INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 6.078
(Volume 6, Issue 6)
Available online at:https://www.ijariit.com

Applications of quantum computing in IT security

Aditi Venkatesh Bhat
aditivenky2003@gmail.com
Independent Researcher, Bengaluru, Karnataka

Aditya Iyer
adityaiyer.m@gmail.com
Independent Researcher, Bengaluru, Karnataka

ABSTRACT

Observing the developments in the quantum computing, our current encryption-decryption systems such as the RSA are insubstantial. Therefore, to maintain the security and protection of data, quantum cryptography through the understanding of quantum key distribution is the future of secured user-data.

Keywords:*Quantum Computing, Cryptography, Quantum Key Distribution, Shor's Algorithm, Encryption, Decryption, Logic Gates, Boolean Tables, Qubit, Quantum Gates*

1. INTRODUCTION

A computer acts as a machine that makes a human's work easier. It is a machine that has the right combination between several hardware and software components that allows it to function with human instructions. The human instructions behave as input while the computer processes those instructions and returns the acquired result as the output.

While all the data we provide to the computer as an input is usually in visual representations or through a programming language, the computer takes in all those different languages as input, converts it to machine language - binary code (the 1s and 0s language) and converts back to a human understanding language and returns the output. In summary this is how a computer processes any command given to it but the complex aspects can only be seen if we give close attention to the processing stage and the tasks done during that point of time.

Once the instructions are received by the computer, as mentioned earlier, the computer converts human language into the machine language. After the machine language has been obtained, the machine code is interpreted in various ways based on different situations:

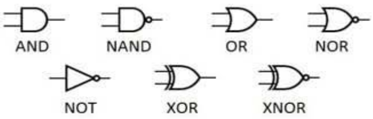
- For example if it is a notification box that has been sent to the user and the computer is processing the user's input in terms of the notification, there can be two ways that a user could have responded. The computer goes to the storage unit to retrieve the two possible inputs of the user in terms of machine language. To make things simpler for understanding, we can assume that the computer has assigned 1 for an acceptance action for that notification and a 0 for canceling the notification. Once the input is received, the computer matches it to the two different possibilities and draws an internal logic gate in order to result in the right output.
- Another common situation that can be taken as an example could be the tasks present for a printer to print out. The input that the computer takes in are the pages that the user wants to print and once the print command is clicked, multiple aspects or attributes of the printer are set by the user in human language but the computer converts all these settings to machine language in order to process the right print task. Just like in scenario 1, even in this situation the computer matches the input of each attribute to binary digit outcomes and then draws an internal logic gate to output the right task to the external device (printer).

Observing both the scenarios, something that does turn out to be similar is the computer forming logic gates. In addition to the logic gates, there are boolean tables that assists in understanding the logic gates.

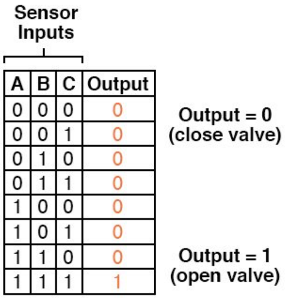
2. MORE ABOUT LOGIC GATES AND BOOLEAN TABLES

Logic gates are defined to be the elementary building blocks of any digital system process. It can be referred to as a circuit for either a process that has one input its corresponding output or a circuit that has more than one input that are connected through specific logic gates to result in one particular output. This is connected to the binary code as logic gates are derived from boolean expressions or boolean tables. Boolean tables are formed between different variables. These logic gate tables show each possible input combination for a specific scenario with a resultant output for each one of those input combinations. TRUE in binary representation of a boolean table is a 1 while FALSE is 0. An example of a boolean table is below:

The logic gates go in handy with the boolean tables:



Each one of these gates has its own use. For example the AND gate is used when talking about procedures in relation to functions needed to be true one after the other. Even if one input does not meet the needs of the process, the output fails. An example of the AND gate through a boolean table is given below.



Sensor Inputs

A	B	C	Output
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Output = 0 (close valve)

Output = 1 (open valve)

This boolean table is drawn to output the results of an 8 bit input of 3 variables: A, B and C. Looking at the table it can be said that when A and B and C are 1 (in other words TRUE), the output is 1 (TRUE) - the process continues. Else, all the other input combinations result in a 0 (FALSE) - not allowing the process to complete.

Quantum Gates

Qubits*:* The most basic unit of computation in regular computers is a bit. Similarly, in a quantum computer the most fundamental unit dealt with is a quantum bit (qubit). A traditional bit can either have the value of 0 or 1, however, a qubit is in a superposition of both 0 and 1 at the same time. Mathematically, the state of the electron can be described in the form of an equation given by:

$$\alpha|0\rangle + \beta|1\rangle$$

It should be noted that the $\alpha$and $\beta$are complex amplitudes of the qubit at that 0 and 1 state respectively. Additionally, it's important that the complex coefficients must be normalised, i.e.

$$|\alpha|^2 + |\beta|^2 = 1$$

As mentioned earlier, the qubit is not in a definite state, rather is a superposition of 0 and 1. The probability of finding the qubit in either states is given by the square of their complex coefficient i.e. the probability of finding the qubit at 0 is $|\alpha|^2$and the probability of finding the qubit at 1 is $|\beta|^2$.

When the system is measured, the state of the superposition is disturbed and the quantum bit takes on a definitive value.

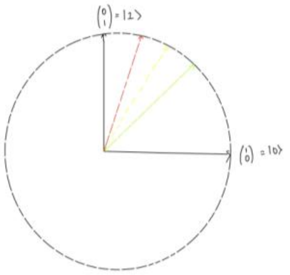
Geometric interpretation: Another notation to represent the superposition is in the form of a vector:

$$\begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

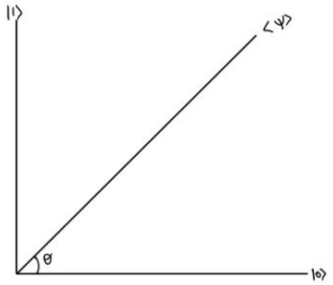
This allows us to interpret the quantum bit as a vector in a two dimensional complex vector space.
Thereby, the 0 and 1 states can also be represented as the following vectors:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Geometrically, we can consider a two dimensional vector space as shown in the figure below. The dotted vectors represent the possible states of the superposition. Here it's important to note that the dotted lines don't represent all the possible states, and the possible states must be within the |0⟩ ,|1⟩bases.


$\begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle$
$\begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle$

Measurement

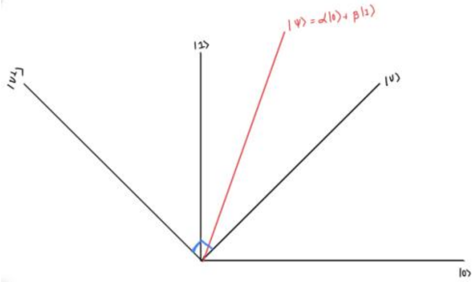


Let’s assume a state ψ(psi) that makes an angle of $\theta$with the unit vector in the x-axis.

$$|\psi\rangle = \cos\theta|0\rangle + \sin\theta|1\rangle$$

When we measure the state of ψ, the superposition collapses and ψtakes on a definitive value of either |0⟩or |1⟩ . The probability that ψwill take the |0⟩ state is given by $\cos^2\theta$. Similarly, the probability that ψwill take the |1⟩ state is given by sin^2. When a measurement is done, the qubit takes on the value of either the excited or the ground state. Essentially, from a geometric standpoint the state $\psi$is projected onto either the |0⟩or |1⟩ with the probabilities mentioned earlier.

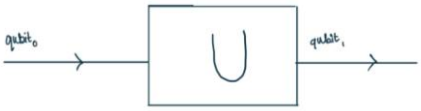
Similarly, instead of measuring it in a |0⟩to |1⟩ basis the, state ψcan be measured on any arbitrary orthogonal basis.



Now, we measure it in an arbitrary basis of |u⟩ and |u⟩-perp. When we measure ψin a |u⟩ and |u⟩-perp basis, the exact same answer holds true as before. Assuming ψmakes an angle of $\theta$with the |u⟩ vector, then we measure it we obtain the following results:

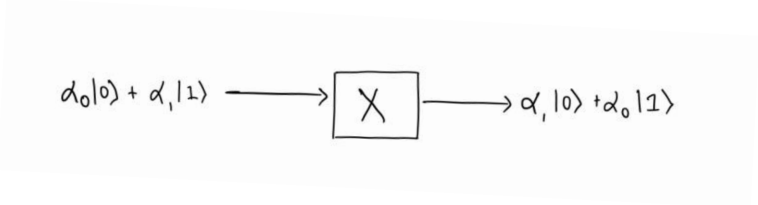
- $\psi$is projected onto the |u⟩ vector with a probability of $\cos^2\theta$
- $\psi$is projected onto the |u⟩-perp vector with a probability of $\sin^2\theta$

Quantum Gates



A single qubit quantum gate takes in a qubit and outputs a modified qubit. Schematically, this is represented by the figure above as qubit$_0$ is transformed to qubit$_1$.

Bit Flip gate: Essentially, the bit flip gate matches the coefficient of the qubit in the |0⟩ state to the qubit in the |1⟩ state.



As seen in the above diagram, the |0⟩ state qubit previously matched to α_0 is now matched to the α_1 and a similar pattern is obtained for the |1⟩ state.

To understand the reason behind the flip, let us equate

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

When X is multiplied with the |0⟩ state, it yields the |1⟩state, so integrally, it's not the coefficients that are getting swapped but rather the state is changing from one to the other causing the illusion of a flip in the coefficients.

$$X\ |0\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \times \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

A similar is observation is made in the |1⟩ state.

$$X\ |1\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \times \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

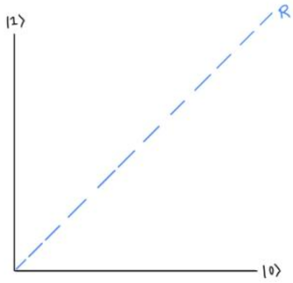
As the name suggests the gate flips a qubit from one state to another as it passes through the gate.
However, before we can call that its a quantum gate, we must first check that X is a unitary transformation. So, in this case we must verify the following statement:

$$X^2=I$$

Where I is the identity matrix.

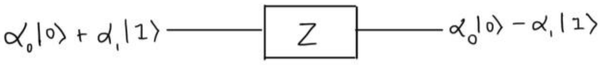
$$X \times X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Ergo, the condition holds true and X is indeed a quantum gate. The unitary transformation in essence is a rotation of the space.



We know that the bit flip gate, maps the 0's to the 1's and vice versa. Therefore, the quantum gate can be visualised as a rotation of the qubit around the R-axis which makes an of 45∘with both the axes. Additionally, it should be noted that the space mentioned is not a real space rather a 2-dimensional complex space.

Phase flip gate: True to its name, the phase flip gate changes the sign of one of the quantum bits.



To understand the reason behind the phase flip , let us equate

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

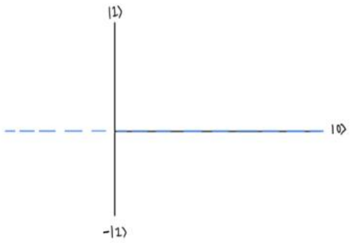
When the |0⟩ state is passed through the phase flip gate,

$$Z \times |0\rangle = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \times \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

As seen by the above equation there is no change on the |0⟩ state qubit. However, when the |1⟩ state qubit is passed through the phase flip gate, a sign change is observed, as shown by the following equation.

$$Z \times |1\rangle = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \times \begin{pmatrix} 0 \\ 1 \end{pmatrix} = -\begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Using the same logic as above, Z acting on a negative gate will change that into a positive.



The phase flip gate maps the 1 state to the -1 state and can be considered as a rotation along the x-axis in a 2-dimensional complex plane.

Hadamard transform

A Hadamard transform labels the $|0\rangle$ state as a positive state and $|1\rangle$ state as the negative state.

$$H = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}$$

$$H \times |0\rangle = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \times \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}$$

$$H \times |1\rangle = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \times \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix}$$

In the above equations, we see that once the $|0\rangle$ state passes through the hadamard transform it yields in the positive state. Consequently, the opposite is seen for the $|1\rangle$ state as it yields in the negative state.

Quantum Cryptography

What is Cryptography?

Cryptography is referred to as an external medium through which secure information is communicated and protected. Cryptography secures information through two different processes:

- Encryption
- Decryption

Encryption & Decryption: Encryption is a stage of cryptography that is used to encode information. It uses various algorithms of discrete security levels in order to encode important information. Encryption follows the methodology of converting the original information, in other words known as plaintext to its encrypted/hidden form known as ciphertext. To convert plaintext to ciphertext, encryption algorithms use specific keys. These keys are meant to only be given to the decrypter for keeping the information secure.

Similar to encryption, decryption is also a process of cryptography. After plaintexts are encrypted into ciphertexts, it is essential to convert them from ciphertexts back to plaintexts to make sense of the messages. As a result, decryption works towards converting the ciphertexts to plaintexts through a key.

Going deeper into the way encryption works, it is important to know the various types of encryption present. The two main types of encryption are:

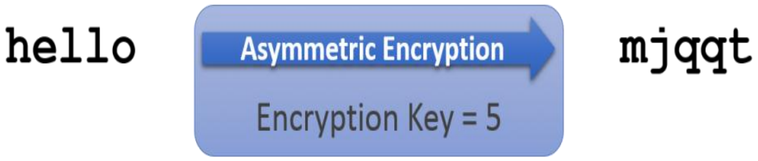
- **Symmetric Encryption:** This form of encryption uses a single key to encrypt and decrypt data.
- **Asymmetric Encryption:** This form of encryption uses two different keys to encrypt and decrypt data. The key that is used to encrypt data is generally a public key that is made available to the public users while the decryption key is a private key and is not made available to all the public users, rather only to the user that has to decrypt the data (who has to decode a certain message).

On the other hand decryption only works with one specific key.

Other than these main classifications, there are different algorithms that are used for encrypting and decrypting data.

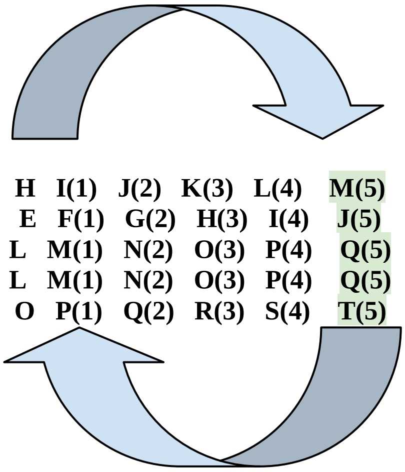
1) Data Encryption Standard (DES)
2) Triple DES
3) RSA
4) Advanced Encryption Standard (AES)

An example of encryption can be seen in this image:



"Hello" (plaintext) has been encrypted through the Data Encryption Standard algorithm, where the key is 5 for encrypting. This is a simple example to understand as to the word "Hello", 5 letters have been added to each corresponding letter of the word.

3. ENCRYPTION



H	**I(1)**	**J(2)**	**K(3)**	**L(4)**	**M(5)**
E	**F(1)**	**G(2)**	**H(3)**	**I(4)**	**J(5)**
L	**M(1)**	**N(2)**	**O(3)**	**P(4)**	**Q(5)**
L	**M(1)**	**N(2)**	**O(3)**	**P(4)**	**Q(5)**
O	**P(1)**	**Q(2)**	**R(3)**	**S(4)**	**T(5)**

4. DECRYPTION

Complex algorithms encrypt messages through a more secure key to reduce hacks and access or leaking of important data. Though the encryption and decryption algorithms are very complex and are done with huge numbers as keys for encrypting messages, the developing technology in computers is beating the algorithmic keys. To ensure safe and secure data, we begin to bring in Quantum Cryptography.

4.1 More about Quantum Cryptography

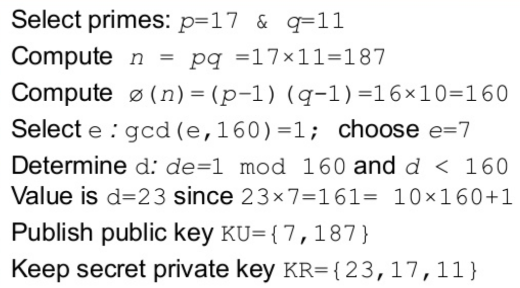
Quantum cryptography, just as its name suggests, uses quantum mechanical properties to secure and keep information protected. It also follows the basic cryptography procedure of encryption and decryption, with an extremely intricate process for encrypting messages.

Before we go further into quantum cryptography, it is essential to know about the RSA form of encryption as Quantum Key Distribution (an example of Quantum Cryptography) works on the basis of the RSA algorithm.

4.2 RSA Encryption

RSA encryption is one amongst the most common algorithms used to encrypt data. We see RSA encryption in all mediums of social media to protect our messages and data. This algorithm works just like the standard encryption and decryption, but for the encrypting and decrypting key it has a complex process.

If you multiply two large prime numbers you will obtain a larger number as the result of multiplication.
For example:



Select primes: p=17 & q=11

Compute $n = pq$ =17×11=187

Compute $\varnothing(n)=(p-1)(q-1)$=16×10=160

Select e : gcd(e,160)=1; choose e=7

Determine d: de=1 mod 160 and d < 160

Value is d=23 since 23×7=161= 10×160+1

Publish public key KU={7,187}

Keep secret private key KR={23,17,11}

As a simple explanation to the above seen example, there are two prime numbers which are multiplied and from the public key, there are factors that the decryptor has to derive to. When the number is small, trial and error is easier. But as the number gets larger it gets harder to try all possible factors. But with fast computers today, it is said to be easier to decrypt messages.

4.3 Quantum Cryptography and Quantum Computing

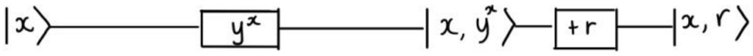
As previously mentioned, the quantum gate is in a superposition of multiple states and in fact this superposition of states helps in performing parallel calculations at a much faster rate. In theory, quantum computers can compute the prime factors of extremely large numbers at a much faster rate than even the most powerful supercomputers. Internet cryptography works on prime factoring, and quantum computers can be used to easily break even complex encryptions of the classical internet cryptography. This is primarily important as the public key used in the RSA encryption process is formed by the multiplication of two large prime numbers. Due to the limitations of classical computation, it is much harder to factorise two prime numbers as compared to multiplying them. However, due to the immense processing power of quantum computing this limitation might soon be broken, and decrypting messages would become a tremendously simple process. One of the key algorithms that can aid in the process of breaking encryption is that of shor's factoring algorithm.

4.5 Shor's Algorithm

To factor out large number using Shor's algorithm, the following 3 steps are undertaken:

1) A random guess (y)
2) Pass that random guess (y) through a quantum computer to receive a value p
3) $y^{(p/2)} \pm 1$ will lead to a much more accurate guess of the prime factor of an enormous number

To factor a number, for example let's say: 6745123, we make a random guess, say y. We then try to find the value of p. To find p, the condition that we look for is y^p is one more than a multiple of 6745123. In order to perform this computation, we use a quantum computer that raises y to the power x and calculates how much more that value is than a multiple of n.



If we start with a superposition of all the numbers up to 6745123, then from the quantum computation we get the superposition of $y + y^2 + y^3$ and so on along with the superposition of all the remainders. If we measure the state of the remainders, the wave function collapses, and the remainders take on one definitive value, say z. The remaining superpositions are separated by a frequency, of $\frac{1}{p}$.

When the superposition:

$$|x\rangle + |x+p\rangle + |x+2p\rangle + \ldots$$

Is passed through a quantum fourier transform,

$$F_N = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \omega^3 & \cdots & \omega^{N-1} \\ 1 & \omega^2 & \omega^4 & \omega^6 & \cdots & \omega^{2(N-1)} \\ 1 & \omega^3 & \omega^6 & \omega^9 & \cdots & \omega^{3(N-1)} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 1 & \omega^{N-1} & \omega^{2(N-1)} & \omega^{3(N-1)} & \cdots & \omega^{(N-1)(N-1)} \end{bmatrix}$$

where

$$\omega = e^{\frac{2\pi i}{N}}$$

When we measure, the superposition of frequencies, we get a random output

$$|\frac{u}{p}\rangle$$

After doing this a few more times, we get more random outputs such as

$$|\frac{c}{p}\rangle, |\frac{e}{p}\rangle, |\frac{f}{p}\rangle$$

On repeating this process sufficiently, we deduce the common factor of

$$|\frac{1}{p}\rangle$$

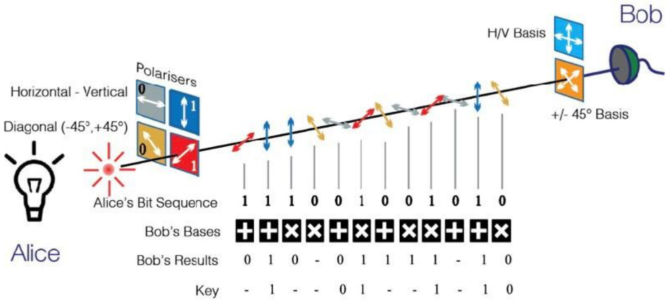
from which we can deduce the value of p.

Using the equation earlier, $y^{(p/2)} \pm 1$ we now have a much better estimate of the factors. On multiplying the two possibilities $y^{(p/2)} + 1$ and $y^{(p/2)} - 1$ we are able to find common factors with the numbers that are to be cracked.

This is a lengthy process as it involves a lot of guesswork and does not contain one specific formula of prime number factorisation. As quantum computing can easily break our current encryption systems, there arises a need for quantum cryptography.

4.6 Quantum Key Distribution

This is where we come back to Quantum Cryptography. As we know through Quantum Mechanic concepts, a particle's position and momentum is hard to predict. There is always probability involved. Quantum key distribution uses this concept to its advantage. It involves the passing through of millions of photons (polarized light particles) into a fiber optic cable from one source to another. Since each photon has a random state, being passed into different polarized filters lets the receiver derive a combination of 0s and 1s. To do this, the receiver uses beam splitters - (horizontal, vertical and diagonal filters). But the key here is to get the right combination of 0s and 1s, the receiver will have to use the same filters as the sender did. If not there will be a completely different stream of 0s and 1s formed.


Bob
H/V Basis
+/- 45° Basis
Polarisers
Horizontal - Vertical
Diagonal (-45°,+45°)
0
1
0
1
Alice
Alice's Bit Sequence 1 1 1 0 0 1 0 0 1 0 1 0
Bob's Bases + + × × + × + × × + + ×
Bob's Results 0 1 0 - 0 1 1 1 1 - 1 0
Key - 1 - - 0 1 - - 1 - 1 0

If observed clearly only a few digits match Bob's results in this example which means the rest of the numbers can be ignored and the key will still hold valid.

4.7 Advantages of Quantum Cryptography

Since the formation of the keys are on the basis of the random states of a stream of delicate photons, if an external intruder attempts to hack the key, it will change the state of the photons thereby changing the entire sequence of 0s and 1s resulted by the photons which will not decode the message accurately. Furthermore, this change will be notified to the sender and receiver of the stream of photons, so a new stream will be sent and the old one will be discarded. This makes this form of key formation for encrypting messages unbreakable, thereby increasing the efficiency and security.

To bring quantum cryptography into action, it is essential we change the way our network works. All of it will be based on the random state of particles but it will result in an extremely secure network society.

5. REFERENCES

[1] harmoush, ed. "Asymmetric Encryption." *Practical Networking .Net*, 15 Mar. 2019, www.practicalnetworking.net/series/cryptography/asymmetric-encryption/.

[2] Song, JooSeok. "The RSA Algorithm ." *SlideShare*, 2 Feb. 2014, www.slideshare.net/natemiller67/the-rsa-algorithm-jooseok-song.

[3] "Quantum Key Distribution (QKD)." *Quantum Technology*, qt.eu/discover-quantum/underlying-principles/quantum-key-distribution-qkd/.

[4] Roell, Jason. "Demystifying Quantum Gates - One Qubit At A Time." *Medium*, Towards Data Science, 28 Feb. 2018, towardsdatascience.com/demystifying-quantum-gates-one-qubit-at-a-time-54404ed80640.

[5] Bækkegaard, T., et al. "Realization of Efficient Quantum Gates with a Superconducting Qubit-Qutrit Circuit." *Nature News*, Nature Publishing Group, 16 Sept. 2019, www.nature.com/articles/s41598-019-49657-1.

[6] Cho, Adrian. "The Biggest Flipping Challenge in Quantum Computing." *Science*, 9 July 2020, www.sciencemag.org/news/2020/07/biggest-flipping-challenge-quantum-computing.

[7] Chu , Jennifer. "The Beginning of the End for Encryption Schemes?" *MIT News | Massachusetts Institute of Technology*, 3 Mar. 2016, news.mit.edu/2016/quantum-computer-end-encryption-schemes-0303.

[8] Katz, Natan. "Quantum Factorization." *Medium*, Towards Data Science, 15 Sept. 2020, towardsdatascience.com/quantum-factorization-b3f44be9d738.

[9] Ristè, Diego, et al. "Real-Time Processing of Stabilizer Measurements in a Bit-Flip Code." *Nature News*, Nature Publishing Group, 21 Aug. 2020, www.nature.com/articles/s41534-020-00304-y.

[10] Hao, Y. M., et al. "Quantum Controlled-Phase-Flip Gate between a Flying Optical Photon and a Rydberg Atomic Ensemble." *Nature News*, Nature Publishing Group, 12 May 2015, www.nature.com/articles/srep10005.

[11] Hui, Jonathan. "QC-Quantum Fourier Transform." *Medium*, Medium, 21 Jan. 2019, jonathan-hui.medium.com/qc-quantum-fourier-transform-45436f90a43.

[12] Steane, Andrew. "Quantum Computing." *Reports on Progress in Physics*, vol. 61, no. 2, 1998, pp. 117–173., doi:10.1088/0034-4885/61/2/002.

[13] Scarani, Valerio. "Quantum Computing." *American Journal of Physics*, vol. 66, no. 11, 1998, pp. 956–960., doi:10.1119/1.19005.

[14] Ladd, T. D., et al. "Quantum Computers." *Nature*, vol. 464, no. 7285, 2010, pp. 45–53., doi:10.1038/nature08812.

[15] Amico, Mirko, et al. "Experimental Study of Shor's Factoring Algorithm Using the IBM Q Experience." *Physical Review A*, vol. 100, no. 1, 2019, doi:10.1103/physreva.100.012305.

[16] Zhou, S. S., et al. "Quantum Fourier Transform in Computational Basis." *Quantum Information Processing*, vol. 16, no. 3, 2017, doi:10.1007/s11128-017-1515-0.

[17] Tsokos, K. A., and Mark Farrington. *Physics for the IB Diploma*. Cambridge University Press, 2014.

[18] "How Does Quantum Key Distribution Work?: QKD Explained." *Quantum Xchange*, quantumxc.com/how-does-quantum-key-distribution-work/.

[19] Written by Alison Grace Johansen for NortonLifeLock. "What Is Encryption and How Does It Protect Your Data?" *Norton*, us.norton.com/internetsecurity-privacy-what-is-encryption.html.

[20] "What Is Cryptography and How Does It Work?" *Synopsys*, www.synopsys.com/glossary/what-is-cryptography.html.

[21] "Converting truth tables into boolean expressions." *All about circuits*,

[22] www.allaboutcircuits.com/uploads/articles/converting-truth-tables-into-boolean-expressions.jpg

[23] "ANd9GcSsML6J8ySJ3h0LdBH8IrxpY9Jk1ufEYMxx9k0gZyDlee-F3cfhAhIPzpUbhq1tqyzbGwCjcvjdcGXPfLJ9gWl65pDk7DD6qQhsPA&usqp=CAU&ec=45732303." *Encrypted-tbn0*,

[24] www.encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcSsML6J8ySJ3h0LdBH8IrxpY9Jk1ufEYMxx9k0gZyDlee-F3cfhAhIPzpUbhq1tqyzbGwCjcvjdcGXPfLJ9gWl65pDk7DD6qQhsPA&usqp=CAU&ec=45732303

[25] Bäumer, Elisa, et al. *Shor's Algorithm*. *qudev.phys.ethz*, qudev.phys.ethz.ch/static/content/QSIT15/Shors%20Algorithm.pdf.

[26] Gao, Jie, et al. "Realizing Quantum Controlled Phase-Flip Gate through Quantum Dot in Silicon Slow-Light Photonic Crystal Waveguide." *Optical Nanostructures Laboratory*, doi:https://arxiv.org/pdf/0803.1017.pdf.

[27] Berkeley, University of California. "QFT, Period Finding & Shor's Algorithm." *EdX*, CS, no. 191x. *Chap 5*, doi:https://courses.edx.org/c4x/BerkeleyX/CS191x/asset/chap5.pdf.

[28] Styles, Iain. "Lecture 6: The Quantum Fourier Transform." *CS Bham Courses*, Lesson 6, doi:https://www.cs.bham.ac.uk/internal/courses/intro-mqc/current/lecture06_handout.pdf.