IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 11, Issue 2 (May. - Jun. 2013), PP 64-71
www.iosrjournals.org

Simulation of Quantum Cryptography and use of DNA based algorithm for Secure Communication

Sharvari Gogte[1], Trupti Nemade[2], Shweta Pawar[3], Prajakta Nalawade[4]
1-4(B. Tech, Computer Technology Department, Veermata Jijabai Technological Institute (V.J.T.I.)/ Mumbai University, India)

Abstract: *Quantum Cryptography (QC) is an emerging security technique in which two parties communicate via a quantum channel. The fundamentals of quantum cryptography are No-cloning theorem and Heisenberg's uncertainty principle. This research paper proposes a simulation of quantum key exchange and authentication followed by an implementation of DNA based algorithm for secure message exchange.*
Keywords: *BB84 protocol, DNA based algorithm, Quantum Cryptography (QC), Secure key exchange by simulation*

I. Introduction

Traditional cryptosystem has been in use for many years. It includes public as well as secret key cryptography. With the advent of supercomputers, even the computationally intensive algorithms like RSA, triple DES, AES, RC4, etc. are easily breakable. In secret key cryptography, secure key exchange through public channel is difficult. One time pad technique which was widely used and was proved to provide immense security fails to generate unique random keys after certain period of time. These shortcomings of classical cryptography led to the development of QC. ([1], [2])

QC is based on laws of physics namely No-cloning theorem and Heisenberg's uncertainty principle. ([2], [3])

No-cloning theorem:

It states that an exact copy of a quantum sized particle cannot be made. An attempt of reproducing the quantum particle would not result in exact copy of its true state. Due to this property of photons (a quantum sized particle), replay attack is prevented. An attempt to reproduce the current key will be detected, resulting in termination of communication.

Heisenberg's Uncertainty principle:

It states that at a given time, measuring the velocity and the location of a quantum sized particle is impossible. Hence, an eavesdropper cannot measure the stream of transmitted photons accurately. This principle also helps communicating parties to determine whether the stream of photons has been modified or not.

Based on these principles, the quantum channel is resistant to attacks like replay attack, man-in-the–middle attack, sniffing attack etc.

In order to implement QC, we require fiber optic cables for transmission, photon generator and photon polarizer to generate polarized stream of photons. Since the hardware requirements are very expensive, the real life implementation of QC is limited. Hence we propose the simulation of QC. ([4])

We have used QC for authentication and secure key exchange followed by message encryption using DNA based algorithm. This algorithm is based on the central dogma of molecular biology which describes the flow of generic information within a biological system. Dogma provides a framework for understanding transfer of sequence information in living organisms containing sequence information carrying biopolymers such as DNA, RNA and Protein. As per the normal flow of biological information, DNA can be copied to DNA by DNA replication. Information can be copied into mRNA by transcription and proteins can be synthesized using the information in mRNA as a template by translation. ([5], [6])

Transcription:

It's the process that creates a new RNA piece by transferring information from a section of DNA strand. A DNA segment is read, non-coding areas are removed, and the remaining ones are re-joined and transcribed into a single strand of RNA.

Translation:

The RNA sequence is translated first, into a sequence of amino acids, then into a protein, according to the genetic code table. This central dogma framework is simulated in DNA module of our encryption algorithm. (Fig. 1)

II. Proposed System

We have proposed a secure message transfer protocol which consists of the following modules.

2.1. BB84 Protocol

This protocol was invented by Charles Bennett and Gilles Brassard in 1984.The description of protocol is as follows: (Fig. 2) ([7] - [12]) (A and B are communicating parties)

1. A sends a stream of polarized photons to B. The photons can be polarized in rectilinear (0° or 90°) or diagonal direction (45° or 135°). A stream of photons is string of zeros and ones with direction specified for each bit.
2. B measures this stream of photons with the help of randomly selected basis i.e. rectilinear or diagonal, and records the results.
   Note: It is not necessary that A and B will use same basis for measurement.
3. B informs A, his basis used for measurement of photons, through public channel.
4. A then compares the received basis with the actual basis and informs B over the public channel, the correct bits.
   Note: The correct bits are those whose basis are same.
5. Then A and B discard the incorrect bits and the correct ones are considered as a key.

2.2. Authentication

The authentication protocol makes use of classical as well as quantum cryptographic protocols. Since our main focus is on quantum cryptography, we have assumed that the identities of the communicating parties have already been established with a Third Party authentication server via classical cryptographic protocols through public channel. The procedure stated below describes how A and B authenticate to each other. (Fig. 3) ([13])

1. Initially, when A wishes to communicate with B, A sends its own identity and the identity of B (the identity of the party with which the communication is intended) to authentication server via public channel.
2. After verifying the identities of A and B, the authentication server (AS) generates a random stream of basis and sends it to both the parties.
3. The AS then generates a random stream of photons, sends it to A and its complement to B.
4. A then sends a subset of received photons to B, and B checks whether he has complement of the bits or not.
5. After this successful verification, B repeats the same procedure but sends a disjoint stream of photons. Thus, A and B have authenticated themselves to each other.

2.3. Secure Key Exchange

We have implemented the secure key exchange module using the BB84 protocol ([7] – [12]) simulation. All the steps of the protocol are followed as-is. The only difference is in the method by which random bits and basis are generated.

A random stream of bits (zeroes and ones) which represents the stream of photons is generated as follows. To get n random bits, the function 'random' (which returns a random value between 0 and 1) is called n times. For a particular call, if the 'random' function returns a value less than or equal to 0.5, then that particular bit is considered '0' else it is considered '1'.

The method for generating the corresponding basis is similar. The only difference is in the last step. If the 'random' function returns a value less than or equal to 0.5, then that particular basis is considered '+' else it is considered 'x'.

2.4. DNA based algorithm

The DNA based algorithm is symmetric block cipher. The inputs to this algorithm are a 128 bit key and plain text divided into blocks of 128 bits. ([6])

2.4.1. Encryption

The encryption procedure is as follows:

1. The user gives input in the form of a string which is then converted to its hexadecimal equivalent. If the size of this hexadecimal equivalent is less than 32 (128 bits /4) hexadecimal digits, then a padding of zeros is added.
2. Now these hexadecimal digits are transferred into 4X4 byte matrix M.
3. The key exchanged in the 'secure key exchange' step is used to generate further 10 sub-keys (K_0-K_9) with the help of AES sub-keys generator.

4. XOR operation is performed between matrix M and first sub-key $K_{0.}$
5. Then the plain text undergoes 8 rounds of transformation based on the DNA module.
6. The result obtained is then XORed with the last key (K_9) and thus we get the encrypted text.

Pseudo-code for encryption:

```
Encryption (Block b (block of the plain text), key k):
  Begin
    Transform b into a 4x4 matrix M
    Generate 10 sub-keys (K0-K9) from the principal key K
    M = M XOR K0
    For i=1 to 8 do:
      M = DNA (M, Ki)
    End For
    M = M XOR K9
  End
```

2.4.2. DNA Module

The DNA module treats every line of matrix M and the round keys as a DNA strand (A for 00, C for 01, G for 10 and T for11). Then the following three steps are applied:

1. Transcription:

This process simulates the transcription process of the central dogma. This is mono-alphabetic substitution. A DNA strand is converted to RNA strand by changing A to T, C to G, G to C and T to A.

2. BIO_XOR:

Given a Matrix M and a sub-key K_i , we compute M BIO_XOR K_i using BIO_XOR (TABLE 1)

3. Translation:

This process simulates the translation of the central dogma which is also a substitution process performed using amino acid table (TABLE 2). The value in each block of matrix is divided into left and right part which is then used to look up the amino acid table for substitution.

2.4.3. Decryption

The steps for decryption are reverse of the encryption algorithm.

1. Initially, the encrypted matrix is XORed with the last key.
2. Then the matrix undergoes eight rounds of DNA module with the round keys.
3. The matrix so obtained is XORed with the first key to get the block of plaintext.
4. This block is read row wise to get back the original text.

The transcription and the BIO-XOR steps of the DNA module remain the same. For the translation process, an inverse of the amino acids translation table is made and then the lookup takes place.

For creating the inverse table, we traverse the original table row wise. The value obtained by reading the entry is divided into left and right half which transforms into the row and column index of the inverse matrix respectively. At that position, the concatenation of the row and column indices of the entry is inserted. This procedure is repeated for all the entries of the table. Thus we get the inverse amino acids table.

For example, if aminoAcid[AA,AT] = TAAT, then inverseAminoAcid[TA,AT] = AAAT

Pseudo-code for decryption:

```
Decryption (Block b (block of the ciphered text), key k):
  Begin
    Transform b into a 4x4 matrix M
    Generate 10 sub-keys (K0-K9) from the principal key K
    M = M XOR K9
    For i=1 to 8 do:
      M = Inversed_DNA (M, Ki)
    End For
    M = M XOR K0
  End
```

III. Methods and Results

The following are the screenshots of our implemented system along with the space and time complexity.

Time and space complexity:

The space complexity of our system is n^3 as we need to store ten sub-keys used for encryption-decryption in a 3-dimensional array. (An array of 10 keys which are in a matrix form) The time complexity of our system is n^3 since the encryption and decryption operations are applied on three dimensional arrays.

According to the System calculated time, the total time required for authentication, key-exchange and encryption-decryption is approximately 10 seconds. (Fig. 4-7) ([14])

IV. Figures and Tables

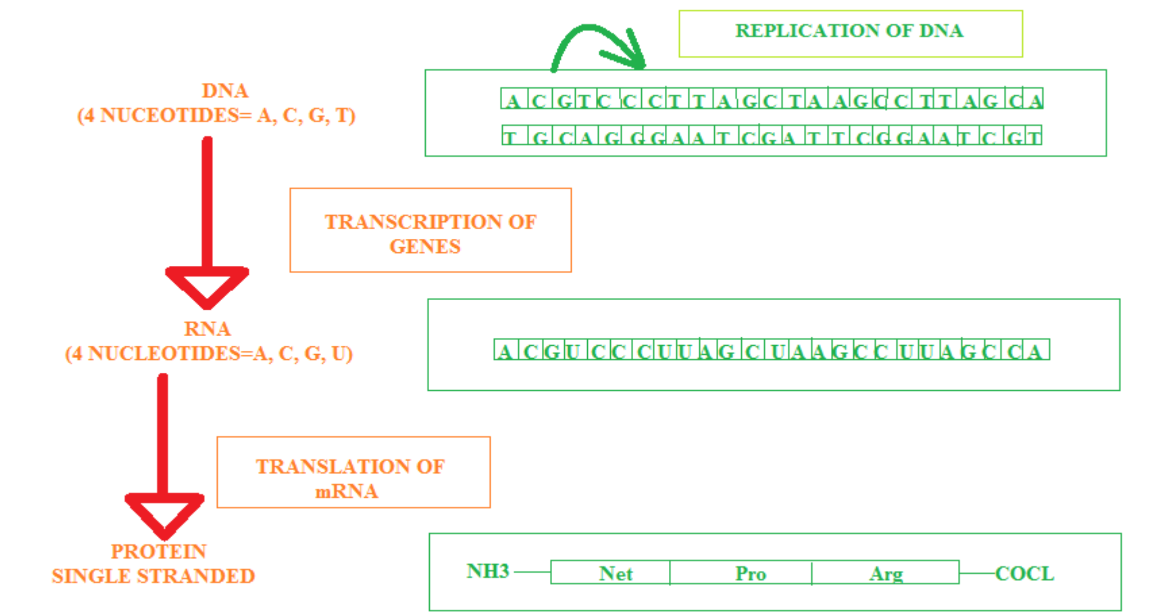

Fig. 1: central dogma of molecular biology ([5])

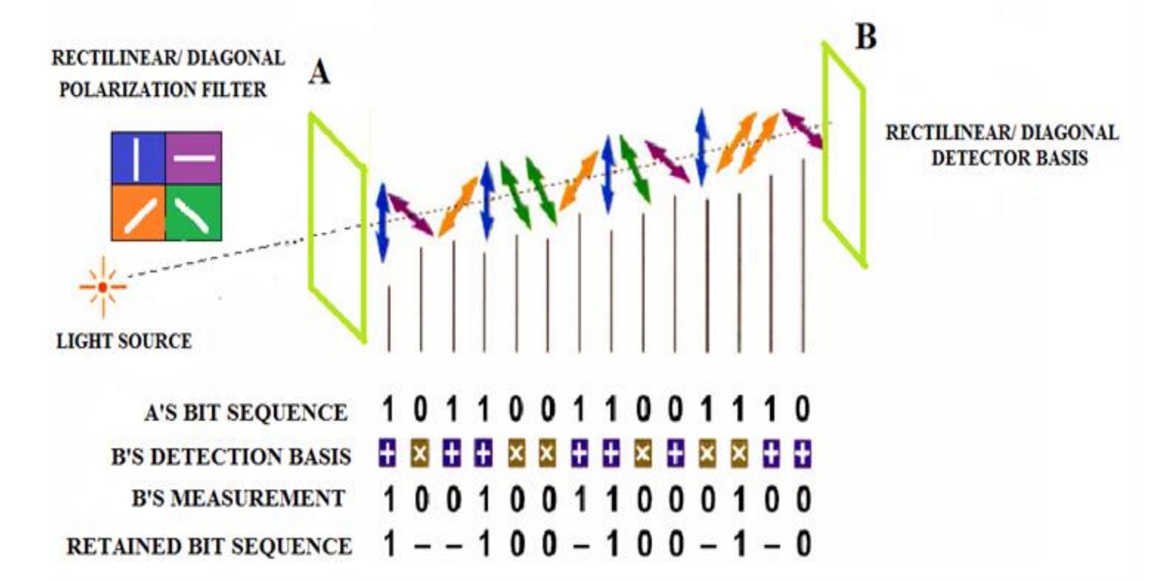

Fig. 2: BB84 protocol ([7] – [12])

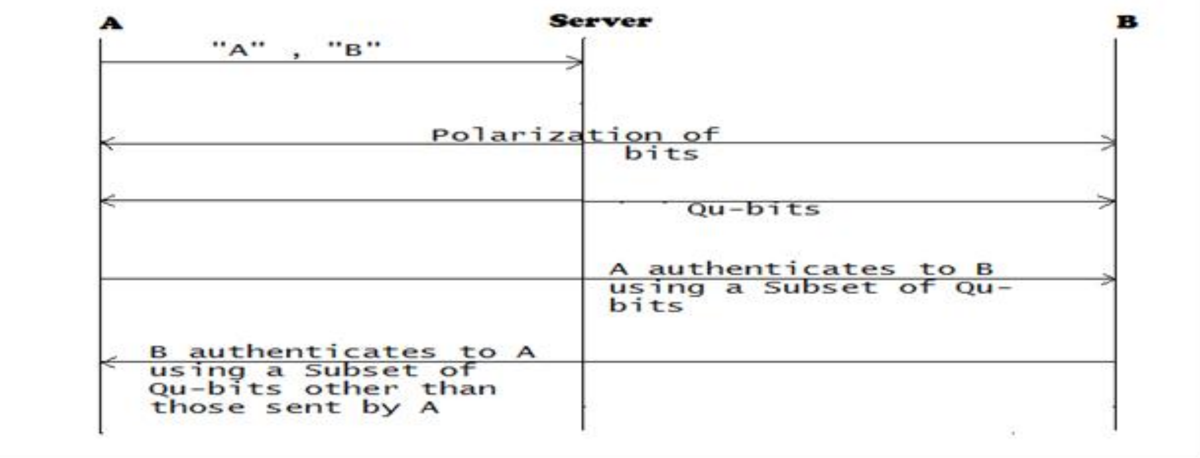

Fig. 3: authentication protocol ([13])

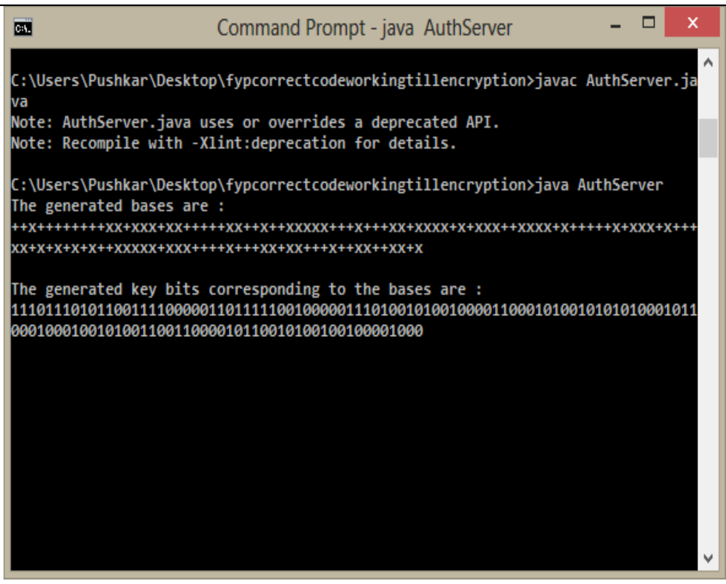

Fig. 4: authentication server

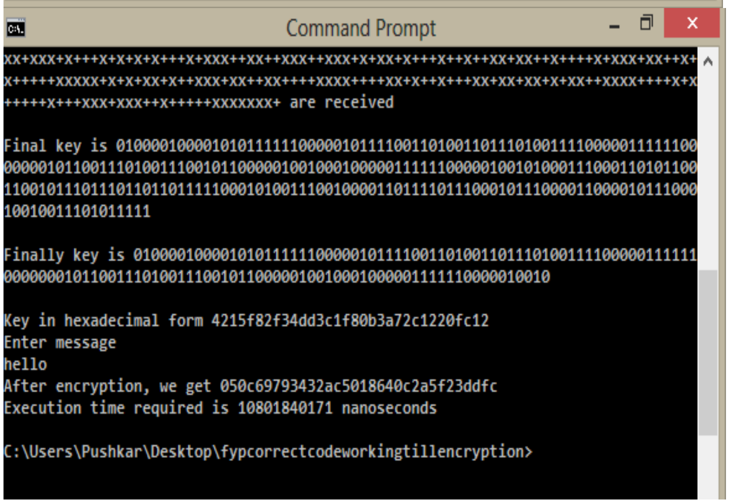

Fig. 5 and 6: communicating party A

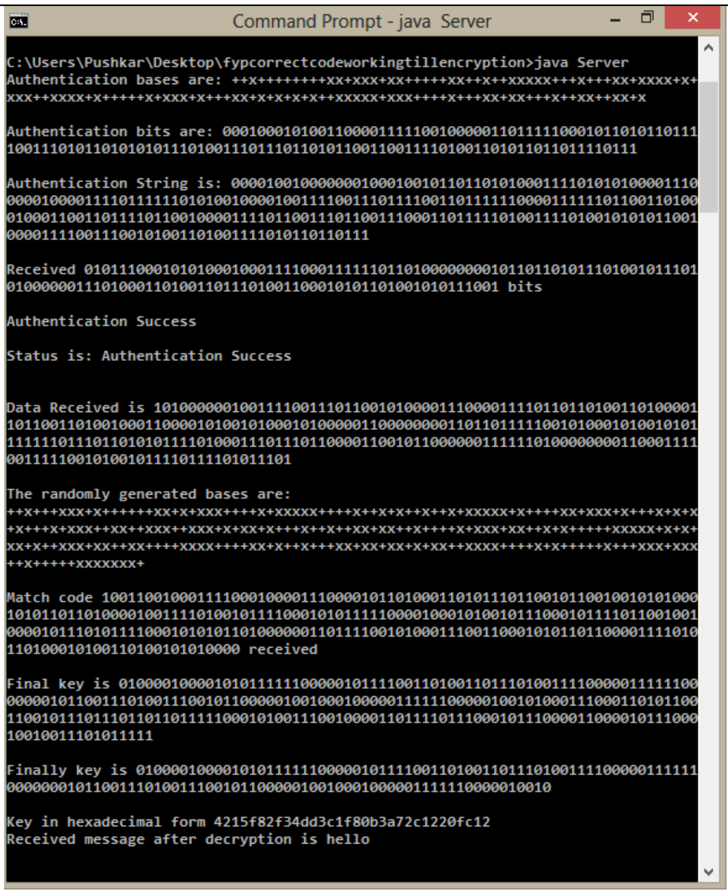

Fig. 7: communicating party B

TABLE 1: BIO_XOR ([6])

BIO_XOR	A	C	G	T
A	T	G	C	A
C	G	T	A	C
G	C	A	T	G
T	A	C	G	T

TABLE 2: Amino Acids ([6])

	AA	AC	AG	AT	CA	CC	CG	CT	GA	GC	GG	GT	TA	TC	TG	TT
AA	TAAA	TAAC	TAAG	TAAT	TACA	TACC	TACG	TACT	TAGA	TAGC	TAGG	TAGT	TATA	TATC	TATG	TATT
AC	TCAA	TCAC	TCAG	TCAT	TCCA	TCCC	TCCG	TCCT	TCGA	TCGC	TCGG	TCGT	TCTA	TCTC	TCTG	TCTT
AG	TGAA	TGAC	TGAG	TGAT	TGCA	TGCC	TGCG	TGCT	TGGA	TGGC	TGGG	TGGT	TGTA	TGTC	TGTG	TGTT
AT	TTAA	TTAC	TTAG	TTAT	TTCA	TTCC	TTCG	TTCT	TTGA	TTGC	TTGG	TTGT	TTTA	TTTC	TTTG	TTTT
CA	GAAA	GAAC	GAAG	GAAT	GACA	GACC	GACG	GACT	GAGA	GAGC	GAGG	GAGT	GATA	GATC	GATG	GATT
CC	GCAA	GCAC	GAAG	GCAT	GCCA	GCCC	GCCG	GCCT	GCGA	GCGC	GCGG	GCGT	GCTT	GCTC	GCTG	GCTT
CG	GGAA	GGAC	GGAG	GGAT	GGCA	GGCC	GGCG	GGCT	GGGA	GGGC	GGGG	GGGT	GGTA	GGTC	GGTG	GGTT
CT	GTAA	GTAC	GTAG	GTAT	GTCA	GTCC	GTCG	GTCT	GTGA	GTGC	GTGG	GTGT	GTTA	GTTA	GTTG	GTTT
GA	CAAA	CAAC	CAAG	CAAT	CACA	CACC	CACG	CACT	CAGA	CAGC	CAGG	CAGT	CATA	CATC	CATG	CATT
GC	CCAA	CCAC	CCAG	CCAT	CCCA	CCCC	CCCG	CCCT	CCGA	CCGC	CCGG	CCGT	CCTA	CCTC	CCTG	CCTT
GG	CGAA	CGAC	CGAGA	CGAT	CGCA	CGCC	CGCG	CGCT	CGGA	CGGC	CGGG	CGGT	CGTA	CGTC	CGTG	CGTT
GT	CTAA	CTAC	CTAG	CTAT	CTCA	CTCC	CTCG	CTCT	CTGA	CTGC	CTGG	CTGT	CTTA	CTTC	CTTG	CTTT
TA	AAAA	AAAC	AAAG	AAAT	AACA	AACC	AACG	AACT	AAGA	AAGC	AAGG	AAGT	AATT	AATA	AATC	AATG
TC	ACAA	ACAC	ACAG	ACAT	ACCA	ACCC	ACCG	ACCT	ACGA	ACGG	ACGC	ACGT	ACTA	ACTC	ACTG	ACTT
TG	AGAA	AGAC	AGAG	AGAT	AGCA	AGCC	AGCG	AGCT	AGGA	AGGC	AGGG	AGGT	AGTA	AGTC	AGTG	AGTT
TT	ATAA	ATAC	ATAG	ATAT	ATCA	ATCC	ATCG	ATCT	ATGA	ATGC	ATGG	ATGT	ATTA	ATTC	ATTG	ATTT

V. Conclusion

Our system can guard against man in the middle attack, eavesdropping, spoofing, packet sniffing and replay attack. This is because the fundamentals of QC are based on the laws of physics; thus the eavesdropper would be unable to generate the exact replica of the photons exchanged between the communicating parties as well as record the stream of photons exchanged between the parties. These factors increase the security of our system. ([1], [2])

Our further research includes:

1. Overcoming distance limitations
2. Protection against denial of service attack
3. To devise cost effective methods for generation, polarization, transmission of photons and retaining their polarizations over long distances

Acknowledgements

We would like to thank our project mentor Prof. Seema Shrawne for guiding us throughout our project. Her suggestions have helped us in effective planning of our work and have added value to it. We appreciate the time that she invested in our project in spite of her busy schedule. We are grateful for her constant encouragement and co-operation which was instrumental in driving our project work.

References

[1] Karen Hunter, QuantumCryptography, *SCI 510, Quantum Todd Duncan*

[2] Sheila Cobourne , *Quantum Key Distribution Protocols and Applications*, Royal Holloway, University of London, Egham, Surrey TW20 0EX, England

[3] Kelly Koenig, *Bleeding edge: Nano-technological application in Quantum Cryptography*, CSC grants, December 2008

[4] Serge Massar, Fiber Optics Protocols for Quantum Communication, *Quantum Information Processing, Vol. 5, No. 6, December 2006*

[5] Wikipedia, http://en.wikipedia.org/wiki/Central_dogma_of_molecular_biology

[6] Souhila Sadeg , Mohamed Gougache, Nabil Mansouri, Habiba Drias, An encryption algorithm inspired from DNA,*IEEE , 978-1-4244-8611-3/10, 2010*

[7] Charles H. Bennett, Gilles Brassard, Quantum cryptography: public key distribution and coin tossing, *International Conference on Computers, Systems and Signal Processing, Banglore, India, December 10-12, 1984*

[8] Mehrdad S. Sharbaf, Quantum Cryptography: A New Generation of Information Technology Security System, *2009 Sixth International Conference on Information Technology: New Generations*

[9] Mehrdad S. Sharbaf, Quantum Cryptography: An Emerging Technology in Network Security, *IEEE, 978-1-4577-1376-7/11, 2011*

[10] Marcin Sobota, Adrian Kapczy_ski, Arkadiusz Banasik, Application of Quantum Cryptography protocols in Authentication process, *The 6th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, 15-17 September 2011, Prague, Czech Republic*

[11] Yoshito Kanamori, Seong-Moo Yoo, Don A. Gregory, Frederick T. Sheldon, On Quantum Authentication Protocols, *IEEE GLOBECOM 2005 proceedings*

[12] http://swissquantum.idquantique.com/?Key-Sifting

[13] D. Richard Kuhn, A quantum cryptographic protocol with detection of compromised server, *Quantum Information and Computation, Vol. 5, No.7 (2005) 551-560*

[14] Stack Overflow, http://stackoverflow.com/questions/6789385/calculation-of-execution-time