ISSN No. 0976-5697

Volume 8, No. 5, May – June 2017

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

A Novel Approach for a Multi-level Secure DNA Based Cryptographic Model

Bismi Beegom S
PG Scholar, Dept. of Computer Science and Engineering
Government Engineering College, Idukki
Kerala, India

Dr. Sangeetha Jose
Assistant Professor, Dept. of Information Technology
Government Engineering College, Idukki
Kerala, India

Abstract: Information technology is growing day by day and security of information from unauthorized access is a major concern in today's security requirements. Cryptography is the art of hiding secret message to a format called cipher text in which no person other than sender or receiver is able to decode it. We already proposed a method of DNA digital coding technology and this paper elaborates the detailed design, implementation and analysis of the model enhancing the message security. The crucial attraction of this extended paper is providing multi-level security of 3 levels with round key selection and message encryption in level 1 followed by 16×16 matrix manipulation using asymmetric key encryption in level 2 and shift operations in level 3. By using 16x16 expansion matrix as key, the scheme improves the independence and the strict avalanche effect without compromising complexity and size of the cipher text. Although this model increases the computation amount because of using the matrix operation, the round key generation ensures better and enhanced security of the secret message.

Keywords: Authenticity; Confidentiality; Cryptography; Decryption; DNA Computing; Encryption

I. INTRODUCTION

Cryptography refers to the science and art of protecting information by encrypting plaintext into incoherent cipher text. To enhance the security of information, several cryptographic approaches are proposed consisting of various kinds of complex mathematical computations on the data. They are not secure enough to provide better security, so still an effective cryptographic algorithm is required for today's security requirements. DNA based encryption technique is one of the recent research area in cryptographic field that can provide higher security and confidentiality to the secret message.

Deoxyribo Nucleic Acid is the carrier of genetic information present in almost all living organisms [1]. In 1953, James Watson [2] discovered the structure of DNA. It consists of two single strands, a Deoxiribose sugar and a phosphate group. This forms a double helical structure consisting of the four DNA bases, Adenine (A) Cytosine (C) Guanine (G) and Thymine (T). Watson-Crick [3] proposed a complimentary rule for DNA sequences that A only joints with T through double hydrogen bond (A = T) and C only joints with G through triple hydrogen bond (C≡ G). In DNA digital coding technology we can represent the DNA nucleotides into two bit binary as A: 0(00), C: 1(01), G: 2(10), T: 3(11). These DNA nucleotides are responsible for the transfer of complex information. Today, DNA based cryptography is taken as a most promising area of research by several researchers due to having the complex structural features and several special characteristics of the DNA [4]. Out of them some used DNA computing, while some other incorporated biological properties of DNA strands and DNA sequence in their algorithms. In this paper a DNA based cryptographic approach is proposed where a 256-bit randomly generated round key is shared among the sender and the receiver. Also a 16 x 16 matrix table is created that will be used as the encryption phase, which uses the asymmetric key encryption technique.

We have designed and implemented a new DNA cryptographic algorithm and elaborated its security compared to other current DNA cryptography algorithms. Section 2 presents related research studies in area of DNA cryptography. The proposed algorithm to encrypt and decrypt binary information of secret message based on DNA cryptography is clarified in section 3. Its detailed implementation is given in section 4. In section 5, we focus on the result analysis based on the strength of the proposed cryptosystem along with key-strength in section 6. Conclusions are presented in section 7.

II. RELATED WORKS

Large numbers of researchers are working on DNA encryption methods and a lot of proposals, directly or indirectly based on the DNA concept were raised. Some of them incorporated the complex characteristics of DNA into their algorithms, while some have used the concept of DNA computing in their algorithms [5]. DNA cryptography focuses on DNA computing in which secret message is encrypted using the DNA nucleotide sequence. DNA computing can be used as conceptual platform for data encryption and decryption by using the two cryptographic approach either symmetric or asymmetric key [6]. The ultimate aim is to scramble data in a way that the adversary can't read or modify the data. Several proposals based on quantitative and qualitative analysis on DNA based cryptography as well as many new cryptographic techniques has been proposed by the researchers. Atanu Majumder et al. [7] have given a new method of encryption to provide better security and reliable data transmission. Abhishek Majumder et al. [1] have derived a method to provide reliable and secured data transmission using the combined technology of DNA based cryptography along with steganography. Snehal Javheri and Rahul Kulkarni [2] have given the concept that how researchers now days can use DNA as a medium for encrypting the messages in a very secure manner. Sanjeev Dhawan and Alisha Saini [8] proposed a new way as to how cryptography can work with DNA computing and how to transmit the secret message securely and effectively. Tushar Mandge and Vijay Choudhary [9] designed a DNA encryption technique based on 4 x 4 matrix manipulations and using a key generation scheme which makes data much secure.

III. PROPOSED METHOD

To provide better security and reliable data transmission, an efficient method of DNA based cryptographic approach is used. The proposed algorithmic model works on block cipher with a key of 256 bit. The detailed design of the approach is elaborated in [7]. Not only the proposed key selection method but also several encryption methods are applied at multiple levels to provide enhanced security aspects. In this model a unique cipher text generation procedure as well as a new key generation procedure which followed by 16 x 16 matrix manipulation is applied to provide high security to the secret information [6]. The abstract figure of proposed scheme is provided in Fig. 1.

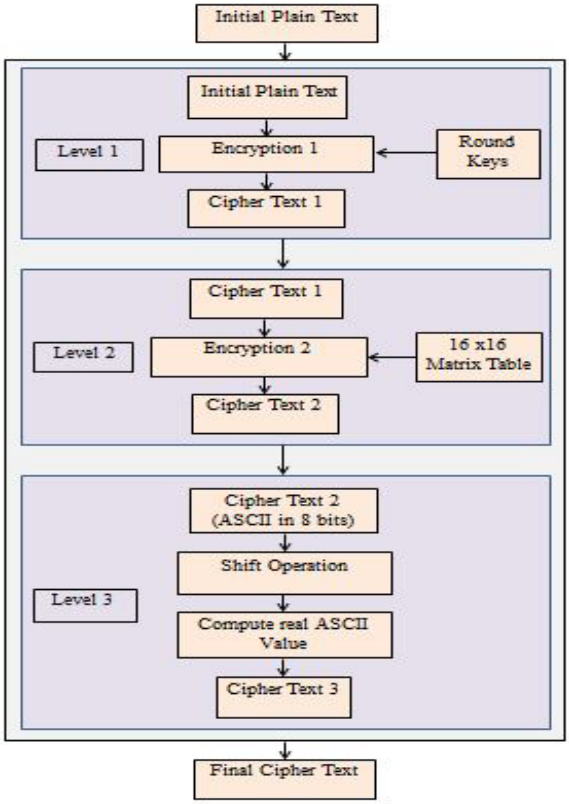

Figure 1. Block Diagram of Proposed Model [6].

IV. IMPLEMENTATION DETAILS

We have successfully implemented the encryption and decryption using C++ code and Visual Studio 2010 Ultimate software. We achieved all the desired results of encryption as well as correct decryption of text message given as input and produce original message as output.

First of all, we have 2 entities, one set as sender and the other as receiver. Connection must be established for the proper communication. We can input a message of desired length with all the 256 ASCII characters including alphabet, number, symbol or blank space and is encrypted into 8 bit sequence finally outputs the correct result.

For the level 1encryption, generation of a 256 bit random key and the encryption works only if the inputted message length is a multiple of 32 characters (256 bit). So we do padding for making the input a multiple of 256 bit. After that we have to select a DNA combination from the total of 24 combinations. Divide each block of inputted message and the 256 bit level 1 key into a length of 64 bits for further processing in level 1.

In level 2, asymmetric key encryption is used and for the implementation, we use RSA encryption scheme. Level 2 key is the 16 x 16 matrix and it contains all the 256 ASCII characters including NULL (binary value 00000000). But a problem in encryption is that, we can't use the Null character and for avoiding such a difficulty, we adopted a method in which first character is encrypted using binary value 00000000 and the rest 255 blocks is encrypted by using the 255 ASCII values except NULL.

In level 3, consecutive left and right shifts are applied, so for each character there is a mapping of 2 characters for each bit and the mapping is done respective of their positions in the intermediate message.

The proposed method provides a better level of security and reliable data transmission. Here the overall method is done in 3 sub-phase, round key generation scheme and message encryption in level 1, 16x16 matrix manipulation in level 2 using asymmetric key encryption followed by shift operations in level 3.

We explain the detailed implementation as follows. Consider the Plain text as

#Wêlcómëtö DÑÅÇRYPTÖgRÄphY2{µ

A. Round Key Generation

Step 1: Randomly generate a 256-bit key.
Let K be the key.
K=1010110010110011001010101010101100110010101000
1101111111100111001010101100000011010100101010101
0101010101010010101010101101011001010101111110010
0000001100000011001100110011001100101100110001001
0111010010100001111111100000000000000101011111111
11010111101010

Step 2: Transform the 256-bit key into an 8 x 8 matrix with each cell having 4-bit key value.

Step 3: Transform the 256-bit key into matrix in row wise.

Table I. Key Values in Matrix Row Wise.

1010	1100	1011	0011	0010	1010	1010	1011
0011	0010	1010	0011	0111	1111	1001	1100
1010	1011	0000	0011	0101	0010	1010	1010
1010	1010	1010	1010	0101	0101	1010	1100
1010	1011	1111	0010	0000	0011	0000	0011
0011	0011	0011	0011	0010	1100	1100	0100
1011	1010	0101	0000	1111	1111	0000	0000
0000	0010	1011	1111	1111	0101	1110	1010

Step 4: Read key bits in column wise, two columns at a time.
1010001110101010101000111011000011000010101110101
011001110100010
1011101000001010111100110101101100110011001110100
010001100001111
0010011101010101000000101111111110101111001001010
011110011110101
1010100110101010000011000000111010111100101011000
011010000001010

Step 5: Label each sub-key block with one of DNA bases A, T, C or G.
A:1010001110101010101000111011000011000010101110101
01011001110100010
T:1011101000001010111100110101101100110011001110100
00010001100001111
C:0010011101010101000000101111111110101111001001010
10011110011110101

G:10101001101010100000110000001110101111001010110000110101000001010

B. Level 1 Encryption Algorithm

The encryption algorithm for the inputted plaintext is as follows.

Step 1: From the input message, read the byte values and transform each byte value into 8-bit binary representation.

Table II. Inputted Plaintext and its 8-bit Binary Values.

#	35	00100011	m	109	01101101	Ñ	165	10100101	T	84	01010100	H	104	01101000
W	87	01010111	ë	137	10001001	Å	143	10001111	Ö	153	10011001	Y	89	01011001
ê	136	10001000	T	116	01110100	Ç	128	10000000	g	103	01100111	2	50	00110010
l	108	01101100	Ö	148	10010011	R	82	01010010	R	82	01010010	{	123	01111011
c	99	01100011		32	00100000	Y	89	01011001	Ä	142	10001110	µ	230	11100110
ó	162	10100010	D	68	01000100	P	80	01010000	p	112	01110000			

Step 2: From binary representation, make plaintext blocks of 256 bits.
0010001101010111100010000110110001100011101000100110110001001011101001001001100100000010001001010010110001111100000000101001001011001010100000101010010011001011001110101001010001110011100000110100001010010001100111101111100110

Step 3: If 256 bits are not present in a block, we should do padding.
Here padding bits: 00001010 00111010 11001110

Step 4: Repeat step 5 to 7 for each block of plaintext.

Step 5: Split the 256-bit block into four 64-bit blocks P1, P2, P3, P4.
P1:0010001101010111100010000110110001100011101000100110110110001001
P2:0111010010010011001000000100010010100101100011111000000001010010
P3:0101100101010000010101001001100101100111010100101000111001110000
P4:0110100001011001001100100111101111100110000010100011101011001110

Step 6: Perform level 1 encryption for each 64-bit block to obtain 64-bit cipher text.

Round 1:

Let, Temporary variables be T11, T12, T13, T14
Compute T14= P4 ⊕ Kl, T13= P3 ⊕ T14, T12= P2 ⊕ T13, T11=Pl ⊕ T12
Let, randomly selected DNA sequence be 'TGCA'.
Fig. 2 shows the Round 1 encryption operation.

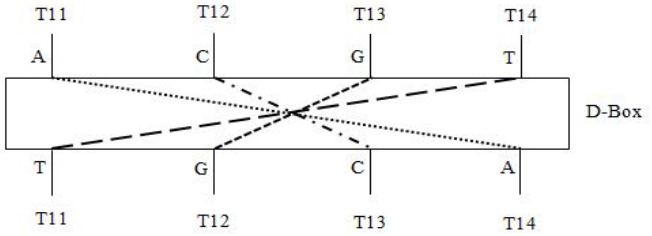

Figure 2. Round 1 Encryption Operation.

K1:1011101000001010111100110101101100110011001110100010010001100001111
T14:P4⊕K1=1101001001010011111000001001000001101010100110000000110011100000 1
T13:P3⊕T14=1000101100000011100101011011100110110010011000010100101111011 0001

T12:P2⊕T13=1111111110010000101101011111110100010111111011010001011111100011
T11:P1⊕T12=1101110011000111001111011001000101110100010011110111010100101001

Round 2:

Let, Temporary variables T2l, T22, T23, T24
Compute T21= T11 ⊕ K2, T22= T12 ⊕ T2l, T23= T13 ⊕ T22, T24= T14 ⊕ T23
Let, randomly selected DNA sequence be 'AGTC'
Fig. 3 shows the Round 2 encryption operation.

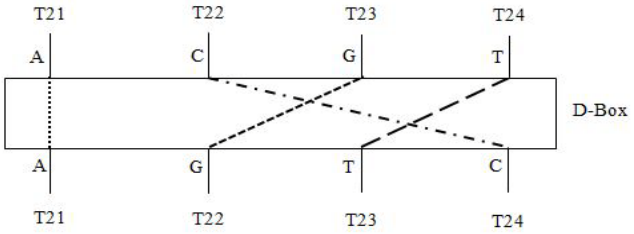

Figure 3. Round 2 Encryption Operation.

K2:1010001110101010101000111011000011000010101110101011100111010010
T21:T11⊕K2=0111111101101101100111100010000110110110111101011100100111001000
T22:T12⊕T21=1000000011111101001010111101110010100001000110001101111000101011
T23:T13⊕T22=0111111101101101100111100010000110110110111101011100100111001000
T24:T14⊕T23=1010110100111110010111110000000101100011110001011101000000001001

Round 3:

Let, Temporary variables T31, T32, T33, T34
Compute T34= T24 ⊕ K3, T33= T23 ⊕ T34, T32= T22 ⊕ T33, T3l= T21 ⊕ T32
Let, randomly selected DNA sequence be 'GATC'
Fig. 4 shows the Round 3 encryption operation.

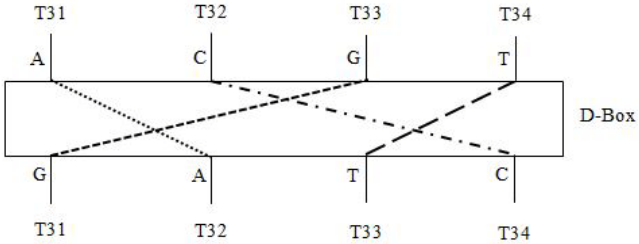

Figure 4. Round 3 Encryption Operation.

K3:10101001101010100000110000001110101111001010110000011010000001010
T34:T24⊕K3=00000100100101000101001100001111110111110110100111110010000000011
T33:T23⊕T34=01111011111110011100110100101110011010011000111000010110111001011
T32:T22⊕T33=00000100100101000101001100001111110111110110100111110010000000011
T31:T21⊕T32=01111011111110011100110100101110011010011000111000010110111001011

Round 4:

Let, Temporary variables T4l, T42, T43, T44
Compute T41= T3l ⊕ K4, T42= T32 ⊕ T4l, T43= T33 ⊕ T42, T44= T34 ⊕ T43
Let, randomly selected DNA sequence be 'CGTA'
Fig. 5 shows the Round 4 encryption operation.

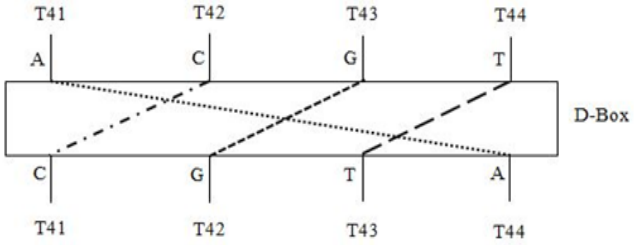

Figure 5. Round 4 Encryption Operation.

K4:00100111010101010000001011111111101011110010010100111110011110101
T41:T31⊕K4=01011100101011001100111111010001110001101011100100010001000111110
T42:T32⊕T41=01011000001110001001110011011110000011001110100001111010100111101
T43:T33⊕T42=00100011110000010101000111110000001110000001001100110110001111010
T44:T34⊕T43=00100111010101010000001011111111110101111001001010011110011110101

Step 7: Combine all the 64-bit cipher blocks to form 256-bit cipher text block.

Step 8: Combine all the 256-bit cipher text blocks to form the final cipher text.

Level 1 output will be:
0101110010101100110011111101000111000110101111001000100010011111001011000001110001001110011011110000011001110100001111010100111101001000111100000101010001101111000001110000010011001101100011111011000100111010101010000001011111111110101111001001010011110011110101

C. *Level 2*

Level 1 output will be the input to level 2.

Step 9: Divide all the 256-bit blocks into 32 8-bit blocks each.
01011100 10101100 11001111 11010001 11000110 10111001 00010001 00111110 01011000 00111000 10011100 11011110 00011001 11010000 11110101 00111101 00100011 11000001 01010001 11110000 01110000 01001100 11011000 11110110 00100111 01010101 00000010 11111111 10101111 00100101 00111100 11110101

Step 10: Generate a 16 x 16 random matrix table.

Step 11: Map each 8-bits with corresponding character in the matrix table

01011100: Ý 10101100: f 11001111: 9 11010001: 2 11000110: Ë 10111001: À 00010001: 0 00111110: ñ 01011000: m 00111000: N 10011100: Œ 11011110: î 00011001: O 11010000: = 11110101: SI 00111101: Ö 00100011: ¬ 11000001: „ 01010001: / 11110000: b 01110000: ¡ 01001100: ´ 11011000: • 11110110: z 00100111: T 01010101: Æ 00000010: ACK 11111111: R 10101111: W 00100101: H 00111100: Â 11110101: SI

Step 12: Take each character's real 8-bit binary values
Ý: 11011101 f: 01100110 9: 00111001 2: 00110010 Ë: 11001011 À: 11000000 0: 00110000 ñ: 11110001 m: 01101101 N: 01001110 Œ: 10001100 î: 11101110 O: 01001111 =: 00111101 SI: 00001111 Ö: 11010110 ¬:10101100 „:10000100 /: 00101111 b: 01100010 ¡: 01101001 ´:10010001 •: 10010101 z: 01111010 T: 01010100 Æ: 11000110 ACK: 00000110 R: 01010010 W: 01010111 H: 01001000 Â: 11000010 SI: 00001111

Here this random matrix table is the level 2 key and is exchanged by using asymmetric key encryption. For this we use RSA Algorithm [10].

D. *RSA Algorithm*

This is an asymmetric cryptographic algorithm suggested by Rivest, Adi Shamir, and Len Adleman in 1977 [11]. The algorithm uses two keys, public (which is announced to all) and private (which is kept secretly with the entity) .

The public key 'e' is used to encrypt the plain text 'P' into cipher text 'C' and a private key 'd' converts cipher text back to plain text.[12]

Encryption Algorithm:

- C = P^e modulo n

This generates the cipher text 'C' from the plain text 'P' using public key 'e'.

Decryption Algorithm:

- P = C^d modulo n

This generates the plain text 'P' from the cipher text 'C' using private key 'd'.

Key Generation:

- Select two large prime numbers 'p' and 'q' where $p \neq q$
- Calculate n = p x q
- Calculate $\phi(n) = (p-1) \times (q-1)$
- Choose integer 'e' such that $1< e < \phi(n)$ and GCD(e, $\phi(n)$) = 1
- Calculate 'd' such that (d x e) modulo $\phi(n) = 1$; $d \equiv e^{(-1)} (\mathrm{mod}\ \phi(n))$

Both sender and receiver must know the value of n. The sender knows the value of e, and only the receiver knows the value of d. Thus, this is a public-key encryption algorithm with a public key of PU = {e, n} and a private key of PR = {d, n}.

The RSA algorithm is very secure because if p and q are very large prime numbers then it is very difficult to find the values of p and q from n. Anyone can use the public key 'e' to encrypt the message but only an authorized user can decrypt it using the private key 'd'.

E. *Level 3*

Step 13: Applying periodic left and right shifts for all the intermediate 8-bits.

All the odd positioned digits are allowed for a left shift and all the even positioned digits are allowed for a right shift.

Step 14: Combine all the 32 character blocks; we will get the final cipher text.

Table III. Generated 16 x 16 Random Matrix Table.

	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0000	™	:	Š	?	…	A	BEL	¡	Y	‹	+	x	P	=	‰	b
0001	Ž	0	W	€	¢	/	—	Ž	†	8	D	Ÿ	„	2	š	BS
0010	ACK	¿	,	CAN	º	µ	NAK	M	®	S	²	STX	ª	‘	)	œ
0011	‡	§	¬	Ç	(	U	R	‘	Ü	&	›	Þ	X	¾	Ú	Q
0100	-	%	Û	}	©	±	¨	°	FF	Ä	1	a	|	ETB	å	ì
0101	Y	¹	H	C	NUL	Æ	$	Á	3	~	.	B	Ã	E	ô	SI
0110	£	DC2	Í	*	¸	p	Ì	SOH	½	Ð	K	ETX	Ë	Î	J	z
0111	¥	5	T	“	SYN	Ó	;	o	É	RS	Å	Ô	\	!	GS	í
1000	>	¶	³	N	<	m	Õ	i	»	ˆ	·	DC1	ä	•	à	g
1001	¤	O	ENQ	Ï	Z	EM	del	â	EOT	nbs	I	À	#	’	Ù	æ
1010	US	c	”	V	¦	Ò	]	È	_	k	d	u	*f*	ß	e	HT
1011	Á	è	4	L	¯	VT	V	ê	ç	,	LF	ã	˜	ESC	ë	osc
1100	h	n	DLE	Â	´	Ý	õ	FS	û	Œ	f	–	6	Ê		q
1101	Ñ	Ø	¼	Ö	`	7	ý	«	F	Ï	“	ù	SO	þ	[	ö
1110	Sst	DC3	Xx	Ñ	ó	É	ctrl	@	ø	×	ð	{	÷	î	ÿ	Sh
1111	T	ò	G	^	CR	ú	j	s	DC4	l	W	rlf	9	ü	SUB	R

The output of the final cipher text is

»3rEM—``øÚ'EMwŽŽRSkYB^1ÒÈ+=¨cFF)®$…‡

From the final output it is clear that randomness in characters increased several times better than the existing algorithms. Here same inputted plaintext results in two entirely different cipher texts. Also same result in intermediate output may results different cipher texts. So it is very difficult for an intruder to guess what the plain text is or to get any hint about the plaintext.

V. RESULT ANALYSIS

Several papers [1,2,7,8,9,13] based on the concept of DNA cryptographic methods were studied. Almost all the researchers in this field used the concept of DNA computing in their algorithms [13]. Operations like XOR, Shifting, Matrix manipulations, Base 10 – Base 4 conversions, Amino acid conversions, DNA digital coding etc. were used in all the papers which resulted in excellent security of the secret message.

In paper [7], symmetric key is employed using an 8x8 matrix for the key selection. Using this there performs 4 XORing in all the 4 rounds resulted in a total of 16 XOR operations. The resulted cipher text is then allowed for a 4x4 matrix manipulation. The final cipher text contains only 16 characters which results a decreased randomness. The resulted final cipher text size is double the size of plain text size.

In paper [1], an 8 x 8 matrix is used for key selection. Symmetric key mechanism is used and in phase 1, a total of 16 XOR operations along with 3 right shift operations are performed. A random mapping using ASCII table along with DNA digital coding is used in phase 1. So the phase 1 result contains only the 4 DNA bases. Result of phase 1 cipher text size is four times the plain text size. In phase 2, the result is embedded into an image where XOR operations are used.

In paper [8], for generating the key, a substitution array has to be generated by entering a total of 5 values through symmetric method. Here performs a division operation to find the quotient and remainder. Then a sequence of Base 10 – Base 4 conversions, DNA Digital coding followed by an Amino Acid conversion is performed. Each 3 characters represent an amino acid and the final cipher text size will be 3 times that of plain text size.

Table IV. Applying Shift Operations.

Left shift		Right shift	
Ý: 11011101	10111011 : »	f: 01100110	00110011 : 3
9: 00111001	01110010 : r	2: 00110010	00011001 : EM
Ë: 11001011	10010111 : —	À: 11000000	01100000 : `
0: 00110000	01100000 : `	ñ: 11110001	11111000 : ø
m: 01101101	11011010 : Ú	N: 01001110	00100111 : '
Œ:10001100	00011001 : EM	î: 11101110	01110111 : w
O: 01001111	10011110 : Ž	=: 00111101	10011110 : Ž
SI: 00001111	00011110 : RS	Ő: 11010110	01101011 : k
¬: 10101100	01011001 : Y	„: 10000100	01000010 : B
/: 00101111	01011110 : ^	b: 01100010	00110001 : 1
¡: 01101001	11010010 : Ò	´:10010001	11001000 : È
•: 10010101	00101011 : +	z: 01111010	00111101 : =
T: 01010100	10101000 : ¨	Æ: 11000110	01100011 : c
ACK: 00000110	00001100 : FF	R: 01010010	00101001 :)
W: 01010111	10101110 : ®	H: 01001000	00100100 : $
Â: 11000010	10000101 : …	SI:00001111	10000111 : ‡

In paper [2], asymmetric key transfer is performed. The plain text is transformed into intermediate cipher text using level 1 private key and is then again transformed into primary cipher text using level 2 private key. Primary cipher text is appended with a starting primer at the MSB and ending primer at LSB. It is then applied for DNA digital coding resulted in a large cipher text size than the plain text.

In paper [9], a separate symmetric key generation procedure along with mini cipher generation is performed. In mini cipher generation phase, a 4x4 matrix manipulation, row shifting, left to right flipping, up to down flipping, and XOR operations were performed. It is then applied for a sequence of Base 10 to Base 4 conversion, data reshaping and DNA digital coding. Primers were appended at the first and last of the generated sequence and are applied for an amino acid conversion. So the generated final cipher text size will be greater than the plain text size.

In our proposed method, we performed a symmetric key selection through an 8x8 matrix. Level 1 consist a total of 16 XOR operations from the four rounds. In level 2 a 16x16 matrix manipulation is done and the key is exchanged using asymmetric key mechanism. Level 3 employs consecutive left and right shift operations. The final cipher text contains all the 256 ASCII characters resulted in an improved randomness. Thus we achieve a final cipher text size, which will be the same size of plain text size along with the padding bits of length less than or equal to 31 characters, through multi-level encryption. In level 1, we divide the plaintext block into multiples of 32 characters and if it is not a multiple of 256 bits we should do padding. So the final cipher text length includes the size of padding bits (less than or equal to 31 characters) along with the text message. Our model results best for large sized messages rather than small packets since padding is applying here.

VI. KEY STRENGTH ANALYSIS

In level 1, a 256 bit round key is randomly generated and is divided into four 64-bit sub-keys. Thus if an attacker tries to apply Brute-Force attack on the cipher text then there are,

$2^{256} = 1.157921 \times 10^{77}$ numbers of possible combinations of keys for cryptanalysis.

Possible number of DNA Bases can be selected in 4! which is equal to 24 combinations.

Thus in level 1, if an attacker goes for a Brute-Force attack, there are,

$2^{256} \times 4! = 2.779010 \times 10^{78}$

In level 2, along with the complexity of using asymmetric key method for the key transfer, a 16 x 16 matrix with all the 256 ASCII Characters are selected as the level 2 key. So if attacker goes for a Brute-Force attack here, total number of possible keys for cryptanalysis be

$256! = 8.578178 \times 10^{506}$

Table V. Result Analysis.

Level 1 256-bit random key generation DNA Sequence Selection	$2^{256} = 1.157921 \times 10^{77}$ $4! = 24$
Level 2 16x16 Matrix Table Manipulation	$256! = 8.578178 \times 10^{506}$
Level 3 Possible combinations for a character	2
In Total	$2^{256} \times 4! \times 256! \times 2$ $= 4.767769 \times 10^{585}$ $\approx \infty$

In level 3, consecutive left and right shift results in a total of 2 possible combinations for each character.

So the total possible number of key combinations for cryptanalysis be

$$2^{256} \times 4! \times 256! \times 2 = 4.767769 \times 10^{585} \approx \infty$$

Therefore it could be concluded from the above large sized keys for level 1 and level 2, it is almost impossible for an attacker to break the system and predict the plaintext or the message that is being sent by the sender.

VII. CONCLUSION

The proposed approach in this paper is more secure and faster than other cryptographic algorithms. The key used in this algorithm is a randomly generated 256 bit key along with 16x16 matrix table, which provides enhanced security to secret message. As there are 256! possible combination for level 2 key, it is almost impossible for an intruder to predict the matrix table sequence. The level 1 message encryption approach is far better than the available cryptographic algorithms based on special operations performed by the round keys on the data. So the intruder will face very difficult to apply different cryptanalysis on the cipher text. Hence we can say that our proposed method is the better method among all the existing methods.

VIII. REFERENCES

[1] Abhishek Majumder et.al. , “DNA Based Cryptographic Approach Toward Information Security”, Springer India 2015.

[2] Snehal Javheri and Rahul Kulkarni, “Secure Data communication and Cryptography based on DNA based Message Encoding” International Journal of Computer Applications (0975-8887) Volume 98-No.16, July 2014.

[3] Genetic home reference, a service of the U.S. National Library of Medicine, Available: http://ghr.nlm.nih.gov/handbook/basics/dna, 2012.

[4] Guangzhao Cui Limin Qin Yanfeng Wang Xuncai Zhang, “An encryption scheme using DNA technology.” Bio-Inspired Computing: Theories and Applications, 2008. BICTA 2008 3rd International Conference, Publication Date: Sept. 28 2008-Oct. 1 2008 ISBN: 9781-4244-2724-6, page(s):37-42; Adelaide, SA.

[5] D. Prabhu and M. Adimoolam, Bi-serial DNA Encryption Algorithm (BDEA), Cornell university library, Available: http://arxiv.org/abs/1101.2577, 2011.

[6] Bismi Beegom S and Dr. Sangeetha Jose, “An Enhanced Cryptographic Model Based on DNA Approach” International Conference on Electronics, Communication and Aerospace Technology (ICECA), IEEE Conference 2017., in press.

[7] Atanu Majumder et.al. , “Secure Data Communication and Cryptography Based on DNA Based Message Encoding”, IEEE Conference, 2014.

[8] Sanjeev Dhawan and Alisha Saini, “A New Encryption Technique for Secure Data Transmission”, IJETCAS Journal, 2012.

[9] Tushar Mandge and Vijay Choudhary, “A DNA Encryption Technique Based on Matrix Manipulation and Secure key Generation Scheme”, ICICES Journal, 2013.

[10] Wang, X., Zhang, Q.: DNA computing-based cryptography. In: Proceedings of the IEEE International Conference, ISBN: 978-1-42443867-9/09 (2009).

[11] William Stallings, *Cryptography and Network Security Principles and Practices*, Fourth Edition, Prentice Hall Publications, November 2005.

[12] Behrouz A Forouzan, *Cryptography and Network Security*, Tata McGraw Hill, Special Indian Edition 2007.

[13] Bibhash Roy et.al. , "An improved Symmetric Key Cryptography with DNA Based Strong Cipher", ICDeCom-2011, BIT Mesra, Ranchi, India, Feb 2011.