**Anbar Journal for Engineering Sciences © AJES / 2007**

# Unconditionally Secure Authentication in Quantum Key Distribution*

**Sufyan T. Faraj**
*College of Computers, University of Anbar, Iraq*
*E-mail: sufyantaih@ieee.org*

**ABSTRACT**
Quantum key distribution (QKD) is a method using some properties of quantum mechanics to create a secret shared cryptographic key even if an eavesdropper has access to unlimited computational power. All QKD protocols require that the parties have access to an authentic channel. Otherwise, QKD is vulnerable to man-in-the-middle attacks. This paper studies QKD from this point of view, emphasizing the necessity and sufficiency of using unconditionally secure authentication in QKD. In this work, a new technique of using unconditionally secure authentication is proposed for quantum cryptosystems. This technique is based on a hybrid of normal application of authentication codes and the so-called "counter-based" authentication method such that to achieve a better trade off between security and efficiency (in terms of the required size of initially shared secret data). Based on this strategy, an authenticated version of a typical QKD protocol (the well-known BB84 protocol) is described. Some advantages of our protocol in comparison to other proposals are also highlighted.

**Keywords: Authentication, Cryptography, Key Distribution, Quantum, Universal Hashing.**

# التوثيق ذو الامنية غير المشترطة في انظمة التوزيع الكمي لمفاتيح التشفير

**سفيان تايه رجب**
**كلية الحاسبات ، جامعة الانبار**

**الخلاصة**
تقوم عملية التوزيع الكمي لمفاتيح التشفير على استخدام خواص الميكانيك الكمي لبناء مفاتيح تشفير أمينة مشتركة حتى في الظروف التي يمتلك فيها المتنصت قدرات حاسوبية غير مقيدة . غير ان كافة بروتوكولات التوزيع الكمي لمفاتيح التشفير تحتاج لوجود قناة علنية كاملة الموثوقية بين الأطراف المخولة للاشتراك في البروتوكول . وبعكسه ستكون هذه المنظومات معرضة لخطر الهجوم من نوع " رجل في الوسط " .
ويقوم هذا البحث بدراسة التوزيع الكمي لمفاتيح التشفير من هذه الناحية وابراز ضرورة وكفاية استخدام طرق التوثيق ذات الأمنية غير المشروطة فيها . تم في هذا العمل اقتراح ستراتيجية هجينة للتوثيق الغير مشروط تناسب تطبيقات التوزيع الكمي لمفاتيح التشفير . وتقوم هذه الستراتيجية على مزيج من الاستخدام الاعتيادي لترميز التوثيق وما يعرف بالتوثيق المعتمد على العداد . وقد حقق هذا توازناً أفضل بين متطلبات الأمنية العالية غير المشروطة وبين كفاءة الاسلوب من حيث تقليل استهلاك عملية التوثيق للبيانات السرية المشتركة .
وبناءً على هذه الستراتيجية تم تقديم نسخة كاملة التوثيق لأحد أشهر بروتوكولات التوزيع الكمي لمفاتيح وهو BB84 . كما تم توضيح الايجابيات التي يقدمها البروتوكول المقترح من قبلنا قياساً الى المقترحات السابقة .

## 1. INTRODUCTION

Data transmission has always been vulnerable to eavesdropping. Conventional cryptography has provided many security services in data communication; however, it has serious limitations when dealing with passive eavesdropping. The recent application of the principles of quantum mechanics to cryptography has led to remarkable new dimension in secret communication [1]. The most important contribution of quantum cryptography or, more precisely, quantum key distribution (QKD) is a mechanism for detecting eavesdropping. This is totally new contribution to the field of cryptography. Neither symmetrical cryptographic systems nor public–key systems have such a capability [2].

The first QKD protocol, called BB84, was invented in 1984 by C. H. Bennett and G. Brassard [3]. Here, the authorized parties (Alice and Bob) have access to two channels: a one-way quantum channel for sending quantum signals and a two-way classical public channel for verification and reconciliation. This is depicted in Figure 1, where eavesdropping actions by Eve (an eavesdropper) are also shown. Also, there are QKD protocols other than BB84.However, there are three main reasons for choosing BB84 in this work. First, it is the first of QKD protocols. Secondly, it is the most widely studied and implemented one. And finally, given that the known laws of quantum physics hold, there is more than one proof on the unconditional security of BB84 [4, 5, 6].

It is well known that QKD requires a classical public channel with trusted integrity as otherwise a potential eavesdropper (Eve) can easily amount a man-in-the-middle attack. In case that Eve can manipulate messages on the public channel, it is clear that she could sit between Alice and Bob impersonating each of them to the other. As a result, Eve would thus share two independent keys with the two the legitimate parties and gain full control of all the subsequent communication, without being noticed [7, 8]. It was suggested that this crucial property of the public channel can be implementation using either of the followings [8]:

**i.** An inherently unjammable public channel.
**ii.** An information–theoretically (i.e., unconditionally) secure authentication scheme to certify that the public messages have not been altered in transit.

It is obvious that the first case above is not feasible for most practical situations. Hence, we left with the second case in which Alice and Bob need to initially share secret information to serve as an authentication key. Subsequently in each QKD session they repeatedly renew the mutual secret by reserving part of the newly generated key. This is used to authenticate communication in the next session. Hence in this case the protocol implements key expansion or key growing rather than key distribution [7, 8, 9].

This paper presents a new hybrid technique for applying unconditionally secure authentication for protecting public channel transmissions in quantum cryptosystems. Based on this technique, an authenticated version of the BB84 protocol is fully described. The rest of the paper is organized as follows: Section 2 describes the basic BB84 protocol. In Section 3, the theoretical aspects of unconditionally secure authentication (or authentication codes) are explained. While certain examples of authentication codes are presented in Section 4. The problems of multiple authentication in QKD are discussed in Section 5, where the proposed authentication strategy is also presented in detail. The authenticated version of the BB84 protocol

based on this strategy is fully described in Section 6. Finally, some important concluding remarks are given in Section 7.

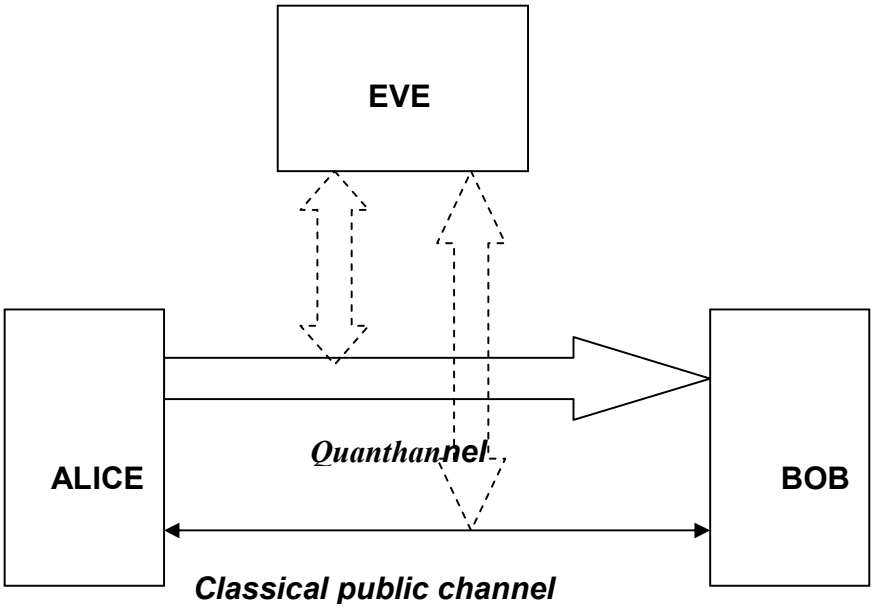


**Fig. 1. A schematic for the basic BB84 QKD setting**

## 2. THE BASIC BB84 PROTOCOL

As mentioned previously, the BB84 protocol is considered in this work. In this section, we give a brief description of the BB84 protocol steps, as it is usually described in the literature (for example, see [1, 2, 8]). It can be noted that only a sketch for solving the authentication problem of QKD is given in most cases. It is really difficult to find an accurate treatment for this problem that considers its practical consequences as a real-life communication protocol. This fact is just what made some researches recently claim that authentication in QKD is "a vital but often neglected part of the method" [9, 10].

Conventionally, the BB84 QKD protocol when extended to a noisy environment, Alice and Bob must adopt the assumption that all errors in raw key are caused by Eve. This is simply because they cannot distinguish between errors caused by noise and errors caused by eavesdropping. Now this protocol can be described in terms of polarization states of a single photon (in fact, it can be described in terms of any other two-state quantum system).

Let Ң be the two dimensional Hilbert space whose elements represent the polarization state of a single photon. The BB84 uses two different orthogonal bases of Ң. Let them be the linear polarization basis which consists of the vertical and horizontal polarization states; and the circular polarization basis consisting of the right and left circular polarization states. It is now possible to compose the required two alphabets. First, the linear polarization quantum alphabet is constructed by interpreting the vertical polarization state as binary "1" and the horizontal

polarization state as binary "0". Then, the circular polarization quantum alphabet is constructed by interpreting the right-circular polarization state as binary "1" and the left-circular polarization state as binary "0".

Keeping in mined that Alice and Bob have access to a one-way quantum channel and two-way public channel (as illustrated in Figure 1), the protocol proceeds as follows:

1- Using the quantum channel, Alice sends a random string of bits. For each bit, she uses randomly with equal probability one of the orthogonal quantum alphabets. For each photon sent by Alice, Bob randomly and independently uses one of the orthogonal polarization bases to perform his measurement (detection). He records his sequence of used bases and measurement results. This is the so-called quantum transmission phase.
2- Using the public channel, Bob announces the measurement operators he used for each of the received pulses. Alice then publicly tells Bob which of his measurements operators were correct. This step is the so-called "sifting" or raw key extraction.
3- Alice and Bob use the public channel to estimate the error rate in raw key. They can select a random sample of raw key, and then publicly they compare these sample bits to obtain an estimate of the so-called quantum bit error rate (QBER).
4- Using the public channel, Alice and Bob apply a reconciliation (i.e. error elimination) procedure to produce an error free common key, called reconciled key. There are many possible procedures for reconciliation, for example, see [8, 11, 12, 13, 14].
5- Alice and Bob now have a common reconciled key which is only partially secret from Eve. They now begin the process of "privacy amplification", which is the extraction of the common final secret key from a partially secret one [8, 15, 16].

## 3. UNCONDITIONALLY SECURE AUTHENTICATION

In spite of that Eve is unable to gain any non negligible information about the final key material from passively monitoring public channel communications, it is essential that these messages are authenticated. Thus, Alice and Bob can verify that they are communicating with each other, and that their public messages have not been altered in transit. This is essential to prevent Eve performing a "man-in-the-middle" attack.

Although using public–key authentication techniques (e.g. digital signatures) for authentication the public channel messages in QKD may still offer some security advantages over traditional (i.e. non QKD-based) approaches [10], this work is dedicated for using unconditionally secure authentication method. It is important to notice that all currently existing unconditionally secure authentication schemes requires an initially shared secret key. The first unconditionally secure authentication scheme was invented about 25 years ago by J.L. Carter and M.N. Wegman who published their discoveries in [17] and [18]. It is commonly referred to as Wegman-Carter authentication.

One important difference between unconditionally secure encryption (the one-time-pad) and unconditionally secure authentication is that with unconditionally secure encryption, the required key needs to be at least as long as the message to be encrypted. This is the main problem with the one-time-pad [8, 9]. Fortunately, Wegman-Carter authentication dose not share this problem. The shared key required is only logarithmic in the size of the message being authenticated. The fact that required keys can be mach shorter than the message to be authenticated is crucial for any QKD protocol. Each round of QKD generates a certain amount of

newly shared secret key bits and requires far more communication which needs to be authenticated. If the key consumed by the authentication process is larger than the generated key, then the process would not be "quantum key expansion" but "quantum key shrinking" which is quite pointless [9].

### 3.1 Authentication Codes

In the usual model for authentication (without secrecy) [19, 20], there are three participants: a transmitter, a receiver, and an opponent. The transmitter wants to send information using a public communication channel. The source state (i.e. a plaintext message) in concatenated with an authenticator (i.e. a tag) to obtain a message (i.e. an encoded message or an authenticated message), which is sent through the channel. An authentication rule (or key) **e** defines the authenticator **e(s)** to be appended to the source state **s**. It is assumed that the transmitter has a key source from which a key is obtained. Before any (authenticated) message is sent, this key is transmitted to the receiver by means of a secure channel. If we compare this model with that of QKD illustrated in Figure 1, we may notice two things. First, in general any of Alice and Bob can be a transmitter or a receiver. This depends on the direction of communication required on the two–way public channel. The second is that the quantum channel is used subsequently for the establishment of the required secret key.

Let us assume that the same key (authentication rule) is used to authenticate up to **w** consecutive source states, where **w** is some fixed positive integer. Also assume that an opponent observes $\mathbf{i} \le \mathbf{w}$ distinct message which are sent using the same key. Suppose the opponent has the ability to introduce messages into the channel and/or to modify existing messages. Assume the opponent places a message $m' = (s', a')$ into the channel by either of these methods, where $m'$ is distinct from the **i** messages already sent. If **e** is the key being used, then the opponent is hoping that $a' = e(s')$. This is sometimes called a spoofing attack of order **i**. The specific case $\mathbf{i} = 0$ and $\mathbf{i} = 1$ have received the most attention. The case $\mathbf{i} = 0$ is called *impersonation*, and the case $\mathbf{i} = 1$ is called *substitution*.

Let $\xi$ be a set of authentication rules. It can be assumed that there is some probability distribution on the source states, which is known to all participants. Given this, the transmitter and receiver choose a probability distribution for $\xi$, called and "authentication strategy". This strategy is also assumed to be known to the opponent. Then, for each $\mathbf{i} \ge 0$, it is possible to calculate $\mathbf{P_{di}}$, which is the probability that the opponent can deceive the transmitter/receiver with a spoofing attack of order **i**. The following theorem gives a lower bound on $\mathbf{P_{di}}$ [20, 21].

*Theorem 1*. Suppose we have an authentication code (without secrecy) with **n** authenticators. Then $\mathbf{P_{di}} \ge \mathbf{1/n}$ for all $\mathbf{i} \ge 0$.

### 3.2 Universal Hashing

Cryptographically secure hash functions are widely used today in cryptography. However, they are only computationally secure, i.e. they can be broken with enough computation power or good enough algorithms (if they exist) [9]. Despite that hash functions cannot be unbreakable, message authentication can. It is important to not that although the fundamental block of unconditionally secure authentication (such as Wegman–Carter scheme) is called universal families (also classes or sets) of hash functions, those hash functions are quite different from the cryptographically

secure hash functions just mentioned above. They have similarities, but the individual hash functions of Wegman–Carter (and other similar schemes) are not, and need not be, cryptographically secure in the classical sense [9].

Since this work is concerned with (unconditionally secure) authentication codes obtained from universal hash families. Some relevant definitions of various types of hash families are recalled from [19, 20, 22] below.

**Definition :**

- An **(N; m, n)** "hash family" is a set $\mathfrak{I}$ of **N** functions such that $f: A \to B$ for each $f \in \mathfrak{I}$, where $|A| = m$ , $|B| = n$ . There will be no loss in generality in assuming **m ≥ n**.

- An **(N; m, n)** hash family is "ϵ-universal" (ϵ-**U**) provided that for any two distinct elements $x_1, x_2 \in \mathbf{A}$ , there exist at most ϵ**N** functions $f \in \mathfrak{I}$ such that $f(x_1) = f(x_2)$.

- Suppose that the functions in an **(N; m, n)** hash family, $\mathfrak{I}$, have range **B = G**, where **G** is an additive abelian group (of order **n**). $\mathfrak{I}$ is called "ϵ-Δ universal" (ϵ-Δ**U**) provided that for any two distinct elements $x_1, x_2 \in \mathbf{A}$, and for any element $y \in \mathbf{G}$, there exit at most ϵ**N** functions $f \in \mathfrak{I}$ such that $f(x_1) - f(x_2) = y$.

- An **(N; m, n)** hash family is "ϵ–almost–strongly–universal" (ϵ–**ASU**) provided that the following two conditions are satisfied :
  - **i.** For any $x \in \mathbf{A}$ and $y \in \mathbf{B}$, there exit exactly **N/n** functions $f \in \mathfrak{I}$ such that $f(x) = y$.
  - **ii.** For any two distinct element $x_1, x_2 \in \mathbf{A}$, and for any two (not necessarily distinct) elements $y_1, y_2 \in \mathbf{B}$, there exist at most ϵ **N/n** functions $f \in \mathfrak{I}$ such that $f(x_i) = y_i$ , $i = 1, 2$.

- An **(N; m, n)** hash family $\mathfrak{I}$ of functions from **A** to **B** is "strongly-universal" (**SU**) provided that, for any two distinct elements $x_1, x_2 \in \mathbf{A}$, and for any two (not necessarily distinct) elements $y_1, y_2 \in \mathbf{B}$, we have

$$\left| \left\{ f \in \mathfrak{I} : f(x_i) = y_i \ , i = 1, 2 \right\} \right| = \frac{N}{n^2} \qquad (1)$$

An **SU** hash family is also called "pairwise independent random variables". It is obvious that a hash family is **SU** if and only if it is $\frac{1}{n} - \mathrm{ASU}$ . For authentication a definition of ϵ–**ASU** is

sufficient, where each function in the family corresponds to a key. In this case, elements of **A** are considered as source states, elements of **B** are considered as authenticators, each hash function gives rise to an authentication rule, and the authentication rules are used with equal probability. The proof of the following theorem is straightforward [19, 20].

*Theorem 2.* If there exists an $\epsilon$–**ASU (N; m, n)** hash family, $\Im$, then there exists an authentication code without secrecy for **m** source states, having **n** authenticators and **N** authentication rules, such that $\mathbf{P_{d_0} = 1/n}$ and $\mathbf{P_{d_1} \leq} \epsilon$.

From Theorem 1, it can be noted that **SU** families achieve the minimum possible deception probability $\mathbf{P_{d_1}}$. Wegman and Cater began with this stronger requirement in [17] but the keys needed to be for too big for authentication to be practical. In [18] they showed that is possible to construct $\epsilon$–**ASU** hash families, having $\epsilon$ a bit larger than **1/n**, that are much smaller than **SU** hash families. This means that by allowing a slightly larger deception probability $\mathbf{P_{d_1}}$, the length of the key required for authentication can be reduced significantly. Since then other papers have used this approach either explicitly or implicitly. Typically, the construction of **ASU** hash families is accomplished by one of two means [20]:

1- Composition of a **U** family and a (smaller) **ASU** family (this is the approach used by Wegman and Carter).
2- Composition of a $\Delta$**U** family with a one–time–pad.

## 4. AUTHENTICATION CODE EXAMPLES

Authentication codes have a level of security that does not depend on any unproven assumptions. In this section, three examples of unconditionally secure authentication schemes are presented. The first is the original Wegman–Cater scheme. While the second, due to M. Peev *et.al.* [7], uses a two step procedure to achieve an efficient authentication for small messages. Finally, a scheme due to R. Taylor [23] uses a similar approach to the first; however, it achieves improved characteristics in terms of both the required key material and authentication computations.

### 4.1 Wegman–Carter Authentication Scheme

In [17], Wegman and Carter proposed several **SU** hash families. Then they proposed one $\epsilon$–**ASU** family (with $\epsilon = 2/\mathbf{n}$) in [18]. In fact, these hash families are by no means unique or most effective. However, they are often referenced because they are the original ones, they are quite easy to understand, and their performance is although not optional, good enough for many applications [9].

This **2/n-ASU** family works by picking several hash functions from a mach smaller but **SU** family and applying them in a hierarchical manner. Assuming the smaller family consists of hash functions mapping bit strings of length $2\lambda$ to bit strings of length $\lambda$, where $\lambda$ is slightly larger than the length of the tag we want to produce. Now, implement the following steps:

- Divide the message into substrings of length **2**$\lambda$, padding the last substring with zero if necessary.
- Pick a hash function from the small family, apply that function to each of the substrings and concatenate the results.
- Repeat until only one substring of length $\lambda$ left, using a new hash function each repetition.
- Discard the most significant bits that won't fit into a tag. What is left is the first tag (with length of **n** bits).

It can be seen that regardless of the size of the plaintext message, each round of hashing halves its length. And this requires using only one hash function, and only one small key to pick that hash function. Thus, the total key length required grows with approximately the logarithm of the plaintext message length. This means that a QKD system can always be designed with large enough rounds to make the key used for authentication acceptably small in comparison of the created shared secret [9].

### 4.2 Peev Authentication Scheme

So far in the literature, the authentication of the public channel in QKD is almost assumed to be exclusively based on the above discussed Wegman–Carter authentication primitive (we call it a primitive because it is independent of the context in which authentication is applied [7]). This includes the choice of the basic intermediate class of **SU** (**2**$\lambda$ to $\lambda$) hash functions. This primitive is suitable for authentication long (plaintext) messages. For example, with authentication tags of 64 bits long and (plaintext) messages longer than 20 000 bits, the message length exceeds the required key length by a factor of four.

Based on the above argument, M. Peev *et. al.* proposed in [7] an authentication primitive that is also efficient for short messages. To achieve this goal, they suggested a two step procedure. In the first step, one maps the initial (plaintext) message from **A** to **Z**, where **Z** is the set of all binary strings of length **r** (**m > r > n**), by means of a single publicly known hash function $\mathbf{f_o}$. While in the second step, one uses a randomly chosen secret **SU** hash function from $\mathbf{H_z}$ mapping **Z** into **B**.

It can be noted that the secret key required for this latter approach is exactly the number of bits needed to index the family $\mathbf{H_z}$. However, the security analysis of this approach is based on certain assumptions that can not always be granted.

### 4.3 Taylor Authentication Scheme

For a given probability of successful attack, the efficiency of an unconditionally secure authentication scheme may be considered in terms of [23]:

- the amount of the required shared secret key,
- the computations required by the sender and receiver, and
- the length of codewords used to convey source messages.

In other words, the characteristics of any authentication code can be studied in terms of the authentication cost and the burdens these schemes place on computational and communications resources. The authentication cost is the number of shared secret bits that need to be sacrificed in order to guarantee that the protocols perform correctly. This cost has a direct impact on the rate at which keys can be generated from a QKD protocol. Unlike the authentication cost, estimation of the required computational and communications resources has no direct effect on the rate of key generation of QKD protocols. However, estimating these resources required to support key generation results in constraints on the rate of key generation for a given set of resources. These costs can be crucial when considering earth to satellite QKD setting [24].

The following authentication scheme, due to R. Taylor in [23] is closely related to that of Wegman–Carter discussed previously. However, in comparison with the Wegman–Carter scheme, the Taylor authentication scheme requires about 1/4 of the required secret key and 1/2 of the authentication computations while maintaining the same codeword lengths.

Let **p** be a prime number. Let a (plaintext) message **M** be divided up into **d**–bit words $\mathbf{m_1}, \mathbf{m_2}, \ldots \mathbf{m_z}$. Let $\mathbf{a_1}, \mathbf{a_2}, \ldots \mathbf{a_{j+2}}$ where $j = \lceil \log_2(z) \rceil$, be integers modulo **p** that form a secret shared key between a sender and receiver. The authentication function **F** of the message **M** (which correspond to **e(s)** and **s** in Sub-section 3.1 respectively) is defined below. The sequences $\mathbf{s_o}, \mathbf{s_1}, \ldots \mathbf{s_j}$ are initialized by $\mathbf{s_o}=\mathbf{M}$ and defined recursively in a way that approximately halves the length of successive $\mathbf{s_i}$. All the arithmetic below is modulo **p**, and the value of **F** is in the range 0 to **p**–1.

$$\mathbf{s_o} = (\ \mathbf{m_1}, \mathbf{m_2}, \mathbf{m_3}, \ldots \mathbf{m_z}\ ) \tag{2}$$

If $\mathbf{s_i} = (\ \mathbf{r_1}, \mathbf{r_2}, \mathbf{r_3}, \ldots\ldots \mathbf{r_t}\ )$, define

$$s_{i+1} = \left\{ \begin{array}{l} \left(a_{i+1}\ r_1 + r_2\ ,\ a_{i+1}\ r_3 + r_4\ , \ldots\ldots,\ a_{i+1}\ r_{t-1} + r_t\ \right) \\ \qquad\qquad\qquad\qquad\qquad\qquad t \text{ even}, \\ \left(a_{i+1}\ r_1 + r_2\ ,\ a_{i+1}\ r_3 + r_4\ , \ldots\ldots,\ a_{i+1}\ r_{t-2} + r_{t-1}\ ,\ r_t\right) \\ \qquad\qquad\qquad\qquad\qquad\qquad t \text{ odd} \end{array} \right\} \tag{3}$$

then $s_j = (\upsilon)$ Let

$$F\left(M, p, a_1, a_2, \ldots\ldots, a_{j+1}, a_{j+2}\right) = a_{j+1}\,\upsilon + a_{j+2} \tag{4}$$

In this scheme, the value of the authentication function **F** is simply appended to the (plaintext) message **M** and sent with it in a way similar to a message authentication code (MAC).

We note that the amount of key required in calculating **F** as at most $(\log_2(z)+3)\log_2(p)$ bits. Also, the calculation of **F** requires at most **z** multiplications modulo **p** and **z** additions modulo **p**. This is probably the fewest number of multiplications possible in any multiplication based scheme. In comparison, Wegman–Carter authentication scheme requires about $4\log_2(z)\log_2(p)$ bits of key, and involves approximately **2z** multiplications and **3z** additions modulo **p** [23]. Thus, the Taylor unconditionally secure authentication scheme has been chosen as an authentication primitive in our work. Finally, it can be noted that the Taylor scheme is identical to a previous construction of D.R. Stinson [19, 20].

## 5. MULTIPLE AUTHENTICATION

In this section, we discuss the situation where we would like to authenticate a sequence of messages with the same key. The authentication schemes discussed in the previous section (Section 4) do not allow us to tag more than one (plaintext) message using the same function (authentication rule), since once Eve knows two message–tag pairs she may be able to determine more such pairs. The definition of $\epsilon$-**ASU** families makes no guarantees about the hardness of such a guess. Therefore the keys must never be reused [9, 18].

To get around this problem, Wegman and Carter suggested in [18] an approach so as to authenticate multiple messages using any $\epsilon$-**ASU** class of hash functions. To apply this technique, the **i** th message in the sequence must be labled with a counter (message number) having the value **i**, $\mathbf{1 \le i \le w}$. This is the so-called counter-based multiple authentication.

### 5.1 Counter–Based Multiple Authentication

Following [20], let $\mathfrak{I}$ be an $\epsilon$-**ASU** **(N; m, n)** hash family, where each function in $\mathfrak{I}$ has domain **A** and range **B**, and suppose it is required to authenticate a sequence of at most **w** source states. Assume that **B** is an abelian group. A key **e** is specified by a function $\mathbf{f} \in \mathfrak{I}$, together with $(\mathbf{w-1})$-tuple $(\mathbf{b_1, \ldots, b_{w-1}}) \in \mathbf{B^{w-1}}$. This $(\mathbf{w-1})$-tuple acts like a sequence of $\mathbf{w-1}$ one–time pads. Let $\mathbf{s_i}$ denotes the **i** th source state in the sequence. The authentication for $(\mathbf{i}, \mathbf{s_i})$ is defined to be:

$$e(i, s_i) = \begin{cases} f(s_i) & \text{if } i = 1 \\ f(s_i) + b_{i-1} & \text{if } 2 \le i \le w \end{cases} \qquad (5)$$

The one–time pad encryption makes it impossible to leak any information about the hash function **f** to Eve. Hence, the hash function **f** can be safely reused an arbitrary number of times as long as new one–time pads are used each time. The following theorem from [20] can be proved in a manner similar to that of [18].

*Theorem3*. Suppose there exists an $\epsilon$-**ASU (N; m, n)** hash family, and let $\mathbf{w \geq 1}$. Then there exists an authentication code without secrecy for **m** source states, having **n** authenticators and $\mathbf{Nn^{w-1}}$ authentication rules, such that $\mathbf{P_{do} = 1/n}$ and $\mathbf{P_{di} \leq \epsilon}$, $\mathbf{1 \leq i \leq w}$.

This counter–based scheme is much more efficient than simply using **w** independent keys, since we need only add $\mathbf{\log_2 n}$ new key bits for each extra message to be authenticated. However, this scheme has some drawbacks (as will be discussed in the next subsection). For example, when a message is lost in transmission, then subsequent messages will not authenticate properly. Thus there in an interest in achieving multiple authentication without counters. In [20], Atici and Stinson had generalized the theory of universal hashing to construct authentication codes that allow the authentication of a sequence of plaintext messages without the use of counters. However, their construction requires considerably more key bits than the counter–based scheme described above.

One additional remark about counter–based multiple authentication is that it can be implemented using the Taylor authentication scheme described previously by simply re–using the same $\mathbf{a_1, a_2, \ldots a_{j+1}}$ and using a new value of $\mathbf{a_{j+2}}$ for each message (see Sub-section 4.3). This gives an average amount of key approaching just one integer modulo **p** or $\mathbf{\log_2(p)}$ bits, per message [23].

### 5.2 The Proposed hybrid Authentication Strategy

The counter–based authentication method makes authentication of a constant stream of messages works fine and requires a minimal amount of previously shared data on condition that completely secret one–time pads are available. However, when the one–time pads are not guaranteed to be totally secret, Eve will learn some thing about the hash function for each massage/tag pair she sees [9].

In QKD, information leakage in the quantum transmission phase is unavoidable. Thus, privacy amplification is used to significantly reduce Eve's knowledge of the key, but not to exactly zero. However, in subsequent QKD sessions, Alice and Bob will start using some of the data obtained from some previous QKD sessions as initial shared information for further authentication. This situation represents counter–based authentication with (not completely) secret key. A possible attack for Eve in this case is to passively eavesdrop the messages and the encrypted tags and combine that information with whatever she knows about the one–time pads until she feels that her information on the used hash function enables her adjust an active attack that succeeds with acceptable probability. This off course does not work in the normal situation when a new hash function is used for authenticating each message.

In [9] two solutions had been suggested to counter feat the above attack scenario on QKD systems that use counter–based authentication. The first solution is for Alice and Bob to have synchronized clocks such that Alice when sends a (plaintext) message waits for enough time interval before sending the tag. Thus, Eve will not forge a message/tag pair before she sees the real tag, but by then it will be too late to change the message. We may note that the use of synchronized clocks is generally assumed in many QKD implementations.

The second solution is based on introducing a new parameter, which is a random fixed–size temporary bit string called the "salt", to the authentication protocol and thus increasing the number of messages need to be exchanged. In this case, when Alice sends a (plaintext) message

to Bob, he responds with the salt, which Eve must not be able to guess before she sees it. Then, Alice calculates a tag based on the concatenation of the (plaintext) message and the salt. She sends the tag to Bob. As the first one, this solution forces Eve to make her attack before she knows that it will succeed. A QKD system might already have similar properties (usually a number of public messages are exchanged, then a single tag is calculated to authenticate some of them after a while), but this highly dependent on the details of the implementation.

Each of the above two solutions has its own drawbacks. They both increase the system complexity and seem to be rather *ad hoc* treatments. More system analysis is required to prove the sufficiency of either of them to other possible attack scenarios on QKD systems using counter–based authentication.

In this work a more fundamental solution is proposed. This solution deals with the problem of using (not completely) secret one–time pads in counter– based authentication, as well as some other possible drawbacks of QKD systems. The solution is based on using a hybrid authentication strategy of normal authentication (where a new hash function is used for each message) and counter–based authentication. The system starts with the counter–based authentication mode. In this mode, a new one–time pad is used for each authenticated message. However, this does not continue indefinitely. Instead, we define a new parameter, $\boldsymbol{\gamma}$ , which represents the number of successive times that the value of the counter, in the counter-based mode, has been changed (i.e. the number of the new one-time pads used). When $\boldsymbol{\gamma}$ reaches a certain pre-defined maximum value, i.e. $\boldsymbol{\gamma = \gamma_{max}}$, the system changes its state from the counter-based authentication mode to the second mode, which is the normal authentication mode.

Now, the system starts using a new hash function for authentication. As soon as this happens, the system automatically returns back to the counter–based authentication mode and the value of $\boldsymbol{\gamma}$ resets to zero. The value of $\boldsymbol{\gamma}$ is increased by one for each a new one–time pad used in this mode until it reaches $\boldsymbol{\gamma_{max}}$, whereby the systems goes to normal authentication mode, and so on. This is illustrated in Figure 2. In both modes, the system uses the Taylor scheme (see Sub-section 4.3) as the unconditionally secure authentication primitive.

It is obvious that $\gamma_{max}$ in this case is a security parameter. Alice and Bob need to agree on a certain value of $\gamma_{max}$ depending on system details (particularly the privacy amplification protocol) and the required level of security. The proposed authentication strategy, defeats possible attacks on quantum cryptosystems that use counter–based authentication by suitably adjusting $\gamma_{max}$ such that Eve never reaches to a situation that she can launch an active attack on authentication tags with any promising probability of success. For a typical situation of grouping (plaintext) messages to 3000-bit blocks, Figure 3 shows a typical effect of the used value of $\gamma_{max}$ on the average amount of the additional authentication cost for each new block. In this figure, the parameters of Taylor scheme were set such that $p=2^{31}-1$ and $d=30$. It is obvious that as the value of $\gamma_{max}$ increases, the average amount of key bits required to authenticate each new message block approaches $log_2(p)$ bits.

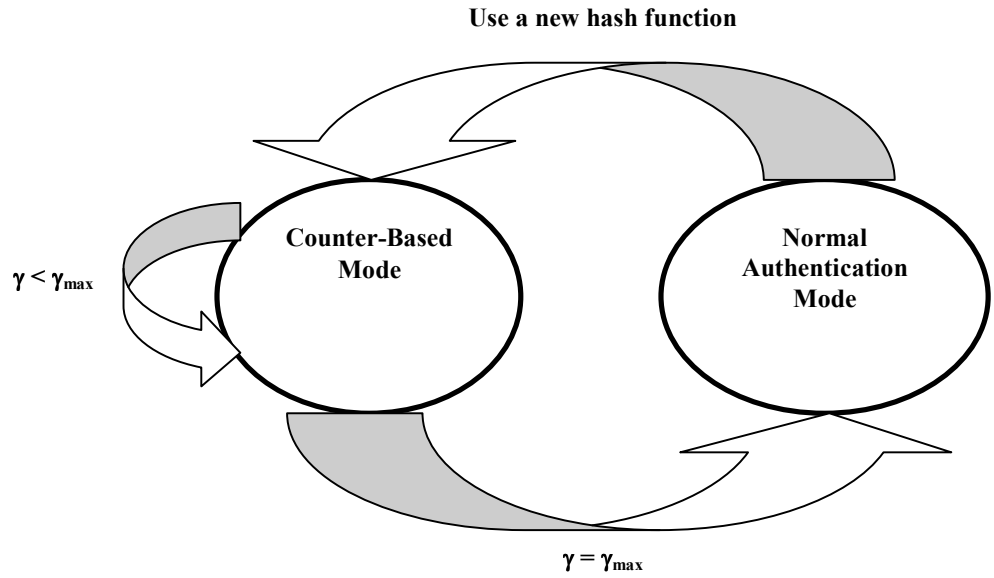


**Fig. 2. A state diagram for the proposed hybrid authentication strategy**

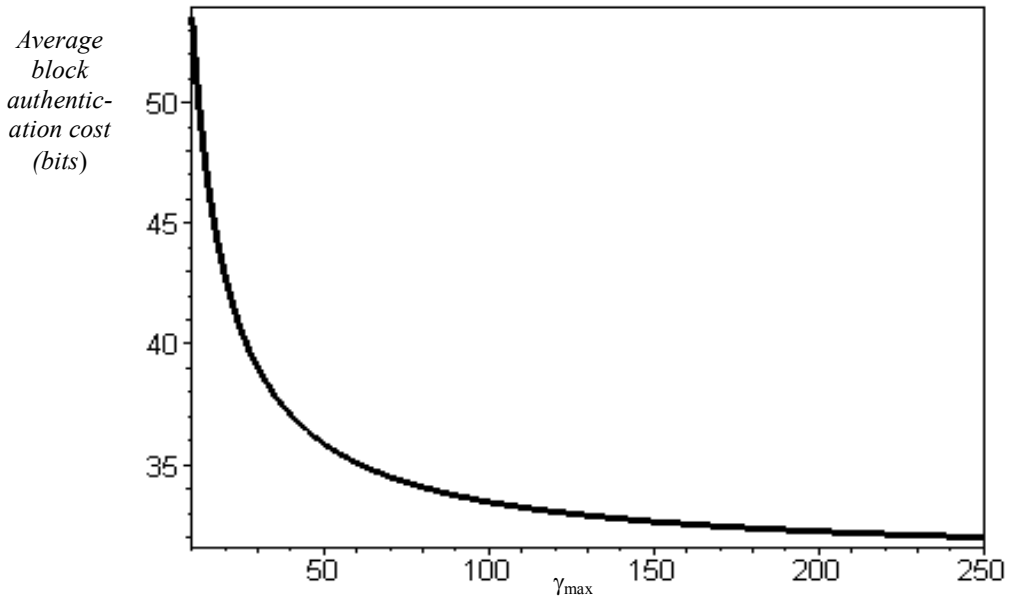


**Fig. 3.** Typical effect of the used value of $\gamma_{max}$ on the average amount of authentication cost of each additional message block (block size=3000 bits, $p=2^{31}-1$, $d=30$).

## 6. THE AUTHENTICATED QKD PROTOCOL

This section describes a specific implementation of the BB84 protocol with all the required unconditionally secure authentication steps. This is on its right is an important achievement of

this work since a description of a complete authentication extracts for QKD protocols are rarely found in enough details in the literature.

It is obvious that quantum channel transition need not to be authenticated because its security is protected by deep physical laws. Instead, the messages exchanged between Alice and Bob on the public channel have to be considered for authentication. Certainly authenticating all public channel messages is an inefficient extreme possibility. In fact, it is not necessary to authenticate all individual messages sent along the public channel. It is sufficient to authenticate some essential steps.

Recalling back the BB84 protocol steps described previously (see Section 2), the following remarks can be made on the essential steps that are necessary and sufficient to authenticate them:

**i-** The quantum transmission phase needs not to be authenticated since it is done using the quantum channel.

**ii-** All sifting phase messages have to be authenticated. This is crucial since otherwise Eve can exchange separate shifted keys with Alice and Bob and then choose only the bits where their all three choice of bases coincide. This is a serious man-in-the-middle-attack situation. In our implementation of the sifting phase, the following two messages have to be authenticated:

- Bob's message indicating the indices of photon pulses detected by him and his choice of basis for each one of them.
- Alice's message indicating her choice of basis for pulses detected by Bob.

Let $\mathbf{N_i}$ be the total number of photon pulses that are initially sent by Alice in quantum transmission phase of a QKD protocol session and let $\mathbf{N_s}$ be the length (in bits) of the obtained sifted key (the string obtained after the sifting phase). Thus, usually $\mathbf{N_i >> N_s}$. Then, the first message requires $\mathbf{N_i}$ bits for indicating whether a photon pulse was detected or not for each pulse index in addition to $\mathbf{2N_s}$ bits for indicating the choice of basis for each detected pulse (this is because that there is on average a probability of 50% that Bob's choice of basis coincides with Alice's). This gives a total of $\mathbf{N_i + 2N_s}$ bits of message length. The second message requires Alice sending $\mathbf{2N_s}$ bits. In order to reduce these messages length, and hence reducing the authentication requirements or cost (i.e., number of exhausted initially shared secret bits), a form of run-length encoding is used. This also reduces the communication overhead in the system.

**iii-** The steps used for the estimation of the QBER (and hence the estimation of Eve's knowledge on the raw quantum transmission) have to be authenticated. Both of the selection of the random subset from the sifted (or raw) key and the process of comparison need authentication. This is particularly important in order not to underestimate Eve's knowledge. It is useful to note that it was stressed elsewhere [25] on the necessity of performing the check on

QBER as the first step of the public discussion, even before obtaining the sifted key. However, the situation here is different, especially that we authenticate all messages of the sifting phase. Thus, there is no security compromise in the sequence of steps mentioned here. Indeed, our approach (of doing authenticated estimation of the QBER after the authenticated sifting) is more efficient in terms of the authentication cost since we deal with the sifted key rather than the initially transmitted block of photon pulses (remembering that $N_s << N_i$ ).

**iv-** There are many techniques for key reconciliation. In our work, two alternatives have been implemented. The first one is a variant of that proposed in [8]. While the second is based on error–correcting codes. Despite the details of implementation of the reconciliation phase, Eve's interaction with system during this phase would not give her additional information about the secret bits; however she could fool Alice and Bob into correcting the wrong set of bits. Thus the reconciliation procedure could actually fail while they think it works. This is can be crucial since it is well understood that the privacy amplification technique cannot work if there is even a one bit error in the reconciled strings. Accordingly, there are at least two possible solutions for the problem of Eve interaction with the system in the reconciliation phase. The first is for Alice and Bob to verify the equivalence of their strings at the end of the reconciliation phase. The second is verifying equivalence of the final keys (i.e. after privacy amplification). As a result, both solutions effectively authenticates the prior communications between Alice and Bob (namely the reconciliation phase communications). This authentication can either be made asymmetric or symmetric.

One good technique for accomplishing the equivalence check is using the set equality tester proposed by Wegman and Carter in [18]. This technique is based on using a hash function from a suitable strongly universal set and its probability of error can be set to be less than a predetermined specified value. However, there is a price that Alice and Bob have to pay for using this equality tester that is they must sacrifice an additional portion of their initially shared secret bits as an index to select the required hash function, to indicate where their strings match, and to authenticate their transmissions. An asymmetric scenario for implementing this equality tester is proposed in [24]. While a symmetric authentication of final key had been proposed in [11]. In this work, a scenario similar to that of [24] is used since symmetric authentication would almost duplicate the authentication cost of this step without an obvious benefit for the application. Also, the equivalence check is done before privacy amplification in order that we can adjust the privacy amplification phase to deal with any (even small) additional information leakage to Eve during the equivalence check step.

**v-** It was noted previously in [24] that the privacy amplification phase need no authentication. This is because that there no need to exchange public messages

for privacy amplification. The trick is that the sifting phase supplies random strings of sufficient length to define the hash index required to implement privacy amplification.

### 6.1 The Proposed Protocol

Remembering the basic BB84 protocol described previously in Section 2, the proposed authenticated version of the protocol consists of the following steps:

**1-** The quantum transmission phase is done as in the basic protocol.

**2-** The messages of the sifting phase have to be authenticated as follows :
- Bob sends one authenticated message indicating the indices of the pulses detected by him and his choice of basis for each one of them.
- Alice responds with an authenticated message indicating her choice of basis for those pulses.

**3-** In order to estimate the QBER, Alice and Bob use authenticated messages for the selection of the required random subset and for performing the comparison process. If the estimated QBER is below a defined maximum value (this value practically is highly dependent on the system implementation details), they continue the procedure. Otherwise, they abort the protocol.

**4-** The key reconciliation (or error–elimination) procedure is done as usually done in the basic protocol.

**5-** Alice and Bob use an asymmetric authentication technique to apply the Wegman–Carter set equality tester for checking the equivalence of their strings after reconciliation. If they find that their strings are equal, they proceed to the privacy amplification stage. Otherwise, the abort the protocol.

**6-** The privacy amplification technique is applied as usual without need for public discussion, and hence no need for further authentication. Thus the final key is obtained.

We note that all messages are authenticated using the unconditionally secure hybrid authentication strategy (proposed in Sub-section 5.2), which is based on an authentication primitive built using the Taylor authentication code (presented in Sub-section 4.3). This protocol is now implemented using software modules written in C++ as a part of the "Quantum cryptography" package. The first version of this package had been developed several years ago to simulate the process of QKD in multiple-access networks [26].

### 6.2 The Protocol Authentication Cost

Here, we will not present a detailed analysis of the authentication cost of the proposed protocol. However, a brief discussion of some important related issues is given. The first issue is that it is necessary to send a sufficient large number of photon pulses during each session of QKD. This would not compensate for system losses only, but it would also results in a better key expansion rate (considering QKD as a quantum key expansion). Indeed, the possibility of aborting more than one QKD session due to denial-of-service attack should be taken into consideration.

The second of these issues is that in accordance to the theory of authentication codes, the probability of Eve's success to forge a tag can be made vanishingly small as desired by increasing the space of hash functions chosen by Alice and Bob. However, this increase of confidence would also increase the size of the set of indices required for selection from this large space of hash functions. Hence, the authentication process, in this case, would need a larger size of shared secret bits to be sacrificed.

Finally, the third issue is concerned with the number of successive times the hybrid authentication primitive stays in counter-based mode before making a transition to normal authentication mode, i.e. the value of $\gamma_{max}$ . As this value increases, the authentication cost decreases and vice versa. However, a higher value of $\gamma_{max}$ may increase the chances of Eve for launching a successful active attack. Thus, there is a number of parameters that have to be adjusted into suitable values to make a trade-off between higher level of security and efficiency.

## 7. CONCLUSIONS

In order to maintain the flavor of unconditional security of QKD, an unconditionally secure authentication primitive has been used in this work for authenticating public channel communications. However, developing an authenticated QKD protocol with a practical level of efficiency was not a trivial task. In addition, there are some proposed applications of QKD (e.g. via satellites) that impose critical limitations on hardware (and software) resources. Thus, building authenticated QKD protocol with low or moderate computational requirements is of high benefit. We believe that our proposed protocol is a good candidate for such missions.

## REFERENCES:

[1] M.I. Khan and M. Sher, "Protocols for secure quantum transmission: a review of recent developments," *Pakistan J. of Information and Technology*, Vol. 2, No. 3, pp. 265-276, 2003.

[2] S.J. Lomonaco, "A quick glance at quantum cryptography," *Dept. of Comput. Sci. and Elect. Engr., Univ. of Maryland Baltimore County*, quant-ph/9811056, Nov. 1998.

[3] C.H. Bennett and G. Brassard, "Quantum cryptography: public key distribution and coin tossing," *International Conference on Computers, Systems and Signal Processing*, India, pp.175-179, Dec. 1984.

[4] D. Mayers, "Unconditional security in quantum cryptography," *J. of the ACM*, Vol. 48, No. 3, pp.351-406, May 2001.

[5] D. Gottesman, H-K. Lo, N. Lutkenhaus, and J. Preskill, "Security of quantum key distribution with imperfect devices," quant-ph/0212066, Dec. 2002.

[6] D. Gottesman and H-K. Lo, "Proof of security of quantum key distribution with two-way classical communications," quant-ph/0105121, Sept. 2005.
[7] M. Peev, M. Nolle, O. Maurhardt, T. Lorunser, M. Suda, A. Poppe, R. Ursin, A. Fedrizzi, and A. Zeilinger, "A novel protocol-authentication algorithm ruling out a man-in-the-middle attack in quantum cryptography," quant-ph/0407131, June 2005.
[8] C.H. Bennett, F. Bessette, G. Brassard, L. Salvail, and J. Smolin, "Experimental quantum cryptography," *J. of Cryptology*, Vol. 5, pp. 3-28, 1991.

[9] J. Cederlof, "Authentication in quantum key growing," *M. Sc. Thesis, Department of Applied Mathematics, Linkopings University*, Sweden, June 2005.
[10] K.G. Paterson, F. Piper, and R. Schack, "Why quantum cryptography," *Department of Mathematics, Univ. of London*, UK, quant-ph/0406147, Sept. 2005.
[11] N. Lutkenhaus, "Estimates for practical quantum cryptography," *Phys. Rev.*, Vol. A59, pp. 3301-3319, 1999.
[12] U.M. Maurer, "Secret key agreement by public discussion from common information," *IEEE Trans. on Information Th.*, Vol. 39, pp. 733-742, 1993.
[13] G. Brassard and L. Salvail, "Secret-key reconciliation by public discussion," *Advances in Cryptology:EUROCRYPT'93*, Lect. Notes in Comput. Sci., Vol. 765, pp. 410-423, 1994.
[14] W.T. Buttler, S.K. Lamoreaux, J.R. Torgerson, G.H. Nickel, C.H. Donahue and C.G. Peterson, "Fast, efficient error reconciliation for quantum cryptography," *Univ. of California, Los Alamos National Lab.*, quant-ph/0203096, Aug. 2005.
[15] C.H. Bennett, G. Brassard, and J-M. Roberts, "Privacy amplification by public discussion," *Siam J. Comput.*, Vol.17, No. 2, pp. 210-229, April 1988.
[16] C.H. Bennett, G. Brassard, C. Crepeau, and U. Maurer, "Generalized privacy amplification," *IEEE Trans. on Information Th.*, Vol. 41, pp. 1915-1923, 1995.
[17] J.L Carter and M.N. Wegman, "Universal classes of hash functions," *J. Comput. and System. Sci.*, Vol. 18, pp. 143-154, 1979.
[18] M.N. Wegman and J.L Carter, "New hash functions and their use in authentication and set equality," *J. Comput. and System. Sci.*, Vol. 22, pp. 256-279, 1981.
[19] D.R. Stinson, "Universal hashing and authentication codes," *Advances in Cryptology-CRYPTO'91*, Lect. Notes in Comput. Sci., Vol. 576, pp. 74-85, 1992.
[20] M. Atici and D.R. Stinson, "Universal hashing and multiple authentication," *Advances in Cryptology-CRYPTO'96*, Lect. Notes in Comput. Sci., Vol. 1109, pp. 16-30, 1996.
[21] J.L. Massey, *Cryptography- a selective survey*, in "Digital Communications", E. Biglieri and G. Prati, eds., North-Holland, pp. 3-21, 1986.
[22] D.R. Stinson, "On the connections between universal hashing, combinatorial designs and error-correcting codes," *Congressus Numerantium*, Vol. 114, 1996.
[23] R. Taylor, "Near optimal unconditionally secure authentication," *EUROCRYPT'94*, Lect. Notes in Comput. Sci., Vol. 950, pp.244-253, 1995.
[24] G. Gilbert and M. Hamrick, "Practical quantum cryptography: a comprehensive analysis (part one)," *MITRE Technical Report*, USA, Sept. 2000.
[25] M. Dusek, O. Haderka, M. Hendrych, and R. Myska, "Quantum identification system," *Phys. Rev.*, Vol. A60, pp. 149-157, 1999.
[26] S.T. Faraj, F. Al-Naima, and S.Y. Ameen, "Quantum cryptographic key distribution in multiple-access networks," *Proceeding of 16th IFIP World Computer Congress*, China, August 2000.