Inspired Pseudo Biotic DNA Based Cryptographic Mechanism Against Adaptive Cryptographic Attacks

E. Suresh Babu[1], C. Naga Raju[2], and Munaga HM Krishna Prasad[3]
(Corresponding author:E Suresh Babu)

Department of Computer Science and Engineering, K L University[1]
Green Fields, Vaddeswaram, Guntur District, A.P. 522 502, India
Department of Computer Science and Engineering, YSR Engineering College of Yogivemana University, India[2]
Department of Computer Science and Engineering, University College of Engineering, Kakinada[3]
Pithapuram Road, Nagamallithota, Kakinada, Andhra Pradesh 533003, India
(Email: sureshbabu.erukala@gmail.com)

(Received July 2, 2014; revised and accepted Mar. 23 & July 4, 2015)

Abstract

DNA Cryptography is a new cryptographic paradigm from hastily growing bio molecular computation, as its computational power will determine next generation computing. As technology is growing much faster, data protection is getting more important and it is necessary to design the unbreakable encryption technology to protect the information. In this paper, we proposed a biotic DNA based secret key cryptographic mechanism, seeing as DNA computing had made great strides in ultra-compact information storage, vast parallelism, and exceptional energy efficiency. This Biotic Pseudo DNA cryptography method is based upon the genetic information on biological systems. This method makes use of splicing system to improve security, random multiple key sequence to increase the degree of diffusion and confusion which makes resulting cipher texts difficult to decipher and makes to realize a perfect secrecy system. The formal and experimental analysis not only shows that this method is powerful against brute force attack and chosen cipher text attacks, but also it is very efficient in storage, computation as well as transmission.

Keywords: Brute force attack, chosen cipher text attack, DNA based symmetric cryptography

1 Introduction

DNA Computing is a Bio-molecular Computation (BMC) which makes use of biological methods for performing massively parallel computation. This can be a lot quicker than a conventional Silicon Chip computer, for which large quantities of hardware needed for performing parallel computation. These DNA computers [1, 29, 32, 37] don't just make use of massively parallel computation, but also uses ultra-compact information storage in which large amount of information that can be stashed in a more compact away with, which massively exceeds in conventional electronic media, (i.e., A single gram of DNA [1, 8, 14] comprises 10^{21} DNA bases which equals to 108 terabytes. A hardly few grams of DNA, possibly contains all data stored in world. This cross-topical field of DNA Computing [33] combines the ideas from biological sciences, computer science and chemistry. In 1994, Adleman [8] designed a study to solve the Travelling Salesman problem that attempts to visit each city exactly once and try to find every possible route using molecules of DNA. Hence, this inspired model provides the potential ability of working out many problems that were previously thought impossible or exceedingly difficult to solve out with the traditional computing paradigm such as encryption breaking, game strategy etc.

As Power of the parallel processing is increasing day to day, modern cryptosystems can be easily cryptanalyzed by the cryptanalyst, the world is looking for new ways of information and network security in order to safeguard the data as it carries. The purpose of using cryptography in the areas of bio-molecular computation to bring up a promising technology for providing of unbreakable algorithms, but these DNA cryptography lacks the related theory which is nevertheless still an open problem to model the good DNA cryptographic schemes.

In this paper, we used pseudo DNA based cryptographic technique which is based on central dogmas of biological system, which is not same as original DNA cryptography [12, 18, 23]. This proposed method only makes use of DNA mechanisms and terminology of DNA function rather than actual biological DNA sequences (or oligos). The encryption and decryption processes are initiated with DNA transcription, splicing system and RNA translation [28].

The remainder of the paper is organized as follows. In Section 2 specifies the related work. Section 3 and Section 4 describes the scope of research and overview of DNA. The proposed Pseudo DNA-Based Symmetric Cryptosystem mechanism and its security analysis are discussed in Section 5 and Section 6. Section 7 describes the simulation results. Finally Section 8 concludes this paper with future work.

2 Related Work

The domain of information and network security is persistently looking for unbreakable cryptosystem to protect the information while transmitting on to the network, but it seems that every cryptographic encryption technology comes across its end game as the new computing technologies are evolving.

DNA is very potent and exciting study direction from a cryptographic point of view which requires simple and effective algorithms, of late, many scientists have projected a various DNA-based encryption algorithms, but it is too early to decide the perfect complete model for some cryptographic functions, such as DNA authentication methods, digital signature and secure data storage as these cryptographic models is still in the initial phase. Adleman [1, 3] proposed the hypothetical model of DNA computing for any bio-molecular computational problem which provides vast parallel computing. As his background stemmed from computer encryption, he particularly envisioned DNA computing in helping to create encryption and decryption algorithms in the area of cryptography.

Gehani et.al from Duke University had investigated the procedure of DNA based Cryptography [18] for one time pads (OTP). They proposed the large number short sequence of message can be encrypted using one-time pads. These small sequences of DNA can avail from massive one time pad using public key infrastructure (PKI) [34]. Leier [23] proposed the data hiding procedure predicated on DNA binary sequences to achieve DNA encryption scheme; Applying DNA Computation, Kazuo [22] resolved the trouble for generation of key distribution, he also proposed DNA based secret key encryption system; Amin [4] proposed DNA YAEA encryption algorithm which is a conventional secret key encryption algorithm. Ning [28] proposed pseudo DNA based cryptography along with the initial Secret key to build DNA cryptosystem which is also a symmetric encryption algorithm.

3 Scope of Research

The Intellectual property, which is being transferred over the internet, can be easily acquired and is vulnerable to many security attacks [6, 7, 8, 10] such as Worm Hole attack, Man in the middle attack, IP Spooning, Black Hole Attack etc. Securing all the information passed through networked computers is primarily more important for any application or system, Already a great heap of effort had been put on the cryptology, As a result, various security mechanisms have been designed such as DES,RSA, ECC, DSA, etc., to achieve very high level of security. But these mechanisms require complex factorization of large prime numbers and the elliptic curve problem, for which still a lot of investigation is required to find a proper solution. Moreover, the RSA cryptosystem is based on the intractability of large prime factorization as there are no known efficient algorithms to find largest prime factors.

DNA cryptography is a techniques which have been devised to break RSA scheme. This techniques is used to self-assembly of DNA tiles to fully break RSA scheme [3, 9, 11]. If these techniques are able to break RSA, RSA will no more remain practical. Further, DNA-based Methods had also been developed to break the cryptosystems based on elliptic curves. These methods utilize a parallel multiplier to perform basic biological operations and for adding the points on elliptic curves, it uses both parallel divider and a parallel adder [24]. Moreover, so far many researchers had concentrated on breaking the cryptosystem using several DNA-based methods which are presently being practiced.

4 Overview of DNA

In order to understand the rudimentary principles of Deoxyribonucleic Acid (DNA) Cryptography in a emerge area of DNA Computing, it is necessary to address the background details of central dogma of molecular biology, that is, how a DNA sequence is actually transcripted and translated into a protein sequence as shown in Figure 1. DNA (Deoxyribo Nucleic Acid) is the fundamental hereditary material that stores genetic information found in almost every living organisms ranging from very small viruses to complex human beings. It is constituted by nucleotides which forms polymer chains. These chains are also known as DNA strands. Each DNA nucleotides contains a single base and usually consists of four bases, specifically, Adenine (A), Guanine (G), Cytosine(C), and Thymine (T) represent genetic code. These bases reads from the start promoter which forms the structure of DNA strand by forming two strands of hydrogen bonds, one is A with T and another is C with G; These DNA sequences are eventually transcripted and interpreted into chains of amino acids, which constitutes proteins.

4.1 Transcription

Transcription is a process of newly prepared intermediary copy of DNA called mRNA instructions that transpires in the nucleus of the cell, these instructions are contained and created in DNA i.e. DNA sequence [40] which con-

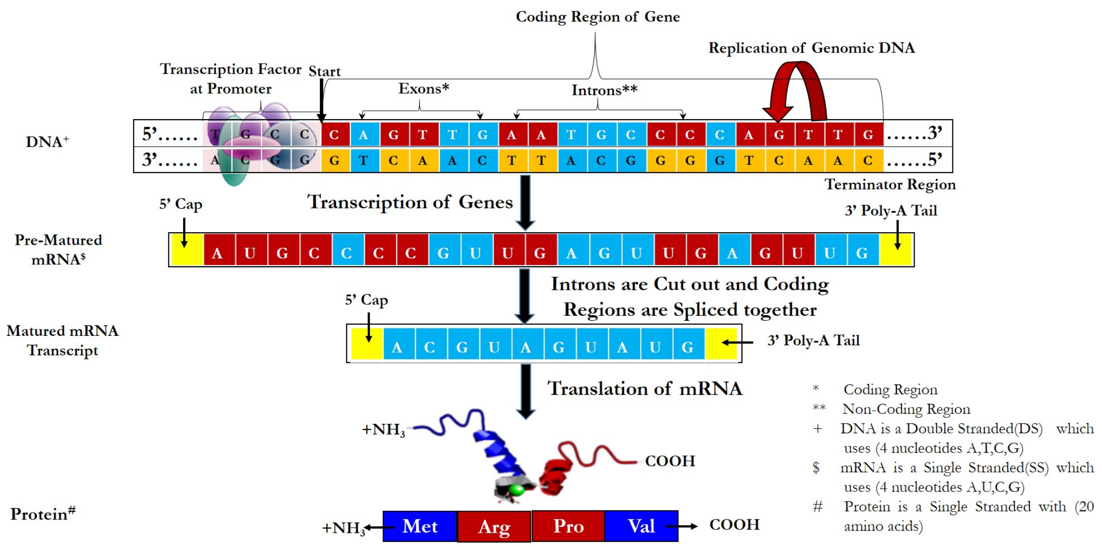

Coding Region of Gene
Replication of Genomic DNA
Transcription Factor at Promoter
Start
Exons*
Introns**
DNA+
5'...... T G C C C A G T T G A A T G C C C C A G T T G ......3'
3'...... A C G G G T C A A C T T A C G G G G T C A A C ......5'
Terminator Region
Transcription of Genes
5' Cap
3' Poly-A Tail
Pre-Matured mRNA$
A U G C C C C G U U G A G U U G A G U U G
Introns are Cut out and Coding Regions are Spliced together
5' Cap
Matured mRNA Transcript
A C G U A G U A U G
3' Poly-A Tail
Translation of mRNA
+NH3
COOH
Protein#
+NH3
Met
Arg
Pro
Val
COOH
* Coding Region
** Non-Coding Region
+ DNA is a Double Stranded(DS) which uses (4 nucleotides A,T,C,G)
$ mRNA is a Single Stranded(SS) which uses (4 nucleotides A,U,C,G)
# Protein is a Single Stranded with (20 amino acids)

Figure 1: Central dogma of molecular biology

tains the nucleotides A, G, C and T. are transcript into mRNA sequence, here mRNA is a single stranded that contains the nucleotides A, G, C and Uracil (U). This intermediary mRNA polymerase that binds the enzyme which is responsible for copying DNA into RNA.

4.2 Translation

Translation is also a process that contains the RNA copy of DNA to make a protein. i.e, the mRNA copy of DNA sequence is translated into a distinct amino acids that can be chained together to form protein. There are 20 distinct amino acids which is a basic building block of a protein. On the mRNA, there are also certain ending three-bases to sign the end of the translation. Essentially, RNA transcript process undergoes the processing step called splicing steps, in which INTervening (introns) sequences are cut out and discarded, keeping the Expressed (extrons) sequences to form mRNA, In this mRNA translation process, a grouping of three nucleotides [33, 35] called codons are translated into the amino acids according to the genetic code table [25].

5 Pseudo DNA-based Symmetric Cryptosystem

The pseudo DNA cryptography [17, 26, 28, 39] method just takes the standard principle ideas of central dogma of molecular biology method i.e. the pseudo DNA cryptosystem (Encryption/Decryption) process is similar to the DNA transcription, splicing system and RNA translation of the real organisms, but it is different from existing DNA based cryptography [5, 13, 15, 21, 36], as this method only make use of DNA mechanisms and terminology of DNA function rather than actual biological DNA sequences (or oligos); therefore, this proposed method is a kind of pseudo biotic DNA based cryptography method.

The pseudo DNA cryptography technique consists of transcription/splicing system and translation processes which is similar to central dogma of molecular biology essentially, in the transcription process undergoes the processing step called splicing steps, in which INTervening (introns) sequences are cut out and discarded, keeping the Expressed (extrons) sequences to form Messenger, Ribonucleic Acid (mRNA). In the translation process of mRNA, a grouping of three nucleotides called codons are translated into the amino acids according to the genetic code table [25, 31].

In order to make the statistics of the cipher text and the multiple rounds of random keys of the encryption as complex as possible to decipher, we have modified the original splicing system process. Originally, the starting codes of the introns and ending codes of the introns are very easy to guess. In this proposed work, we have modified start codes and the pattern codes to specify the introns. The non-continuous pattern codes are used to confuse the adversary and hard to guess the introns, by defining which parts of the DNA frame to be removed, and which DNA frame to be kept. Further, the no of the splices, the starting code of the frame and removed length of the pattern codes can be used to determine the key, the ending codes of the DNA frame are not required.

5.1 Symmetric Cryptography Principles

Generally, Modern Cryptography [27] solves many cryptographic algorithm with the help a KEY. The cryptosystem which comprises of Encryption and Decryption functions using the same Key(K) that can be interpreted as symmetric cryptography, which is represented with two functions: $E_k(M) = C$ and $D_k(C) = M$. In this cryptosystem, first, both the sender and receiver must agree on a key as well as cryptosystem in order to communicate securely. Hence, the success of such symmetric cryptosystem is mainly depends upon its Key.

5.2 Communications Using Pseudo DNA Symmetric Cryptography

The conventional secret key encryption scheme $\prod = (\mathcal{E}_{\mathcal{K}}, \mathcal{D}_{\mathcal{K}})$ is usually represented with two algorithms; one is $\mathcal{E}_{\mathcal{K}}$ function, which is a stateful encryption algorithm with k randomized key generation algorithm. It takes the plaintext 'M' along with random key 'K' and returns a cipher text 'C'; usually represented $\mathrm{E_K(M) = C}$ and another is $\mathcal{D}_{\mathcal{K}}$ function, which is a deterministic decryption algorithm, which takes a string 'C' and the same random key 'K' and returns the equivalent plaintext 'M' ; usually represented as $\mathrm{D_K\ (C) = M}$ where $\mathrm{M} \in \{\mathrm{A, T, C, G}\}^*$, finally we perform that $\mathrm{D_k(E_k(M)) =}$ M or all $\mathrm{M} \in \{\mathrm{A, T, C, G, 0, 1}\}^*$.

Let us assume Alice want send the message to Bob; both agree on a key and a cryptosystem; Alice takes her plaintext message and performs two different conversions i.e., First the plaintext information is converted into in the binary numerical representation, and Second, she transforms binary forms into DNA form (A for 00, C for 01, G for 10, T for 11) and encrypts it with the random key (Here, the Key will number of the splices, the starting code of the frame and removed length of the pattern codes). This creates a cipher-text; Alice sends the cipher text message to Bob through public channel; Bob decrypts the cipher text message with the decryption algorithm and random key (No. of the splices, the starting code of the frame and removed length of the pattern codes) received from secure channel and reads it. Therefore, to perform above Communications model using Symmetric Pseudo DNA based Cryptography the following steps can be described briefly:

1) Alice takes the plaintext and converts to binary form and then converts into DNA form as shown in Algorithm-1.

2) Alice will scan DNA form of information to generate the variable length random key by generating the No of the splices from the specified DNA pattern, the starting code of the DNA frame to find out the introns, introns places and removed length of the pattern codes i,e. introns are removed from the specified DNA sequence as the first round of Key Generation, which is shown in Algorithm-2 and Algorithm-3.

3) With the help random key (splicing system), Alice will transcript the DNA sequence into the mRNA strand, as shown in Algorithm-4.

4) After Generating mRNA Strand, Alice also generate the variable length random sub key by generating the No of the splices from the specified mRNA pattern, the starting code of the mRNA frame, introns places and removed length of the pattern codes as the Second round of Processing.

Algorithm 1 Generate binary value

```
BEGIN
Binary Text required to search the DNA patterns
X ⇐ 0
for  i ⇐ 0 to n do +1 do
    Take an initial quotient variable and while it is not zero do
    while (Quotient is not equal to Zero) do
        t[i]⇐ quotient mod 2 +Zero
        Divide the Quotient by 2;
        Increment i
    end while
    while (Variable Y is greater than Zero) do
        Reverse the string to get resultant binary code
        Increment x;
        Decrement y;
    end while
    if String length Mod 2 ==1)  then
        The string length should be a multiple of 2
    else
        Padding Zero at beginning of the String
    end if
end for
End
```

Algorithm 2 Generate DNA strand

```
Begin
DNA patterns
Assign the 2-bit patterns of Binary String to convert in to DNA SEQUENCE
for  (i ⇐ 0 to stringlength do +2 ) do
    if  (String Cmp(Binary,"00") ==0)  then
        DNA-Code[i]⇐ 'A';
    else
        if  (String Cmp(Binary,"01") ==0)  then
            DNA-Code[i]⇐ 'C';
        else
            if  (String Cmp(Binary,"10") ==0)  then
                DNA-Code[i]⇐ 'G';
            else
                if  (String Cmp(Binary,"11") ==0)  then
                    DNA-Code[i]⇐ 'T';
                end if
            end if
        end if
    end if
end for
End
```

5) Again, the Spliced mRNA strand are translated into the amino acids according to the genetic code table (61 codons to 20 amino acids) which forms protein sequence, as shown in Algorithm-5.

6) The protein sequence (Cipher Text) will be sent to the to Bob through public channel.

7) The Random Variable Length Key such as no of

Algorithm 3 Generation of variable random key

```
Begin
Input: DNA patterns, Print the length of DNA Strand for Generating Variable Random Key, No of Splices of Sender choice for Slicing for M-RNA code generation, The starting indices of Sender choice, The lengths of DNA Strand to be deleted
Output: Generating Variable Random Key using Splicing System
for i and j ⇐ 1 to n do +1 do
  Performing the Sub Key Patterns
end for
for i ⇐ 0 to j do +1 do
  Converting patterns key to binary format
  Quotient ⇐ key[i];
  while (i is less then Length of DNA Strand ) do
    key[n] ⇐ quotient mod 2;
    Divide the Quotient by 2
  end while
  if (quotient==0) then
    No Sub Key is Present in DNA Strand and Generate Sub Key in mRNA
  end if
  for i ⇐ 1 to nj do +1 do
    Stored Splices in a Key Space
  end for
end for
End
```

Algorithm 4 Generation of mRNA strand

```
Begin
Input: DNA patterns, Random Key
Output : Generating mRNA Strand
for i ⇐ 0 to n do +1 do
  Extract the slices part from DNA code using slices process
end for
for i ⇐ 0 to Length(DNA Strand) do +1 do
  Except the slices part sort the remaining part from DNA code to form M-RNA code
end for
End
```

Algorithm 5 Protein code generation

```
Begin
Input:mRNA Strand
Output: Protein Code (Cipher Text)
for i ⇐ 0 to Length(mRNA Strand) do +3 do
  Copy the 3-bit patterns from DNA code to protein code array to match the codon table formats
  Compare and replace appropriate protein value from codon table
  Finally print the PROTEINCODE which will be our final CIPHER TEXT
end for
End
```

splices, the starting index, pattern codes length of the introns, the positions and places of the introns, the cut out the introns, random mapping of codon-amino acids will form the key to decrypt the cipher text (protein sequence) and also sent to the Bob through a secure channel as shown in Figure 2.

8) On Bob (Receiver) side, when he receives random keys and protein form (Cipher text) of data from Alice through the secure channel.

9) Bob decrypts the cipher text message using the random key reversible translation to recover mRNA sequence from protein sequence, and then recover DNA form of information, in the reverse order as Alice encrypt the information.

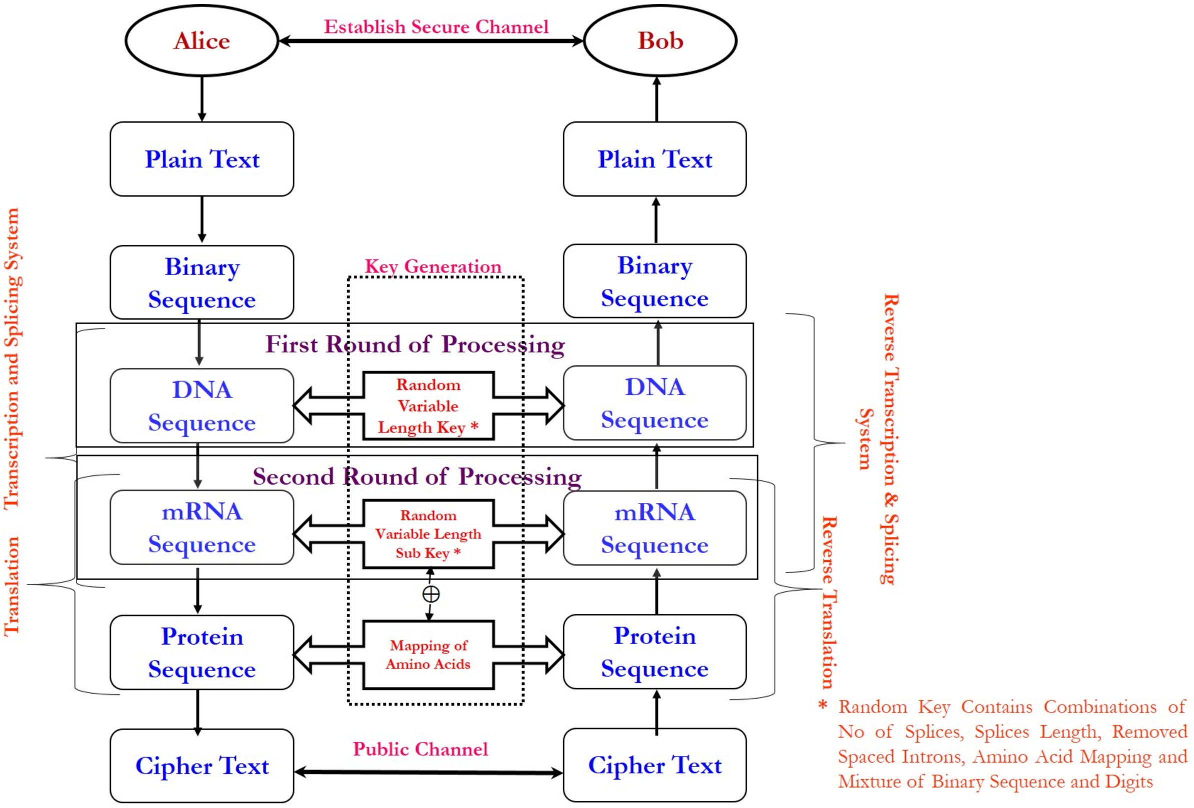
10) Bob can then recover then binary form of information, and finally gets what Alice sent him.

5.3 Key Generation Using Splicing Systems

Head [2, 19] proposed the splicing system which captures mathematically $\Sigma_{DNA} = \{$A, C, G, T$\})$. Where DNA strands are referred as strings over the finite alphabet. However, these splicing systems were introduced more than twenty years ago, that is when nobody spoke about DNA computing. In fact, only in 1995-thus, after Adleman's paper - Splicing Systems [30, 38] have been suggested to represent DNA computations and their computational properties, by various authors. The central operation of the splicing systems: Given an alphabet S and two strings, $y \in \sum^*$, it is defined the splicing of x and y, as indicated by the rule r. formally, a splicing rule r defined on the alphabet Σ is a word of the form $\alpha_1 \# \beta_1 \$ \alpha_2 \# \beta_2$ where α_1, β_1, α_2, $\beta_2 \in \sum^*$, while # and \$ are special symbols that are part of Σ. If we have $x = x_1\alpha_1$, β_1, $\alpha_1' y = y_2\alpha_2$, β_2, y_2' and $r = \alpha_1\#\beta_1\$\alpha_2\#\beta_2$, we write: $(x, y) \rightarrow_r (p, q)$ to indicate that the strings p and q are obtained from the values of x and y applying the splicing rule r.

5.4 Key Selection Using Splicing Systems

In order to improve the security of the proposed algorithm, we had designed random keys of key generator based upon splicing system of central dogma, the random key information will be selected from DNA sequence and mRNA sequence, the user random generated key sequence of DNA strand and random generated key sequence of mRNA will be XORed and resultant random key is shared between Alice and Bob through private or secure channel. As shown in Figure 1, the Biotic DNA symmetric cryptosystem is designed in such way that, the adversary cannot decrypt the encryption algorithm without the information of the key, it is very difficult to



* Random Key Contains Combinations of No of Splices, Splices Length, Removed Spaced Introns, Amino Acid Mapping and Mixture of Binary Sequence and Digits

Figure 2: Pseudo DNA symmetric cryptosystem

find the Random DNA secret key sequence and Random mRNA key sequence. Suppose, If the adversary applies brute force search for finding the random key in order to decrypt the cipher-text, then, the attacker should spend numerous time and resources because DNA has an extremely large amount of data storage capacity, which requires tens of millions of nucleotides in order to find the correct no of splices, cutoff introns, starting position of DNA/mRNA strand, removed DNA/mRNA strand and mapping of Amino acids. Hence, the algorithm is secure and safe enough against Brute force attack and Chosen cipher text (CCA) [16].

6 Security Analysis of Biotic Pseudo DNA Cryptography

The main objective to Strength any security technique is to protect the network and information from any malicious activities. Mainly, Time and computational complexity are two of the most significant parameters for whatever sort of cryptographic schemes. Semantic Security and Message Indistinguishability are the two fundamental computational -complexity analog of Shannon's definition of perfect privacy [20]. Former one represents the infeasibility to learn anything about the plaintext from the cipher text and the later one represents the infeasibility of distinguishing between the given pair of messages.

6.1 Formal Definitions of Semantic Security (SS) and Message Indistinguishability (MI)

Definition 1. *Semantic security ensures that nothing can be learned just by looking at a cipher text. i.e., cipher text reveals no information about the message. For every distribution X over $\{0,1\}^n$ and for every partial information h: $\{0,1\}^n \rightarrow \{0,1\}^n$. For every interesting information f: $\{0,1\}^n \rightarrow \{0,1\}^n$. For every attacking algorithm A, running time complexity $t' \leq t(n)$, $t(n)$ is a polynomial in n, there exists algorithm S such that:*

$$Pr_{m \leftarrow X, \left(P_k, {}^{S_k}\right) \leftarrow G(n)} \left[A\left(E\left(m, {}^{P_k}\right), P_k, h\left(m\right)\right)\right] \\ \leq Pr_{m \leftarrow X}\left[S\left(h\left(m\right)\right) = f\left(m\right)\right] + \epsilon\left(n\right)$$

where $\varepsilon(n)$ is a negligible quantity which depends upon value of n. For example, $\varepsilon(n)$ may be $\frac{1}{\mathrm{P(n)}}$ where $p(n)$ is a polynomial in 'n' of a large degree.

Definition 2. *Given two encryptions of messages m_0 and m_1, the probability of guessing the message is very close to the random probability of guessing the correct message $(\frac{1}{2})$. The security of message indistinguishability states that the inability to distinguish two plaintexts (of the same length). i.e., the cipher texts are computationally indistinguishable. For every two messages m_0, $\mathrm{m}_1 \in \{0,1\}^n$,* *for every algorithm A that runs within time $\leq t(n)$.*

$$Pr_{i\epsilon\{0,1\}}(P_k, S_k) \leftarrow G\left(n\right)\left[A\left(E\left(^{m_i}, P_k\right), P_k\right) = i\right] \\ \leq \frac{1}{2} + \epsilon\left(n\right).$$

Theorem 1. *If the symmetric-key encryption scheme constitutes indistinguishable encryptions then it is semantically secure.*

Proof. If $X = [m_0, m_1]$, $f(m_0) = 0$, $f(m_1) = 1$, $h(\cdot)$: empty output string From Semantic security, for every opponent A there exist a simulator S, such that

$$Pr_{m \leftarrow X, \left(^{P_k}, ^{S_k}\right) \leftarrow G(n)}\left[A\left(E\left(m, P_k\right)\right) = i\right] \leq \\ Pr_{m \leftarrow X}\left[S\left(h\left(m\right)\right) = i\right] + \epsilon(n).$$

Now since the simulator receives no information: Pr[S( ) = i] = $\frac{1}{2}$, regardless of S. Thus,

$$\mathrm{Pr}_{\mathrm{i}\ \in\ \{0,1\}\quad (\mathrm{P_k},\ \mathrm{S_k})\ \leftarrow \mathrm{G(n)}}\left[\mathrm{A}\left(\mathrm{E}\left(\mathrm{m_i}\ \ \mathrm{p_k}\right)\ \ \mathrm{p_k}\ \right) = \mathrm{i}\ \right] \\ = \frac{1}{2} + \in(n).$$

Now, For every $\mathrm{m_0}$, $\mathrm{m_1}$ $\in$ $\{0,1\}^{\mathrm{n}}$, for every algorithm A that runs within time = t(n), for every a $\in$ $\{0\ \ 1\}^*$

$$Pr_{(P_k, S_k) \in G(n)}\left[A\left(E\left(m_1, P_k\right), P_k\right) = a\right] \\ -Pr_{(P_k, S_k) \in G(n)}\left[A\left(E\left(m_1, P_k\right), P_k\right) = a\right] \leq 2\epsilon(n).$$

Let us call above equation as (*) then we can say that

$$(\mathrm{t}, \in) - \mathrm{MI} \rightarrow (*) \equiv \sim (\mathrm{t}, \in) - \mathrm{MI}.$$

Define $A'(c,p) = \begin{cases} 1, & if A(c,p) = a \\ 0, & otherwise. \end{cases}$

So,

$$\begin{aligned} & Pr_{i\in\{0,1\},(P_k,S_k)\leftarrow G(n)}\left[A\left(E\left(m_i, P_k\right), P_k\right) = i\right] \\ = & \frac{1}{2}Pr_{(P_k,S_k)\leftarrow G(n)}\left[A'\left(E\left(m_0, P_k\right), P_k\right) = 0\right] \\ & +\frac{1}{2}Pr_{(P_k,S_k)\leftarrow G(n)}\left[A'\left(E\left(m_1, P_k\right), P_k\right) = 1\right] \\ = & \frac{1}{2}(1 - Pr_{(P_k,S_k)\leftarrow G(n)}\left[A'\left(E\left(m_0, P_k\right), P_k\right) = a\right]) \\ & +\frac{1}{2}Pr_{(P_k,S_k)\leftarrow G(n)}\left[A'\left(E\left(m_1, P_k\right), P_k\right) = a\right] \\ = & \frac{1}{2} + \frac{1}{2}Pr_{(P_k,S_k)\leftarrow G(n)}\left[A'\left(E\left(m_0, P_k\right), P_k\right) = a\right] \\ & -Pr_{(P_k,S_k)\leftarrow G(n)}\left[A'\left(E\left(m_1, P_k\right), P_k\right) = a\right] \\ \leq & \frac{1}{2} + \in(n) - MI \end{aligned}$$

is violated. □

The theoretical result of Symmetric DNA based encryption function gives a diffusion cipher text, which is hard to compute plaintext without random key Therefore, security analysis of Symmetric DNA based cryptography is efficient and very powerful against certain cryptographic attacks.

Definition 3. *A polynomial-time-computable predicate b is called a hard-core of a function f, if every efficient algorithm, given $f(x)$, can guess $b(x)$ with success probability that is only negligibly better than one-half. Formally speaking, we define a hard-core predicate as follows: A polynomial-time-computable predicate $b\left\{0,1,G,T,A,C\right\}^* \rightarrow \left\{0,1,G,T,A,C\right\}$ is called a hard-core of a function f if for every probabilistic polynomial time algorithm $A^{'}$, every positive polynomial $p(\cdot)$, and all sufficiently large n's,*

$$\mathrm{P_r}\left[\mathrm{A}^{'}\ \left(\mathrm{f}\left(\mathrm{U_n}\right)\right) = \mathrm{b}\left(\mathrm{U_n}\right)\right] < \frac{1}{2} + \frac{1}{p(n)}.$$

Note that, for every b: $\{0, 1, A, G, T, C\}^* \rightarrow \{0$, 1, A, G, T, $C\}$ and f: $\{A$, G, T, $C\}^* \rightarrow \{A, G, T, C\}$ there exist obvious algorithms that guess $b\left(U_n\right)\ from\ f\left(U_n\right)$ with success probability at least one-half, e.g. the algorithm that obliviously of its input, outputs uniformly chosen DNA Strand. Also if b is a hardcore predicate for any function, then $b\left(U_n\right)$ must be almost unbiased (i.e. $|P_r\ \left[b\left(U_n\right) = 0\ \right] - \ \left[b\left(U_n\right) = 1\right]$ must be a negligible function of n). Now our encryption scheme make use hard-core predicate (hp) and we analyze the security of the scheme.

6.2 Encryption Algorithm

Assume the Encryption Function (F_{bin}, F_{dna} F_{rna}, F_{pro}) and a hard core predicate $B(X,k)$ for FKEY.Here we want to encrypt a plaintext p and b is a key, which is the secret information.

Theorem 2. *Symmetric DNA based encryption scheme for Message, i.e. Encryption $E_{F_{bin},F_{dna},F_{rna},F_{pro}}(b,k)$ is MI secure.*

```
SCHEME (($F_{bin}$, $F_{dna}$ $F_{rna}$, $F_{pro}$, $F_{Key}$), B, b)
/*** Encryption $E_{F_{bin}}$, $F_{dna}$ $F_{rna}$, $F_{pro}(b,k)$ ***/

  1) /*** Encryption $E_{F_{bin}}$, $F_{dna}$ $F_{rna}$, $F_{pro}$ (b, k) ***/
     Pick $X \overset{U}{\leftarrow} \{A,G,T,C\}^n$;
     Return (F(X,k) ,b,B(X,k));

  2) /*** key generation ***/
     Generate the Combination of pairs (kb, Kd, $K_{dna}$)
     using $F_{dna}$ and Rdna;

  3) /*** Decryption $D_{F_{bin}}$, $F_{dna}$ $F_{rna}$, $F_{pro}(c, F(X,k))$
     ***/
     $X = D[F(X,k), K_b, K_d, K_{dna}]$
     Return (c, B(X, k)).
```

Proof. Suppose the encryption scheme is not (t, ϵ)-MI secure, So it exists a PPT algorithm A' such that

$$Pr_{\substack{b\in\{0,1,G,T,A,C\} \\ (P_k,S_k)\leftarrow G(n),X\overset{U}{\leftarrow}\{A,G,T,C\}^n}}[\mathrm{A(F(X,\ k),\ b,\ B(X,\ k),\ k)}].$$

Consider the following algorithm:

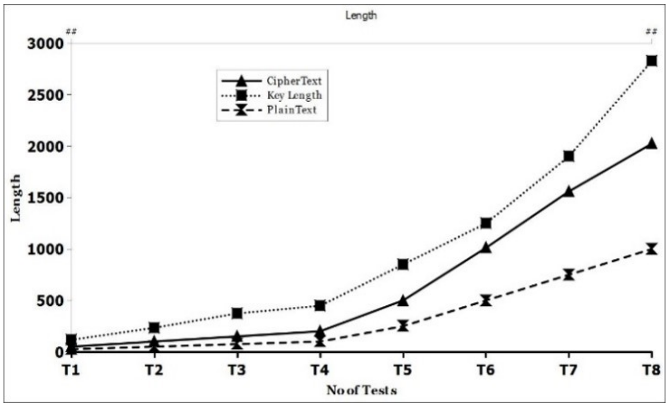


Figure 3: Performance analysis between plaintext, cipher text and key length

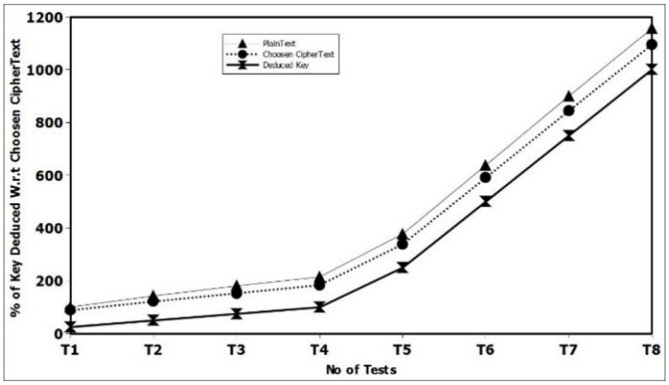


Figure 4: Performance analysis between plaintext, chosen cipher text and its deduction of key

A” (y,k)
{
1). Pick random c ∈ protein form;
2). Return (c,A (y,c,k))[
}

$$Pr_{X \overset{U}{\leftarrow} \{A,G,T,C\}^n}[A'(F(X,k),b,B(X,k),k)]$$

$$Pr_{\substack{c\in ProteinsForm\\ (P_k,S_k)\leftarrow G(n), X\overset{U}{\leftarrow}\{A,G,T,C\}^n}}\text{[A(F(X,k),c,B(X,k),c)].}$$

Since A’ is a PPT algorithm just as A.So B is not a hardcore predicate (hp) according to definition. This is a contradiction. Hence the primary assumption was wrong. Hence SCHEME ((F_{bin}, F_{dna} F_{rna}, F_{pro}), B, b) F_{Key} is MI secure. Hence proved □

7 Cipher Text Indistinguishability

Cipher text indistinguishability is a one of the important security property for numerous encryption schemes. Instinctively, if a cryptosystem has the property of indistinguishability, then an opponent will be not able to distinguish pairs of cipher texts focused around the message they encrypt. The property of indistinguishability is viewed as an essential requirement for most of the provably secure key cryptosystems under chosen cipher text attack, chosen plaintext attack and adaptive chosen cipher text attack. A cryptosystem is viewed as ”secure in terms of indistinguishability” if no opponent A, given an encryption of a message haphazardly chosen from a two-component message space controlled by the opponent, can distinguish the message decision with likelihood better than that of random guessing (1/2). If any opponent can succeed in recognizing the chosen cipher text with likelihood fundamentally more noteworthy than . There are numerous security definitions in terms of indistinguishability, depending on presumptions made about the abilities of the attacker. At this point, when the cryptosystem is viewed as secure. if, no opponent can guess randomly with significantly probability more prominent better than half. The most well-known definitions used in cryptography are indistinguishability with various attacks [16] such as (non-adaptive) chosen cipher text attack (IND-CCA), chosen plaintext attack (IND-CPA), adaptive chosen cipher text attack (IND-CCA2). The convenient way to sort out above definitions to secure DNA based Encryption is by considering different conceivable objectives and attacks models. The objective here is to make an opponent’s powerlessness to realize any data about plaintext underlying a challenge cipher text. In this conception, the adversary cannot determine from which plaintext the challenge cipher text came from.

The attack models are considered here are Adaptive Chosen cipher text Attack, Non-Adaptive Chosen Cipher text Attack and Chosen Plain text Attack (CPA). In IND-CPA is characterized between an opponent and a challenger. For schemes focused around computational security, the adversary is modelled in such a way; he must finish inside a polynomial number of time steps to guess. In IND-CCA1, the adversary has a right to access to unscrambling oracle O. Nevertheless, the opponent can utilize this oracle only before it gets the challenge cipher text y. Finally, In IND-CCA2, Adversary has a right to gain the access of oracle O and his inquiry to the oracle may rely on upon the challenge cipher text y. however, the only restriction with this attack is that the opponent cannot query the oracle to the challenger to decrypt the cipher text y.

In formalizing IND-Atk, An opponent A as a pair of probabilistic polynomial time algorithm $= (_{1}, _{2})$. Here, A runs in two stages. Whereas, A1 generates a message pair, encrypt one of them and send to A2 as challenge cipher text. We say A2 is successful depending on its goal; the goal is here to tell which message is in encrypted form.

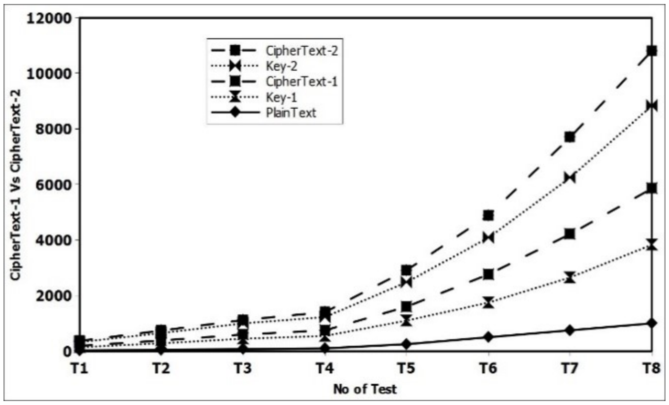


Figure 5: Analysis of message indistinguishability (MI) of plaintext, cipher text with different random keys

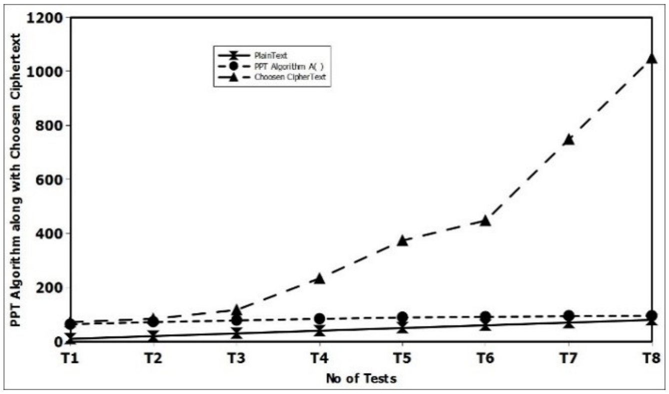


Figure 6: Percentage of chosen cipher text w.r.t PPT algorithm)

7.1 Indistinguishability of IND-CCA1 or IND-CCA2

Definition 4. *Let $\prod = \{\mathcal{E}, \mathcal{D}, \mathcal{K}\}$ be a secret key encryption scheme. For an opponent A and $b = \{0, 1\}$ characterize the experiment*
Experiment:

$$\mathrm{Exp}_{\pi}^{\mathrm{ind-cca}}(\mathrm{A\ \ b\ });$$

$$\mathrm{a} \leftarrow \mathrm{K}; (\mathrm{x}_1,\ \mathrm{x}_2,\ \mathrm{s}) \leftarrow \mathrm{A}^{\mathrm{E_a D_a}}(\mathrm{Find});$$

$$\mathrm{y} \leftarrow \mathrm{E_a}(\mathrm{x_b});$$

$$d \leftarrow A^{E_a D_a}(Guess, y, s);$$

$$\text{Return d};$$

It is assigned that $|\mathrm{x}_0| = |\mathrm{x}_1|$ above and that opponent A does not query for decryption oracle $\mathrm{D_a}(\cdot)$ on cipher-text y in the supposition phase. Characterize the advantage between opponent A and function π respectfully, takes as follows:

$$\mathrm{Adv}_{\pi}^{\mathrm{ind-cca}}(\mathrm{A}) = \mathrm{P_r}\left[\mathrm{Exp}_{\pi}^{\mathrm{ind-cca}}(\mathrm{A\ \ 0\ }) = 0\right] - \mathrm{P_r}\left[\mathrm{Exp}_{\pi}^{\mathrm{ind-cca}}(\mathrm{A\ \ 1\ }) = 1\right]$$

$$\mathrm{Adv}_{\pi}^{\mathrm{ind-cca}}(\mathrm{t,q_e,q_d},\ \mu,\ \mathrm{v}) = \overset{\mathrm{MAX}}{\mathrm{A}}\mathrm{Adv}_{\pi}^{\mathrm{ind-cca}}(\mathrm{A}).$$

The maximum time-complexity t with at most q_e and q_d encryption and decryption oracle queries and totaling these queries with at most μ bits and finally choosing $|\mathrm{x}_0| = |\mathrm{x}_1| =$ v bits. Hence, the worst-case time-complexity for this experiment is $\mathrm{Exp}_{\pi}^{\mathrm{ind-cca}}(\mathrm{A\ })$ plus the total size of the code of opponent A.

The analogy of the above definition E (P$_K$, "M") which represents the encrypted message "M" under the random key "P$_K$": The challenger produces encrypts arbitrary cipher texts and the opponent is offered to access the decryption oracle, which decrypts self-assertive cipher texts at the opponent's request, retaining the plaintext. The opponent may keep on query the decryption oracle significantly even after it has received a challenge cipher text, but it may not pass the cipher text for further processing:

Step 1. The challenger generates a key P_K in multiple rounds of transcription (first key), spicing system (second key) and translation process (third key) (e.g., a key size in K_{dna}, Kmrna, Kmap) which produces cipher text and given to the opponent.

Step 2. The opponent calls to the decryption function based on haphazard cipher texts.

Step 3. The challenger selects the key $P_k = \{K_b, K_d, K_{pdna}\}$ randomly and sends the challenge cipher text $C = E(P_k, M)$ back to the opponent.

Step 4. The opponent is free to execute any number of encryptions or computations.

Step 5. Once again, the opponent may further calls to the decryption function, but this time he may not submit the cipher text "C".

Step 6. Finally, the opponent generates the outputs by guessing for the value of message "M". This scheme is secure against IND-CCA2 if no opponent can guess with non-negligible time.

A DNA based private key scheme ((F_{bin}, F_{dna} F_{rna}, F_{pro}, F_{Key}), B, b) is (t, q, ϵ) secure in IND-Atk sense. If for all pair of different messages of same length and any opponent A, that runs within given time t and performs at most q queries to the decryption oracle O, $\epsilon(n)$ denotes the advantage of the algorithm over a random guess.

$$Pr_{(P_k,S_k)\leftarrow G(n)}\left[A^0\left(P_k, E_{pk}\left(m_1\right)\right) = 1\right] - Pr_{(P_k,S_k)\leftarrow G(n)}\left[A^0\left(P_k, E_{pk}\left(m_0\right)\right) = 1\right] \leq \in (n)$$

where the oracle is

$$O = \begin{cases} - & \text{if IND} - \text{CPA} \\ D_{sk} & \text{if IND} - \text{CCA2} \end{cases}$$

and the adversary cannot query the decryption oracle at $E_{pk}(m_i)$. Therefore, Informally an pseudo DNA based

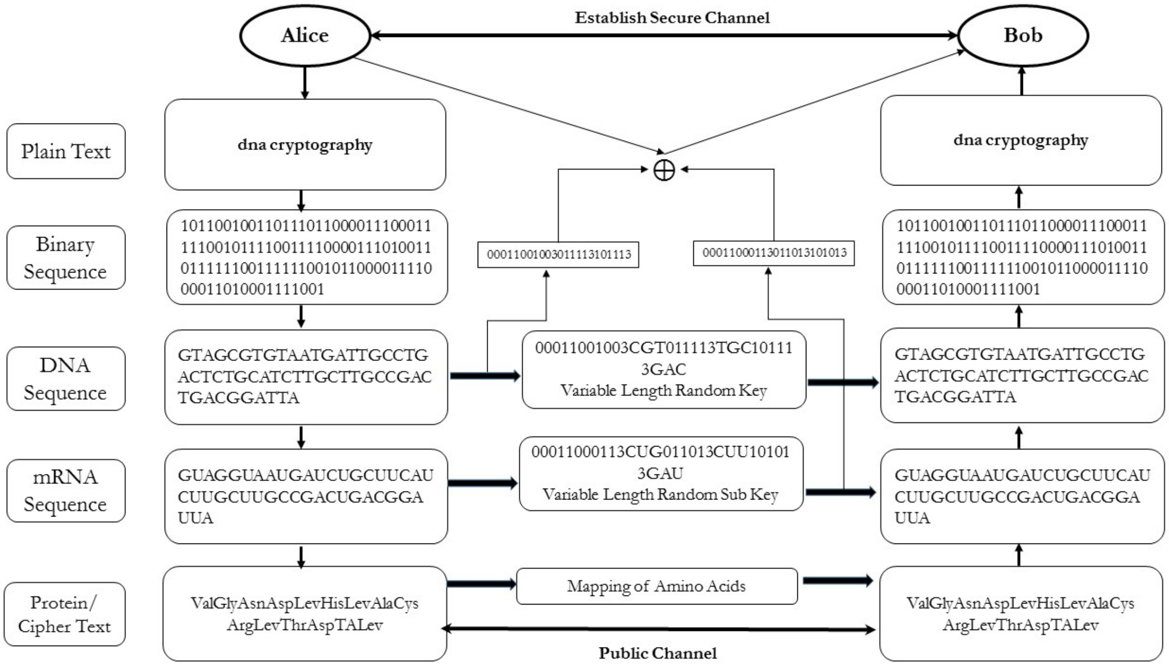


Figure 7: Flowchart of biotic DNA based symmetric cryptosystem

encryption scheme is secure if for each adversary A and for every polynomial $P(\cdot)$, there exist a 'N' such that, $P_r\left(A \text{ succeeds in the attack}\right) < \frac{1}{P(n)}\ \forall\ n > N$.

From the definition of Semantic Security, for all distribution over $\{A, T, G, C\}$; For All Partial information h: $\{Proteins\}^n \rightarrow \{Proteins\}^n$; For all interesting information f: $\{0, 1, A, T, C, G\} \rightarrow \{O, 1, digits, DNAStrands\}$; Adversary A with time complexity $t' < t(n)$, $t(n) = \sum t_d n^d$; $\exists$ Simulating algorithm S such that

$$P_{r}{}^{(P_k,S_k)\leftarrow G(n)}_{X\leftarrow\{A,T,G,C\}^n}\left[A\left(E\left(m,{}^{P_k}\right), P_k, h\left(m\right)\right) = f\left(m\right)\right]$$
$$\leq P_{r}{}^{(P_k,S_k)\leftarrow G(n)}_{X\leftarrow\{A,T,G,C\}^n}\left[S\left(h\left(m\right)\right) = f\left(m\right)\right] + \epsilon\left(n\right)$$

where $\epsilon(n)$ is a negligible quantity; then $E(\cdot)$ is called semantically (t, ϵ) is secure.

From the definition of Message Indistinguishability; For all messages m_0, $m_1 \in \{0, 1\}^n$; For all Adversary A with time complexity $t' < t(n)$, $t(n) = \sum t_d n^d$:

$$Pr^{(P_k,S_k)\leftarrow G(n)}_{i\in\{0,1\}}\left[A\ \left(E\left(m_i\ \ p_k\ \right)p_k\ \right)\right)=i\ ]\ \ = \frac{1}{2} + \in\left(n\right)$$

where $\epsilon(n)$ is a negligible quantity; $E(\cdot)$ is called (t, ϵ) MI secure; n is the security parameter such as key length; $\epsilon(n)$ is a negligible quantity.

8 Results and Simulation Analysis

To study the feasibility of our theoretical work, we have implemented and evaluated the pseudo Biotic DNA cryptography method in C++ and conducted a series of experiments in a network simulator [NS2] to evaluate its effectiveness. The experiment results show that this method is more efficient and its increase the power against certain adaptive cryptographic attacks. The experimental values were obtained by evaluating the multiple running times of the pseudo Biotic DNA cryptography on a software program running Uduntu-13.04. Our simulations are based on sender and receiver programs. On the sender side, the sender first converts the plaintext into the binary sequence, which in turn translated into the DNA Strand. Indeed, necessary padding is performed at the time of translation in order to have the compatibility DNA strand. After translation, the sender will generate the random variable length key using the splicing system process of the central dogma. In other words, the sender will generate the random key with a mixture of binary sequence, decimal digit and DNA Strand, which makes the adversary hard to guess the key and translates into mRNA sequence. Next, the sub key generation is chosen at mRNA sequence using pseudo random key generator. Subsequently, these two random keys will be XORed with random mapping of codon-amino acids to produce the protein sequence. To put in another way, the mRNA is translate into the amino acid sequence called codons, which produces the proteins sequence. Eventually, the whole transcription and translation process of central dogma creates enciphered information. These enciphered information and Random Key are transferred to receiver through different channels, i.e., enciphered information through public channel, and Random key through secure channel.

On the destination side, the receiver receives the enciphered information and random key from different channels. Consequently, the receiver uses decryption algorithm and the same key information to decipher the enciphered information. To be more specific, first, the receiver performs reverse translation process to recover from protein sequence into mRNA form using same sub-key with the help of pseudo random generator. Next, reverse transcription process is performed using reverse splicing process to recover from mRNA to DNA form. Finally, he recovers the plaintext using the recovery translator that the sender had send him.

Figure 7 illustrates the proposed biotic cryptography method. Let us exhibit with an example; how this proposed cryptographic protocol works. Alice creates a cipher text and public key converts into the DNA Strand. Moreover, she also generates the variable length random key (splicing system) “00011001003CGT011113TGC101113GAC” from DNA Strand of cipher text. However, DNA form of public key will be converted into equivalent numerical form for clear understanding of the key. The main specific reason of converting the public key into DNA form is to have optimized key size. Subsequently, the sub key 00011000113CTG011013CTT101013GAT” is chosen from mRNA sequence “GUAG GUAA UGAU CUGC UUCA UCUU GCUU GCCG ACUG ACGG AUUA”using pseudo random key generator. Finally, these mRNA Strand is translate into amino acid sequence (codons), which produces proteins sequence “Val Gly Asn Asp Lev His Lev Ala Cys Arg Lev Thr Asp TA Lev”. This encoded proteins sequence will be sent to the Bob. Bob decrypts the cipher text using the same random key to recover the plain text.

I verified experimentally that, the encryption and decryption can be performed effectively a given key. Moreover, different plaintexts with the combination of alphabets, digits and few special characters are chosen with increasing size that includes short-text, average-text and long-text. Indeed, each plaintext is stored in ASCII format and number of bits are calculated to that of 8 or 16 times that of the length of the plaintext. The original plaintext size is calculated with different 64, 128, 256, 512, 1024 and 2048-bits random key and the resulting cipher text size are examined. These random key are used to examine the efficiency of the algorithm in terms of computation, storage and transmission. Furthermore, we also investigated that the proposed algorithm needs the 264, 310, 410, 575 chosen cipher texts to find the message without key for different key size.

As shown in Figure 3. The length of cipher texts is proportional to that of the corresponding plaintexts lengths with varying key length. However, this method requires less storage space than that of the plaintext, thus, it is more efficient in the storage capacity. Another reflection is that, the size of the random key length increase as the size of the plaintext increase, which greatly reduces size of the key length. Moreover, key as well cipher text can be transmitted much faster through the secure channel and public channel respectively. Therefore, the method is also more efficient items of storage and transmission.

As shown in Figure 4. The adversary requires more than 65% of chosen cipher texts for the corresponding plaintexts to recover 78% of the random key length. Hence, it requires more chosen cipher text to retrieve the key. The figure also shows that different tests are performed to experiment the robustness of this proposed method. Therefore, it is more efficient and effective method.

Figure 5 indicates, for the same plaintext length, it generates different cipher text, namely cipher text-1 and cipher text-2 with different random key. Thus, this method satisfies the Message Indistinguishability (MI) because the probability of guessing these two cipher text is more than half of the random probability of guessing the right message.

Figure 6 shows that the adversary requires more chosen cipher text for a given plaintext, which takes more than half of the time to retrieve the key. Therefore, PPT algorithm satisfies Message Indistinguishability (MI), according to the definition.

9 Conclusion

In this paper, we addressed a biotic DNA based secret key cryptographic mechanism, which is based upon the genetic information of biological system. Moreover, this cryptographic prototype is motivated from bio-molecular computation, which is rapidly growing field that has made great strides of ultra-compact information storage, vast parallelism, and exceptional energy efficiency. Over the last two decades, Internet technology is growing much faster, which permits the users to access the intellectual property that is being transferred over the internet can be easily acquired and is vulnerable to many security attacks. Hence, network security is looking for unbreakable encryption technology to protect the data. This motivated us to propose biotic pseudo DNA cryptography method, which makes use of splicing system to improve security and random multiple key sequence to increase the degree of diffusion and confusion that makes resulting cipher texts difficult to decipher and to realize a secure system. Furthermore, Moreover, we also modelled Hybrid DNA cryptosystem that make use of proposed work by assembling DNA based public key cryptography for effective storage of public key as well as double blinded encryption scheme for a given message. The formal and experimental analysis not only shows that, this method is powerful against chosen cipher text attacks, but also very effective and efficient in storage, computation as well as transmission; To conclude, DNA cryptography is an new emerge area and extremely guaranteeing field, where research is possible in incredible development and improvement.

References

[1] L. Adleman, *On Constructing a Molecular Computer*, University of California, U.S.C draft, Jan. 1995.

[2] L. M. Adleman, "Molecular computation of solutions to combinatorial problems," *Science*, vol. 266, no. 5187, pp. 1021–1024, 1994.

[3] L. M. Adleman, P. W. K. Rothemund, S. Roweis and E. Winfree, "On applying molecular computation to the data encryption standard," in *Proceedings of the Second DIMACS Workshop*, pp. 31–44, 1999.

[4] S. T. Amin, M. Saeb, S. El-Gindi, "A DNA-based implementation of YAEA encryption algorithm," in *Proceedings of the Second IASTED International Conference on Computational Intelligence*, pp. 32–36, 2006.

[5] B. Anam and W. Yorkshire, "Review on the Advancements of DNA cryptography," in *4th International Conference on Software, Knowledge, Information Management and Applications (SKIMA'10)*, Aug. 2010.

[6] E. S. Babu, "An implementation and performance evaluation study of AODV, MAODV, RAODV in mobile Ad hoc networks," *International Journal of Scientific & Engineering Research*, vol. 4, no. 9, pp. 691–695, 2013.

[7] E. S. Babu, C. Nagaraju, and MHM. K. Prasad, "An implementation and performance evaluation of passive DoS attack on AODV routing protocol in mobile Ad hoc networks," *International Journal of Emerging Trends & Technology in Computer Science*, vol. 2, no. 4, pp. 124–129, 2013.

[8] E. S. Babu and MHM K. Prasad, "An implementation analysis and evaluation study of DSR with inactive DoS attack in mobile Ad hoc networks," in *International Journal of Engineering Innovations and Research*, vol. 2, no. 6, pp. 501–507, 2013.

[9] D. Beaver, "Factoring: The DNA solution," in *4th International Conferences on the Theory and Applications of Cryptology*, pp. 419–423, 1994.

[10] M. Borda and O. Tornea, "DNA secret writing techniques," in *8th International Conference on Communications (COMM'10)* pp. 451–456, 2010.

[11] Y. Brun, "Nondeterministic polynomial time factoring in the tile assembly model," *Theoritical Computer Science, Science Direct*, vol. 395, no. 1, pp. 3–23, Apr. 2008.

[12] J. Chen, "A DNA-based, biomolecular cryptography design," in *IEEE International Symposium on Circuits and Systems (ISCAS'03)*, pp. 822–825, 2003.

[13] G. Cui, L. Cuiling, L. Haobin, and L. Xiaoguang, "DNA computing and its application to information security field," in *IEEE Fifth International Conference on Natural Computation*, pp. 43–47, Aug. 2009.

[14] G. Cui, Y. Liu, and X. Zhang, "New direction of data storage: DNA molecular storage technology," *Computer Engineering and Application*, vol. 42, no. 26, pp. 29–32, 2006.

[15] G. Cui, L. Qin, Y. Wang, and X. Zhang, "An encryption scheme using DNA technology," in *IEEE 3rd International Conference on Bio-Inspired Computing: Theories and Applications (BICTA'08)*, pp. 37–42, 2008.

[16] A. Desai, *Secure Against Chosen-Ciphertext Attack Department of Computer Science and Engineering*, University of California at San Diego, USA, 2000.

[17] E. Fujisaki and T. Okamoto, "Secure Integration of asymmetric and symmetric encryption schemes," in *Advances in Cryptology (CRYPTO'99)*, LNCS 1666, pp. 537–554, Springer, 1999.

[18] A. Gehani, T. H. LaBean, and J. H. Reif, "DNA-based cryptography," in *Proceedings 5th DIMACS Workshop on DNA Based Computers*, pp. 233–249, 1999.

[19] T. Head, "Splicing schemes and DNA," in *Lindenmayer Systems; Impact on Theoretical Computer-science and Developmental Biology*, pp. 371–383, 1992.

[20] T. Head, "Formal language theory and DNA: an analysis of the generative capacity of specific recombinant behaviors," *Bulletin of Mathematical Biology*, vol. 49, no. 6, pp. 737–759, 1987.

[21] M. Hirabayashi, A. Nishikawa, F. Tanaka, M. Hagiya, H. Kojima, K. Oiwa, "Analysis on Secure and Effective Applications of a DNA-Based Cryptosystem," in *Sixth International Conference on Bio-Inspired Computing: Theories and Applications*, pp. 205–210, 2011.

[22] T. Kazuo, O. Akimitsu, S. Isao, "Public-key system using DNA as a oneway function for key distribution," *Biosystems*, vol. 81, no. 1, pp. 25–29, 2005.

[23] A. Leier, C. Richter, W. Banzhaf, H. Rauhe, "Cryptography with DNA binary strands," *Biosystems*, vol. 5. no. 7, pp. 13–22, 2000.

[24] K. Li, S. Zou, and J. Xu, "Fast parallel molecular algorithms for DNA based computation: Solving the elliptic curve discrete logarithm problem over GF(2n)," in *Frontiers in the Convergence of Bioscience and Information Technologies (FBIT'07)*, pp. 749–752, 2007.

[25] H. Lodish, A. Berk, P. Matsudaira, et al., *Molecular Cell Biology, 5th Ed.*, Chap. 4, pp. 101–145, 2006.

[26] MX Lu, XJ Lai, GZ Xiao, L Qin, "Symmetric-key cryptosystem with DNA technology," *Science in China Series F: Information Sciences*, vol. 50, no. 3, pp. 324–333, 2007.

[27] A. Menezes, P. Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1996.

[28] K. Ning, *A Pseudo DNA Cryptography Method*, 16 Mar 2009. (http://arxiv.org/abs/0903.2693)

[29] G. Paun, G. Rozenberg and A. Salomaa, *DNA Computing: New Computing Paradigms*, Springer-Verlag, 1998.

[30] D. Pixton, "Regularity of splicing languages," *Discrete Applied Mathematics*, vol. 69, no. 12, pp. 101–124, 1996.

[31] P. Rakheja, "Integrating DNA computing in international data encryption algorithm," *International Journal of Computer Applications*, vol. 26, no. 3, pp. 1–6, 2011.

[32] J. H. Reif, "Parallel molecular computations: Models and simulations," in *Seventh ACM Symposium on Parallel Algorithms and Architecture*, pp. 217–236, 1995.

[33] C. E. Shannon, "Communication theory of secrecy systems," *Bell System Technical Journal*, vol. 28, no. 4, pp. 656—715, 1949.

[34] B. Shimanovsky, J. Feng, and M. Potkonjak, "Hiding data in DNA," in *5th International Workshop on Information Hiding (IH'02)*, pp. N373–386, 2002.

[35] C. T. Taylor, V. Risca, and C. Bancroft, "Hiding messages in DNA microdots," *Nature*, 399, pp. 533–534, 1999.

[36] O. Tornea and M. E. Borda, "DNA cryptographic algorithms," in *International Conference on Advancements of Medicine and Health Care Through Technology*, pp. 223–226, 2009.

[37] J. Watada, R. binti Abu Bakar, "DNA computing and its applications," in *Eighth International Conference on Intelligent Systems Design and Applications (ISDA'08)*, pp. 288–294, 2008.

[38] M. Yarus, "RNA-ligand chemistry: A testable source for the genetic code," *RNA*, vol. 6, pp. 475–487, 2000.

[39] Z. Yunpeng, W. Zhong, and R. O. Sinnott, "Index-based symmetric DNA encryption algorithm," in *4th International Congress on Image Signal Process*, pp. 2290–2294, Oct. 2011.

[40] M. Zhang, L. Sabharwal, and W. Tao, "Interactive DNA sequence and structure design for DNA nanoapplications," *IEEE Transactions on Nanobioscience*, vol. 3, no. 4, pp. 286–292, Dec. 2004.

E. Suresh Babu received his B.Tech degree in Computer Science Engineering from RGM College of Engineering, Nandyal, M.Tech degree in Computer Science from V.T.University Belgaum and pursuing PhD in Computer Science and Engineering from J.N.T.University Kakinada. Currently, he is working as an Associate Professor in the Department of CSE in K L University Vijayawada; He has 12 years of teaching experience. He has published 12 research papers in various International Journal and 10 research papers in various National and International Conferences. He has attended 32 seminars and workshops. His areas of interests are Wireless Networks, Network Security, and MANETs, Cogntive Radio Networks, Software Radio Networks.

C. Naga Raju is currently working as Associate Professor and Head of the Department of Computer Science and Engineering at YSR Engineering College of Yogivemana University, Poddatur, Kadapa District, and Andhra Pradesh, India. He received his B.Tech Degree in Computer Science from J.N.T.University, Anantapur, and M.Tech Degree in Computer Science from J.N.T.University Hyderabad and PhD in digital Image processing from J.N.T.University Hyderabad. He has got 18 years of teaching experience. He received research excellence award, teaching excellence award and Rayalaseemavidhyaratna award for his credit. He wrote text book on and Data structures. He has six PhD scholars. He has published fifty three research papers in various National and International Journals and about thirty research papers in various National and International Conferences. He has attended twenty seminars and workshops. He is member of various professional societies like IEEE, ISTE and CSI.

Munaga HM Krishna Prasad is currently an Associate Professor of the Department of Computer Science and Engineering, University College of Engineering, Kakinada (Autonomous), JNTUK, Andhra Pradesh. He did his B.E. from Osmania University, Hyderabad, M.Tech. and Ph.D. Computer Science and Engineering from JNTU, Hyderabad.He successfully completed a two year MIUR fellowship at University of Udine, Udine, Italy. He has about 50+ research papers in various International Journals and Conferences, and attended many national and international conferences in India and abroad. He is a member of Association for Computing Machinery (ACM), ISTE and IAENG (Germany) is an active member of the board of reviewers in various International Journals and Conferences. His research interests include data mining, Big Data Analytics and High Performance Computing