 + 4a + 1d + 02 + 00$	*6c =l*
2	$x_4 = 4 + 5d$	*61= a*
	$x_2 = 3 + 5d + 0c$	*6c = l*
	$x_3 = 3 + 5d + 0c + 00$	*6c = l*
	⋮	⋮
7	$x_7 = 3 + 4a$	*4d = M*
	$x_2 = 2 + 4a + 1d$	*69 = i*
	$x_4 = 4 + 4a + 1d + 08$	*73 = s*
	$x_4 = 4 + 4a + 1d + 08 + 00$	*73 = s*
	$x_2 = 2 + 4a + 1d + 08 + 00 - 08$	*69 = i*
	$x_4 = 4 + 4a + 1d + 08 + 00 - 08 + 08$	*73 = s*
	$x_4 = 4 + 4a + 1d + 08 + 00 - 08 + 08 + 00$	*73 = s*
	$x_7 = 2 + 4a + 1d + 08 + 00 - 08 + 08 + 00 - 08$	*69 = i*
	$x_6 = 6 + 4a + 1d + 08 + 00 - 08 + 08 + 00 - 08 + 03$	*70 = p*
	$x_6 = 6 + 4a + 1d + 08 + 00 - 08 + 08 + 00 - 08 + 03$+00	*70 = p*
	$x_7 = 2 + 4a + 1d + 08 + 00 - 08 + 08 + 00 - 08 + 03$+00-03	*69 = i*

TABLE 6: Plaintext Recovery for "Kill all Hippopotamus in the river Mississippi"

Table 7 shows the result of the deciphered text from the enciphered text (Equation (8))

Position of Word s_k	$s_k = v_s + \sum_{n=1}^{K} \delta_n$, where v_s = variable solution	*From Table 1 (R.C)*
1	$x_1 = 1 + 42$	*43 = C*
	$x_2 = 2 + 42 + 2e$	*72 = r*
	$x_3 = 3 + 42 + 2e - 0e$	*65 = e*
	$x_4 = 4 + 42 + 2e - 0e - 02$	*64 = d*
	$x_5 = 5 + 42 + 2e - 0e - 02 + 04$	*69 = i*
	$x_6 = 6 + 42 + 2e - 0e - 02 + 04 + 0a$	*74 = t*
2		
	$x_7 = 7 + 3a$	*41 = A*
	$x_8 = 8 + 3a - 11$	*2f = /*
	$x_9 = 9 + 3a - 11 + 33$	*63 = c*
3	$x_{10} = a + 44$	*4e = N*
	$x_1 = 1 + 44 + 2a$	*6f = o*
	$x_2 = 2 + 44 + 2a + 33$	*3a = :*
	. . .	. . .
10	$x_5 = 5 + 2c$	*31 = 1*
	$x_{10} = a + 2c - 04$	*32 = 2*
	$x_8 = 8 + 2c - 04 + 03$	*33 = 3*
	$x_6 = 6 + 2c - 04 + 03 + 03$	*34 = 4*
	$x_7 = 7 + 2c - 04 + 03 + 03 + 00$	*35 = 5*

	$x_3 = 3 + 2c - 04 + 03 + 03 + 00 + 05$	*36 = 6*
	$x_4 = 4 + 2c - 04 + 03 + 03 + 00 + 05 + 00$	*37 = 7*
	$x_9 = 9 + 2c - 04 + 03 + 03 + 00 + 05 + 00 - 04$	*38 = 8*
	$x_1 = 1 + 2c - 04 + 03 + 03 + 00 + 05 + 00 - 04 + 09$	*39 = 9*
	$x_5 = 5 + 2c - 04 + 03 + 03 + 00 + 05 + 00 - 04 + 09 - 0c$	*31 = 1*
	$x_{10} = a + 2c - 04 + 03 + 03 + 00 + 05 + 00 - 04 + 09 - 0c - 06$	*30 = 0*
	$x_5 = 5 + 2c - 04 + 03 + 03 + 00 + 05 + 00 - 04 + 09 - 0c - 06 + 06$	*31 = 1*
	$x_5 = 5 + 2c - 04 + 03 + 03 + 00 + 05 + 00 - 04 + 09 - 0c - 06 + 06 + 00$	*31 = 1*
	$x_5 = 5 + 2c - 04 + 03 + 03 + 00 + 05 + 00 - 04 + 09 - 0c - 06 + 06 + 00 + 00$	*31 = 1*
	$x_{10} = a + 2c - 04 + 03 + 03 + 00v05 + 00 - 04 + 09 - 0c - 06 + 06 + 00 + 00 - 04$	*32 = 2*

TABLE 7: Plaintext Recovery: Credit A/c No: 6711645138110 with my VISA debit card No: 123456789101112

In decompression, the characters are written as they appeared coupled with spaces between them and where there is an index value in place of new character, the index is interpreted and written in the place they appeared. Each time a word is written, a space is allowed between them.

4. DISCUSSION

Examination of Tables 5, 6 and 7, showed that simple attack identified by [9] to find the decryption key by the cryptanalyst requires solving the systems of nonlinear equations in equations (1b), (5b) and (8), obtaining the δ_x values associated with the variable index in Tables 2, 3 and 4, the formula $S_k = V_s + \sum_{n=1}^{k} \delta_x$, and the lookup character position in table 1. This is notoriously difficult to obtain due to their high mathematical formulation. A good encryption algorithm should be designed so that, when used with sufficiently long keys, it becomes computationally infeasible to break as reported [12,13]. This is in accordance with another related literature that revealed that the strength of an encryption algorithm relied on the mathematical soundness of the algorithm [3]. It is also in agreement with an earlier study by [8] who revealed that resources required for revealing a secret message should be strong and complex enough through a hiding key. This study is designed on similar encryption techniques that use sufficiently long keys.

The study also indicated that one key is used to encipher plaintext into ciphertext and another different key to decipher that ciphertext into plaintext as depicted in Tables 5, 6, and 7 [14,15,16]. The proposed scheme avoids the problem of sharing keys associated with the symmetric cryptography [16,17] that there is less risk associated with a public key than the symmetric key and the security based on that key is not compromised [18]. This study is designed on similar encryption techniques that use asymmetric key.

The study showed that, the decipher keys were transmitted to the intended receiver secretly through a different medium such as email, short message service or fax machine to the receiver before the receiver can have access to the plaintext. On the basis of this results, it is evident that unauthorized user will find it difficult to decrypt the message without the knowledge of the secret keys [3], since they were not transmitted together with the ciphered message. The strength of an encryption scheme is relies on the secrecy of the key [12]. This placed another level of security on the data in store or transit.

From the results of the study it is clear that there is confidentiality, non- repudiation and integrity of our sensitive and classified information over the Internet from the hands of Internet terrorist as highlighted by [2] and [16]. This is due to the robustness design of the proposed algorithms.

5. CONCLUSION

This paper has practically demonstrated how people can secure their vital and sensitive information stored or transmitted via insecure communication channels from cryptanalysts by using strong encryption and decryption keys. The proposed algorithm has proven to withstand any type of the attack.

7. REFERENCES

[1] B. Figg. (2004). Cryptography and Network Security. Internet: http:/www.homepages.dsu.edu/figgw/Cryptography%20&%20Network%2 0Security.ppt. [March 16, 2010].

[2] A. Kahate, *Cryptography and Network Security (2nd ed.*). New Delhi: Tata McGraw Hill, 2008.

[3] M. Milenkovic. *Operating System: Concepts and Design*, New York: McGrew-Hill, Inc., 1992.

[4] P.R. Zimmermann. An Introduction to Cryptography. Germany: MIT press. Available: http://www.pgpi.org/doc/pgpintro, 1995, [March 16, 2009].

[5] W. Stallings. *Cryptography and Network Security (4th ed.*). Englewood (NJ):Prentice Hall, 1995.

[6] V. Potdar and E. Chang. "Disguising Text Cryptography Using Image Cryptography," International Network Conference, United Kingdom: Plymouth, 2004.

[7] S.A.M. Diaa, M.A.K. Hatem, and M.H. Mohiy (2010). "Evaluating The Performance of Symmetric Encryption Algorithms" *International Journal of Network Security,* 2010, 10(3), pp.213-219

[8] T. Ritter. "Crypto Glossary a*nd* Dictionary of Technical Cryptography'. Internet: www.ciphersbyritter.com/GLOSSARY.HTM , 2007, [August 17, 2009]

[9] K.M. Alallayah, W.F.M. Abd El-Wahed, and A.H. Alhamani."Attack Of Against Simplified Data Encryption Standard Cipher System Using Neural Networks". *Journal of Computer Science*,2010, 6(1), pp. 29-35.

[10] D. Rudolf. "Development and Analysis of Block Cipher and DES System". Internet: http://www.cs.usask..ca/~dtr467/400/, 2000, [April 24, 2009]

[11] H. Wang. (2002). Security Architecture for The Teamdee System. An unpublished MSc Thesis submitted to Polytechnic Institution and State University, Virginia, USA.

[12] G.W. Moore. (2001). Cryptography Mini-Tutorial. Lecture notes University of Maryland School of Medicine. Internet: http://www.medparse.com/whatcryp.htm [March 16, 2009].

[13] T. Jakobsen and L.R. Knudsen. (2001). Attack on Block of Ciphers of Low Algebraic Degree. *Journal of Cryptography*, New York, *14*(3), pp.197-210.

[14] N. Su, R.N. Zobel, and F.O. Iwu. "Simulation in Cryptographic Protocol Design and Analysis." Proceedings 15th European Simulation Symposium, University of Manchester, UK., 2003.

[15] C.K. Laudan, and C.G. Traver. *E-Commerce .Business .Technology .Society (2nd ed.).* New York: Pearson Education, Inc., 2004.

[16] G.C. Kessler. *Handbook on Local Area Networks: An Overview of Cryptography*. United Kingdom: Auerbach. Available http://www.garykessler.net/library/crypto.html. 2010, [January 3, 2010].

[17] M.A. Yusuf. Data Security: Layered Approach Algorithm. An unpublished MSc Thesis submitted to Abubakar Tafawa Balewa University, Bauchi, Nigeria, 2007.

[18] J. Talbot and D. Welsh. *Complexity and Cryptography: An Introduction*. New York: Cambridge University Press, 2006