A Survey of Key distribution issue: from classical to quantum solution

Rima DJELLAB
Computer Science Department
LAMIE laboratory
University of Hadj Lakhdar
1, rue Boukhlouf Med EL-Hadi, 05000 Batna
Algeria
rima.djellab@gmail.com
Tel: +213 5 49 08 62 53

Mohamed BENMOHAMMED
Computer Science Department
LIRE laboratory
University of Mentouri
BP 325, Route Ain El Bey 25017 Constantine
Algeria
ben_moh123@yahoo.com

Abstract—**Cryptography seems to be the main solution to secure information. In the taxonomy of cryptographic algorithm, two categories are distinguished, symmetric and asymmetric algorithm. In the first case only one key is used to encrypt and decrypt data. Whist the asymmetric algorithm uses a pair of keys, respectively, to encrypt and decrypt data. In both case, the main issue is how to distribute the key in a secure manner so that it can be used for encryption concerns. In this paper, we will review the key distribution issues in both cases (symmetric and asymmetric). We will present the vulnerabilities of classical key distribution techniques and then introduce the concept of quantum key distribution (QKD) by presenting the BB84 standard and some of its applications.**

Keywords-security; cryptography; key distribution; quantum; BB84;

I. INTRODUCTION

Nowadays, many applications like e-commerce, e-banking… need certain level of security that can be achieved using cryptographic mechanisms. Transforming a message M, to another one M', is encryption. By encrypting data only authorized parties can retrieve M from M' knowing special other information, which is the key.

Encrypting data can be achieved either by using symmetric or asymmetric algorithm. When symmetric algorithm is used, the same key is used either for encryption and decryption. Whilst, a pair of public/private keys is needed when an asymmetric algorithm is used. In both cases, we need to distribute encryption key in secure manner between authorized entities.

Many protocols were proposed to resolve the key management issue. In this paper we overview the classical techniques that were proposed in the literature. We will show the vulnerabilities of such protocols, which are based on mathematics. Then, we introduce the quantum key distribution protocol, proposed in 1984 by C.Bennett and G.Brassard. Furthermore, we review the quantum mechanics principles on which this protocol is based. Finally, we conclude with some actual applications of the quantum key distribution.

II. KEY MANAGEMENT ISSUE

For A.J Menzer [1] the key establishment is any process whereby a shared secret key becomes available to two or more parties, for subsequent cryptographic use. In the other hand, he define the key management as the set of processes and mechanisms which support key establishment and the maintenance of ongoing keying relationships between parties, including replacing older keys with new keys as necessary [1].

Ensure the replacement of the old key used between two parties, or more, is extremely important to achieve the forward and backward security.

Indeed, when a new party is about to join a group, the group's key need to be refreshed such that the new party cannot accede data that have been shared before the "join" operation. In the same way, when a party leaves the group, this party is no more allowed to access the data that will be shared between the rest of the group. This is respectively the backward and the forward security.

The shared key can be either symmetric or asymmetric one, depending on the encryption algorithm.

A. Secret key distribution

We recall that a symmetric encryption algorithm, is one within the same key is used either for encryption and decryption. The main issue when using such an algorithm is the establishment of pairewise secret key when each two parties in a group need to communicate securely. If N is the number of parties in the group, then N(N-1)/2 keys is need. It is clear that if N is a large number, it is impracticable to manage the number of needed keys.

The intuitive solution to distribute a secret key is the manual one. Two persons who want to communicate secretly need to meet each other and exchange the secret key, or send it with a trusted one. Nevertheless, the scheme needs a mutual trust to ensure that the key that had been exchanged is the same that will be used to encrypt data. If third part is the one how will transport the key, the security of this later must be ensured in case he is kidnapped or killed [2]. Or that the third part is a trusted one and will not corrupt the key.

Moreover, the solution is not so practical, because it implies travelling of the parts that need to exchange the key.

The second solution for the symmetric key distribution issue is the use of a pre-shared key [2] [3]. Hence, the pre-shared key is used to encrypt the secret key that will be used for the encryption purpose.

Even if the key distribution issue remains for the pre-shared key solution, different enhanced schemes of the later were proposed.

In [3] an enhanced scheme is presented. The aim is to ensure the authentication of protagonist. Thus, when A wants to exchange a secret key with B, A should add a Nonce or signature, and the identity of B to the secret message, then the packet is encrypted with the pre-shared key.

A variant of the same scheme can be found in [4]. It is based on the KDC (Key Distribution Center). Thereby, A addresses a request to the KDC the later response by sending a packet containing two parts, the first one is intended to A and encrypted with the shared key between the KDC and A, the second part is for B and is encrypted with the corresponding key shared between B and the KDC, so that either A cannot decrypt it. A will send this packet to B, and the later should response and confirm that it get the secret key retrieved from the packet received from A.

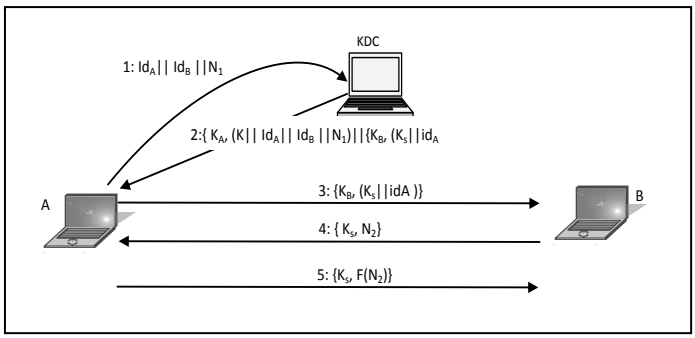

Figure 1. Secret Key Distribution using the KDC.

In fact, using the KDC implies a hierarchy of keys. The keys used between the KDC and the other entities are called master keys; the keys used between the entities to encrypt data are called session keys [4]. The KDC is an important entity in such scheme, if it is corrupted, an opponent can retrieve all the secret key of the group.

Moreover, if the number of the entities is large, the KDC can be a bottleneck, hence, for an extended group it is possible to use more than one KDC where each one is in charge to distribute a secret key in a restricted group of entities [4] [5]. Using such a scheme if the master key of a group is corrupted, the security of the hall group is not jeopardized.

The third scheme for secret key distribution is based on the use of public key (see "Fig.2"). A generates a pair of public/private keys [2] (e.g using the R.A.S algorithm). B can use the public key (K_{pb}) to encrypt the secret key and send it to A. A is the only one how can retrieve the secret key (K) using its private key (K_{pr}).

Such scheme is called digital envelope [3].

The scheme can be adopted when the protagonists do not trust any third part. However, the security of such scheme is based on the calculating power of any opponent. Indeed, if this later can retrieve the private key of A from the public one, then it is easy to intercept the message send from B and read the secret key that A and B want to exchange.

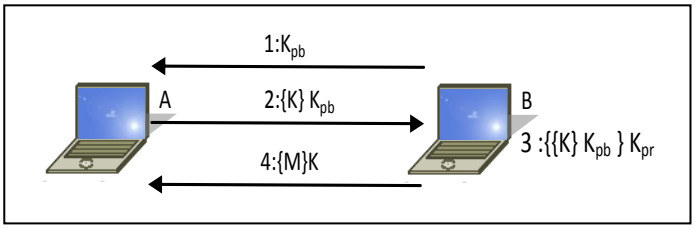

Figure 2. Secret key distribution using digital envelope scheme.

The fourth scheme is a hybrid one. A public key can be used to exchange a master key, then this one can be used to exchange the encryption secret key.

Another scheme based on the collaboration of all entities to generate and distribute a secret key. In [3], a mechanism for agreement is a process that establishes a shared key between entities so that none of them could establish its value in advance. The aim of the agreement mechanism is that every entity contributes to the generation of the secret key. One of the most important schemes is the Diffie-Hellman one. The protocol was proposed in 1976, by Whitfield DIFFIE et Martin E. HELLMAN [6]. The details of the protocol can be found in the RFC 2631. We introduce the principle of the Diffie-Hellman by the following example:

1. Let A and B the entities to share the secret key; P is a prime public number, and W primitive number from Z_p^x / 1<W<P. Let W=7 et P=11 ;
2. A generates random number a / 1 < a < P, Let a=3. B do the same, let b=6 ;
3. A calculates $\alpha = W^a mod\ P = 7^3 mod\ 11 = 2$, B calculates $\beta = W^b mod\ 11 = 7^6 mod\ 11 = 4$;
4. A et B exchange the results of the step before;
5. A calculates $4^3 mod\ 11 = 9$, and B calculates $2^6 mod\ 11 = 9$;
6. A and B have the same results because of the mathematical property :

$$[W^a]^b mod\ P = [W^b]^a mod\ P$$

The security of Diffie-Hellman algorithm is based on the fact that the equivalence class Zn / n∈ N is infinite. So, in the example above, if an opponent intercepts 2, it is impossible to retrieve any information either if the opponent known W and P.

The Diffie-Hellman can be adopted in an extended group of N entities. In this case, N-1 rounds must be performed to exchange the secret key.

The secret sharing technique can also be used to exchange a secret key. The aim idea is to divide a secret into N parts and distribute the parts through different ways. To retrieve the secret, the entities must collaborate. All the entities must

collaborate if a Xor scheme is used, or K entities can collaborate to do it like in Shamir's secret sharing scheme [7] [8] or Blakely one. Shamir's and Blakely's scheme was, independently, proposed in 1979.

B. Public key distribution

In [4] the used techniques to distribute a public key are classified into four categories:

Public announcement;

Publicly available directory;

Public-key Authority;

Public-key Certificates.

The first solution is reliable with the idea of public key encryption. A public key is broadcasted so that every entity has a copy of the key and can use it to encrypt data then send it to the corresponding entity.

In this scheme all entities can send an encrypted data, thing that can lead to saturation. In the other hand, such scheme is not sheltered against mascara attack. An entity X can broadcast a key K_x pretending that it is A's key. B will use K_x to transmit to X the data that should be transmitted to A.

The situation persists till A denounces X.

The second solution, publicly available directory "Fig. 3", is a structure where is specified the identity of each entity and the corresponding public key. The directory is publicly available so that A can retrieve the B's public key from it. A trusted third part is in charge to record the pair (identity, public key) and update the directory each time an entity need to change its public key.

The entity updates its key, when this later was used to encrypt a large amount of data.

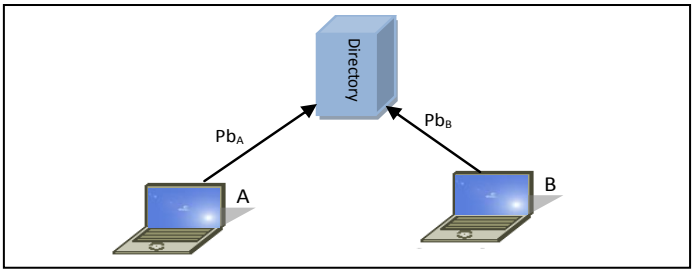

Figure 3. Public key distribution via publicly available directory

The drawback of this scheme is the centralized management. If the central unit is compromised, an opponent can counterfeit public keys and subsequently impersonate any participant and eavesdrop on messages sent to any participant [4].

The third solution is the enhancement of the second one. The third part, C, generates a pair of private/public key. When A wants to communicate with B using a public key encryption, it sends the request to C. C will response by encrypting the public key of B, and other information, using its private key. A can retrieve B's public key when decrypting the message send by C using its public key.

The "Fig. 4" resumes the scheme.

Any entity even corrupted one, which has the C's public key can retrieve B's public key. Moreover, soliciting the central entity can quickly become a bottleneck if the traffic is intense. It is then possible to choose another alternative, the certificates.

Hence, the fourth solution for public key distribution is the public key certificate. Each entity can request a certificate from the certificate entity. Thus, the entity A will not broadcast its public key, but its certificate. When B wants to communicate with A, B retrieve the A's public key from A's certificate. B can also verify the authenticity of A with the certificate authority.

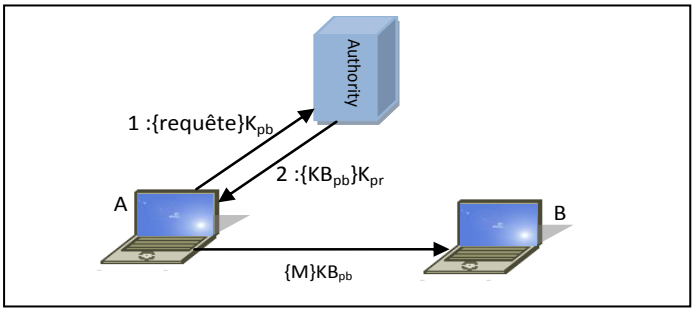

Figure 4. Public key distribution via public-key Authority

III. QUNATUM KEY DISTRIBUTION

It is clear that the classical techniques used to deal with the key distribution issue are based on the trust of a third part, or the complexity of calculation. If any computing power is available to break the algorithms on which is based the classical solution, the security become more vulnerable.

Based on quantum mechanics principals, a new paradigm was proposed to solve the key distribution issue. It is the quantum key distribution.

A. Quantum mechanics principales:

The quantum mechanics is based on four principals:

The superposition;

The non-cloning principal;

The Heisenberg principal;

The entanglement;

1) The superposition:

Classical information is represented on the *bit*. A bit can hold only one value, 0 or 1, whilst the quantum information is represented on the *Qbit*. The qubit can hold both the value 0 and 1 with corresponding probabilities.

Formally, the qubit is represented by the Dirac's function that denotes the state of the qubit :

$$|\psi >= \alpha|1 > +\beta|0 > \quad (1)$$

Where: $|1>=\begin{pmatrix}0\\1\end{pmatrix}$ and $|0>=\begin{pmatrix}1\\0\end{pmatrix}$

α, β are complex numbers, representing, respectively, the probability of getting 1 or 0 after the measurement process, and $|\alpha|+|\beta|=1$.

2) The Non-cloning principal:

The superposition principal stipulates that the qubit holds the values 0 and 1 till the measurement is performed. When measured, the qubit is either 0 or 1. Nevertheless, to clone a stat an adversary should have the initial state $|\psi>$ which is completely destroyed when measured. Hence, it is impossible for him/her copy the initial state so to colon it.

3) The Heisenberg principal:

Because a state of Qubit is a combination of different 'observables' (speed, polarization,…) the Heisenberg principle (known also as the uncertainty principle) stipulates that it is impossible to make, at the same time, a high precise measurement of more than one property (e.g: position and momentum of a moving object). In another words, measuring one of this 'observables' involves inevitably change on the other ones. Such changes are not visible at macroscopic level but so they are at microscopic one.

The Heisenberg principle improves the no-cloning one, since it is impossible to reproduce the initial Qubit's state after the measurement is done, because the observables that constitute the initial state will change inevitably by measurement.

4) The entanglement principal:

The quantum entanglement principle stipulates that, if two photons are entangled, and are distant only a few nanometers or a few kilometers, then the measurement of a photon will inevitably change the state of the second one.

B. The BB84

In this section we introduce the main steps of the BB84.

In 1984, Charles Bennett and Gilles Brassard, based on Wisner's work, proposed a key distribution protocol [9].

The BB84 protocol deals with the public key distribution issue by generating a random and secure binary string after exchanging photons and series of computations applied on the obtained string.

The BB84 used two channels; classical one and quantum one (free air, optical fiber…).

Two polarization bases are used; the diagonal one, noted '×', where '/' represents the polarization 45° and traduced at the measurement by the value 0. The '\' represents the polarization 135° and it is traduced by the value 1. The second polarization base is the rectilinear one '+', it includes also two polarization axes: '-'which is the 0°, and represents the '0' when measurement occurs. The 90° is represented by '|' and traduces the value '1'.

It is possible to distinguish two cases in the BB84 protocol; BB84 with and without eavesdropper.

In the following we will used the names of the traditional protagonists Alice and Bob. Eve is the eavesdropper.

1) Without eavesdropper:

Alice generates a chain of values representing the photons' bases polarizations. Bob choose randomly a polarization base for each received photon trying to measure its polarization.

Bob succeed in 50% of the cases. He will choose the same polarization base used by Alice and so obtain the value send by her.

The case is schematized in Fig 5.

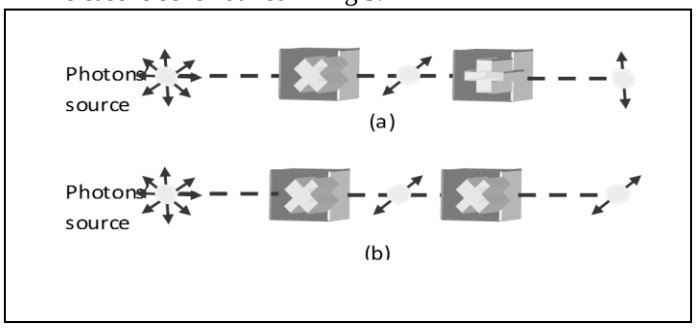

Figure 5. BB84 Protocol case without eavesdropper: in (a) Alice and Bob already choose two different bases, in (b) Alice and Bob choose the same bases[11].

2) With eavesdropper

In the second case, the eavesdropper, Eve, will act the same way as Bob, trying to intercept the photon send by Alice. Eve will succeed in 50% of the case to choose the right polarization. She will send again the photon to Bob who will proceed as in the first case. In this case Bob will succeed in 25% on the case to choose the right polarization base used by Alice.

The following figure resumes this case.

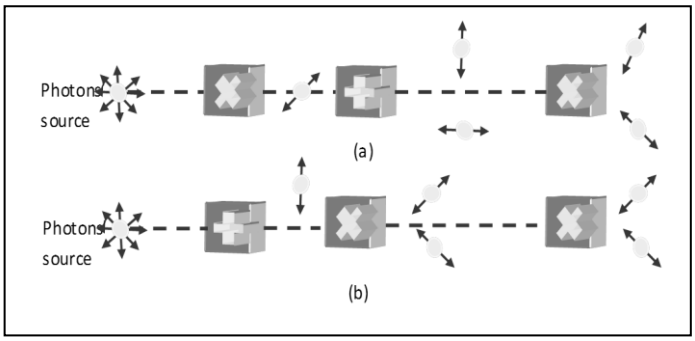

Figure 6. BB84 protocol case with eavesdropper; in the first case (a) Alice and Bob choose the same bases but got different value; the bit will be discarded. In the second case (b) Alice and Bob used already different bases so the bit will be discarded anyway[11].

After the first phase of photons distribution, both Alice and Bob will perform a set of operations.

Firstly, Alice and Bob will exchange the lists of the polarized based that they used respectively when sending and receiving the photons. All non-equal bases will be discarded; hence, the lists will be reduced.

TABLE I. BB84 EAVESDROPPING CASES.

Alice's polarization	-	|	/	/
Bit sent	0	1	0	0
Eve's Base	+	×	+	+
Eve's Polarization	-	/	-	|
Bit measured	0	0	0	1
Bob's base	+	+	+	+
Bob's polarization	-	|	-	|
Bit measured	0	1	0	1

After elimination of the non-equal bases, Alice and Bob will proceed to Error Correction phase using CASCADE algorithm [12]. CASCADE algorithm in the BB84 is based on the dichotomy process. Alice and Bob, both calculate the polarities of the respective chains obtained after elimination of the non-equal bases. They exchange the polarities, if they are different then dichotomy process is run in both sides to detect and correct eventual errors.

The last phase in the BB84 protocol is the amplification. Different methods are proposed for this phase. In [12] Claude Crépeau proposes the use of universal function; in [13] the authors propose the use of binary matrix K of size (n-t)*n, where n is the length of the chains. Alice transmits publicly the matrix K to Bob. The final key (Key_{final}) is calculated in both sides as follows:

$$key_{final}=K.k_{reconciliation} mod2 \quad (2)$$

IV. USING THE QUANTUM KEY DISTRIBUTION

Nowadays, quantum technologies have made a step out of the laboratories. Many companies propose appropriate material to perform such process (e.g : IdQuantique and MagiQ technologies which propose sophisticate material to perform a quantum key distribution, quantum generator for random number…).

In fact, the first quantum key distribution was implemented in 2007 over Lausanne. In 2007, an integration of the quantum key distribution was proposed by T.M.Trang et al. In [13], we have proposed in 2009 a new scheme of the integration of the QKD in the 802.11i standard [10] and in 2012 we enhanced your proposed scheme [11]. In [14] authors proposed a QKD network as an infrastructure allowing the realization of key distribution cryptographic primitive over long distances based on trusted repeaters.

Secret sharing was also inspected. In [15] R. Cleve et al. investigate the concept of the quantum secret sharing. D. Gottesman presented in [16] results on the theory of quantum secret sharing. Later, in 2008, Lian-Fang Han et al, proposed a multiparty quantum secret sharing of secure direct communication (QSSSD) using single photon technique, in the other hand, Tian-yin Wang et al proposed a scheme of multiparty quantum secret sharing of classical messages (QSSCM) also based on the single photon technique, and the unitary operations [17].

V. CONLUSION

Information security is an issue of any information system. Nowadays, many applications need an adequate level of security according to the importance of the transmitted information. In this paper, we have presented the importance of the key distribution issue that can be meet using symmetric or asymmetric algorithm. We overview different techniques used to overcome this issue and discuss the vulnerabilities of such classical technique. Hence, we introduced the quantum key distribution technique, and overview some of the actual application of this technique.

REFERENCES

[1] **Alfred, J. Menezes, Paul C, van Oorschot and Scott A, Vanstone.** *Handbook of Applied cryptography.* s.l. : CRC Press, Août 1996.

[2] **Nigel, Smart.** *Cryptography:An introduction.* s.l. : McGraw Hill, 2009.

[3] **Martin, Bruno.** *Codage, cryptographie et applications.* s.l. : Presses polythechniques et universitaires romandes, 29 avril 2004.

[4] **Willam, Stallings.** *Cryptography and Network security Principles and practice.* New York : Pearson, 2011.

[5] **Ralph C, Merkle.** Protocols for public key cryptosystems. s.l. : IEEE Symp. on Security and Privacy, 1980.

[6] **Whitfield, Diffie and Martine E, Hellman.** Multiuser cryptographic techniques. s.l. : AFIPS '76: Proceedings of the June 7-10, national computer conference and exposition, 1976.

[7] **Adleman, Shamir.** *How to share a secret.* 11, s.l. : Communications of the ACM, 1979, Vol. 22.

[8] **George Robert, Blakley** *Safeguarding cryptographic key*.**.** Arlignton. VA : National Computer Conference, 1979, Vol. 48.

[9] **Charles, Bennett and Brassard, Gilles**, *Quantum cryptography: Public-key distribution and coin tossing.***.** 1984, In proceedings of IEEE International Conference on Computers, Systems and Signal Processing, pp. 175-179.

[10] **Rima, Djellab,** New Scheme Of Integrating Quantum Key Distribution In 802.11i", IEEE ICMCS'09, 2–4 Avril, Ouarzazete, Maroc. Publisher: IEEE, pp. 46-50

[11] **Rima, Djellab and Mohammed, Benmohammed**, *Enhancing 802.11i key distribution using quantum key distribution.***.** 3, 2012, IJARITAC, Vol. 2, pp. 14-22.

[12] **Claude, Crépeau.** *Réconciliation et destillation publique de secret.* 1995.

[13] **Thi Mai Trang, Nguyen, Mohamed Ali, Sfaxi and Solange, Ghernaouti-Hélie,** *802.11i Encryption key distribution using quantum cryptography.* 5, September/October 2006, Journal of Networks, Vol. 1, pp. 9-20.

[14] **R, Alléaume, et al.** *Topological optimization of quantum key distribution networks.* July 2, 2009, New Journal of Physics.

[15] **Richard, Cleve, Daniel, Gottesman and Hoi-Kwong, Lo.** How *to share a quantum secret.* 3, July 19, 1999, PHYSICAL REVIEW LETTERS , Vol. 83, pp. 648-651.

[16] **Daniel, Gottesman.** *On the Theory of Quantum Secret Sharing.* 1999.

[17] **Tian-yin, Wanga, et al.** *An efficient and secure multiparty quantum secret sharing scheme based on single photon.* [ed.] Elsevier. 24, 2008, Optics Communications, Vol. 281, pp. 6130–6134.