Revista de Sistemas de Informação da FSMA
n. 7 (2011) pp. 39-48

SISTEMAS DE INFORMAÇÃO
http://www.fsma.edu.br/si/sistemas.html

TUTORIAL

Quantum Cryptography: a direct approach

A. J. B. Mendes, E. H. Paulicena, W. A. R de Souza

Abstract— **This study aims to answer directly to four questions relating to knowledge of what Quantum Cryptography is. The text is developed through a historical overview of the encryption of messages, reaching asymmetric encryption, a solution to the problem of production and distribution of keys. Afterwards, inserted in a quantum scope, it defines and exemplifies protocols of quantum cryptography, showing, in conclusion, the responses required.**

Index Terms-**Quantum Cryptography, QKD, Quantum Key Distribution, BB84, Reconciliation Information, Privacy Amplification.**

INTRODUCTION

THis paper intends to establish an accessible and easily understandable explanation on the limits and possibilities of Quantum Cryptography for those who do not work on this field.

It is important to make it clear this paper's goals: the term Quantum Cryptography will be considered only in the aspects concerning the concept of Quantum Key Distribution.

We intend to bring them to an objective level of understanding that will allow the reader to understand the fundamental concepts and be able to answer to a few selected questions.

In order to achieve this goal, Section II will address the fundamental issue in cryptography, that is, the guarantee of the secrecy of a message. We will start with a brief history that explains how algorithm and key are related in cryptographic systems and the limits to the security they provide.

Section III presents some differences between Quantum Mechanics and Classical Mechanics, showing that it is feasible to build a computer whose processing is based on quantum states.

Section IV will present the distinction between Quantum Computing and Quantum Key Distribution and will follow with a presentation on the BB84 protocols and its procedures, with the pertinent conclusions to these.

Section V established a theoretical example that complies with the original academical definition of the procedures relating to Information Reconciliation and Privacy Amplification and makes some comments on the current state of the art of the subject of this paper.

Section VI, the conclusion answers the following questions:

a) is quantum computing necessary for Quantum Key Distribution?

b) is Quantum Key Distribution a self sufficient system for ciphering and deciphering messages?

c) what is the purpose of Quantum Key Distribution?

d) what does it means when we claim that Quantum Key Distribution is 100% safe?

CLASSICAL CRYPTOGRAPHY

Three words are commonly used in the ciphering and deciphering universe: cryptology, cryptography and cryptanalysis. The word etymology is clear: *Cripto* is a Greek word that means secret, *Logos* means discourse or study, *Grafos* means writing and *Analisis*, means separation (as opposed to synthesis). Therefore, it is simples to understand the meaning of those three important words.

Cryptology is the study of secrets, including cryptography and cryptanalysis; cryptography is the writing of secrets, here understood as two process: the ciphering and the deciphering of a message; and cryptanalysis is the separation of secrets, that is, the search to find the original message that originated a ciphered text or the key used to cipher it, without any previous knowledge.

The art of message ciphering was known since ancient era and Roman emperor Caesar's Cipher [1] consisted on writing the original message, called clear text, in the cryptographic context through a simple substitution of each letter for another one three positions ahead. Therefore, the letter "A", when ciphered, became "D" and the word "CAESAR" became "FDHVDU".

Manuscript received march 25, 2011.

Alvaro Jorge Braga MENDES, Centro de Análises de Sistemas Navais – CASNAV. Rua da Ponte, Ed 23 do AMRJ, Ilha das Cobras, Centro, 20091-000 Rio de Janeiro, RJ, Brasil, alvaro@casnav.mar.mil.br

Edésio Hernane PAULICENA, Centro de Análises de Sistemas Navais – CASNAV. Rua da Ponte, Ed 23 do AMRJ, Ilha das Cobras, Centro, 20091-000 Rio de Janeiro, RJ, Brasil, edesio@casnav.mar.mil.br

William Augusto Rodrigues de SOUZA., Centro de Análises de Sistemas Navais – CASNAV. Rua da Ponte, Ed 23 do AMRJ, Ilha das Cobras, Centro, 20091-000 Rio de Janeiro, RJ, Brasil, william@casnav.mar.mil.br

In the twentieth century, electromechanical machines for starters and computers afterwards, gave a big boost to cryptography, given their huge power to change and scramble a clear text.

It is interesting to remark that cryptographic processed usually have two fundamental elements: algorithm and key. The algorithm is the procedure to cipher the message and the key is a specification for that process. In Caesar's Cipher, fro instance, the algorithm is "replace the letter for one that is a few positions ahead" and the key is "3, or third position". Please notice that knowing the algorithm is not enough to solve the problem, for we still do not know how many positions ahead we must stop. In order to find that out, we must enter the field of cryptanalysis, with a method called "brute force", that consists of testing every possible key. For an alphabet of 26 letters, Caesar's Cipher would have 25 possible keys, since one of them (0 or 26) would not make sense, given that it would not chance the clear text. Therefore, with only 25 attempts we would be sure to break that cipher and discover the clear text.

This small example has a single purpose: to show how important is the cryptographic key, specially because most cryptographic algorithms widely used are open source and publicly known. Therefore, the only way to keep the secret is through the key.

This fact has already been established explicitly by Auguste Kerckhoffs in 1873 in his opus *La cripographie militaire*, when he proposed that the security of cryptographic systems must rely solely on the secrecy of the key and not in the algorithm [2]

Current cryptographic keys usually have 128 or 256 bits (sequences of zeros and ones). In the case of a key with 256 bits, that means 2^{256} possible keys. In order to understand the magnitude of this number, one must consider that it is equivalent to the number one followed by 77 zeros. Therefore, it is a lot better than the 25 attempts needed to solve Caesar's Cipher.

Table 1 shows an example of the average time required for an attack by brute force to perform a complete search for a key, based on the execution of a fixed amount of decryptography task per time unit [1]. The result shown in the last column contemplates the possibility of using massively parallel processor architectures. As a comparison, we can stress that the age of the universe, since the Big Bang to our days, is estimated as 1,37 x 10^{10} years.

Symmetric cryptography has always been used to assure message confidentiality and is an algorithmic procedure that uses a single key, both in ciphering as in deciphering the message. This way, is some moment the key used by the emitter to cipher the clear text must be sent to the message receiver, so that he can understand the ciphered text. Classic transmission channels of the used keys include oral communication, phone communication or even e-mail. If on one hand informing orally and personally the key is a reasonably secure method, on the other hand it is not efficient and as agile as automated processed may require. Agile transmission methods such as e-mails, for instance, imply of the possibility of an invasion of that channel by an intruder and his taking hold of the information (or even his corrupting the key).

This is the central issue of cryptography: assure that only authorized parts can access the transmitted information. Symmetric cryptography assures the safety of the clear text when using a 128 or 256 bits keys. But who assures the security of the key?

In the final decades of the previous century, this question was solved using asymmetric or public key cryptography. As Columbus egg, it is simple after you know it. Asymmetric cryptography also consists on an algorithm and a key that now consists of two parts, one of which is used to cipher the clear text and the other one to decipher it.

Therefore, if a user intends to receive ciphered messages he only needs to inform, publicly what key must be used by those who want to send him messages. This key is called his public key. Once the ciphered message is received, the user will decipher it using a key that is known only by him, called his private key. Only this private key will be able to decipher the received message and, hence, each user involved in a cryptographic transmission process will have his own pair of keys: his public key will be used to cipher messages addressed to him and using his private key he will be able to decipher them.

TABLE 1 –AVERAGE TIME TO SEARCH FOR KEYS

Key size (bits)	Number of possible keys	Time needed to search for key (considering one decryptography each µs)	Time needed to search for key (considering a million decryptography each µs)
32	2^{32}=4,3 x 10^{9}	35,8 minutes	2,15 miliseconds
56	2^{56}=7,2 x10^{16}	1.142 years	10,01 hours
128	2^{128}=3,4 x10^{38}	5,4 x 10^{24} years	5,4 x 10^{18} years
168	2^{168}=3,7 x10^{50}	5,9 x 10^{36} years	5,9 x 10^{30} years

Some concepts must be made clear:

a) The asymmetric algorithm is structurally different from the symmetric one. The calculations that support it are based on mathematical functions that demand two keys, one as input to cipher and one as an input to the deciphering process.

b) Any transmission user can calculate his own pair of keys;

c) Once the pair of keys is calculated, what is ciphered with the public key can only be deciphered with the private one;

d) It is not possible, except for who calculated his own pair of keys, to obtain the private key once in possession of the public key and vice versa.

It may seem strange that someone can calculate two Keys that are mathematically related, inform one of them and it is not feasible for anyone to calculate the other key. The reason for this lack of feasibility is the fact that some additional information pertaining only to the person who generates the keys is used to calculate them. Hence, without them it is not feasible to calculate the unknown key.

As a figurative description, we may say that there are some mathematical functions which are easy to calculate (the ciphering process) but which are very hard to invert (the deciphering process). In order to perform this inversion it is necessary to use additional information known only to the key owner.

We can use RSA [3] as an example, since it is a widely used system in asymmetric cryptography. Since it is quite hard to factor two huge numbers generated by the product of two equally large prime numbers, the information made public on the product is not enough to make it feasible to find the private key corresponding to a known public key. Nevertheless, for the person who knows how to factor that number, it is easy to create a pair of keys.

This model was supposed to replace completely single key cryptography. Nevertheless, the calculations needed for this public key cryptography are very expensive and make processing very slow, when compared to the symmetric model. Therefore, instead of ciphering a huge clear text with asymmetric cryptography the common rule was to cipher only a small text (128 or 256 bits, for instance, because this is the size of the key often used in symmetric cryptography). That means that usually asymmetric cryptography is used to cipher only the key that is going to be used. The ciphering of a full clear text is performed by symmetric cryptography and this model is widely known as hybrid cryptographic system.

Actually, asymmetric cryptography performs other roles, such as digital signature. Nevertheless, for the scope of this paper, public key cryptography will be considered only or key production.

Therefore, since asymmetric cryptography performs the role of key producer for the symmetric model, cryptanalysts have changed their focus for another possibility: instead of using brute force in the search for the key using the ciphered text, now it was also possible to apply brute force in the asymmetric algorithm that generated the key, due to the intrinsic unresolved mathematical problems. Nevertheless, it was also realized that this task was also not feasible for the computational power now available. This means that based on the data available I the public key and without the efficient solution of the mathematical algorithms, the most powerful computers available nowadays would take such an absurdly long time to find the key that we could consider this task not feasible.

QUANTUM COMPUTING

“The final goal of quantum computing is to build a computer that is unthinkably faster than the currently available computers” [4].

Classic computer used daily can be thought of, in a very simplified way, as machines able to read input coded as zeros and ones, performing calculations and generating outputs also coded as zeros and ones. Those zeros and ones can be physically represented and a low voltage state (the zero bit) of a high voltage state (the one bit). The basis of classic computing is the common sense notion that a low potential state and a high potential state are mutually exclusive, so both of them cannot occur simultaneously.

This way, two consecutive and independent operations on those bits will happen normally, through two consecutive logic steps.

In quantum mechanics, on the other hand, this common sense does not apply. There are no classic states, but quantum states to which are associated a probability distribution which indicates the chances of finding each possible value when measuring.

In quantum computing the bit is replaced by the *q-bit* and the values 0 and 1 of a bit are replaced by vectors, which are here represented in a nomenclature known as Dirac notation [5]:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

It is usual to represent the probability function of a generic *q-bit* $|\psi\rangle$ as a linear combination of the vectors $|0\rangle$ and $|1\rangle$, such as [5]:

$|\psi\rangle = m\,|0\rangle + p\,|1\rangle$, where m and p are complex numbers.

The most important thing in the physical interpretation of the *q-bit*, is the fact that it can be simultaneously in states $|0\rangle$ and $|1\rangle$.

That is the major difference between the classic and quantum views. In the classic world, different states cannot coexist simultaneously, but in the quantum world they can, and this coexistence is called superposition. Therefore, those independent operations that had to be done step by step in class computing now can be done simultaneously in quantum computing, in a single step inside that specific quantum state.

We can make an analogy [4]: imagine a car moving along a street that has two choices: turn left or keep on straight. In the classic world, it cannot do both simultaneously. Nevertheless, a “quantum car” would be able to do both actions at the same time. In this theoretical example, two new versions of this “quantum car” would have been generated. Each of these versions would again be able to make a choice between the two options, generating two new versions and so on. The question is: would each of these versions of the “quantum car” be able to run an errand? Quantum computing answers affirmatively and so it is possible to execute an exponentially big list of errands, even if we cannot get the results of them separately.

With this example in mind it is easy to understand the purpose of quantum computing stated at the beginning of this section: to find a computer “unthinkably” faster.

Nevertheless, there are some obstacles to overcome, that make the way to quantum computing a long one. As seen, there is a physical reality in the classic mechanic that is intrinsic to the phenomena and is independent of the observer. In quantum mechanics, the opposite is true. There is no intrinsic reality in quantum computing, but a superposition of possibilities that, when measured by an observer, will collapse into one of those states. Therefore, after performing the execution of the simultaneous calculations, when extracting the desired information, there is the possibility that the result achieved is not the correct one.

With the advancement of nanotechnology, overcoming the sensitivity of these systems to external interferences and the discovery of new materials and processes, it is expected that, as soon as science allows for it, a quantum computer as commercial item will be manufactured.

PROTOCOL BB84

Different from what one may expect, Quantum Key Distribution is not a type of cryptography that must be used in quantum computers.

While the quantum computer does not exist as a commercial product, Quantum Key Distribution has already established its communication protocols and has already been used publicly.

Actually, Quantum Key Distribution as known today does not need a quantum computer. It uses only a quantum and a classic communication channel.

By quantum communication channel, one can understand fiber optics, for instance, that allow for the transmission of photons, the particles that make up light. And for classic communication channel, one can understand communication through any other channel, such as e-mail, radio waves, etc.

In Quantum Mechanics it is possible to establish the concept of a single photon polarization, in a binary way. Therefore, the light polarization can be understood as a quantum property that can be represented as a vector in the bi-dimensional space, as shown is the orthogonal axis of **Figure 1**.

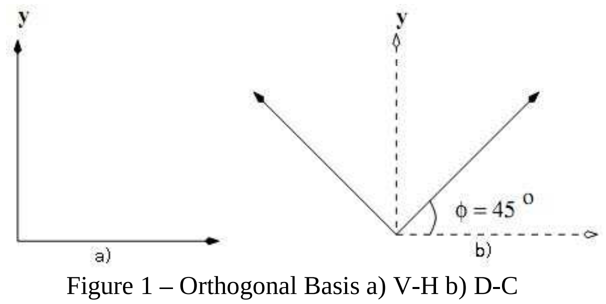

Figure 1 – Orthogonal Basis a) V-H b) D-C

Long before the relatively recent articles on quantum cryptography that present the subject in a broader way, such as [6] and [7], the first protocol on Quantum Key Distribution, called BB84, came to life in the eighties. The letters in the acronym reference the names of Charles H. Bennett and Gilles Brassard and the number, the year of 1984 [8].

The protocol BB84 is based on the fact that an emitter transmits, in a quantum channel, polarized photons to a receiver. Besides, the two orthogonal based defined in **Figure 1** will be used for the emission and reception of these photons, that is, the V-H (Vertical-Horizontal) and D-C (Diagonal-Counter diagonal) orthogonal bases.

In order to make it easier for the reader to understand and without any loss of correction, we will use a graphic symbolism to define the photon polarization possibilities, replacing the classical Dirac notation. It is a graphic symbolism similar to the one used by Brassard and Bennet. In the same lie, within the scope of this paper, polarized photon and *q-bit* will mean the same thing.

Using base V-H, there are two possible polarizations (or *q-bits)*: → ↑

The first one defines the direction established when $\phi = 0$; and the second $\phi = \pi/2$

Using base D-C, there are two possible polarizations (or *q-bits*): ↗ ↖

The first one defines the direction established when $\phi = \pi/4$; and the second, $\phi = 3\pi/4$.

We can summarize, then, the following conclusions verified through a communication in a quantum state:

1. The emitter can polarize a photon based on bases V-H and D-C, in four different positions:

   ↑ → ↗ ↖

2. The receiver, to capture the photon, uses one of the following bases:
   - (V-H)
   - (D-C)
3. If the receiver used base V-H, there are four possibilities:

   a) If the polarized photon is ↑ , it is captured exactly as ↑;

   b) If the polarized photon is → , it is captured exactly as →;

   c) If the polarized photon is ↗ , this information is lost and the photon is captured as ↑ or → , with probability ½ for capturing ↑ and ½ for capturing →;

   d) If the polarized photon is ↖ , this information is lost and the photon is captured as ↑ or → , with probability ½ for capturing ↑ and ½ for capturing →;

4. If the receiver used base D-C, there are four possibilities:
   a) If the polarized photon is ↗ , it is captured exactly as ↗;
   b) If the polarized photon is ↖, it is captured exactly as ↖;
   c) If the polarized photon is ↑, this information is lost and the photon is captured as ↗ or ↖ , , with probability ½ for capturing ↗ and ½ for ↖;
   d) If the polarized photon is →, this information is lost and the photon is captured as ↗ or ↖ , with probability ½ for capturing ↗ and ½ for ↖.

It must be pointed out that the choice of two orthogonal bases being such that D-C is a rotation of exactly π/4 in relation to V-H, assures the probabilities of ½ for each of the possible capturing with the wrong base (items 3.c, 3.d, 4.c and 4.d).

In pursuit of the protocol, emitter and receiver must establish a binary convention, as illustrated by **TABLE 2**

TABLE 2 –BINARY CONVENTION

BASIS	0	1
V-H	↑	→
D-C	↗	↖

Therefore, one can imagine a hypothetical situation that Will result in the following steps according to protocol BB84:

1st. part: Protocol BB84 – Quantum channel

a) The emitter intends to send the following message:

0 1 1 0 1 0 0 1

b) The emitter chooses randomly 8 bases do code the polarized photons:

V-H V-H D-C V-H D-C D-C D-C V-H

c) Polarizations generates 8 *q-bits*, that are sent to the receiver:

↑ → ↖ ↑ ↖ ↗ ↗ →

d) The receiver, in order to capture the *q-bits*, chooses randomly a basis sequence::

V-H D-C D-C D-C V-H D-C V-H V-H

e) An example of the reading made by the receiver(capture of the polarized photons) may be the following:

1st	**2nd**	**3rd**	**4th**	**5th**	**6th**	**7th**	**8th**
↑	↖	↖	↗	→	↗	↑	→

- It is important to observe that the 1st, 3rd, 6th, e 8th *q-bits* are necessarily captured that way, because the emitter and received bases are equal;
- The 2nd *q-bit* could have been captured either as ↖ or ↗ (½ probability each);
- The 4th *q-bit* could have been captured either as ↖ or ↗ (½ probability each);
- The 5th *q-bit* could have been captured either as ↑ or → (½ probability each);
- The 7th *q-bit* could have been captured either as ↑ or → (½ probability each);

2nd. part: Protocol BB84 – Classic channel

a) When using a classic channel, the emitter and receiver inform the basis sequence they used, so that the second line presents the basis used by the emitter and the third, those used by the receiver.

1st	**2nd**	**3rd**	**4th**	**5th**	**6th**	**7th**	**8th**
V-H	**V-H**	**D-C**	**V-H**	**D-C**	**D-C**	**D-C**	**V-H**
V-H	**D-C**	**D-C**	**D-C**	**V-H**	**D-C**	**V-H**	**V-H**

b) Only the *q-bits* coming from the positions were equal Will be considered. In the example, only those coming in the 1st, 3rd, 6th, e 8th positions:

↑ ↖ ↗ →

c) According to binary convention, this becomes the following bit sequence:

0 1 0 1

Receiving the bit sequence (in the example above0101) terminates the protocol BB84.

There are some comments that are important to make it clear the competence and purpose limits of the Quantum Key Distribution:

a) Quantum Key Distribution cannot work without a classic communication channel for the Exchange of information on the used basis;

b) In Quantum Key Distribution there is no clear text, original, to be transmitted and afterwards received and deciphered. Therefore, while restricted to the quantum channel, it is not offered the confidentiality and privacy services (nor it is intended to provide them), here understood as "keeping the secret of the information from everyone, except those authorized to receive the information" [9].

c) It is recommended that the Quantum Key Distribution is used to distribute the key (0101, in the example above), to be used in symmetric cryptography;

d) Since the probability of the receiver using the correct basis to receive a photon is ½, it is convenient to transmit a number of polarized photons that is at least the double the size of the string that originates the desired key.

e) Even if in the public channel there is a leak of the basis used by the emitter and the receiver, this information will not be enough for the interceptor to discover the key, given that there are two options of polarized photons for each coincident pair of basis.

f) Quantum Key Distribution is, therefore, a competitor of the asymmetric cryptography, when the latter is used to transmit the key to be used in conventional ciphering done by symmetric cryptography;

g) Principles of quantum mechanics assure that there will always occur an interference when there is an observation in a quantum state. That means that an invader of a quantum channel will be limited to a certain mathematical probability to use all the correct basis for "perfect capture". In a transmission of 256 polarized photons, this probability would be $1/2^{256}$. Differently from asymmetric cryptography, in which cryptanalysis may arrive to the mathematical function that originated the key, in Quantum Key Distribution cryptanalysis rate of success is as small as desired.

h) Quantum states, as shown in [10] and [11], cannot be cloned from an original emission, and this makes it unfeasible to make a quantum security copy;

The next question to analyze is the eventual interception of the message in a quantum state by an invader. Given the principles of quantum mechanics, the fact will cause, with high probability, the change of the message sent.

Since only the *q-bits* coming from the coincidental pair of basis will compose the key, only the *q-bits* that are sent in those positions will interest the analysis.

Analyzing the polarized photons obtained in the positions where emitter and receiver basis coincide and supposing that there was an invasion in the quantum channel by an invader using the same protocol of orthogonal basis as the transmitter, the probability of the intruder not having altered the *q-bit* captured by the receiver for each analyzed photon is ¾, that is, 75%, since:

a) If the invader used the same basis as the emitter, nothing will be changed and the invasion will not be perceived (½ probability);

b) If the invader used a base different from the emitter (½ probability), he will change the photon reception; this altered photon, when captured by the receiver basis (which is the same as the emitter) has ½ probability of returning to the original *q-bit* and, therefore, rendering the intrusion undetectable (½ of ½ = ¼).

c) The total probability of rendering the *q-bit* unchanged is ½ + ¼ = ¾, in spite of the intrusion.

It is important to stress that the higher the number of *q-bits* under analysis, the smaller the chance of not occurring a change by the intruder.

TABLE 3 below shows the probability of not changing the forwarded information.

TABLE 3 –PROBABILITY OF CHANGE NOT HAPPENING

Bits	Probability (unit)		Probability (%)
1	$(3/4)^1$	¾	75%
2	$(3/4)^2$	9/16	56%
3	$(3/4)^3$	27/64	42%
4	$(3/4)^4$	81/256	32%
8	$(3/4)^8$	10^{-1}	10%
16	$(3/4)^{16}$	10^{-2}	1%
32	$(3/4)^{32}$	10^{-4}	0,01%
64	$(3/4)^{64}$	10^{-8}	10^{-6} %
128	$(3/4)^{128}$	10^{-16}	10^{-14} %
256	$(3/4)^{256}$	10^{-32}	10^{-30} %

Table 4 shows an hypothetical quantum transmission with invader intrusion. The polarized photons that are captured by the invasion are, afterwards, forwarded to the receiver.

TABLE 4 – QUANTIC TRANSMISSION WITH INTERCEPTION

Position	1º.	2º.	3º	4º	5º	6º	7º	8º
Bits	0	1	1	0	1	0	0	1
Emitter base	V-H	V-H	D-C	V-H	D-C	D-C	D-C	V-H
Polarized photons	↑	→	↖	↑	↖	↗	↗	→
Invader base	D-C	D-C	V-H	V-H	D-C	D-C	V-H	V-H
Invaded captured photons	↖	↗	→	↑	↖	↗	→	→
Receiver base	V-H	D-C	D-C	D-C	V-H	D-C	V-H	V-H
Received captured photons	→	↗	↖	↗	↑	↗	↑	→
Bits after comparing emitter and receiver bases	1		1			0		1

In the example given above, the basis used coincides only in the 1st, 3rd, 6th and 8th positions. Therefore, only those will be analyzed.

In the 6th and 8th positions, since the invader base in equal to the emitter, there is no change in the receiver capture. Nevertheless, in the 1st and 3rd, where the invader base is different from that of the emitter, there may be a change in the reception, with the probability of change of ½ per polarized photon.

Emitter and receiver can discover the invasion through data check in a classic channel.

Therefore, in order to discover if there was an intrusion, after verifying that the 1st, 3rd, 6th and 8th positions are the ones where there are coinciding basis, emitter and receive inform, also through classic channel, that which are their bits in those positions, that are compared:

Emitter:	0 1 0 1
Receiver:	1 1 0 1

The difference between the bits in the first position assures that there was an error (either invasion or noise in the quantum channel).

Once the invasion is discovered, the whole transmission is discarded and another one is begun.

It is evident that by making the conference of the bits themselves through the classic channel the parts expose that string of bits. Therefore, they will verify through the classic channel only part of the string of bits. If they come to the conclusion that there was neither invasion nor significative noise, only that part of the string will be discarded and all the rest of the string (that was not submitted through the classic channel) is used to create the key.

Therefore, it is possible to come to the conclusion that the protocol for Quantum Key Distribution has a final goal of obtaining two strings of bits that can originate a common key used in symmetric cryptography.

The principles of quantum mechanics assure that any intervention by an invader in the quantum channel Will be most likely perceived at both ends of the communication.

Differently from asymmetric cryptography that has a strong mathematical foundation to deal with key distribution but can be broken any given moment, Quantum Key Distribution assures the possibility of an unbreakable key distribution, restricted to the knowledge only of the parts involved.

Therefore, after both parts have obtained the quantum key, there is no other possibility for the cryptanalyst than to resort to brute force in order to understand the symmetric cryptographed ciphered text. That means, therefore, that the process is maximally secure when key supply is concerned.

It must be mentioned that the classic symmetric protocol, used in quantum protocols, is the one that uses same sized keys and message, called *"one-time-pad"*.

Reconciliation of Information and Privacy Amplification

Whatever the quantum channel used to create the key, it will not be perfect. Noise will necessarily cause both parts to come to different results [12].

Assume the following facts that will induce a reconciliation protocol, that is, a procedure whose goal is to achieve identification and correction of errors occurred during quantum transmission through a defined set of steps.

a) Quantum transmission has already happened;

b) Basis comparison was done in a classic channel;

c) There is a string of bits S(e) at the emitter and a string of bits S(r) was obtained by the receiver.

The question that arises is: how, given S(e) and S(r), can we come to a final string Sf(e) at the emitter that has maximum probability of being equal to Sf(r) at the receiver, correcting for eventual transmission noise?

A reconciliation protocol was presented by G. Brassard e L. Salvail, in [13]. Known as *Cascade Protocol*, it is a procedure verified in classic channel, that continues in "Quantum Key Distribution" [14].

Now we outline a purely theoretical procedure that is an example and whose purpose is merely to allow the reader to visualize the logic of what is intended:

1.Through classic channel, emitter and receiver Exchange the following information:

a)*k* and *i*, size and position of a block of bits from the *strings S(e)* and *S(r)* that will be analyzed.;

b) The bits in that block are compared.

Therefore, assuming a string S of 1000 bits and a size k=100, starting from position i=145, the 100 consecutive bits starting from position 145 will be informed through classic channel.

2.Emitter and receiver come to the conclusion of the percentual amount of errors *p* in that block calculating:

a)*p* = number of bits that are different in that block;

b) Assume that 10 errors are found. In that case, *p* = 10/100 =1/10;

3.The block under analysis is totally discarded, remaining a new string *S'*. The new string S' is formed by the 144 bits before position 145 and by all the bits posterior to position 245, inclusive. (245 = 145 + 100). It is clear that the emitter Will have a new string *S'(e)* and the receiver will also have a new string *S'(r)*;

4.The purpose of finding *p* is to know in how many block the string S' can be broken so that there will **probably** remain on error per block. Some recommendations that increase the block size (Ko) can be implement. An usual recommendation is:

- *Ko* = 1/p + 1/4p, which means increasing block size by 25%.
- In the example, new strings S'(e) and S'(r), blocks would be therefore be have size *Ko* = 1 / (1/10) + 1 / (4/10) = 10 + 2,5 = 12,5.

5.The next step is for the emitter to divide his string *S'(e)* into blocks of *Ko* size (in our example, this size would be 12 or 13 bits) and verify the bit parity for each block. That means executing a XOR operation for all bits in each block which will result in a parity bit.

6.Through this public channel these parity bits (and only them) are transmitted to the receiver that can then verify each block of his string *S'(r)*.

7.Once this verification is performed by the emitter, he also divides his string S'(r) into blocks of size Ko bits and also calculates the parity bits executing a XOR operation over each block. The receiver then verifies each parity bit he calculated with the ones he received from the emitter.

8.Through classic channel, the receiver informs the emitter which parity blocks were not equal. Using this information, the emitter knows which blocks have errors and subdivides these blocks into two halves. He then performs the XOR operation over each block, calculating new parity bits that are then informed to the receiver once again over classic channel.

9. The process is repeated until each block consists of a single bit whose error will be identified and corrected. It should be clear now why the initial size Ko was calculated using the percent of errors found in the sample – this intend to assure that we will probably have a single error per block, in order to isolate this error at the end of the process of subdivisions so that he can be corrected.

10. Since every exchange of parity bits was done publicly through a classic channel, the protocol recommends that the last bit of every analyzed block be discarded.

11. At the end of the procedure, emitter and receiver have their respective strings made of blocks with the same parity.

12. Since the parity identity does not assure that the blocks are exactly equal (0110 and 1001 are different blocks with the same parity – for they get the same result when the XOR operation is performed over their bits: 0 xor 1 xor 1 xor 0 = 0; and 1 xor 0 xor 0 xor 1 = 0). Therefore, reconciliation can proceed in the following way:

a)A block size K1 is chosen, K1 being double the size of Ko, and the same process is repeated.

b) A block size K2 is chosen, K2 being double the size of Ko, and the same process is repeated.

c)The operation proceed until a block sized *Kn* used in the process is bigger than ¼ of the original *string S'* size.

d) Two additional iterations with sizes close to ¼ of the *string* size are recommended to finalize the procedure.

Since the reconciliation process is all made over classic channel (which is public), many pieces of information on the parity bits may have been captured by an intruder. This way, the intruder may have a lot of information on the transmission.

There are some algorithms known as *Hash functions* [9]. These functions are known for generating a fixed size outline (*hash value*, *message digest*, *digital fingerprint*) based on messages of any size. Hence, they are known as compressing or condensing functions. These functions should comply with the following principles:

a) Resistance to pre-imaging, which means that based on the hash value it is not feasible to find the original message.

b) Non-collision, which means that two different messages cannot generate the same *hash value*.

Privacy amplification, as described in [15] and [16], proposes the application of a Hash function [17] that transforms the whole final string after reconciliation in order to assure message integrity, verifying if the hash value of the emitter, hv(e), is equal to the hash value of the receiver, hv(r).

This action is intended to amplify the privacy of the parts involved in the communication and preclude the knowledge of an eventual intruder from obtaining any useful knowledge through the reconciliation process.

The following schema, extracted and adapted from [18] and shown in **Figure 2**, presents a complete communication process that uses Quantum Key Distribution to exchange keys.

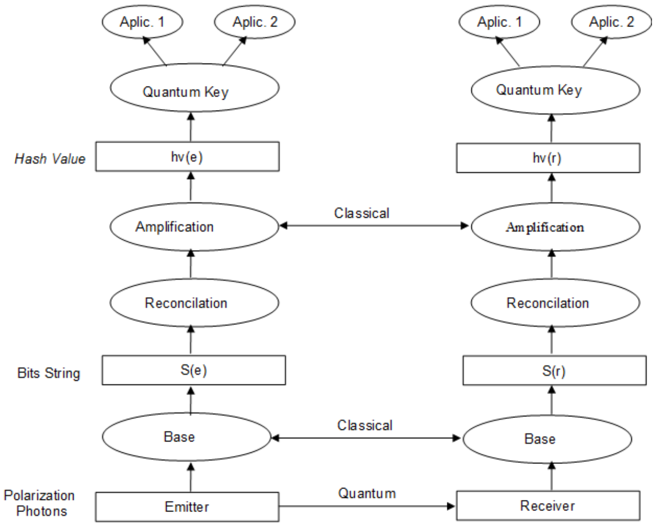

Figure 2 – Complete Process for **Quantum Key Distribution**

There are some important comments on the state of the art of this subject that we find necessary.

Quantum computing has not generated a commercial product (a quantum computer), and is still in a research and development phase. On the other hand, Quantum Key

Distribution has already passed that stage – there are many commercial products available for users [19].

Nowadays, there is a machine that is offered to everyone interested that performs a process of Quantum Key Distribution (*QKD – Quantum Key Distribution*), to be used together with symmetric ciphering systems. Those products can work with fiber optics over a distance close to 100 km.

CONCLUSION

Even if it still may be deemed incipient, specially when compared to classic cryptography, the process of using quantum principle assures the resolution of two crucial aspects in the issue of secret communication.

The first issue concerns the intrusion of unauthorized parts in a transmission. How one can know if the key exchange is not being intercepted and discovered? Which environment can assure that a key informed orally or through mail, radio waves or whatever means will not be intercepted in such a way that the communicating parts will not be made aware of it?

Asymmetric cryptography presented a solution to this problem: the creation of two keys, so that when one is used to cipher, only the other can decipher the message. Everything is informed explicitly, without any concerns on the environment. In this case, the intruder may have access to all public information and yet will not be able to discover the private key. The question that arises is: how long will this process will remain efficient? The creation of a quantum computer may turn the discovery of a private key into a rather easy task, as it will turn procedures that are too complex into feasible and rather ordinary tasks.

Principles of quantum mechanics assure that any observation on a quantum state will necessarily interfere with that state. Therefore, transmissions over quantum channels cannot be passively observed without the legitimate parts being made aware of the observations.

Therefore, if a quantum communication channel is created, even if it does not guarantee that it will not be broken, it will assure that the legitimate parts will be made aware of a possible intrusion.

The second problem concerns the issue of the distribution of cryptographic keys among the legitimate communicating parts. The same quantum channel that informs the intrusion is also capable of supplying the necessary elements for the establishment of a protocol for the generation of cryptographic keys.

This way, given all that was explained in this paper, we may answer the following questions:

a) Is quantum computing necessary for Quantum Key Distribution?

No. While the former has not presented a commercial product (the quantum computer), the latter already operates in commercial scale.

b) Is Quantum Key Distribution a self suficient technique for the ciphering and deciphering of messages?

No, and it does not intend to be. Quantum Key Distribution requires for its operation a quantum communication channel *pari passu* with a regular communication channel.

In spite of most research efforts being directed to keys distribution, there are some studies on quantum cryptography on message ciphering and quantum authentication.

c) What is the goal of Quantum Key Distribution?

Establish a reliable key distribution process among the legitimate communication parts. Quantum Key Distribution intends to define the bits for a key to be used in conventional symmetric cryptography.

d) What does the sentence “Quantum Key Distribution is 100% safe” mean?

It means, basically, two different things: first, that there is a communication channel that is safe against intrusion and espionage (the quantum channel); second, that using this channel, it is possible to create safe cryptographic keys. The next point is quantifying safety. The quantum issue is a matter of probability – therefore the expression “100% safe” means that there is a safety probability as high as the parts wish for.

We can use a hypothetical thought as a complement to the reasoning process exposed here. A cryptographic key with the size of a single bit (either 0 or 1), has a safety probability when face with a brute force attack of ½: 50% safe, meaning that the odds of finding the secret is ½. In no more than two attempts the secret would be exposed. Since the time for each attempt is equal to one time unit, in no more than two time units the secret would be uncovered.

Hence the need for bigger keys. Given the processing and execution time for each attempt by brute force, one can establish tie key size that can be considered as 100% safe.

Therefore, as usually happens with scientific progress, the question of knowledge is time dependent and its results refer to the certainties available at that time frame.

In this work we tried, therefore, to show some aspects of Quantum Key Distribution in an objective and clear way, including its limits and possibilities. We hope to have motivated the scientific spirit that moves the investigation, research and transformation of every scientific area.

REFERENCES

[1] W. Stallings, “Criptografia e segurança de redes – Princípios e práticas”, Pearson Brasil, 4ª edição, pp 21, 2008.

[2] Auguste Kerckhoffs, "La cryptographie militaire," Journal des sciences militaires, vol. IX, pp. 5–83, Jan. 1883, pp. 161–191, Feb. 1883.

[3] Rivest, R; Shamir, A; e Adleman, L. “A method for obtaining digital signatures and public key cryptosystems”. Communications of ACM, fevereiro de 1978.

[4] R. Portugal, “Computação quântica - III Ciclo de Estudos Desafios da Física para o Século XXI: o admirável e o desafiador mundo das

nanotecnologias". Available at the internet address http://www.ihuonline.unisinos.br/index.php?option=com_content&view=article&id=1309&secao=235., Acesso em Março de 2011.

[5] R. Portugal, Uma Introdução à Computação Quântica - São Carlos, SP: SBMAC, 2004.

[6] G. Rigolin, A. A. Rieznik, "Introdução à criptografia quântica". In Revista Brasileira de Ensino de Física, vol. 27, n. 4, p. 517 - 526, 2005.

[7] N. Gisin et al, "Quantum Cryptography", Rev. Mod. Phys, 74, 145, (2002).

[8] C. H. Bennett, G. Brassard, "Quantum Cryptography: Public Key Distribution and Coin Tossin". In International Conference on Computers, Systems & Signal Processing. December, 1984.

[9] A. J. Menezes, P. C. V. Oorschot, S. A. V. "Handbook of Applied Cryptography", CRC Press, Boca Raton, 1996.

[10] W.K. Wootters and W.H. Zurek, A Single Quantum Cannot be Cloned, Nature 299, pp. 802–803, 1982.

[11] D. Dieks, Communication by EPR devices, Physics Letters A, vol. 92(6), pp. 271–272, 1982.

[12] G. V. Assche, J. Cardinal, J. Nicolas, "Reconciliation of a Quantum-Distributed Guassian Key", In proceedings of IEEE Transactions on Information Theory, vol. 50, no. 2, p. 394, 2004.

[13] G. Brassard, L. Salvail. "Secret-key reconciliation by public discussion".In Advances in Cryptology — Eurocrypt 93, Ed. Berlin, Germany: Springer-Verlag, pp 411-423, 1993.

[14] C. H. Bennet, F, Bessete, G. Brassard, L. salvail, J. Smolin, "Experimental Quantum Cryptography". In Journal of Cryptology, Vol. 5, no 3, 1992.

[15] C. H. Bennett, G. Brassard, C. Crépeau, and U. M. Maurer, Generalized privacy amplification,IEEE Transactions on Information Theory, vol. 41, pp. 1915–1923, November, 1995.

[16] C. H. Bennett, G. Brassard; J. M. Robert. "Privacy amplification by public discussion", SIAM J. Comput., vol. 17, no. 2, pp 210-229, 1988.

[17] J. L. Carter, M. N. Wegman. "Universal classes of hash functions", Journal of Compuier and System Sciences, Vol. 18, 1979, pp. 143-154.

[18] A. Mink, S. Frankel and R. Perlner. "Quantum Key Distribution (QKD) and Commodity Security Protocols: Introduction and Integration". In International Journal of Network Security & Its Applications (IJNSA), Vol 1, No 2, 2009.

[19] Idquantique. "A fast and secure solution: high speed encryption combined with quantum key distribution". Available at the internet address http://www.idquantique.com, last access in march/2011.

Alvaro Jorge Braga Mendes has an undergraduate degree in Mathematics and a specialization degree on Mathematics both given by the Federal Fluminense University (UFF). Nowadays he is a technologist at the Cryptology Division at the Naval Systems Analysis Center (CASNAV). He has experience on Applied Mathematics, working mainly on Cryptography and Financial mathematics.

Edésio Hernane Paulicena has an undergraduate degree in Computer Science given by the Federal University of Goiás and a masters degree given in Computers and Electronic Engineering given by the Technological Institute of the Air Force (ITA). He also has a DSc in Telecommunications given by ITA. Nowadays he is a Technologist at CASNAV, where he works at the Cryptology Division. He is experienced in Electrical Engineering, specially in Computer Networks, Cryptography and Information Security.

William Augusto Rodrigues de Souza has a masters degree in Systems and Computer Science given by the Engineering Military Institute (IME) in 2007 and a DSc in Systems and Computers Engineering given by COPPE-UFRJ. He is currently the head of the Cryptology Division at CASNAV. He is experienced in the fields of Computer Science and Applied Mathematics.