Data Security with DNA Cryptography

Anupam Das, Shikhar Kumar Sarma, Shrutimala Deka

***Abstract*—In the present day world, a lot of works have been done in making the data communication safe and secured. But an illegal professional practice, i.e. stealing the data during communication is still going on and the efforts in this field are constantly made to intrude into the network to crack the encrypted data before reaching it to the authenticated destination by some black hat persons. On the other hand, there are lots of researches are going on for making the data secured by encrypting them during communication and an efficient way of generating key to decrypt the encrypted data. There are so many techniques for encryption-decryption, i.e. cryptographic methods are used to make the data safe and secured during transmission. Here we are analyzing a cryptographic technique which is used earlier by some eminent scholars. But in those works the input-output fragments, analysis of the nature of the output generated and the details of the findings from the entire mechanism were missing. Here we discussed it clearly so that it made easy for the future researchers in this field and now they can take this work further more. This is the primary motive of this work and in this paper, we worked very hard to explain the various aspects of the DNA cryptography and its working and also an honest attempt is made to provide the important modules which are used. The algorithm is implemented using C++ and finally some examples of input-output are given for final analysis and conclusion.**

***Index Terms*—Codon, DNA, encryption, decryption, cryptography**

I. Introduction

THIS paper deals with Data security with DNA cryptography. DNA is the abbreviated name for deoxyribonucleic acid which is the store house of all the genetic information of living organisms. The information stored in the genes within the DNA are instructions that tell the body how to construct that organism cell by cell. DNA is shaped in a double helix consisting of two complementary strands that bond to form the final structure. The most basic building block of DNA are the four nitrogen bases, namely, Adenine, Guanine, Cytosine and Thymine. These can bond in a particular fashion and form unique sequences of protein strands. The complementary bases are Adenine and Thymine, and Guanine and Cytosine.

Manuscript received Fen 07, 2019; revised March 31, 2018. Anupam Das is currently an assistant professor in the Department of Computer Science & Information technology, Cotton University India; Email adas_arya@rediffmail.com. Sikhar kumar sarma is currently a Professor in the Department of Information Technology, Gauhati University, India; Email sks01@gmail.com. Shrutimala Deka is a post-graduate student in Cotton University and currently doing project in mobile computing, at CDAC, India; Email: sruti.mail20@gmail.com.

DNA cryptography is the latest technology in cryptographic methods where the natural process of DNA formation has been used to encrypt information and then retrieve them by decrypting it. The biological structure of DNA is such that once information has been transformed into the basic forms of the four nitrogen bases, the process of protein formation

II. The Problem definition

A. Discussion on the problem

The cryptographic algorithms that already exist have the common strategy to have a large keyspace and a complicated algorithm. For symmetric cryptography, the use of one time pad is the most simple solution to the key distribution problem. However, with increasing advancement in technology, it is getting easier to break the algorithms that are widely in use. The increasing length of OTPs are also a cause of concern.
For a more secure data hiding and symmetric key generation using genetic database, DNA cryptography has been proposed.

The data security and protected communication among Mobile Node (MN) and Correspondent Node (CN). This algorithm detects and prevents an attacker who intends to modify the data by using a suitable existing encryption algorithm[1]. The research is also done to map CRC cards into stochastic petri net for evaluating and analyzing quality parameter of security[2]. In another way a method is implemented where a model of security, including control of user access to databases of big data with RMS, the multiplicity and the virtual machine to prevent internal threats, deleting data, insecure or incomplete data protection and control of a third-party can be provided to improve the operation according to the rules of Petri net modeling and simulation.[3].

B. What is DNA?

DNA is the abbrviation of deoxyribonucleic acid. It is a moluecule with a long structure which consists of the unique code called genetic code of any living being. As an insrtuction manual contains the steps and rules for any process the DNA holds the instructions of all the proteins of the bodies of any living beings. This unique code reserves all the characteristics of living beings. This DNA makes every individual unique and this uniqueness is carried in the DNA from the parents to the childs and so to the subsequent hierrachies. All individuals have their own DNA structure as no two individuals are equal,

ISBN: 978-988-14048-6-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

even twins are having unique DNA structures.

C. Some distinguished characteristics of DNA

i) DNA is responsible to make GENOME
ii) The four basic block of DNA are: Adenine (A), Cytosine (C), Guanine(G), and Thymine(T).
iii) The GENOME gets instruction from the sequence of the basic bases of DNA.
iv) A,C,G and T make the strand of the DNA
v) Deoxyribonucleic acid is a two stranded molecule.
vi) In DNA, the strands are "double helix" shaped and twisted within.
vii) DNA molecule with its complementary bases form "rungs".
viii) The combining mechanism is always same as A combines with T and C combines with G.
ix) The joining element of the base is hydrogen.
x) Francis Crick and James Watson found the double helix structure of DNA with the help of the two DNA scientists Rosalind Frankline and Mourice Wilkins.
xi) All living beings have different sizes of GENOME, human being's GENOME size is 3.2 billions.

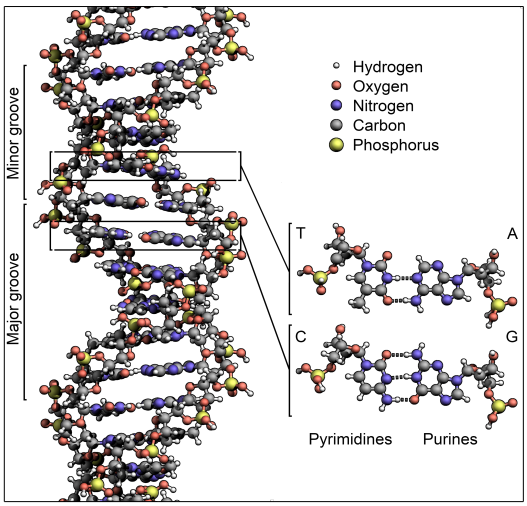


Figure 1a: DNA Structure (source: wikimedia.org)

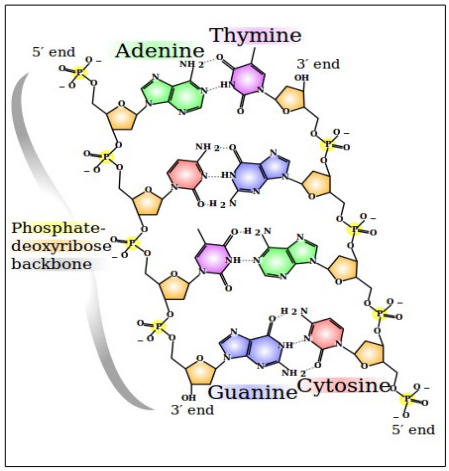


Figure-1b: DNA blocks or 'bases':**Adenine(A), Cytosine (C), Guanine (G) and Thymine (T) (**source: wikimedia.org**)**

D. Advantages of Computing Copyright DNA structure

i) Speed: The conventional computer can compute at the rate of 100 millions of instructions per second (MIPS) approximately but experimentally it is found that DNA strands combinations are generated by combining DNA strands on computing at the rate of 109 MIPS or 100 times faster than a fastest computer.

ii) Storage: The media storage requires 10 12 cubic nanometer to store 1 bit but DNA needs only 1 bit per cubic nanometer.

iii) Power requirements: Since the DNA computing is based on chemical bonds and structures it does not need any outside power.

E. Advantages of DNA storage of data

i) Medium of Ultra-compact Information storage: Very large amounts of data that can be stored in compact volume.ii) A gram of DNA contains 1021 DNA bases = 108 Terabytes of data.
iii) A few grams of DNA may hold all data stored in the world.

III. IMPLEMENTATION

In Implementing the modules of the DNA cryptography, the C++ is used.
A. Key generation

Start
1. Take input string password, lower case with no spaces
2. Store integer value of each character of password
3. Convert to equivalent binary values (7 bits) and store in vector bitset structure named b_key
4. Take pair of binary bits in b_key from the right (LSB) and map them to nucleotides according to table 1. Take the MSB as 0-bit.
5. Store in nucleotide vector
6. Perform annealing by concatenating the nucleotide string with another obtained by using complementary rule in table 2.
7. Perform transcription by mapping each T to U.
8. Parse the string for stop codons UAA, UGA, UAG and record their position.
9. Count the lengths of each string obtained between these stop codons starting from the beginning and ending at the last codon.
10. If multiple strings of various length obtained choose the longest
Else if no codons are obtained choose the entire transcriped string
11. This is the protein key
12. Convert to binary bits and store into fkey
13. Output is fkey

ISBN: 978-988-14048-6-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Stop

B. Encryption

Start

1. Take input string Msg from the user

2. Store integer value of each character of Msg

3. Convert to equivalent binary values (7 bits) and store in vector bitset structure named b_msg

4. Perform circular left shift on each binary blocks in b_msg such that first block is shifted by 1 bit, second block by 2 bits and so on. Blocks which are multiple of 8 (8, 16..) are shifted starting from 1 bit again.

5. Xor b_msg with fkey and store in c_msg.

If fkey is smaller than b_msg

then repeat fkey blocks from the beginning

6. Output is encrypted message c_msg

Stop

C. Bit Encoding to nucleotide

0 0 is used for A *1 1 is used for C*

0 1 is used for G *1 0 is used for T*

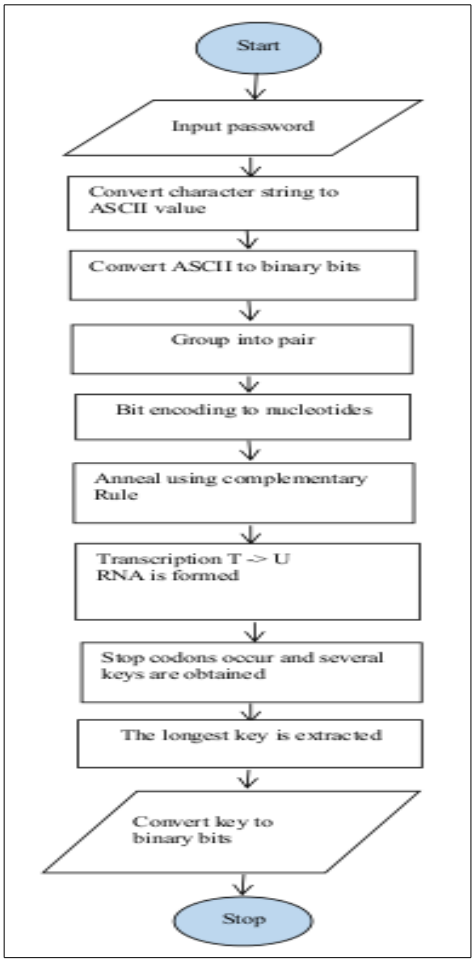
D. Anneal using complementary Rule

A [0 0] its complement is C [1 1]

G [0 1] its complement is T [1 0]

E. STOP Codons

UAG UAA UGA



G. Figure-2a:Flow Charts for Key Generation

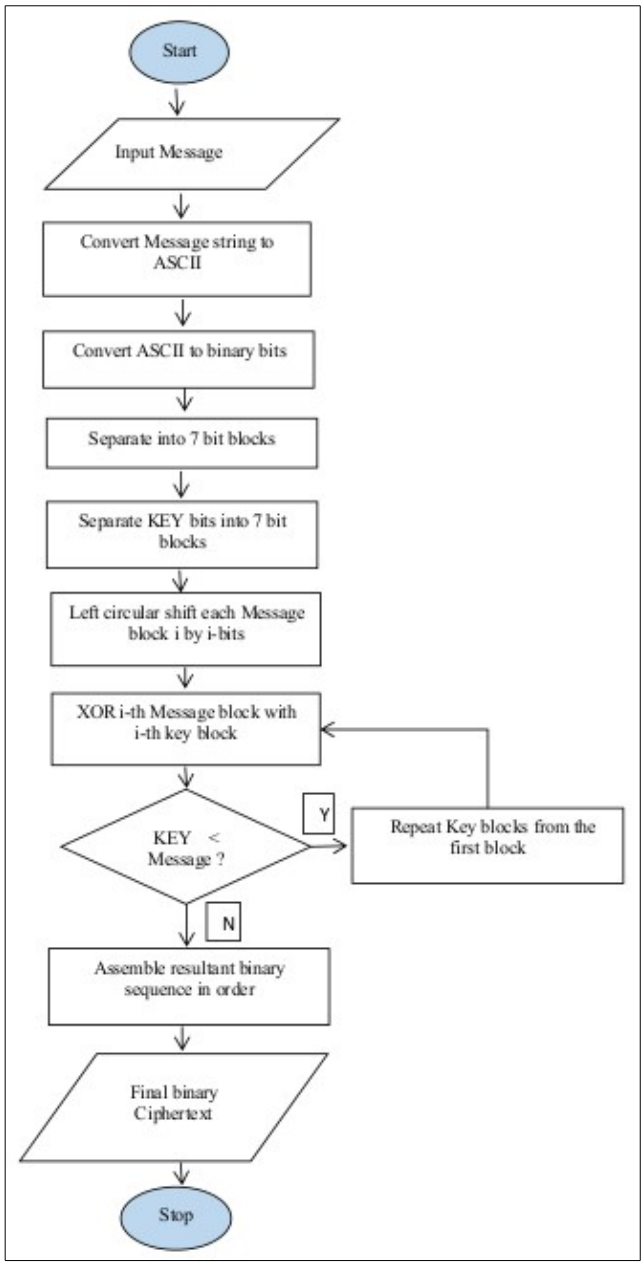


Figure-2b: Flow-chart for Encryption

Key Generation module:

Character string to binary bit string conversion

```
cout << "Enter password [space] Message: ";
        cin >> password;
int len=password.length();
char Mpass[len+1];
 strcpy(Mpass, password.c_str()); //string to char
   for(int t=0; t<len; t++)
   { char letter=Mpass[t];
   bitset<7> b(letter); //to binary

b_key.push_back(b); //store into b_key
}
```

Obtaining nucleotides

(Here, comp parses through the bits)

ISBN: 978-988-14048-6-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

```
int c=0;
 for(int t=0; t<b_key.size(); t++ ) {
            int i=0; while(i < 7) {
           if(i==6) {      comp.clear();
comp.push_back(b_key[t][6]);
if(comp[c]==0) nucleotide.push_back('A');
else if(comp[c]==1) nucleotide.push_back('T');}
 else { comp.clear();         comp.push_back(b_key[t][i]);
 comp.push_back(b_key[t][i+1]);
 if(comp[c]==0 && comp[c+1]==0)
 nucleotide.push_back('A');
 else if(comp[c]==0 &&
comp[c+1]==1) nucleotide.push_back('G');
else if(comp[c]==1 &&comp[c+1]==0)
 nucleotide.push_back('T');
 else if(comp[c]==1 && comp[c+1]==1)
 nucleotide.push_back('C'); } i=i+2; }}
```

Extracting the longest key terminated by $

(Here, p_key stores all the candidate keys that ends with $)

```
int count1=0; //counts to $ encountered to choose zth key
     for(int t=0; t<count; t++)      { //when first key is the
longest       if(z==0) //first key lies before the first $

{ while(p_key[t+1] != '$') {
   key.push_back(p_key[t]);
t++; }  key.push_back(p_key[t]); //for last char
  break;  }  else {  if(p_key[t]== '$')count1++;

if(count1==z) //key is between (z-1)th $ and the zth $
{  while(p_key[t+1] != '$'){  key.push_back(p_key[t+1]);
t++;}  break;} }}
```

Encryption function

```
Circular left shift operation
int e=0;
   for(int t=0; t<b_msg.size(); t++) {
   if(t%7 == 0) { e=0;}     int shift= e+1;     e++;
 b_msg[t]= b_msg[t] << shift | b_msg[t] >> (7-shift);
             cout << b_msg[t] << " "; //print
```

IV. Input-Output

Example-1:
Input password: apassword Input message: a message
Output : Password in binary bits:
11000011110000110000111100111110011111101111101111
11100101100100
nucleotide:TAGTAACTTAGTCACTCACTCTCTCCGT
ACTATGT
Annealed:TAGTAACTTAGTCACTCACTCTCTCCGT
ACTATGTATCATTGAATCAGTGAGTGAGAGAGGC
CTGATACA
Transcription:UAGUAACUUAGUCACUCACUCUCU
CGUGACUAUGUAUCAUUGAAUCAGUGAGUGAG
GAGGCACUGAUACA
Total Stop codons/Number of protein keys: 8
Codon at 0 Codon at 3 Codon at 8
Codon at 27 Codon at 41 Codon at 49
Codon at 53 Codon at 65
$$ CU$
UCACUCACUCUCUCCG$ CUAUGUAUCAU$
AUCAG$ G$ GAGAGGCAC$
Key length 1 : 0 Key length 2 : 0 Key length 3 : 2
Key length 4 : 16 Key length 5 : 11 Key length 6 : 5
Key length 7 : 1 Key length 8 : 9
Longest Key is: key 4 with length: 16
KEY: UCACUCACUCUCUCCG
Final binary key: 1010101 1000011 1000001 1000011
1010101 1000011 1000001 1000011 1010101 1000011
1010101 1000011 1010101 1000011 1000011 1000111
Binary Message: 1100001 1101101 1100101 1110011
1110011 1100001 1100111 1100101
Circular left shifting messages...
1000011 0110111 0101110 0111110 1111100 1110000
1100111 1001011
Encrypted message:
0010110 1110100 1101111 1111101 0101001 0110011
0100110 0001000 1010101
In Decryption....
Message BEFORE XORing with key: 1000011 0110111
0101110 0111110 1111100 1110000 1100111 1001011
0000000
Reversing circular shift(retrieve binary Message):
1100001 1101101 1100101 1110011 1110011 1100001
1100111 1100101 0000000
Decrypted message: a m e s s a g e
Example-2: PASSWORD MESSAGE
pswd End of Conversation
Password in binary bits:
11100001110011111011111100100
nucleotide: AACTCACTCTCTATGT
Annealed:AACTCACTCTCTATGTTTGAGTGAGAG
TACA
Transcription:AACUCACUCUCUAUGUUUGAGUG
GAGAUACA
Total Stop codons/Number of protein keys: 2
Codon at 17 Codon at 21
AACUCACUCUCUAUGUU$
G$ Key length 1 : 17 Key length 2 : 1
Longest Key is: key 1 with length: 17
KEY: AACUCACUCUCUAUGUU
Final binary key: 1000001 1000001 1000011 1010101
1000011 1000001 1000011 1010101 1000011 1010101
1000011 1010101 1000001 1010101 1000111 1010101
1010101
Binary Message: 1000101 1101110 1100100 1101111
1100110 1000011 1101111 1101110 1110110 1100101
1110010 1110011 1100001 1110100 1101001 1101111
1101110
Circular left shifting messages...
0001011 0111011 0100110 1111101 1011001 1100001
1101111 1011101 1011011 0101110 0101110 1111100
1110000 1110100 1010011 0111111 1110110
Encrypted message:
```

ISBN: 978-988-14048-6-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)
```

1001010 1111010 1100101 0101000 0011010 0100000
0101100 0001000 0011000 1111011 1101101 0101001
0110001 0100001 0010100 1101010 0100011
In Decryption....
Message BEFORE XORing with key: 0001011 0111011
0100110 1111101 1011001 1100001 1101111 1011101
1011011 0101110 0101110 1111100 1110000 1110100
1010011 0111111 1110110
Reversing circular shift(retrieve binary Message): 1000101
1101110 1100100 1101111 1100110 1000011 1101111
1101110 1110110 1100101 1110010 1110011 1100001
1110100 1101001 1101111 1101110
Decrypted message: E n d o f C o n v e r s a t i o n

Example-3:PASSWORD MESSAGE
shortpass Meet on the Eastside
Password in binary bits:
11100111101000110111111100101110100111000011000
11100111110011
nucleotide:CACTAGGTCCGTGACTATCTAACTTAGT
ACTCACT
Annealed:CACTAGGTCCGTGACTATCTAACTTAGT
ACTCACTGTGATCCAGGCACTGATAGATTGAATC
GTGAGTGA
Transcription:CACUAGGUCCGUGACUAUCUAACU
AGUCACUCACUGUGAUCCAGGCACUGAUAGAU
GAAUCAGUGAGUGA
Total Stop codons/Number of protein keys: 10
Codon at 3 Codon at 11 Codon at 19 Codon at 24
Codon at 37 Codon at 49 Codon at 52 Codon at 57
Codon at 65 Codon at 69
CAC$ GUCCG$ CUAUC$ CU$ UCACUCACUG$
UCCAGGCAC$ $ AU$ AUCG$ G$
Key length 1 : 3 Key length 2 : 5 Key length 3 : 5
Key length 4 : 2 Key length 5 : 10 Key length 6 : 9
Key length 7 : 0 Key length 8 : 2 Key length 9 : 5
Key length 10 : 1
Longest Key is: key 5 with length: 10
KEY: UCACUCACUG
Final binary key: 1010101 1000011 1000001 1000011
1010101 1000011 1000001 1000011 1010101 1000111
Binary Message: 1001101 1100101 1100101 1110100
1101111 1101110 1110100 1101000 1100101 1000101
1100001 1110011 1110100 1110011 1101001 1100100
1100101
Circular left shifting messages...
0011011 0010111 0101110 1001110 1111011 0110111
1110100 1010001 0010111 0101100 0011100 1111100
0111010 1110011 1010011 0010011 0101110
Encrypted message:
1001110 1010100 1101111 0001101 0101110 1110100
0110101 0010010 1000010 1101011 1001001 0111111
1111011 0110000 0000110 1010000 1101111
In Decryption....
Message BEFORE XORing with key: 0011011 0010111
0101110 1001110 1111011 0110111 1110100 1010001
0010111 0101100 0011100 1111100 0111010 1110011
1010011 0010011 0101110
Reversing circular shift(retrieve binary Message):
1001101 1100101 1100101 1110100 1101111 1101110
1110100 1101000 1100101 1000101 1100001 1110011
1110100 1110011 1101001 1100100 1100101
Decrypted message: M e e t o n t h e E a s t s i d e

V. Theoretical analysis

A. Biological aspect

Placing an encrypted protein sequence among the vast number of protein strands poses the issue where the desired strand can not be randomly located in the DNA. The key, thus, must also include where exactly is the encrypted message kept otherwise searching for the message itself will take too many years. Hence, adversaries with no knowledge of the key can not possibly break the algorithm. Most strands only differ by few nucleotides. Without the key, it is impossible to even guess the ciphertext, let alone decrypt it.
This is a unique property of DNA cryptography and no modern cryptographic algorithm provides this kind of security in data.

B. Mathematical aspect

Mathematical computations are minimal in DNA cryptography. This is because the role of confusion and diffusion are negligible, since the ciphertext will not give away any clues to the plaintext. A large keyspace has been our solution so far to reduce the possibility of breaking cryptographic algorithms. This is eliminated in DNA cryptography as key is well hidden within the DNA. In other cases, we could also use genetic database to generate OTPs and eliminate the need to input password for the key altogether and keep the keyspace small at the same time.

C. Observations

i) A single password can produce multiple protein keys in relation to the number of stop codons formed. On the other hand, a single password may also derive a single protein key. Then, we must use the one and only string as our key.

ii) A given password will produce the same number of codons with the same number of keys and key length. This means that there is consistency in our output and it is not randomly generated each time. This also implies that using the same password again and again may not be such a good idea as the same protein key formed will become common knowledge with the people it is being shared with.

iii) When the number of keys with the same maximal length is more than one, the program chooses the first longest key. This is done for convenience and has no

ISBN: 978-988-14048-6-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

particular reason in the security of the algorithm. If we decide to randomly choose to pick any one of the keys it will make the encryption even stronger with no scope of guessing the key.

iv) The length of the password entered by the user is directly proportional to the number of stop codons or protein keys that we find. Heuristically it has been observed that for shorter passwords the number of key decreases. When the password is longer, the number of protein keys also increases. This does not mean that a password with length three will always produce less keys than that with length say five. On an average, the password length and number of keys are directly proportional.

v) When the password is exceptionally small, as we have tested for the sake of proper output analysis, we may find that the stop codons produced is zero. This means that we may not even have a key. In that case, we take the entire annealed string as our key.

vi) It has been found that usually the first or one of the first three proteins keys are found to be the longest. It is extremely rare that the last protein key be the longest.

vii) In message encryption using the final key, the codons play no role in how the final encrypted string will look like. This is because the codons are part of only the key generation process and does not influence the rest of the processes.

viii) Stop codons are UAA, UAG and UGA. Thus, for every occurrence of Thymine T, it becomes more likely that a stop codon will form in that position since every T will be transcriped into U in the subsequent steps in key generation function.

VI. Conclusion

The DNA Cryptography can now be used as the strong algorithm for data security as its cracking time and key generation are so designed that it seems the time taken to decrypt the ciphered data is quite impossible for a life time. So it should be the first choice for the cyber security researchers for securing data and information. The study made here is comprehensive and the information given here will largely help to the researchers for doing further work in this line of thinking. The modules given for key-generation, encryption, decryption will definitely help the subsequent works for implementing cryptographic techniques. The present work will also help to implement and apply DNA methodologies to cryptography and steganography.

References

[1] A. Mehdizadeh, M. Mohammadpoor, Z. Soltanian, “Secured Route Optimization and Micro-mobility with Enhanced Handover Scheme in Mobile IPv6 Networks”, in International Journal of Engineering (IJE), TRANSACTIONS B: Applications Vol. 29, No. 11, (November 2016) pp. 1530-1538.

[2] H. Motameni a , M. Nemati b, Mapping, “CRC Card into Stochastic Petri Net for Analyzing and Evaluating Quality Parameter of Security”, in IJE TRANSACTIONS B: Applications Vol. 27, No. 5, (May 2014) pp. 689-698

[3] A. S. Abad, H. Hamidi, “An Architecture for Security and Protection of Big Data”, in International Journal of Engineering (IJE), TRANSACTIONS A: Basics Vol. 30, No. 10, (October 2017), pp. 1479-1486

[4] B. B. Raj, J. Frank, T.Mahalakshmi, “Secure Data Transfer through DNA Cryptography using Symmetric Algorithm”, in International Journal of Computer Applications, Vol 133-No 2, pp. 0975-8887, January 2016

[5] A. Roy, A. Nath, “DNA Encryption Algorithms: Scope and Challenges in Symmetric Key Cryptography”, in International Journal of Innovative Research in Advanced Engineering, ISSN: 2349-2763, Issue 11, Volume 3, Nov, 2016.

[6] W. Stallings, “Cryptography and Network Security”, Third Editio, Prentice Hall International, 2003.

[7] N. S. Kolte, K. V. Kulhalli and S. C Shinde, “DNA Cryptography using Index-based Symmetric DNA Encryption Algorithm”, International Journal Of Engineering Research and Technology, ISSN 0974-3154 Vol 10, No1 ,2017.

[8] A. K. Kaundal, A. K. Verma, “DNA Based Cryptography: A Review”, in International Journal of Information & Computation Technology, ISSN 0974-2239 Vol 4, No 7, 2014, pp. 693-698

[9] G. Jacob, A. Murugan, “DNA Based Cryptography: An overview and analysis”, ResearchGate, Jan 2013

[10] S. Karthiga, E. Murugavalli, “DNA Cryptography”, in International Research Journal of Engineering and Technology, p-ISSN 2395-0072, Vol 5, March 2018.

ISBN: 978-988-14048-6-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)