Laboratoire d'Informatique Fondamentale, de Recherche Opérationnelle, de Combinatoire et d'Économétrie

Availaible on line at http://www.liforce.usthb.dz
bulletin-liforce@usthb.dz
Bulletin du Laboratoire 02 (2015) 10 - 17

The Set Partitions: Solution for the sharing secret keys

Sadek BOUROUBI[*], **Fella CHARCHALI**[1], **Nesrine BENYAHIA TANI**[2]

[1]Faculty of Mathematics, Laboratory L'IFORCE,
University of Sciences and Technology Houari Boumediene (USTHB),
B.P. 32 El-Alia, Bab-Ezzouar, 16111 Algiers, Algeria.

[2]Algiers 3 University, Laboratory L'IFORCE,
2 Ahmed Waked Street, Dely Brahim, Algiers, Algeria.

Abstract: Confidentiality was and will always remain a critical need in the exchanges either between persons or the official parties. Recently, cryptology has made a jump, from classical form to the quantum one, we talk about quantum cryptography. This theory, although is perfectly safe, there are still binding limits of implementation. In this paper, we developed a new cryptographic protocol, called $BCB12$ protocol, which will be used to provide random keys shared via a classical channel, using the set partitions. Each key can be long enough that the plain text in question, in purpose, for instance, to hide then to transmit the secret information using the Vernam cipher.

Keywords: Cryptography, Vernam Cipher, $BCB12$ Protocol, Set partitions.

[*]Corresponding author: sbouroubi@usthb.dz.

1 Introduction

Issues such as confidentiality and integrity of information have been solved by cryptography. The certificate that the Vernam cipher is unconditionally secure, has transformed the problem to ensure the confidentiality of information to a problem of distribution of the secret key used in the encryption process between two parties. Until the eighties, one way to distribute the secret key, apart from hand to hand, was to use algorithms whose security is based on the computational complexity. The keys generated by such algorithms are reasonably secret but not unconditionally secret.

In the early seventies, Stephen Wiesner wrote conjugate coding [3], describing the basis for a new concept that will be known to the world in the early eighty by quantum cryptography. Cryptography was attached to a quantum concept by the fact it relies on photons to transmit secret information instead of bits. Security is guaranteed not by mathematical theorems, but by the fundamental laws of physics as the Heisenberg uncertainty principle which asserts that certain quantities cannot be measured simultaneously.

Charles H. Bennett (who knew about Wiesner's idea) and Gilles Brassard took the subject in 1984 [4], where they show up to the world the first protocol of quantum key distribution whose security is unconditional because confidentiality is based on impossibilities imposed by the laws of physics [5]. This protocol was implemented in 1989 over a distance of 32 *cm* by calling efforts of F. Bessette, L. Salvail and J. Smolin, a full description of the prototype was published two years later [6].

All Quantum Key Distribution protocols consist of two phases [7]:

1. Initially one of the two parties sends to the other party ”quantum” signals then perform certain measurements.
2. In a second time the two parties engage in classical treatment of measurement results.

2 Concept of unconditional security - Vernam Cipher

The Vernam cryptosystem, also known as the disposable mask or The One Time Pad Cipher, provides perfect security, despite its simplicity. In its classic form, it is nothing but a very long random sequence of letters, written on pages bound together to form a block. The sender uses each letter of the mask in turn to encrypt exactly one plain text character. The Vernam Cipher text C is a function of both the message M and the key K.

The Vernam cipher was invented in 1917 by an engineer of AT & T, Gilbert S. Vernam [9], who thought it would become widely used for automatic encryption and decryption of telegraph messages. The vernam cipher is a polyalphabetic substitution cipher belongs to secret key cryptosystems. The principle of the encryption algorithm is that if a random key is added to a message, the bits of the resulting string are also random and bear no information about the message. If we use binary logic, unlike Vernam who worked with an alphabet of 26 letters, the encryption algorithm E can be written as:

$$E_K(M) = (M_1 \oplus k_1, M_2 \oplus k_2, \ldots, M_n \oplus k_n)\ mod\ 2,$$

where $M = (M_1, M_2, \ldots, M_n)$ is the message to encrypt, and $K = (k_1, k_2, \ldots, k_n)$ is the key consisting of random bits. The message and the key are added bitwise modulo 2, i.e., the exclusive-OR. Decryption process D of cipher text C is the same as encryption, it is given by:

$$M = D_K(C) = (C_1 \oplus k_1, C_2 \oplus k_2, \ldots, C_n \oplus k_n)\ mod\ 2.$$

Perfect security is ensured via the concept of entropy introduced by Shannon in 1949 [1]. Later, Vernam has been used in almost all military concerns. Vernam fits very well to the definition of a secret system [2], a fact confirmed by the following theorem [8]:

Theorem 1 *The Vernam cipher is unconditionally secure for any distribution of plain text.*

But just like any other cryptosystem, it has significant drawbacks which can cause its vulnerability such as the key must be as long as the message to encrypt; the cryptosystem becomes vulnerable if the same key is used more than once and the safest way to transport the key is the diplomatic bag which requires users from the diplomatic sector once.

To remedy major drawbacks of this cipher, we propose here a new protocol, called $BCB12$ "Bouroubi Charchali Benyahia 2012", which is based on the set partitions concept.

3 $BCB12$ Protocol

The protocol $BCB12$ inspired from the quantum protocol $BB84$ "Bennett Brassard 1984" is based on the set partitioning problem which is NP-hard. The expected objective from the protocol is to product and to distribute a secret key via a classical channel, that will be used to ensure confidential communications between the participants by interchanging messages encrypted by the Vernam cipher.

First, the two parties involved in the protocol must share $\pi = \{A_1, A_2, \ldots, A_k\}$, a partition of a set $[n] = \{1, 2, \ldots, n\}$ into k-disjoint blocks (n is assumed to be large enough). Traditional protagonists who must run an exchange of information in cryptography are Alice and Bob. Both are involved in the sending and receiving secret messages and of course Eve, the intruder who wants to spy on Alice and Bob.

Suppose Alice wants to send a message M to Bob, so steps to follow are:

1. Alice calculates the number of characters L_M of the message M to be encrypted.

2. Alice fixes the parameter m such that $m = L_M \times S$, where S is a positive integer called amplification parameter, which we explain the role later.

3. Alice generates randomly a sequence of integers between 1 and n of size m, and for each integer in the sequence, she sets in a list T_A the index of the block to which this element belongs in the partition π.

4. Alice sends the parameter m to Bob.

5. Bob in turn generates a random sequence of integers between 1 and n of size m, and for each integer in the sequence, he sets in a list T_B the index of the block to which this element belongs in the partition π.

6. Bob sends the list T_B to Alice.

7. Alice receives Bob's list, and compares it with hers. If there is correspondence, i.e, for the same index, she locates the same block in both lists, she puts a $" + "$, if not, she puts a $" - "$. Doing so, she creates a new list, said T, whose elements are $" + "$ and $" - "$, then we have:

$$T(i) = \begin{cases} +, & if\ T_A(i) = T_B(i), \\ \\ -, & if\ T_A(i) \neq T_B(i). \end{cases}, \ \forall i = 1, ..., m.$$

8. Alice interprets each $" + "$ as the result of a function f (defined below) chosen from three functions (for example), acting on the elements of the corresponding block. The concatenation of these results provides the random secret key of length L_C.

   To identify f, Alice takes the first $" + "$ in the list T, writes in binary the block index corresponding to this $" + "$, let be j, then she considers the two first bit to the right.

   If the bits are:

   i) identical ("00" or "11") then the function f is interpreted as the sum of elements of the block A_j.

   ii) "10" then the function f is interpreted as the product of the elements of the block A_j.

   iii) "01" in this case, the function f is interpreted as the maximum element of the block A_j.

9. Alice compares L_M to L_C. If $L_M \leq L_C$, she sends the encrypted message and the sequence T to Bob, and then, Bob performs step 8 to get the same key. Otherwise, Alice must return to step 2 and, at this level, she can keep the size m or modify it.

Note that the generated keys by the protocol have been proved random, using statistical tests.

Absolute confidentiality requires a sharing of the key parameter π. The parameter π ensures that the resulting key is secret and is not known, only by legitimate users of the protocol. Therefore, the generated key by $BCB12$ offers the privacy of information transmitted, encrypted according to the Vernam cryptosystem, ensuring inability to decrypt what was encrypted by a spy.

4 Illustrative example

In this section, we present an illustrative and didactic example to show how $BCB12$ protocol runs, in order to get random secret keys, which will be used for the plain texts ciphering. The first step, consists to generate a random partition of a set $[n]$ into k-blocks (for any n and k to choose). The second consists of unrolling the $BCB12$ protocol then inject the provided key in Vernam, the adopted cryptosystem, to get out finally with the enciphered text.

Suppose now, Alice wants to send an encrypted message to Bob. We consider first the following shared parameters between them: $n = 20$, $k = 13$ and the partition $\pi =$ {{1}; {5}; {14}; {3}; {10}; {2}; {6, 8, 12}; {13, 18}; {20}; {9, 11}; {15, 16, 19}; {7}; {4, 17}}.

Let be "***it rains take the umbrella***" the secret message. The message written in binary form is:

01001001011101000010000001110010011000010110100101101110011100110010000001110100011000010110101101100101001000
000111010001101000011001010010000001110101011011010110001001110010011001010110110001101100110000100101110,
with length $L_M = 216$. Alice sets the parameter S at 2, it follows that $m = 216 * 2 = 432$.

Alice generates her random sequence:

{12, 1, 4, 8, 10, 13, 16, 18, 18, 11, 13, 16, 15, 7, 4, 16, 8, 1, 13, 5, 17, 10, 2, 14, 7, 19, 11, 3, 16, 8, 1, 13, 6, 18, 10, 2, 15, 7, 11, 3, 15, 8, 20, 12, 4, 17, 9, 1, 14, 6, 18, 11, 3, 15, 6, 19, 11, 3, 16, 8, 4, 16, 9, 1, 13, 6, 18, 10, 2, 15, 7, 19, 2, 5, 17, 9, 2, 14, 6, 18, 12, 3, 16, 9, 20, 13, 5, 17, 9, 2, 14, 6, 19, 11, 3, 15, 7, 19, 12, 4, 16, 8, 19, 11, 4, 17, 9, 2, 14, 6, 18, 11, 3, 16, 8, 1, 13, 5, 17, 9, 2, 14, 6, 17, 9, 1, 13, 8, 20, 12, 4, 16, 8, 20, 12, 5, 17, 9, 1, 13, 5, 16, 8, 1, 13, 5, 17, 9, 2, 14, 6, 18, 10, 3, 15, 7, 18, 10, 2, 14, 7, 19, 11, 3, 15, 7, 19, 11, 3, 14, 6, 18, 9, 1, 13, 17, 11, 3, 2, 15, 8, 20, 16, 8, 20, 13, 5,17, 9, 2, 13, 6, 19, 12, 4, 16, 9, 1, 13, 6, 18, 13, 5, 17, 10, 2, 14, 6, 19, 11, 4, 16, 8, 1, 13, 5, 18, 10, 2, 15, 11, 3, 15, 8, 1, 13, 5, 18, 10, 2, 15, 7, 20, 12, 5, 17, 9, 2, 14, 7, 19, 12, 4, 16, 9, 1, 13, 6, 3, 15, 7, 20, 12, 4, 17, 9, 1, 16, 8, 20, 12, 5, 17, 9, 2, 14, 7, 19, 11, 4, 16, 9, 1, 14, 7, 19, 11, 3, 16, 7, 20, 12, 4, 16, 9, 2, 14, 7, 19, 12, 4, 16, 9, 1, 14, 10, 3, 15, 7, 20, 12, 4, 17, 9, 3, 16, 8, 20, 13, 5, 17, 9, 2, 14, 6, 18, 11, 3, 16, 8, 1, 14, 6, 19, 11, 4, 16, 8, 1, 13, 5, 18, 10, 2, 15, 7, 20, 12, 4, 17, 9, 2, 18, 10, 3, 15, 7, 20, 12, 4, 17, 11, 4, 16, 9, 1, 14, 6, 18, 10, 3, 15, 7, 20, 13, 5, 17, 10, 3, 15, 12, 4, 17, 9, 2, 14, 6, 19, 11, 3, 16, 8, 20, 13, 5, 18, 10, 2, 14, 7, 19, 11, 4, 16, 8, 20, 13, 5, 18, 11, 3, 16, 8, 20, 13, 5, 17, 9, 2, 14, 6, 18, 11, 3, 14, 8, 20, 12, 5, 17, 9, 1, 14, 6, 19, 11, 3, 18, 10, 2, 14}.

For each integer in the sequence, she sets in a list T_A the block index to which this element belongs in the partition π.

T_A = {7, 1, 13, 7, 5, 8, 1, 7, 10, 8, 11, 11, 12, 13, 11, 7, 1, 8, 2, 13, 5, 6, 3, 12, 11, 10, 4, 11, 7, 1, 8, 7, 8, 5, 6, 11, 12, 10, 4, 11, 7, 9, 7, 13, 13, 10, 1, 3, 7, 8, 10, 4, 11, 7, 11, 10, 4, 11, 7, 13, 11, 10, 1, 8, 7, 8, 5, 6, 11, 12, 11, 7, 2, 13, 10, 6, 3, 7, 8, 7, 4, 11, 10, 9, 8, 2, 13, 10, 6, 3, 7, 11, 10, 4, 11, 12, 11, 7, 13, 11, 7, 11, 10, 13, 13, 10, 6, 3, 7, 8, 10, 4, 11, 7, 1, 8, 2, 13, 10, 6, 7, 13, 10, 1, 8, 7, 9, 7, 13, 11, 7, 9, 7, 2, 13, 10, 1, 8, 2, 11, 7, 1, 8, 2, 13, 10, 6, 3, 7, 8, 5, 4, 11, 12, 8, 5, 6, 3, 12, 11, 10, 4, 11, 12, 11, 10, 4, 3, 7, 8, 10, 1, 8, 13, 10, 4, 6, 11, 7, 9, 11, 7, 9, 8, 2, 13, 10, 6, 8, 7, 11, 7, 13, 11, 10, 1, 8, 7, 8, 8, 2, 13, 5, 6, 3, 7, 11, 10, 13, 11, 7, 1, 8, 2, 8, 5, 6, 11, 10, 4, 11, 7, 1, 8, 2, 8, 5, 6, 11, 12, 9, 7, 2, 13, 10, 6, 3, 12, 11, 7, 13, 11, 10, 1, 8, 7, 7, 11, 12, 9, 7, 13 , 13, 10, 1, 11, 7, 9, 7, 2, 13, 10, 6, 3, 12, 11, 10, 13, 11, 10, 1, 3, 12, 11, 10, 4, 11, 12, 9, 7, 13, 11, 10, 6, 3, 12, 11, 7, 13, 11, 10, 1, 3, 5, 4, 11, 12, 9, 9, 13, 13, 10, 4, 11, 7, 9, 8, 2, 13, 10, 6, 3, 7, 8, 10, 4, 11, 7, 7, 1, 3, 7, 11, 10, 13, 11, 7, 1, 8, 2, 8, 5, 6, 11, 12, 9, 7, 13, 13, 10, 6, 8, 5, 4, 11, 12, 9, 7, 13, 13, 10, 13, 11, 10, 1, 3, 7, 8, 5, 4, 11, 12, 9, 8, 2, 13, 5, 4, 11, 7, 13, 13, 10, 6, 3, 7, 11, 10, 7, 11, 7, 9, 8, 2, 8, 5, 6, 3, 12, 11, 10, 13, 11, 7, 9, 8, 2, 8, 10, 4, 11, 7, 9, 8, 2, 13, 10, 6, 3, 7, 8, 10, 4, 3, 7, 9, 7, 2, 13, 10, 1, 3, 7, 11, 10, 4, 8, 5, 6, 3}.

Alice sends m to Bob. Bob in turn generates his random sequence of length m:

{6, 15, 20, 3, 6, 11, 13, 16, 18, 20, 8, 10, 14, 12, 3, 16, 8, 20, 11, 4, 15, 7, 19, 11, 3, 15, 6, 18, 10, 2, 14, 5, 17, 9, 1, 13, 4, 16, 8, 20, 12, 3, 15, 7, 19, 11, 2, 15, 7, 19, 11, 3, 15, 7, 18, 11, 4, 15, 8, 20, 13, 4, 16, 2, 14, 6, 19, 11, 3, 16, 8, 20, 12, 4, 17, 8, 20, 14, 9, 1, 13, 4, 17, 10, 2, 15, 7, 19, 11, 4, 16, 8, 1, 13, 5, 17, 10, 2, 14, 7, 19, 11, 3, 16, 6, 4, 17, 9, 1, 13, 6, 18, 10, 3, 15, 8, 1, 12, 5, 18, 10, 3, 15, 7, 20, 12, 4, 16, 9, 1, 13, 5, 18, 10, 2, 15, 7, 20, 12, 5, 17, 9, 2, 14, 6, 19, 11, 5, 17, 10, 3, 15, 7, 20, 12, 5, 17, 10, 1, 14, 7, 18, 11, 4, 19, 11, 4, 17, 3, 13, 7, 20, 12, 5, 18, 11, 3, 16, 9, 6, 18, 12, 19, 14, 7, 20, 13, 5, 8, 20, 13, 5, 17, 10, 2, 14, 6, 18, 11, 3, 15, 8, 20, 12, 4, 16, 8, 1, 13, 5, 18, 10, 3, 15, 7, 19, 12, 5, 17, 9, 1, 13, 6, 18, 10, 3, 15, 8, 20, 12, 4, 17, 9, 1, 14, 6, 18, 11, 3, 15, 7, 20, 12, 5, 17, 9, 1, 13, 5, 18, 9, 2, 14, 6, 18, 11, 2, 14, 6, 19, 11,3, 16, 8, 20, 12, 4, 17, 9, 1, 13, 5, 17, 10, 2, 15, 7, 19, 10, 3, 14, 7, 19, 11, 3, 15, 7, 20, 12, 4, 16, 9, 1, 13, 5, 17, 9, 1, 13, 5, 17, 9, 1, 13, 6, 17, 10, 2, 14, 6, 18, 11, 3, 15, 7, 20, 12, 4, 16, 8, 20, 13, 6, 18, 10, 3, 15, 7, 19, 11, 3, 15, 8, 19, 12, 4, 17, 10, 2, 15, 7, 19, 11, 3, 15, 7, 20, 11, 4, 17, 8, 1, 13, 5, 17, 9, 2, 14, 6, 18, 10, 2, 15, 7, 19, 10, 2, 14, 6, 19, 11, 3, 16, 8, 20, 12, 4, 16, 8, 1, 12, 4, 17, 10, 1, 15, 7, 18, 11, 3, 15, 7, 20, 12, 4, 16, 8, 20, 13, 5, 17, 9, 1, 13, 6, 17, 9, 1, 13, 5, 18, 10, 3, 15, 7, 19, 11, 3, 14, 7, 19, 12, 4, 17, 8, 1, 13, 5, 17, 10, 2, 14}.

For each integer in the sequence, Bob sets in a list T_B the block index to which this element belongs in the partition π:

T_B = {7, 11, 9, 4, 7, 10, 8, 11, 8, 9, 7, 5, 3, 7, 4, 11, 7, 9, 10, 13, 11, 12, 11, 10, 4, 11, 7, 8, 5, 6, 3, 2, 13, 10, 1, 8, 13, 11, 7, 9, 7, 4, 11, 12, 11, 10, 6, 11, 12, 11, 10, 4, 11, 12, 8, 10, 13, 11, 7, 9, 8, 13, 11, 6, 3, 7, 11, 10, 4, 11, 7, 9, 7, 13, 13, 7, 9, 3, 10, 1, 8, 13, 13, 5, 6, 11, 12, 11, 10, 13, 11, 7, 1, 8, 2, 13, 5, 6, 3, 12, 11, 10, 4, 11, 7, 13, 13, 10, 1, 8, 7, 8, 5, 4, 11, 7, 1, 7, 2, 8, 5, 4, 11, 12, 9, 7, 13, 11, 10, 18, 2, 8, 5, 6, 11, 12, 9, 7, 2, 13, 10, 6, 3, 7, 11, 10, 2, 13, 5, 4, 11, 12, 9, 7, 2, 13, 5, 1, 3, 12, 8, 10, 13, 11, 10, 13, 13, 4, 8, 12, 9, 7, 2, 8, 10, 4, 11, 10, 7, 8, 7, 11, 3, 12, 9, 8, 2, 7, 9, 8, 2, 13, 5, 6, 3, 7, 8, 10, 4, 11, 7, 9, 7, 13, 11, 7, 1, 8, 2, 8, 5, 4, 11, 12, 11, 7, 2, 13, 10, 1, 8, 7, 8, 5, 4, 11, 7, 9, 7, 13, 13, 10, 1, 3, 7, 8, 10, 4, 11, 12, 9, 7, 2, 13, 10, 1, 8, 2, 8, 10, 6, 3, 7, 8, 10, 6, 3, 7, 11, 10, 4, 11, 7, 9, 7, 13, 13, 10, 1, 8, 2, 13, 5, 6, 11, 12, 11, 5, 4, 3, 12, 11, 10, 4, 11, 12, 9, 7, 13, 11, 10, 1, 8, 2, 13, 10, 1, 8, 2, 13, 10, 1, 8, 7, 13, 5, 6, 3, 7, 8, 10, 4, 11, 12, 9, 7, 13, 11, 7, 9, 8, 7, 8, 5, 4, 11, 12, 11, 10, 4, 11, 7, 11, 7, 13, 13, 5, 6, 11, 12, 11, 10, 4, 11, 12, 9, 10, 13, 13, 7, 1, 8, 2, 13, 10, 6, 3, 7, 8, 5, 6, 11, 12, 11, 5, 6, 3, 7, 11, 10, 4, 11, 7, 9, 7, 13, 11, 7, 1, 7, 13, 13, 5, 1, 11, 12, 8, 10, 4, 11, 12, 9, 7, 13, 11, 7, 9, 8, 2, 13, 10, 1, 8, 7, 13, 10, 1, 8, 2, 8, 5, 4, 11, 12, 11, 10, 4, 3, 12, 11, 7, 13, 13, 7, 1, 8, 2, 13, 5, 6, 3}.

The first integer obtained by Alice is 12 which belongs to the block A_7, where the second one is 1 which belongs to the block A_1. While, the first integer obtained by Bob is 6 belongs to the block A_7, and the second is 15 belongs to the block A_{11}, and so on. Since we have the same index for the first integer in both sequences, Alice obtains the first ” + ”. By comparing the second integer obtained in both sequences, we can see that we have not the same index, so Alice put a ” − ” in the second position. Doing so, Alice establishes the list T:

T = {+, -, +, -, -, -, -, -, -, -, -, -, -, - , -, -, -, -, -, -, -, -, -, -, -, -, +, +, -, -, -, -, -, -, -, -, -, -, -, -, -, -, -, -, -, +, -, +, -, +, +, +, -, -, +, +, -, +, +, +, +, -, +, -, +, -, -, -, +, +, +, -, -, -, -, -, -, -, -, -, -, -, -, -, -, -, -, +, -, +, -, -, +, -, +, -, -, +, +, +, -, -, -, -, -, +, -, -, -, -, -, -, -, -, -, -, -, -, -, -, -, -,

-, -,
-, +, -, +, +, +, +, -, +, +, -, -, -, -, -, -, -, -, -, -, -, -, -, -, -, -,-, -,
-, +, -, -, -, -, +, +, +}.

As the first " + " refers to the block A_7 and $7 = 00000111$ in binary form, with the two first bit to the right are 11, then the function f will be the sum of elements of blocks, hence:

$$f(A_7) = 6 + 8 + 12 = 26.$$

Therefore the provided key is:

{26, 50, 21, 50, 31, 50, 21, 21, 20, 26, 31, 3, 50, 26, 1, 5, 26, 20, 31, 5, 21, 20,7, 26, 20, 1,31, 26, 20, 14, 26, 31, 10, 50, 7, 26, 10, 2, 14}.

Here the key is written in binary form, with length $L_C = 312$:

0001101000110010000101010011001000011111001100100001010100010101000101000001101000011111100 00001100110010000110
1000000001000001010001101000010100000111110000010100010101000101000000001110001101000010100 0000000100011111100011
0100001010000000111000011010000111110000101000110010000000111000110100000101000000001000001110.

Since $L_C > L_M$, Alice has to encrypt the message. She gets the following encrypted text:

01010011010001100011010101000000001111110010110110111101101100110001101000110111001111111001101000 01010111001110
10011101010110110101111111001101000110101001101000011101110110011001100010011101100111100001100000000110001.

Then, she sends the list T and the enciphered message to Bob.

Using the list T, Bob gets the key, therefore, he obtains the plain text by performing the Xor operation (operating principle of Vernam) between the enciphered message and the key.

5 Conclusion

Quantum cryptography ensures that the secret key is shared in confidential way and an unauthorized party has not copy, the Vernam Cipher, under restriction of eliminating its major drawbacks mentioned above, offers unconditional security of the encrypted message with assurance that without the possession the encryption key, it is impossible to decipher what has been encrypted. The $BCB12$ protocol, carries out two objectives: The first objective being the production of a random key at least as long as the message to be encrypted with assurance of the synchronization between the transmitter and the receiver. So, the constraint mentioned above will be removed. The second is the inability of a third person, said Eve, to determine the secret key generated in a reasonable time. This is because if Eve intercepts all data exchanged between Alice and Bob, she has no information on the partition π and the secret key. To determine π in order to find the key, Eve is opposite to the following problem: Find all the k-blocks of a set in the following form $[k+i] = \{1, 2, \ldots, k+i\}$ $i = 1, 2, ...$, for each obtained partition, unroll the protocol in order to generate all possible random keys and then, lead a exhaustive key search, to find the right key, which is not feasible in a reasonable time, at least during the lifetime of the shared secret between Alice and Bob.

References

[1] C.E. Shannon. Communication Theory of Secrecy Systems. *Bell System Technical Journal*, Volume: 28 , Page: 657-715, 1949.

[2] D. Stebila. Classical Authenticated Key Exchange and Quantum Cryptography. *Thesis for degree of doctor of philosophy in Combinatorics and Optimisation, University of Waterloo, Ontario, Canada*, 2009.

[3] S. Wiesner. Conjugate coding. *unpublished manuscript circa 1970; subsequently made available in SIGACTNews*, Volume: 15 , Number: 1, Pages: 78-88, 1983.

[4] G. Brassard. A Bibliography of Quantum Cryptography. *Département IRO, Université de Montréal, Canada*, 3 December 1993.

[5] C.H. Bennett, G. Brassard. Quantum Cryptography: Public Key Distribution And Coin Tossing. *Proccedings of the IEEE International Conference on Computers, Systems and Signal Processing*, Pages: 175-179, Bangalore, India, 1984.

[6] C.H. Bennett, F. Bessette, G. Brassard, L. Salvail, J.A. Smolin. Experimental Quantum Cryptography. *Journal of Cryptology*, Volume:5, Numéro: 1 Pages: 3-28, 1992.

[7] E. Karpov, T. Durt, F.V. Berge, N.J. Cerf, T. D'Hondt. Cryptographie Quantique. *Publication de Cryptax, Phase 1, cryptax.vub.ac.be*, 2007-2010.

[8] S. Vaudenay. *A Classical Introduction to Cryptography: Applications for Communications Security.* Swiss Federal Institute of Technologies, Springer Science+Bussiness Media, 2006.

[9] G. S. Vernam. Cipher Printing Systems for Secret Wire and Radio Telegraphic Communications. *J. AIEE 45*, Pages: 109-115, 1926.