International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 08 | Aug 2020 www.irjet.net p-ISSN: 2395-0072

QKD Algorithm BB84 Protocol in Qiskit

M Ramachandra Kashyap

ABSTRACT: The privacy of our data is abolishing eventually. We are in search of high encryption standards for any private message. The standard encryption can be only made with the use of key. Most of the encryption techniques like AES, DES has encrypted keys which can be cracked using Quantum Computer. The Quantum Key Distribution (QKD) runs as a simulation in Qiskit library of Python. This simulates the pattern of qubits in IBM's Quantum computer. The qubits are the bits in quantum computer but they are not 0 or 1 instead they take the superposition of 0 and 1. This project also runs online in IBM Q experience website. The working of this project is, it generates an initial key which is transformed into qubits by sender and he applies QKD algorithm to generate the Senders public key using senders private key. The initial key is sent to receiver, with which he generates secret key. The Quantum Key thus generated can be used in Encryption techniques.

LITERATURE SURVEY:

The idea of Quantum computing is first given by Richard P. Feynman[1][2].One of the algorithms of Quantum Cryptography[3][4] is BB84 which is a quantum key distribution scheme developed by Charles Bennett and Gilles Brassard in 1984[5][6][7]. It is the first quantum cryptography protocol. This Protocol is used for Classification of Clusters[8] and Data Base[9] management service in the year 1990 by T. Harder[10]. The Quantum cryptography here opens a new chapter by introducing the entangled photons which was done using polarized filters[11]. Many of the scientists worked under these algorithms in further years had studied about attack strategies[12] and then it has used for the theoretical security for first time at Japan in the year 2010[13].

The survey has initiated in this field by Ali Ibnun Nurhadi[14] and now it has been progressed to build the cost efficient Quantum key distribution here the author makes his way by implementing it in the Quantum channel that is optical fibre and reduces the cost[15] over the Incorporating QKD into existing wavelength-division multiplexing(WDM) network infrastructure provides a practical way to reduce the difficulty and cost of QKD networking.

INTRODUCTION:

QKD system has two separate channels namely quantum channel and public channel.

Quantum Channel is used to transmit information that is generated in quantum process like the secret key in the form of photons and polarised photons, that are called as quantum bits or qubits. Whereas, the public channel is used to coordinate the exchange of qubits transmission and prior agreement on the secret key and the way it is to be transferred. QKD system essentially relies on a photon source that is a light source. So ordinary transmission media are not sufficient. So, a QKD system relies on an optical fibre or any other light transmitting channel to work.

The most important and possibly the most crucial point of the QKD algorithms and process is the concept of a Qubit. Ordinary computers do not understand the native human form of communication. They use many logic circuits and wires to do computation. The basic mode of communication in computers is in Bits that take the values of either zero or one. The Qubit can be the superposition of zero and one.

The property in Quantum Mechanics that deals with the overlapping of bits is called entanglement. Quantum entanglement is a property of two or more qubits that allows a set of qubits to express higher correlation than is possible in classical systems. So, consider two qubits that are identical. When, entanglement is done, the qubits superimpose and can result in four resultant values which is two possibilities for each qubit which would have been only two in ordinary bits using one for each bit.

Now that we have established that QKD system uses a Quantum Computer, the main question that arises is how do we get hold of a quantum computer. The straight answer to the question is that it is not possible to commercially produce and maintain a quantum computer.

The main reason being the property of quantum mechanics called Quantum Decoherence.

Quantum Decoherence is the loss of coherence or balance of a quantum particle on its exposure to the environment.

A Qubit needs to uphold the property of Quantum Coherence to oblige the properties like Superposition and entanglement. Thermal energy of a Qubit is an important factor to consider here. If thermal energy is continuously lost, then the spin of a Qubit slows down and results in inconsistent measurements. For a Qubit to remain Coherent, there must be very less loss of thermal energy over the due course of its measurements. So, to achieve that, Quantum computers are mostly maintained up to zero temperatures. Since it is practically not feasible to maintain a Quantum Computer at a commercial level, the only other option is using the already available quantum computers or simulate the jobs on a local computer. IBM developed a software called QisKit that provides quantum computing

facilities to the users. It also hosted a few Quantum computers at different places scattered across the globe wherein the user can queue their tasks for execution. But, since that depends on a lot of uncertain factors like internet connectivity and vacancy, The better option is to simulate on local machine using QisKit that is built using Python and by IBM.

BB84 protocol:

The BB84 protocol is one of the key distribution algorithms. This was introduced by the Charles Bennett and Gilles Brassard in 1984. This is the first Quantum cryptography protocol.

The key distribution is the process in which a key is distributed among two parties in a private channel. This helps the all symmetric key algorithms to distribute key. Diffie–Hellman key exchange is the key distribution algorithm which uses the multiplicative group of integers modulo and uses the primitive root modulo and prime to distribute the key. In this everyone has a public key and private key where the public key is known to everyone and the private key of the sender is sent to the receiver. However, the distributed key is generated using their public and private keys.

In BB84 protocol, the normal information precisely in ones and zeros is converted into photon or Quantum particles each with a direction such as 0, 45, 90, 135 degrees using two polarizing filters. These polarizing filters are horizontal and vertical filters and they give output in different directions. ` ``These photons are transmitted between the two in private quantum channel, that never let eavesdropper inspect the data and if he finds the data, the uncertainty makes the change in data and enables the receiver to notice the data inconsistency. After transmission the receiver again uses the polarized filters to acquire the data. The received data will not be matched but can be verified after crosschecking the polarized filters provided by sender. The remaining bits in the data will make the key. There are the four steps in this process namely Quantum Exchange, Key Shifting, Information Reconciliation, and Privacy Amplification.

QISKIT

The Qiskit is an open source software which is developed by IBM. This enables the quantum computing till machine level code of QASM (Quantum Assembly level language). We use Qubits and Quantum circuits instead of bits and classical circuits. So, similar to a bit in ordinary computer, a Qubit is the basic unit of information in a quantum computer. The main contradiction between an ordinary bit and a Qubit is the values that they accept. As we have already seen that bits take either zero or one, a Qubit, using the properties of Quantum Mechanics can take the superposition values of zero and one thus resulting in a wider range of possibilities and difficulty in deciphering.

The Qubits must need the Quantum Circuits, we cannot perform the operations using the classical gates. The classical gates like Not- gate, Nand-gate and Nor-gate takes input in Boolean and performs the operation to give the output in Boolean form. Similarly, the Quantum gates takes input Quantum bits performs the operation before measurement and gives output. The output cannot be known until we measure the value.

The Quantum gates used in this algorithm are Hadamard gate, C-Not gate and Unitary2. Most of the quantum algorithms are written for single Qubit operations. The single Qubit operations involve in operating a Qubit. The Hadamard gate involves in superposition of the Qubit. It takes any input that is position of the qubit and it transforms into |1> or |0> randomly. This is done using the probability of occurrence of |1> or |0>. This is completely random thus this cannot be predicted by any eavesdropper. The C-Not gate or CX gate in quantum computer is also called as controlled not which involves in two Qubits. It measures the first Qubit and if the value is one then it changes the other Qubit to reverse direction that is if the value of the other is one it changes to zero and vice-versa. Hence this handles the entanglement of the Qubits. Even though we have the circuits we need to measure each Qubit individually to run the machine or algorithm. It becomes more accurate if we give high number of shots. These shots are number of times the circuit is executed. The Qiskit helps us to execute all the circuits virtually in a simulator or in cloud.

PROCESS:

1. **Encrypting and Decrypting a Message**

Pick Your Secret Message:

The first step of creating a secure process is generating a secret message. The secret message that we want to transmit must have the length less than or equal to the length of the key.

If the generated key is smaller than the length of the message text, we will be compelled to use parts of the key more than once. This will allow the eavesdropper to detect and pickup a pattern that makes it easy to decipher the message.

The sender now generates a random string which acts as the key that the sender wants to transmit to Receiver. The sender then converts their bit string into corresponding qubits. Now that the sender has the qubits ready, those qubits are transferred to the receiver with random superposition that helps in generating random noise. The more the randomness , the more difficult it is to decipher.

The receiver now receives the transmitted qubits and randomly modify some of the qubits in the opposite direction before measuring their turn. The sender and the receiver publicly share their rotated qubits prior to confirmation. This allows both sender and receiver to randomly compare the values of their bits to make sure that the rotated and revealed bits values match. This ensures that there is no tampering happening In between.

Now, if it is confirmed that the sender and the receiver have the same keys after discarding the revealed bits that are in the public domain, both sender and the receiver will have matching keys that can be used for encryption.

2. Detecting is eavesdropper is tampering with the communication network.

If an eavesdropper is spying on sender and receiver's line of communication then this process of random string making and rotations using quantum mechanics is useful and it is strong and unbreakable against eavesdropper.

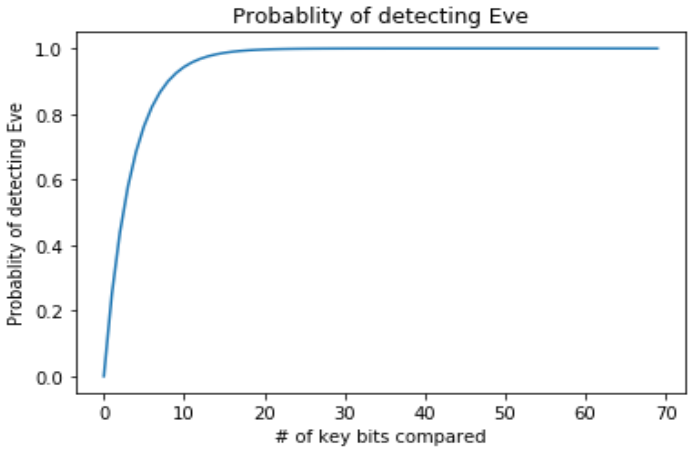
Our enemy eavesdrops by interrupting your transmission to the sender. To be sly, the enemy must send on the intercepted message otherwise the receiver will never receive anything and know that something is wrong.

The enemy can be detected if he intercepts a qubit from the sender, he will not know if sender rotated its state or not. The enemy can only measure a 0 or 1. He can't measure the qubit and then send the same qubit on the transmission medium because his measurement will destroy the quantum state. This can be possible due to the properties of Quantum mechanics. The enemy doesn't know when or where not to rotate to recreate sender's original qubit. He may as well send on qubits that have not been rotated, hoping to get the rotation right half of the time. After he sends these qubits to the receiver, after sender and receiver can compare selected parts of their random keys to see if they have errors or mis comparisons to detect the enemy.

The sender sends his qubit transmission to the receiver and the enemy measures the results. To avoid suspicion, enemy prepares qubits corresponding to the bits that he measured and sends them to receiver. The sender and the receiver make their keys like normal. They both randomly select the same parts of their keys to share publicly. If the selected part of the keys doesn't match the other, they can know someone is eavesdropping them. If the selected part of the keys matches the other, they can be confident that no one is eavesdropping. They throw away the part of the key they made public and encrypt and decrypt secret messages with the portion of the key they have left with.

RESULTS:

The longer the key, the more likely you will detect Eve. In fact, the probability goes up as a function of where n is the number of bits Alice and Bob compare in their spot check. So, the longer the key, the more bits you can use to compare and the more likely you will detect Eve.



REFERENCES:

1. Feynman, R. P.u "Simulating physics with computers". International Journal of Theoretical Physics 21 (6) (1982).: 467–488.

2. Feynman, R. P.u "Quantum mechanical computers ". ASL Journal of Theoretical Physics 27 (1984).: 369-375.

3. Analytic Calculation of Higher-Order Quantum-Chromodynamic Corrections in e+ e− Annihilation William Celmaster and Richard J. Gonsalves Phys. Rev. Lett. 44, 560 – Published 3 March 1980.

4. C. Helstrom "Bayes-cost reduction algorithm in quantum hypothesis testing" Theoretical Computer Science. Theoretical Aspects IEEE pages: 359 – 366.

5. C. H. Bennett and G. Brassard. "Quantum cryptography: Public key distribution and coin tossing". In Proceedings of IEEE International Conference on Computers, Systems and Signal Processing, volume 175, page 8. New York, 1984. http://researcher.watson.ibm.com/researcher/files/us-bennetc/BB84highest.pdf

6. Bennett, Charles H.; Brassard, Gilles (2014-12-04). "Quantum cryptography: Public key distribution and coin tossing". Theoretical Computer Science. Theoretical Aspects of Quantum Cryptography – celebrating 30 years of BB84. 560, Part 1: 7–11. doi: 10.1016/j.tcs.2014.05.025.

7. Branciard, Cyril; Gisin, Nicolas; Kraus, Barbara; Scarani, Valerio (2005). "Security of two quantum cryptography protocols using the same four qubit states". Physical Review A. 72 (3): 032301. arXiv:quant-ph/0505035. Bibcode:2005 PhRvA.72c2301B. doi:10.1103/PhysRevA.72.032301.

8. H. Schöning A. Sikeler "Cluster Mechanisms Supporting the Dynamic Construction of Complex Objects" in: Proc. 3rd

Int. Conf. on Foundations of Data Organization and Algorithms (FODO), Paris, 1989, pp. 31-46.

9. H -B Paul, H -J Schek, M H Scholl, G Welkum, U Depplsch "Architecture and Implementation of the Darmstadt Database Kernel System" Techmcal Umverslty of Darmstadt Computer Science Department Alexander& 24, D-6100 Darmstadt.

10. T. Härder "An Approach to Implement Dynamically Defined Complex Objects"University of Kaiserslautern, Dept. of Computer Sciences, D-6750 Kaiserslautern, West Germany.

11. Thomas Jennewein, Christoph Simon, Gregor Weihs, Harald Weinfurter, and Anton Zeilinger "Quantum Cryptography with Entangled Photons" Institut f¨ur Experimentalphysik, Universit¨at Wien, Boltzmanngasse 5, A–1090 Wien, Austria Sektion Physik, Universit¨at M¨unchen, Schellingstr. 4/III, D-80799 M¨unchen, Germany.

12. Guihua Zeng "A simple attacks strategy of BB84 protocol", National Key Laboratory on ISDN of XiDian University, Xi'an, 710071, P.R.China, year 2000.

13. Yousuke Sano, Ryutaroh Matsumoto, Tomohiko Uyematsu "Secure Key Rate of the BB84 Protocol using Finite Sample Bits", Department of Communications and Integrated Systems Tokyo Institute of Technology Oookayama, Meguro-ku Tokyo, 152–8552, Japan year:2010.

14. Ali Ibnun Nurhadi Nana Rachmana Syambas "Quantum Key Distribution (QKD) Protocols", A Survey Institut Teknologi Bandung Bandung, Indonesia, 2018.

15. Yuan Cao, Yongli Zhao, Jianquan Wang, Xiaosong Yu, Zhangchao Ma, and Jie Zhang "Cost-Efficient Quantum Key Distribution (QKD) Over WDM Networks", IEEE/OSA Journal of Optical Communications and Networking, June,2019.