Technical Report No. 2007-530
Quantum Authenticated Key Distribution*

Naya Nagy and Selim G. Akl
School of Computing
Queen's University
Kingston, Ontario K7L 3N6
Canada
Email: {nagy,akl}@cs.queensu.ca

March 12, 2007

Abstract

Quantum key distribution algorithms use a quantum communication channel with quantum information and a classical communication channel for binary information. The classical channel, in all algorithms to date, was required to be authenticated. Moreover, Lomonaco [8] claimed that authentication is not possible using only quantum means. This paper reverses this claim. We design an algorithm for quantum key distribution that does authentication by quantum means only. Although a classical channel is still used, there is no need for the channel to be authenticated. The algorithm relies on two protected public keys to authenticate the communication partner.

Keywords: quantum key distribution, authentication, entanglement

1 Introduction

Most cryptosystems commercially used today, rely on the principles of public key cryptography. Invariably, such cryptosystems aim to offer the means of exchanging secret messages securely and reliably. Suppose two entities, Alice and Bob, want to exchange secret messages; specifically, Bob prepares a secret message to be sent to Alice. Unfortunately, all they have available is a classical insecure communication channel. This means, a malevolent third party, Eve, makes every effort to ruin the secrecy or content of Bob's message. Eve can listen to the communication channel to find out the content of Bob's message. And also, Eve can tamper with Bob's message, adding, deleting or

*This research was supported by the Natural Sciences and Engineering Research Council of Canada

editing parts of the message. The security of the public key cryptosystem relies on the difficulty of inverting particular algebraic functions, also called "one-way" functions.

Secure communication is achieved using two types of keys: a public key and a private key. If Bob wants to send a secret message to Alice, he uses the public key to encrypt the message. Alice then reads the message after using her private key for decoding. There are a few characteristics worth mentioning about the two keys implied in this communication. Alice's private key is secret, and not shared with anybody else. In particular, Bob does not need to know Alice's private key. This is a major advantage, as the private key is never seen on any communication channel and therefore, its secrecy is ensured. The public key is available to anybody. Bob needs to know it, and also an eavesdropper, Eve, has access to it. In order for the protocol to work, the public key is guaranteed to be protected. This means, there is a consensus about the public key value. Both Bob and Alice are sure that they use the correct, same public key. Eve cannot masquerade as Alice and change the value of the public key, making Bob use a false public key to encrypt his message. This feature of the public key is important. Our authentication protocol for quantum key distribution makes use of this property of the protected public key. It is crucial in both the classical sense of authentication protocols, as well as in our protocol, that such a public key can be published with the guarantee that the key *is and remains* protected from masquerading.

The security of the public key distribution protocol relies on the theoretically unproven assumption that factoring large numbers is intractable on classical computers. As described in [7], quantum computers can break some of the best public key cryptosystems.

Quantum cryptography aims to design mechanisms for secret communication with higher security than protocols based on the public key approach. Privacy of a message and its credibility is well satisfied in a private key cryptosystem setting. Alice and Bob share one and the same secret key, k_s. Bob uses the secret key for encryption and Alice consequently decrypts the message with the same key. As long as k_s is unknown to anybody else, the secrecy of the communication is satisfied. There exist various encryption / decryption functions using k_s, such that the encrypted message reveals no information whatsoever about the content of the message, provided the key k_s is unavailable.

Quantum key distribution protocols establish secret keys via insecure quantum and/or classical channels. Existing quantum key distribution algorithms generally use two communication channels between Alice and Bob: a quantum channel which transmits qubits and a classical channel for classical binary information. The classical channel is used to communicate measurement strategy, or the basis for measurement, and to check for eavesdropping.

Quantum key distribution protocols may derive their efficiency from different quantum properties. The first protocol developed by Charles Bennett and Gilles Brassard, known as the BB84 protocol [2], relies on measuring qubits in two different orthonormal bases. The same idea applies to any two nonorthogonal bases [1]. In [5] the quantum key distribution algorithm is derived from the quantum Fourier transform. Based on the property of entanglement, Artur Ekert [4] gave a quantum key distribution solution using entangled qubits to be shared by Alice and Bob. A simpler version with qubits entangled in the same way, namely in the Bell states, is described in [3].

Note that all quantum key distribution algorithms mentioned above require that the classical channel be authenticated. Authentication is supposed to be done by classical means. The authenticated classical channel prevents Eve from masquerading as someone else and tamper with the communication. It was claimed by Lomonaco [8] that authentication is not possible in quantum computation, that for any secure quantum communication a classical authentication scheme needs to be used.

As will be clear from the algorithm described in this paper, authentication of a quantum communication protocol can be done by the quantum protocol itself. The classical channel in our algorithm is not authenticated. Yet Alice and Bob do have an authentication step at the end of the protocol, with the help of protected public keys. Authentication is derived from the quantum algorithm itself and can catch any masquerading over the classical channel.

Shi et. al [9] describe in their paper a quantum key distribution algorithm that does not use a classical channel at all. Authentication is done by a trusted authority, that provides the entangled qubits to Alice and Bob. In our paper, such a trusted authority is not needed. The entangled qubits may come from an insecure source.

The rest of the paper is organized as follows: Section 2 defines entanglement and describes the particular entanglement based on phase incompatibility used by our algorithm. Section 3 describes the algorithm with authentication and security checking. Section 4 concludes the paper and offers some future directions for investigation.

2 Entangled Qubits

The key distribution algorithm we present in the following sections relies on entangled qubits. Alice and Bob, each possess one of a pair of entangled qubits. If one party, say Alice, measures her qubit, Bob's qubit will collapse to the state compatible with Alice's measurement.

The algorithms mentioned above [4, 3, 9], all rely on Bell entangled qubits. The qubit pair is in one of the four Bell states:

$$\frac{1}{\sqrt{2}}(|00\rangle \pm |11\rangle)$$

$$\frac{1}{\sqrt{2}}(|01\rangle \pm |10\rangle)$$

Suppose Alice and Bob share a pair of entangled qubits described by the first Bell state:

$$\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

Alice has the first qubit and Bob has the second. If Alice measures her qubit and sees a 0, then Bob's qubit has collapsed to $|0\rangle$ as well. Bob will measure a 0 with certainty, that is, with probability 1. Again, if Alice measures a 1, Bob will measure a 1 as well, with probability 1. The same scenario happens if Bob is the first to measure his qubit.

Note that any measurement on one qubit of this entanglement collapses the other qubit to a *classical* state. This property is specific to all four Bell states and is then exploited by the key distribution algorithms mentioned above: If Alice measures her qubit, she *knows* what value Bob will measure. The entanglement employed in this paper and algorithm does not have this property directly.

2.1 Entanglement Caused by Phase Incompatibility

Let us look now at an unusual form of entanglement. Consider the following ensemble of two qubits:

$$\phi = \frac{1}{2}(-|00\rangle + |01\rangle + |10\rangle + |11\rangle)$$

The ensemble has all four components, $|00\rangle$, $|01\rangle$, $|10\rangle$, and $|11\rangle$, in its expression. And yet, this ensemble is entangled.

Consider the following proof. Suppose the ensemble ϕ is not entangled. This means ϕ can be written as a scalar product of two independent qubits:

$$\phi = \frac{1}{2}(\alpha_1|0\rangle + \beta_1|1\rangle)(\alpha_2|0\rangle + \beta_2|1\rangle)$$

Matching the coefficients from each base vector, we have the following conditions:

1. $\alpha_1\alpha_2 = -1$
2. $\alpha_1\beta_2 = 1$
3. $\alpha_2\beta_1 = 1$
4. $\beta_1\beta_2 = 1$

The multiplication of conditions 1 and 4 have the result: $\alpha_1\alpha_2\beta_1\beta_2 = -1$. From conditions 2 and 3, we have: $\alpha_1\alpha_2\beta_1\beta_2 = 1$. This is a contradiction. The product $\alpha_1\alpha_2\beta_1\beta_2$ cannot have two values, both $+1$ and -1. It follows that ϕ cannot be decomposed and thus the two qubits are entangled.

The entanglement of the ensemble is caused by the *signs* in front of the four base vector components. Thus, it is not that some vector is missing in the expression of the ensemble, but the phases of the base vectors keep the two qubits entangled.

2.2 Measurement

Let us investigate what happens to the ensemble ϕ, when the entanglement is disrupted through measurement.

If the first qubit q_1 is measured and yields $q_1 = |0\rangle = 0$ then the second qubit collapses to $q_2 = \frac{1}{\sqrt{2}}(-|0\rangle + |1\rangle)$. This is not a classical state, but a simple Hadamard gate transforms q_2 into a classical state. The Hadamard gate is defined by the matrix

$$H = \frac{1}{\sqrt{2}} \left[\begin{array}{cc} -1 & 1 \\ 1 & 1 \end{array} \right]$$

Applying the Hadamard gate to an arbitrary qubit, we have $H(\alpha|0\rangle + \beta|1\rangle) = \alpha\frac{|0\rangle+|1\rangle}{\sqrt{2}} + \beta\frac{|0\rangle-|1\rangle}{\sqrt{2}}$. For our collapsed q_2, we have $H(q_2) = H(\frac{1}{\sqrt{2}}(-|0\rangle + |1\rangle)) = -|1\rangle$. This is a classical 1.

The converse happens when qubit q_1 yields 1 through measurement. In this case q_2 collapses to $q_2 = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. Applying the Hadamard gate transforms q_2 to $H(q_2) = H(\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)) = |0\rangle = 0$. Again this is a classical state 0.

It follows that by using the Hadamard gate, there is a clear correlation between the measured values of the first and second qubit. In particular, they always have opposite values.

A similar scenario can be developed, when the second qubit q_2 is measured first. In this case, the first qubit q_1, transformed by a Hadamard gate, yields the opposite value of q_2.

3 The Algorithm

Alice and Bob wish to establish a secret key, to be used henceforth to encrypt / decrypt messages. One session is required to establish a binary secret key, called *secret*, such that Alice and Bob are in consensus about the value of the secret key. The secret key *secret* consists of n bits, $secret = b_1 b_2 ... b_n$. Technically, to perform the algorithm, Alice and Bob need a classical communication channel, an array of entangled qubit pairs, and two protected public keys.

On the classical channel, classical binary information can be exchanged. The channel is *unprotected* and *not authenticated*. The channel, being unprotected, is sensitive to attacks of eavesdropping: Eve may attempt and successfully read information from the channel. Also, the channel, not being authenticated, is sensitive to masquerading: Eve may disconnect the channel and then talk to Alice pretending she is Bob, and talk to Bob pretending she is Alice.

The array of the entangled qubits has length l, it consists of l qubit pairs denoted $(q_{1A}, q_{1B}), (q_{2A}, q_{2B}), ..., (q_{lA}, q_{lB})$. The array is split between Alice and Bob. Alice receives the first qubit of each entangled qubit pair, namely $q_{1A}, q_{2A}, ..., q_{lA}$, and Bob receives the second half of the qubit pairs, $q_{1B}, q_{2B}, ..., q_{lB}$. The entanglement of the qubit pair is of the type described in the previous section, namely, phase incompatibility. The array of qubits is unprotected either. There is no guarantee that the qubits of a pair are indeed entangled, Eve may have disrupted the entanglement. Also, Eve may have masqueraded as either Alice or Bob, modifying the entangled qubits, such that Alice's qubit is actually entangled with a qubit in Eve's possession rather than Bob's, and the same holds for Bob.

Two public keys are needed by the algorithm. Alice has a public key key_A and Bob has a public key key_B. The two public keys key_A and key_B are independent. These keys are necessary for authentication. They have some characteristics that are different from the classical public keys. The keys are established *during* the computation. They are not known prior to the key distribution algorithm and are defined in value during the computation according to the measured values of some of the qubits. This means that the keys are available *after* the key distribution protocol. Consequently, the keys have to be posted after the algorithm, which is unlike the classical case, where a public key is known in advance. Also, the two public keys key_A and key_B are valid for one session, for one application of the key distribution algorithm. If Alice and Bob want to distribute a second secret key using the same algorithm, they will have to create new public keys, which are different in value from the public keys of the previous session.

The key distribution algorithm, like all quantum key distribution algorithms, develops the value of the secret key during the computation. Implicitly, the values of the public keys as well are developed *during* the computation. There exists no knowledge whatsoever about the values of the keys (secret and public) prior to running the algorithm.

The algorithm follows the steps below:

- **Step 1 - Establish the value of the secret key**

  For each entangled qubit pair (q_{iA}, q_{iB}) in the array, the following actions are taken. On the classical channel, Alice and Bob decide randomly who is going to perform the first measurement.

  Suppose it is Alice. Therefore, Alice measures her qubit q_{iA} thereby collapsing Bob's qubit q_{iB} to the state consistent with Alice's measurement. If Alice has measured a 0, $q_{iA} = 0$ then Bob's qubit has collapsed to $q_{iB} = \frac{1}{\sqrt{2}}(-|0\rangle + |1\rangle)$. If Alice's measurement resulted in $q_{iA} = 1$ then Bob's qubit collapsed to $q_{iB} = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. Now, Bob transforms his qubit via a Hadamard gate. For Alice's $q_{iA} = 0$, Bob has a $Hq_{iB} = 1$, and conversely for Alice's $q_{iA} = 1$, Bob has a $Hq_{iB} = 0$. Bob now measures and his value will consistently be the complement of Alice's.

  If Bob is the one who measures first his qubit q_{iB}, the procedure is simply mirrored. Alice now has to apply a Hadamard gate on her qubit, thus obtaining Hq_{iA}. Again Alice and Bob will have measured complementary binary digits.

  Ideally, with no interference from Eve, be it through eavesdropping or masquerading, applying this measurement and Hadamard measurement on each qubit is enough to establish the secret key. After going through all the qubit pairs, Alice and Bob will have complements of the same binary number. This, for example Alice's binary number, is the established secret key.

- **Step 2 - Authentication and Eavesdropping Checking**

  Some $2m$ qubits will be sacrificed for security checking, where $2m < l$. The secret key will be formed by the remaining $n = l - 2m$ qubits. Alice and Bob decide via the classical channel, the set of qubit indices to be sacrificed. Alice looks at the first m qubits, and forms a binary number with the values she reads. This is Alice's *public key*. Alice now publishes her public key, which key can be seen by Bob. As the public key is protected, Bob is certain that the public key is safe from masquerading. Note that this is the only step, where Bob is certain to have contact with Alice with no masquerading. Bob now compares Alice's public key with his own measured qubits. If these two binary numbers

> are complementary, then Bob concludes that he has been talking with Alice all the while in the previous step, and also that no eavesdropper has changed some of the qubit values. The same procedure applies to Bob's *public key* formed by the second m sacrificed qubits. If Bob or Alice have encountered a mismatch in the values checked, they discard the secret key and try again.

Let us analyze what Eve can do to get some knowledge about the secret key without being caught.

One option would be that Alice's and Bob's qubits are not really entangled, but Eve has sent qubits of her own choice to both of them. Eve also can listen to the classical channel. The best she can do is send a classical 0 to Alice and a *Hadamard* 1 to Bob. Actually, all other combinations are equivalent to, or less advantageous than, this one. Alice and Bob decide who is to measure first once they already have the qubits. With 1/2 probability, Alice is measuring, in which case the readings are consistent. Bob measures first, again with probability 1/2. Bob will measure a 1 or 0 with equal probability. Then Alice transforms the classical 0 with a Hadamard gate, and also reads a 0 or 1 with equal probability. This means that if Bob measures first, Alice and Bob will read the same value with 1/2 probability. As such, Eve is caught with 1/4 probability. If the number of qubits to be tested is large, this probability can be made arbitrarily large.

Note that, the classical channel does not need to be authenticated. Eve can masquerade, such that she completely severs all connection between Alice and Bob. In this case, Eve will establish one secret key with Alice, and another secret key with Bob. Unfortunately for Eve, she has no control over the value of the secret keys, as their values are determined probabilistically, through quantum measurement. Therefore, the two keys will necessarily differ. As Alice and Bob publish part of their secret keys as protected public keys, they will notice the difference and consequently discard the keys.

This algorithm features a few notable characteristics. Checking for eavesdropping and authentication happens in one and the same step. Until this checking phase, there is no certainty whatsoever about either the validity of the key, the validity of the classical connection or the quantum qubit entanglement. But then, the key is not *useful* or *used* before the checking step.

Another interesting feature is the way in which the two public keys are used in our case. In classical settings, the public key is established and known by both parties, before the algorithm or communication begins. By contrast, in this quantum distribution algorithm, the public keys are determined *during* the computation and they are available only after the secret key is established. The publishing of the public keys is the very last step of Alice and Bob's communication, whereas in previous algorithms this is the first step. The usage of the public keys is in reverse order by comparison with classical secret communications, such as those in the public key cryptosystems.

4 Conclusion

We have shown in this paper that quantum authentication in quantum key distribution can be done through the quantum protocol, and does not need to rely in any way on classical authentication procedures. Moreover, in our algorithm authentication is easily done with the help of two protected public keys.

The algorithm presented performs quantum key distribution based on entangled qubit pairs. The entanglement type is not of the generally used Bell states, but an unusual entanglement based on phase incompatibility.

The algorithm uses a quantum channel and a classical channel, but unlike all previous quantum key distribution algorithms, the classical channel is not authenticated. Authentication and security checking are done at the same time, *after* the algorithm, with the help of two public keys.

Specific to quantum key distribution algorithms, is the fact that the value of the secret key is not known prior to performing the distribution. The key is developed *during* the execution. Likewise, in our algorithm, we have the same behavior of the public keys. They are not known prior to the execution of the algorithm and are developed during the execution. The consequence is that the public keys are session specific, rather than permanent for one person. The public keys are distinct for each application of the quantum key distribution algorithm.

The algorithm presented here can be improved to work without a classical communication channel at all. In this case, Alice and Bob communicate via the quantum channel of entangled qubits only. The tradeoff is a doubling in the number of quantum bits used for a final secret key of the same length. Also, the meaning of the public keys is more complex. This improved algorithm is the subject of [6].

As shown in this paper, quantum authentication can be done with the help of a variation of protected public keys. This might not be the only solution. It is an open problem what other structures can support authentication of quantum channels.

The principle of checking and authenticating at the end of the protocol with public keys, is not restricted to the algorithm described here. The same type of public keys, namely per session keys, posted after the execution of the main body of the algorithm, can be successfully used in authenticating other types of algorithms. This is also a direction worth investigating.

References

[1] Charles H. Bennett. Quantum cryptography using any two nonorthogonal states. *Physical Review Letters*, 68(21):3121–3124, May 1992.

[2] Charles H. Bennett and Gilles Brassard. Quantum cryptography: Public key distribution and coin tossing. In *Proceedings of IEEE International Conference on Computers, Systems and Signal Processing*, pages 175–179, IEEE, New York, 1984. Bangalore, India, December 1984.

[3] Charles H. Bennett, Gilles Brassard, and David N. Mermin. Quantum cryptography without Bell's theorem. *Physical Review Letters*, 68(5):557–559, February 1992.

[4] Artur Ekert. Quantum cryptography based on Bell's theorem. *Physical Review Letters*, 67:661–663, 1991.

[5] Marius Nagy and Selim G. Akl. Quantum key distribution revisited. Technical Report 2006-516, School of Computing, Queen's University, Kingston, Ontario, June 2006.

[6] Naya Nagy and Selim G. Akl. Authenticated quantum key distribution without classical communication. Technical Report 2007-531, School of Computing, Queen's University, Kingston, Ontario, April 2007.

[7] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information.* Cambridge University Press, Cambridge, UK, 2000.

[8] Jr. Samuel J. Lomonaco. A Talk on Quantum Cryptography or How Alice Outwits Eve. In *Proceedings of Symposia in Applied Mathematics*, volume 58, pages 237–264, Washington, DC, January 2002.

[9] Bao-Sen Shi, Jian Li, Jin-Ming Liu, Xiao-Feng Fan, and Guang-Can Guo. Quantum key distribution and quantum authentication based on entangled states. *Physics Letters A*, 281(2-3):83–87, 2001.