DOI: http://dx.doi.org/10.26483/ijarcs.v9i2.5673

ISSN No. 0976-5697

Volume 9, No. 2, March-April 2018

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

QUANTUM KEY DISTRIBUTION SCHEME: AN IMPROVEMENT BASED ON BB84 PROTOCOL

Kausik Saha, Sirshendu Sekhar Ghosh and Dilip Kumar Shaw
Department of Computer Applications, National Institute of Technology Jamshedpur
Jamshedpur-831014, Jharkhand, India

Abstract: Cryptography is being introduced to secure data. Through Classical Cryptographic technique, we can encrypt our data and secure secret key also. But classical cryptography is based on mathematical techniques where reversing one-way function is possible with sufficient computing power and time. Thus, Quantum Cryptography is introduced to overcome those drawbacks of traditional classical cryptography. It is based on the values of quantum mechanics and photon. Scientists and researchers have brought up a new key distribution protocol based on Quantum Cryptography is called Quantum Key Distribution (QKD) Protocol. The first QKD protocol BB84 was introduced by Charles Bennett and Gilles Brassard in 1984. After that several QKD protocols have been created. In this paper we have improved BB84 Protocol by removing the drawbacks of the existing algorithm towards disclosing large portion of secret key or Eaves dropping which may not be detected. Steps of the existing BB84 algorithm along with the modified one is described in detail with example and pseudo code. Limitations of BB84 Protocol and comparative study of both the existing and the improved one based on their security has been made and described in this current work.

Keywords: Quantum Cryptography, Quantum Key Distribution, QKD Protocols, BB84 Protocol, Improved BB84 Protocol.

I. INTRODUCTION

Internet has become a means of communication globally. Day by day amount of data transferred through internet is increasing. We should secure the data transferred through internet along with its transmission channel as per our requirements. Thus, cryptography is being introduced to perform these tasks.

Cryptography is a system to turn normal record into an unreadable form so that it becomes unintelligible to any unauthorized person. The three pillars of cryptography are Confidentiality, Integrity and Availability (CIA) [1].

A. Classical Cryptography

Classical Cryptography and Quantum Cryptography are mainly the two types of Cryptographic Technique. Classical Cryptography is of two types: Symmetric Key Cryptography and Asymmetric Key Cryptography. The base of Classical Cryptography is mathematics. It uses large integer or prime factor to make the computation difficult to the intruder[2].Classical Cryptography generally creates one-way functions with the help of mathematical techniques to make the things complex to the unauthorized person. But with the help of sufficient computing power and time, the one-way function can be reversed which makes Classical Cryptography less secure.

B. Quantum Cryptography

Quantum Cryptography is based on the values of quantum mechanics and on the concept of using light weight particles called photons and the value of classical bits encodes by the polarization of photon [3]. It is guaranteed by the law of physics that demonstrates by non-cloning theorem that supports unconditionally to secure the key and detects an eavesdropper while communicating over quantum channel [4]. It depends on two important components of quantum mechanics [5]: Heisenberg Uncertainty Principle and Photon Polarization.

C. Quantum Key Distribution

The idea of Quantum cryptography was first devised by Stephen Wiesner in 1970. Based on this a new key distribution protocol is introduced called Quantum Key Distribution (QKD) Protocol. QKD will be the most secure way to exchange secret key in the near future of the world of cryptography. It is a technology that provides ultimate security. QKD is used to generate and exchange secret key between at least two parties.

The first QKD protocol is BB84. After that several QKD protocols have been created sequentially. Among them E91, BB92, SARG04, KMB09, S09, COW12, S13, AK15 are most popular.

This paper is structured in the following manner: This section presents a brief idea about Classical and Quantum Cryptography. Section II deals with BB84 Protocol in detail. Section III provides pseudo code of BB84 Protocol with example. Section IV states the loopholes and limitations of BB84 Protocol. Section V describes improvement on BB84 Protocol. This paper concludes with Conclusion in section VI and References in section VII.

II. BB84 PROTOCOL

The BB84 Protocol was developed by Charles Bennett and Gilles Brassard in 1984 [6]. This protocol is the most basic QKD protocol of quantum cryptography and it can be implemented with basic setup available in the market. Most of all the QKD protocols are based on this protocol and much research and implementation has been done on this protocol. Photon polarization state is used by this protocol to transmit bits.

Steps of the BB84 protocol[7]to exchange the secret key has been described here. Alice and Bob both perform these steps as below:

A. Step-1 Communication over quantum channel

1) Alice generate a random string of (0,1) and prepares photons accordingly by polarizing them with either rectilinear (+) or diagonal (×)polarization basis as per the four polarization states (0º (0), 45º (0), 90º (1), 135º(1)) as shown in Table I.

Table I. Bit corresponding to basis and angle used in BB84

Basis	*Angle*	*Bit*	*Photon*
+	0º	0	
+	90º	1	
X	45º	0	
X	135º	1	

2) Alice now transmit those photons to Bob and stores the polarization of each photon.

3) By receiving a photon, Bob randomly chooses the polarization of each photon among rectilinear or diagonal basis. Then Bob measures that photon according to the basis he guessed and records resultant bit. Bob does not know that his measurements are correct or not i.e. he measured as the same basis of Alice or not.

4) At the end of this step, Bob will get a string of sequence (0,1).

5) Now they will start communicating over the public channel.

B. Step 2- Communication over a public channel

1) Phase 1. Raw Key Extraction:

a) Bob sends his recorded basis publicly to Alice i.e. the basis he used for each photon.

b) Alice compare those basis with him and tells Bob about the correct measurements made by him through the public channel.

c) Alice and Bob both then delete all the bits for which basis have not matched in both side.

d) After deleting all the mismatched bit, this will be the raw key. Now, if there is any Eavesdropping, then in both side the key will not be same. Otherwise this will be same in both side and will be their final key.

2) Phase 2. Error Estimation:

Now Alice and Bob take a small portion of their raw key and compare that to check the presence of Eavesdropper and find the error-rate.

- In this comparison, if Alice and Bob find no errors means no mismatch, then they will be sure that there was no eavesdropping.
- If at least one error or mismatch found, then there must be an eavesdropping. In this case, they start the entire process again by discarding all the bits.

At the successful end of this phase they discard those bits they used for comparison to generate the final key, as they disclosed those bits in public.

III. PSEUDO CODE OF BB84

The pseudo code [8-9]of BB84 protocol is written below. After this, one small example is taken for better understanding of this entire algorithm and to map it with the Pseudo Code.

A. Step 1- Communication over quantum channel

```
Sender:
generate string r randomly from (0,1)
FOR each bit from r
    pick randomly from ("R","D") resulting base b[i]
END FOR
FOR each bit from r
    generate a photon
    IF r[i]=0 and b[i]=R polarize the photon in state (0º)
    IF r[i]=1 and b[i]=R polarize the photon in state (90º)
    IF r[i]=0 and b[i]=D polarize the photon in state (45º)
    IF r[i]=1 and b[i]=D polarize the photon in state (135º)
    send q bit p[i] to Receiver
END FOR

Receiver:
FOR each q bit p'[i] received
    generate RANDOM ("R","D") results b'[i]
    measure q bit p'[i] in respect to base b'[i]
    result bit r'[i]
END FOR
```

B. Step 2- Communication over classical channel

1) Communication:

```
Receiver:
FOR each bit r'[i]
    send base b'[i] to Sender
END FOR
Sender:
FOR each bit r[i]
    send base b[i] to Receiver
END FOR
```

2) Raw Key Extraction:

```
Sender:
FOR each bit r[i]
    IF base b[i] ≠b'[i]
        eliminate bit r[i] from string r
    END IF
END FOR
Receiver:
FOR each bit r'[i]
    IF base b'[i] ≠b[i]
        eliminate bit r'[i] from string r'
    END IF
END FOR
```

3) Error Estimation:

Sender and Receiver:

FOR a subset of bites randomly chosen from string r
    IF r'[i] = r[i] and b[i] = b'[i]
        NO eavesdropping
        eliminate r'[i] and r[i]
    END IF
    ELSE
        Eaves dropper is present
        discard all bits and start over again
    END ELSE
END FOR

4) Final Key:

Remaining bits in string r and r' after successful execution of 'IF' block of Error Estimation is final key

C. Example

We have taken a small example below in Fig. 1 which maps the steps of the algorithm and pseudo code as well for better understanding of this BB84 protocol.

Communication and Quantum Transmission over Quantum Channel															
Alice	Alice's random bits	1	1	0	1	0	0	1	0	1	0	0	1	1	0	1
	Photons Alice sends	↕	↘	↗	↕	↔	↗	↘	↔	↕	↗	↗	↕	↘	↔	↘
	Random sending bases	R	D	D	R	R	D	D	R	R	D	D	R	D	R	D
Bob	Random receiving bases	R	R	D	R	D	R	D	D	R	R	D	D	D	R	R
	Bits as received	1		0		1	0	1		1	0	0	0		0	1
Communication and Raw Key Extraction over a public channel																
Bob	Reports bases of received bits	R		D		D	R	D		R	R	D	D		R	R
Alice	Alice says correct bases	OK		OK				OK		OK		OK			OK	
Presumably shared information (if no eavesdrop)		1		0				1		1		0			0	
Error estimation																
Bob	Reveals some key bits at random			0						1						
Alice	Confirms them			OK						OK						
Final Outcome																
Alice-Bob		1						1				0			0	

Figure 1. Example of BB84 Protocol

IV. Limitations of BB84 Protocol

BB84 Protocol has certain limitations. This protocol uses and can be applicable only with single photon sources. In photon preparation stage of this protocol, it is often established in weakening coherent states which gives a reasonable chance to Photon-Number-Splitting attack (PNS).The main loophole of this algorithm is present in the last step, The Error Estimation method, where eavesdropper cannot be detected if the number of comparing bits become small or the eavesdropper become so lucky also. If the number of bit compared in the method is small then the probability of detection of the Eavesdropper presence will be less. To provide more security, we may compare more number of bits. But it is also not a wise decision, as Eavesdropper can easily acquire a large portion of the secret key. It may break the security of the key as Eavesdropper may guess some portion of the key with the help of that acquired information. In the proposed algorithm, these limitations are removed so that secret key can be generated and exchanged in a safer way.

V. Proposed Algorithm

In this proposed algorithm, limitations have been removed by providing two phase securities. All the initial steps of this algorithm like starting from Communication Over Quantum Channel and Photon Distribution up to Error Estimation phase are same as BB84 Protocol. This new algorithm starts after the final key generation of BB84 Protocol. All the steps should be done in both sender and receiver side. Steps of the proposed algorithm are described below:

A. Steps of Proposed Algorithm

1) Step 1- Set a value for Block Division:

We treat the finally generated key of BB84 as a Tentative Final Key in this algorithm. A tentative final key is generated at the end of Error Estimation phase. This key should be divided into blocks. At first Alice and Bob both must have to agree with a number 'n'. Here 'n' defines number of bits in each block.

2) Step 2- Padding:

If the tentative final key is not completely divisible by 'n', then Padding is required for equally division of blocks. This padding is done starting by One (1) and then required number of Zeros (0) followed by that '1'. Now the updated key is divided into blocks with 'n' number of bits in each block.

3) Step 3- Invert the last bit of each block:

Now we will invert the last bit of each block. Inverting means, if the last bit is 0, then change that to 1 and if that is 1 then changed that to 0. This is done to create confusion to Eavesdropper. This will give the key one more level of security in the intermediate steps of key generation.

4) Step 4- XORing:

This is the most key step of this algorithm which will generate a new and totally different key from the tentative one. Now we will do XORing. We keep the first block remain same. Then XOR the next block with the first block and store the result. After that, XOR the next block with the result of the previous block and store the result. We will continue this process up to the last block. This process has been described in Fig. 2.

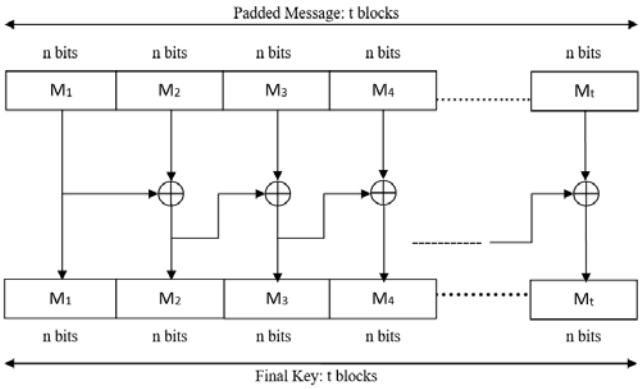

Figure 2. XORing Process

5) Step 5- Final Key generation:

Final key will be generated by sequentially put each block where the result of XORing is stored from the first one. This should be considered as a Final Key for this entire process.

B. Example

Here we are taking a short example to explain the new proposed algorithm. Suppose, tentative final key is generated at the end of Error Estimation step is:

1	0	0	1	1	1	1	0	0	0	1	0			

Let the value of 'n' is 5.Number of bit in the tentative final key is 12. So, padding is required for the last three bits. So, after padding it will be like this:

1	0	0	1	1	1	1	0	0	0	1	0	1	0	0

Now splitting this into block. We will get 3 blocks here:

1	0	0	1	1		1	1	0	0	0		1	0	1	0	0

Now in the next step we invert the last bit of each block. Here is the result after inverting:

1	0	0	1	0		1	1	0	0	1		1	0	1	0	1

In the next step XORing is performed. First block remains same and stored into the resultant array. And We XOR the next block with the first block:

1	0	0	1	0		1	1	0	1	1		1	1	1	0	1
						1	0	0	1	0						

Andstore the result:

1	0	0	1	0		0	1	0	0	1						

Next, we will XOR the third block with the XORing result of the second block:

1	0	0	1	0		1	1	0	1	1		1	1	1	0	1
												0	1	0	0	1

Andstore the result:

1	0	0	1	0		0	1	0	0	1		1	0	1	0	0

Thus, the finally generated secret key will be:

1	0	0	1	0	0	1	0	0	1	1	0	1	0	0

C. Pseudo Code of Modified BB84 Protocol

Starting from Communication over quantum channel upto Final Key generation after Error Estimation phase, all the steps are same as BB84 protocol. We treat the final key generated at the last step of BB84 algorithm as a Tentative Final key.

The entire process should be done in both sender and receiver side.

1) Step 1- Set a value for Block Division:

set 'n' as number of bit in each block
let length of tentative final key is 'i'

2) Step 2- Padding:

IF i completely divisible by n
    do nothing
END IF
ELSE
    remainder=i mod n
    set r[i+1] = '1'
    set all the remaining bit as '0'
END ELSE
divide entire key into equal block of n bits

3) Step 3- Invert the last bit of each block:

FOR block M_1 to M_t
    IF M_i[last bit]= '0'
        M_i [last bit] = '1'
    END IF
    ELSE
        M_i [last bit] = '0'
    END ELSE
END FOR

4) Step 4- XORing:

store M_1 block directly in final key string k
FOR block M_2 to M_t
    XOR M_i with M_{i-1}
    store block sequentially in string k
END FOR

5) Step 5- Final Key generation:

string k will be the Final Key

D. Advantage of this modified BB84 algorithm over the existing one:

The modification is done after the last step of BB84 i.e. after Error Estimation. This means it gives guarantees against Eavesdropping as in that step the presence of Eavesdropper has already been checked and proceeds further only after there is no Eavesdropping. But there is a chance of not detecting Eavesdropper if the number of bit compared is small. So, this proposed algorithm provides securities regarding this as it changes the entire key in the last step. Even if there is an Eavesdropper who may know some portion of the key due to the discloser of bits at the time of comparisons or during the public discussion, but at last the Eavesdropper cannot be able to know the key. Thus, the modified algorithm is also providing securities against Photon-Number-Splitting attack (PNS) as at last the entire key is being modified.

E. Flow chart

A Flow chart of both BB84 protocol and the proposed improvement is in Fig. 3 to understand the steps of both algorithm in a better way.

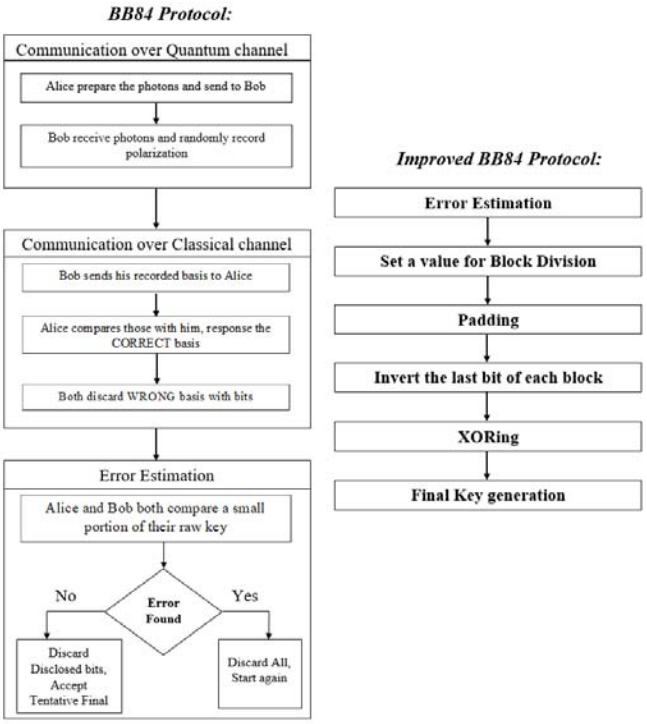

Figure 3. Flow chart of BB84 Protocol and the Improved One

VI. CONCLUSION

This paper gives a detailed analysis of BB84 protocol which is most basic and easy to implement QKD protocol. That is why we have improved this protocol. QKD is based on the quantum physics and can be thoroughly proven by generating secret keys. The modified BB84 Protocol which is described in this paper removes the limitations of the existing algorithms by adding an extra phase of security and helps to generate key securely. By this even if Eavesdropper's presence is not detected or it may acquire a good portion of secret key, then also the shared secret key will be secure enough to use in secret communication.

VII. REFERENCES

[1] W. Stallings, "Cryptography and Network Security: Principles and Practice", 6th ed., Pearson Education Inc.,2014, pp.9-12.

[2] Behrouz A Forouzan, Debdeep Mukhopadhyay, "Cryptography and Network Security", 3rd ed., Tata McGraw Hill Education(India) Pvt. Ltd., New Delhi, 2016.

[3] N. Kaur, "Enhancement of Network Security Techniques using Quantum Cryptography", International Journal on Computer Science and Engineering (IJCSE), 2011.

[4] Zhao Sheng-Mei, Li Fei, and Zheng Bao-yu, "A Proof of Security of Quantum Key Distribution in Probabilistic Clone Scheme", Communication Technology Proceedings (ICCT), vol. 2, 2003, pp. 1507-09.

[5] Miss. Payal P. Kilor, Mr. Pravin. D. Soni, "Quantum Cryptography: Realizing next generation information security", International Journal of Application or Innovation in Engineering & Management (IJAIEM), vol.3, no. 2, February 2014.

[6] C. H. Bennett and G. Brassard, "Quantum cryptography: public key distribution and coin tossing", Proceeding of the IEEE International Conference on Computers, Systems and Signal Processing, Bangalore, India, 1984, pp. 175-179.

[7] Hitesh Singh, D.L. Gupta, A.K Singh, "Quantum Key Distribution Protocols: A Review", IOSR Journal of Computer Engineering (IOSR-JCE), vol. 16, no. 2, ver. XI (Mar-Apr. 2014), pp. 01-09.

[8] C. H. Bennett, F. Bassett, G. Brassard, L. Salvia, and J. Smiling, "Experimental quantum cryptography", J. Cryptal. vol. 5, no. 1, 1992, pp. 3–28.

[9] Akash Shrivastava, Manvendra Singh, "A Security Enhancement Approach in Quantum Cryptography", IEEE 5thInternation Conference on Computers and devices for Communications(CODEC), 2012.