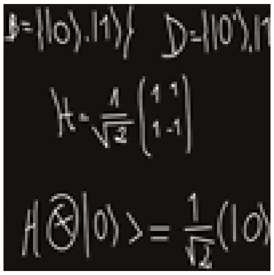
Crossroads
The ACM Student Magazine



An Introduction to Quantum Cryptography

by *Nick Papanikolaou*

Introduction: Perfect Security?

It is widely believed that devising an "unbreakable" cryptographic method is an impossible task. We are still on an age-old quest to find such a method, because virtually all previous attempts have failed. Cryptography was the art of outsmarting a human enemy; today it is concerned more with resisting attack by very powerful computers. Cryptosystems in common use today (e.g., DES, RSA, AES; see [20] for details) are designed to withstand attacks from current generation computers as much as possible. However, the constant increase in available computational power will make these systems more vulnerable to attack.

The amount of computational power used to subvert a particular cryptosystem determines the expected time it takes to recover an encrypted message. This means that it takes much longer to subvert highly sophisticated systems such as AES today than it will take in a couple of decades. Therefore, messages encrypted using such a system can only be considered secure for a limited period of time.

It has turned out to be very difficult to develop a cryptosystem whose security is independent of the time and computational power spent attacking it. Nevertheless, many scientists will insist that it is no longer satisfactory to rely on inabilities of current computer hardware when designing new systems. Consequently, efforts have been made to establish new foundations for cryptography altogether. One of these efforts has led to the development of quantum cryptography, whose security relies not on assumptions about computer power, but on the laws of quantum mechanics.

In the early 1980's, a physicist, S. Wiesner, and two computer scientists, C. H. Bennett and G. Brassard, suggested applying quantum mechanical principles to the task of exchanging secret messages [1, 23]. The "BB84" scheme for quantum key distribution, named after Bennett and Brassard's seminal 1984 paper, makes use of the Heisenberg Uncertainty Principle, from quantum physics, to detect the presence of eavesdropping during communication. According to this principle, it is not possible to measure certain quantities simultaneously, such as the position and momentum of a particle. Physicists call position and momentum **conjugate variables**. Other pairs of conjugate variables do exist. In BB84 they are used to represent binary values; it is thus possible to exchange bit sequences, such as cryptographic keys or even messages, in a highly secure way.

The purpose of this article is to address such questions as "what makes a cryptographic scheme perfectly secure?" and "how can quantum mechanical phenomena be exploited for secure communication?" First, we will investigate the concept of **perfect secrecy** and the well-known Vernam cipher, introducing relevant conventions and notation as we proceed. Then, we present the BB84 scheme for *quantum key distribution,* which uses a train of polarized photons to represent a cryptographic key. This scheme proves to be perfectly secure in theory, while current implementations are prone to noise and errors, this will be discussed later. Finally, a brief account will be given of current computer science research related to quantum cryptography.

Cryptosystems and The Vernam Cipher

Let's start by defining some basic terms (for details, refer to [16, 20]). A **cryptosystem** is a mechanism or convention that allows two or more legitimate users to exchange messages secretly - nobody but these users must be able to learn the content of the messages. Every message, m, is subjected to an **encrypting operation,** E, to produce a so-called **ciphertext** or **cryptogram.** To recover the message corresponding to a given ciphertext, a **decrypting operation** D must be performed. Formally:

$$c = E(m) \text{ and } m = D(c)$$

In a **symmetric** cryptosystem, these two operations require one more argument: the **common key** *k*. The key is a unique sequence of bits known only to the legitimate users of the system. Usually, the procedures *E* and *D* are publicly known, and the key is the only piece of information needed by an enemy to recover the contents of a transmitted message. The basic scenario that arises in most cryptographic applications is the following:

1. The (legitimate) sender of a message *m* uses the key *k* to produce a ciphertext *c = E(m,k)*.
2. An enemy tries to recover the value of *m* by guessing the value of *k*, and performing *D(c,k)*.
3. The (legitimate) receiver of the message uses the key *k* to recover the message *m = D(c,k)*.

Classical cryptography is concerned to a great extent with developing operations *E* and *D* that are practically impossible to compute unless *k* is known. The enemy is assumed to have limited computational power, and limited time on his hands.

A **perfect,** or **unconditionally secure cryptosystem,** cannot be broken even in the face of unlimited time and computational power. The standard example of a perfect cryptosystem is the Vernam cipher, or **one-time pad**.

As an illustration of the one-time pad, consider the message, key and ciphertext as binary strings, such as 010 or 110111. To encrypt a message *m* with a key *k*, we need to perform a bitwise XOR operation on these two values. For example,

if *m* = 010 and *k* = 110, then *c* = *m* XOR *k* = 100

In other words, the encrypting operation for the one-time pad is *E(m,k)* = *m* XOR *k*. To recover the original message from *c*, the XOR operation is applied again, this time on *c* and *k*:

if *c* = 100 and k = 110, then *m* = *c* XOR *k* = 010

So, the decrypting operation is again, an application of exclusive-or: *D(c,k)* = *c* XOR *k*. In the one-time pad:

- Each key is used only once (hence the term "one-time"),
- The key used to encrypt a message *m* is at least as long as *m,*
- Each key is truly random and unpredictable.

As long as these requirements are satisfied, the one-time pad is an unconditionally secure cryptosystem - and this is not a question of computational power, as is the case for most systems in current use.

Why doesn't everyone use the one-time pad? Why do we settle for less secure systems in practice? The one-time pad is difficult to use in practice: a new, secret key must be issued prior to every communication, and the key becomes too long for larger messages. Assuming that the problem of key length does not matter much, knowing the capacity of modern storage media, having to establish a fresh, truly random key secretly all the time is a major problem. Generating large quantities of *truly random* data is not very practical either.

For military and diplomatic applications, it may be possible to have the key delivered manually to all the legitimate users, using a trusted third party such as a courier. But not everybody can afford couriers, and even if they could, there is no guarantee that couriers can be trusted. Anyone who manages to find out the keys being used can decrypt the messages effortlessly, defeating the objective of encryption. This aspect of (symmetric) cryptography is referred to as the **key distribution problem.**

While traditional methods for solving this problem exist, none of them is perfectly secure (except for hand delivery, which is rarely feasible). So although a perfect cryptosystem exists in theory, the difficulty of distributing keys in the first place makes it impossible to devise a complete, practical and perfectly secure cryptographic technique. The most well-known solution to the key distribution problem is **public-key cryptography**, which works as follows.

In public-key cryptography, different keys are used to encrypt and decrypt a message. When sending a message to some user, the message must be encrypted with that user's **public key** *PK*, which is, as its name suggests, publicly available. To decrypt the message, the user only needs his private or **secret key** *SK*. This process can be described as shown below:

$$c = E(m, PK)$$

$$m = D(c, SK)$$

or simply

$$D(\ E(m, PK), SK\ ) = m$$

The public and secret keys of any user are related in such a way that it is computationally intractable to compute the private key from the corresponding public key. The security of the well-known RSA cryptosystem, for example, relies on the difficulty of factoring large numbers into primes [20].

If substantial quantum computers are ever built, it will be possible to perform calculations in massively parallel ways -- leading to the known possibility of factoring prime numbers efficiently (using Shor's algorithm, [21]). This would make it possible to subvert public key cryptosystems, such as RSA, with relative ease. A **quantum computer** is a computational device that uses the phenomena of quantum physics to perform extremely efficient computations. Experimental prototypes of quantum computers have been built, but there is a long way to go before these devices become practical [11, 18]. Given that it is possible to construct devices that "break" classical cryptosystems, it is essential to develop more powerful cryptographic techniques. These new techniques should be secure even when quantum computers become available.

It turns out that we can build a perfectly secure key distribution system using the principles of quantum physics; this is known as **quantum key distribution** (QKD). The keys produced using QKD are guaranteed to be secret, and may be used in conjunction with any cryptosystem.

Quantum Key Distribution

Quantum key distribution takes advantage of certain phenomena that occur at the subatomic level, so that any attempt by an enemy to obtain the bits in a key not only fails, but gets detected as well. Specifically, each bit in a key corresponds to the state of a particular particle, such as the polarization of a photon. The sender of a key has to prepare a sequence of polarized photons, which are sent to the receiver through an optical fiber or a similar medium. In order to obtain the key represented by a given sequence of photons, the receiver must make a series of measurements. A few explanations are necessary before the full implications of this procedure can be understood.

Preliminaries

A **photon** is an elementary particle of light, carrying a fixed amount of energy. Light may be *polarized;* polarization is a physical property that emerges when light is regarded as an electromagnetic wave. The direction of a photon's polarization can be fixed to any desired angle (using a polarizing filter) and can be measured using a calcite crystal.

A photon which is **rectilinearly polarized** has a polarization direction at 0° or 90° with respect to the horizontal. A **diagonally polarized** photon has a polarization direction at 45° or 135°. It is possible to use polarized photons to represent individual bits in a key or a message, with the following conventions:

	0	**1**
Rectilinear	0°	90°
Diagonal	45°	135°

That is to say, a polarization direction of 0° or 45° may be taken to stand for binary **0**, while directions of 45° and 135° may be taken to stand for binary **1**. This is the convention used in the quantum key distribution scheme BB84, which will be described shortly. The process of mapping a sequence of bits to a sequence of rectilinearly and diagonally polarized photons is referred to as **conjugate coding**, while the rectilinear and diagonal polarization are known as **conjugate variables**. Quantum theory stipulates that it is impossible to measure the values of any pair of conjugate variables simultaneously.

The position and momentum of a particle are the most common examples of conjugate variables. If an experimenter tries to measure a particle's position, he or she has to project light on it of a very short wavelength; however, short-wavelength light has a direct impact on the particle's momentum, making it impossible for the experimenter to measure momentum to any degree of accuracy. Similarly, to measure a particle's momentum, long-wavelength light is used, and this necessarily makes the position of the particle uncertain. In quantum mechanics, position and momentum are also referred to as **incompatible observables**, by virtue of the impossibility of measuring both at the same time. This same impossibility applies to rectilinear and diagonal polarization for photons: if someone tries to measure a rectilinearly polarized photon with respect to the diagonal, all information about the photon's rectilinear polarization is lost.

Quantum Key Distribution with BB84

BB84 is the first known quantum key distribution scheme, named after the original paper by Bennett and Brassard, published in 1984 [1]. BB84 allows two parties, conventionally "Alice" and "Bob", to establish a secret, common key sequence using polarized photons. The steps in the procedure are listed below:

1. Alice generates a random binary sequence *s*.
2. Alice chooses which type of photon to use (rectilinearly polarized, "R", or diagonally polarized, "D") in order to represent each bit in *s*. We say that a rectilinearly polarized photon encodes a bit in the R-basis, while a diagonally polarized photon encodes a bit in the D-basis. Let *b* denote the sequence of choices of basis for each photon.
3. Alice uses specialized equipment, including a light source and a set of polarizers, to create a sequence *p* of polarized photons whose polarization directions represent the bits in *s*.
4. Alice sends the photon sequence *p* to Bob over a suitable quantum channel, such as an optical fiber.
5. For each photon received, Bob makes a guess as to whether it is rectilinearly or diagonally polarized, and sets up his measurement device accordingly. Let *b′* denote his choices of basis.
6. Bob measures each photon with respect to the basis chosen in step 5, producing a new sequence of bits *s′*.
7. Alice and Bob communicate over a classical, possibly public channel. Specifically, Alice tells Bob her choice of basis for each bit, and he tells her whether he made the same choice. The bits for which Alice and Bob have used different bases are discarded from *s* and *s′*.

What is important to understand about this procedure is that, only if Bob's guess is correct, is it certain that he will make an accurate measurement. If Bob attempts to measure a rectilinearly polarized photon with a diagonally oriented measurement device (and vice versa), the outcome will be, at random, either **0** or **1**; in this case, the original bit value represented by the photon is encoded in its rectilinear polarization, and all information about the rectilinear polarization is lost. So, an incorrect choice of measurement basis randomizes the outcome of a measurement, which is only accurate in this case with probability 50%. If *n* photons are transmitted in total, there is a probability 0.5^n that Bob will measure all of them correctly.

A similar logical argument allows Alice and Bob to detect the presence of an eavesdropper ("Eve"). Just as Bob, Eve is incapable of knowing which type of photon is used to represent each bit. Therefore Eve must guess which measurement basis to use and, since it is impossible for her to duplicate the state of each received photon (due to the theorem of non-cloneability of quantum states [**24**]), she must create a new photon to send to Bob. Eve's presence is made manifest to Alice and Bob because Eve's measurements necessarily cause a disturbance to the states of the transmitted photons.

The criterion for detecting Eve's presence can be formulated as follows. For the *i*th bit chosen by Alice, *s[i]*, there will correspond a choice of polarization basis, *b[i]*, which is used to encode the bit to a photon. If Bob's chosen measurement basis is *b'[i]* and the outcome of his measurement is *s'[i]*, then

b'[i] = b[i] should necessarily imply s'[i] = s[i]

If an eavesdropper tries to obtain any information about *s[i]*, a disturbance will result and, even if Bob and Alice's bases match, *s'[i] ≠ s[i]*. This allows Alice and Bob to detect an eavesdropper's presence on a noiseless channel, and to reschedule their communications accordingly.

Detailed Walkthrough

Let's consider the following scenario, illustrated in **Figure 1**: Alice and Bob are linked together via a noiseless optical fiber. Eve, the eavesdropper, is capable of making measurements on individual photons passing through the fiber. Consider the case in which Alice wants to communicate the binary sequence 00110 to Bob through this setup, using BB84.

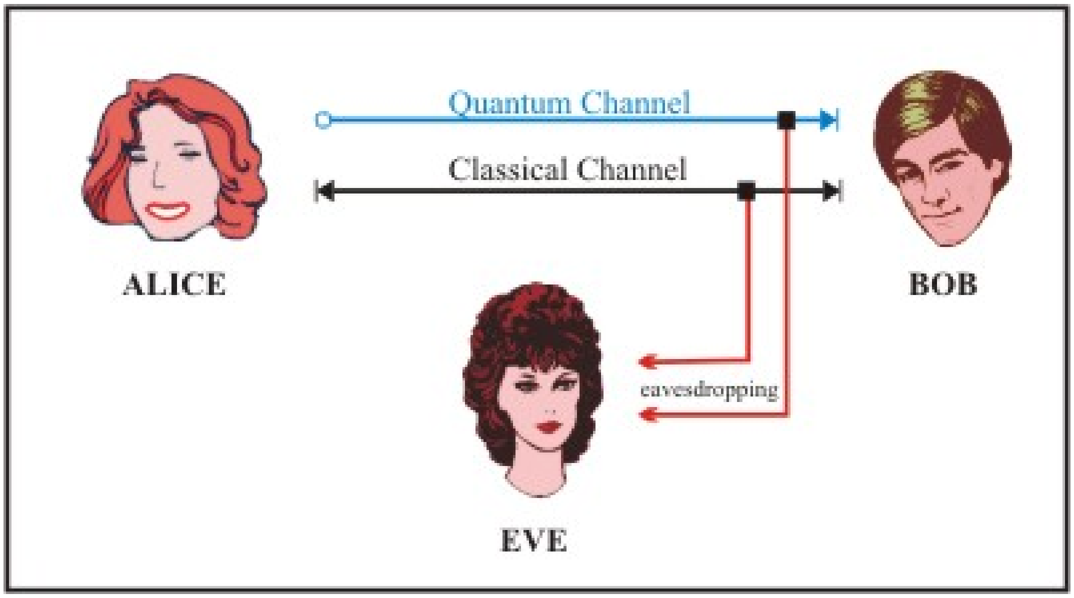


Figure 1: The basic setup for quantum key distribution. The quantum channel is typically an optical fiber, capable of transmitting individual polarized photons.

Alice and Bob perform the steps described in the previous section, detailed below. The question marks indicate bit positions for which measurement will produce a random result (**0** or **1** with equal probability). The whole process is illustrated in **Figure 2**, where instead of question marks, one of the two possible bit values are shown.

1. Alice prepares the binary sequence *s* = **00110**, part of which will be used later as the common cryptographic key with Bob.
2. Alice chooses a sequence of encoding bases at random, say *b* = **RDRDD**. (Remember: "R" = rectilinear polarization (0° or 90°); "D" = diagonal polarization (45° or 135°).
3. Alice encodes *s* using the bases *b*, to produce the sequence of photons with respective polarizations 0°, 45°, 90°, 135°, 45°.
4. Eve makes a random choice of measurement bases, *eb* = **RRDDD***.*
5. Eve intercepts each photon and measures it with her choice of basis, producing a sequence of bits *es* = **0??10**.
6. Eve substitutes the photons she has intercepted, by encoding the bits obtained in the previous step with the bases chosen in step 4. This is known as an "intercept-resend" attack.
7. Bob receives the photons placed on the optical fiber by Eve, and measures them with a set of randomly chosen measurement bases *b'* = RDDRD, obtaining finally a sequence of bits *s'* = **0???0**.
8. Alice and Bob compare their choices of basis and detect Eve's presence with the second bit, for which they used identical bases but obtained different bit values;

they discard the third and fourth bit, leaving *s* = **000** and *s′* = **0?0**.

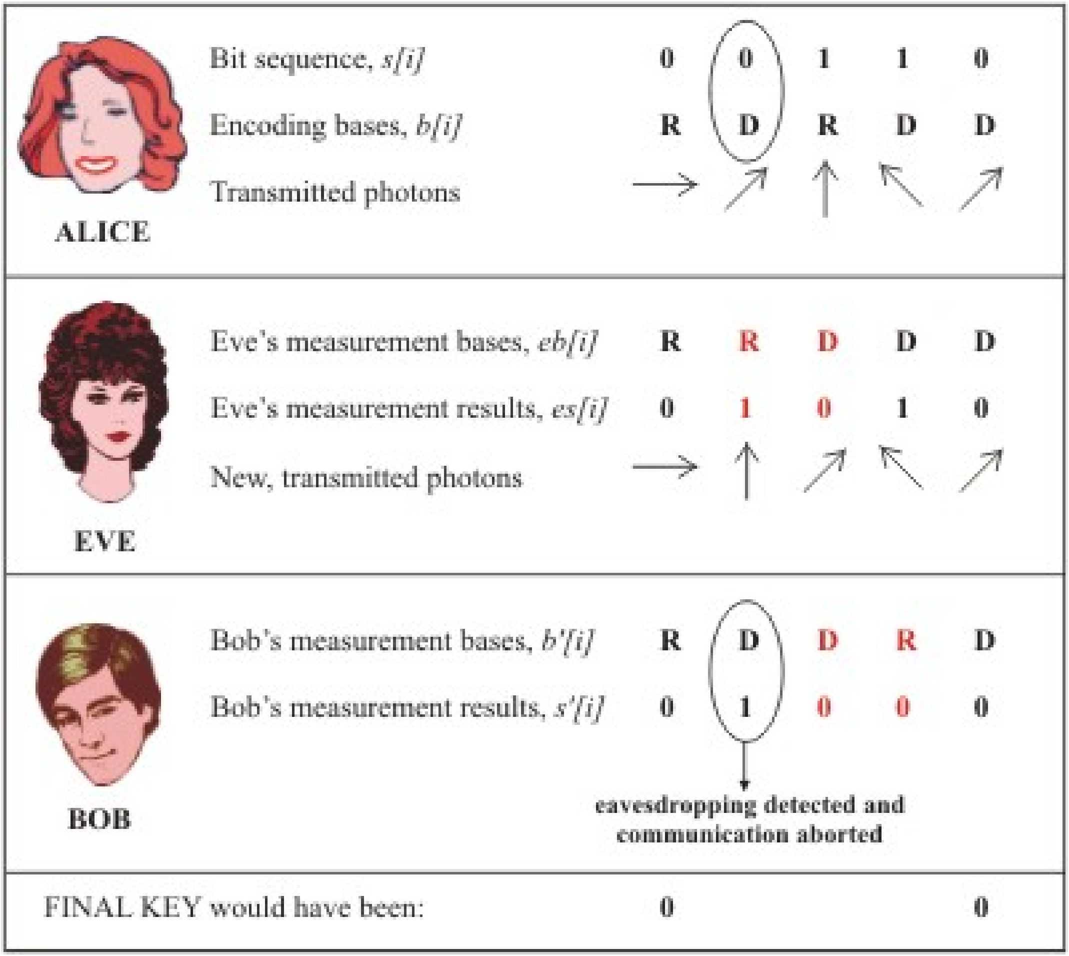

ALICE
Bit sequence, s[i]
0 0 1 1 0
Encoding bases, b[i]
R D R D D
Transmitted photons
EVE
Eve's measurement bases, eb[i]
R R D D D
Eve's measurement results, es[i]
0 1 0 1 0
New, transmitted photons
BOB
Bob's measurement bases, b'[i]
R D D R D
Bob's measurement results, s'[i]
0 1 0 0 0
eavesdropping detected and communication aborted
FINAL KEY would have been:
0 0

Figure 2: The sequence of steps in the BB84 quantum key distribution scheme, in the presence of an eavesdropper. For the second and third bit in this example, Eve makes an incorrect choice of measurement basis, indicated with red colored text. Bob makes an incorrect choice of basis for the third and fourth bit, similarly indicated in red. For the second bit, although Bob has chosen the correct basis (D), the outcome of measurement does not match the original bit encoded by Alice -- this allows Alice and Bob to detect Eve's presence.

Secret Key Reconciliation

The basic BB84 procedure is incomplete in the following sense: whether an eavesdropper is present or not, there will still be errors in Bob's key sequence. The final step of BB84, which was described above merely as a comparison of encoding and measurement bases, is usually much more elaborate. There are two parts involved: *secret key reconciliation* and *privacy amplification.* I will explain the first of the two in this section.

The process of reconciliation is a special error correction procedure which eliminates:

- errors due to incorrect choices of measurement basis;
- errors induced by eavesdropping; and
- errors due to channel noise, if any exists.

Reconciliation is performed as an interactive binary search for errors. Alice and Bob divide their bit sequences into blocks and compare each other's parity for each block. Whenever their respective parities for any given block do not match, they divide it into smaller blocks and compare parities again, repeating this process until the exact location of the error is found. When an error has been located, Alice and Bob may decide to discard the corresponding bit, or agree on the correct value. During this process, Alice and Bob can communicate over a classical (i.e., "non-quantum") channel, which is by definition insecure and accessible to an eavesdropper.

Privacy Amplification

Since valuable information about the key may be obtained by an eavesdropper during reconciliation, Alice and Bob must perform a final step in order to establish a perfectly secret key: this is the process of *privacy amplification*.

The process of reconciliation results in a bit sequence which is common to Alice and Bob, but some of its bits may be known to an eavesdropper who has tapped the classical channel. To eliminate this "leaked" information, Alice and Bob must apply, in common, a binary transformation (usually, a random permutation) to their sequences, and discard a subset of bits from the result. The precise choice of transformation and the number of bits discarded, of course, determine the amount of secrecy of the final key. The objective of this step is to minimize the quantity of correct information which the eavesdropper may have obtained about Alice and Bob's common bit sequence.

At the end of the privacy amplification procedure, Alice and Bob's bit sequences may be shown to be identical and absolutely secret, with arbitrarily high probability. The interested reader is referred to [3, 2, 7, 10] for details.

The Limitations of Quantum Key Distribution and Current Implementations

The exposition, in the previous sections, of quantum key distribution based on BB84, raises several interesting issues. The first observation one makes about the whole procedure is that only part of it involves quantum mechanical phenomena - using polarized photons to represent a binary sequence is only half of the story. In order to distill a perfectly secret binary sequence from the bit values encoded in the photons, it is necessary to perform a set of "non-quantum" communications (namely, key reconciliation and privacy amplification). What is most astounding about these communications is the fact that they may be performed over an *insecure* classical channel, such as a telephone line (which is wire-tappable). Curiously, any information obtained by an eavesdropper through this channel is *useless*. An eavesdropper will learn, for instance, that several of her measurements were performed with the wrong measurement basis; however, quantum measurements are destructive and irreversible, so it is impossible for her to repeat any one measurement.

It has been suggested that an eavesdropper could use a "quantum memory" device to store the photons she intercepts, without performing measurements until the correct choices of basis are made known. It has also been noted that, in practice, all communication channels are prone to noise and, hence, errors will necessarily occur. Importantly, the proof of unconditional security of BB84 [**14**] shows that both these scenarios can be dealt with effectively - neither affects the security of the overall procedure.

Quantum key distribution does have its limitations, however. While a scheme such as BB84 can ensure that an eavesdropper is always detected, it may not always be possible to establish a secret key at the end of the procedure. Generally, as soon as an eavesdropper is detected, the procedure must be aborted and postponed to a later date. That is to say, the legitimate users (Alice and Bob) have to "keep trying" until no eavesdropper is found on the channel. It is now known that, if Alice and Bob share a small amount of information prior to quantum key distribution, it will always be possible for them to establish a secret key [**11**, **18**]. This may sound awkward, since there is little point in performing quantum key distribution if the users are capable of establishing common, secret data via other means -- this is a known issue and remains to be tackled.

Physically breaking into Alice or Bob's communication facilities may still be a practical means of compromising the security of BB84. Should an enemy assault, say, Alice, and attempt to impersonate her toward Bob in BB84, Bob might never notice. There exists a solution to this problem as well. Alice and Bob's equipment could be fitted with a so-called **authentication** mechanism, preventing anyone but the true owner from activating the equipment. An unconditionally secure authentication scheme is known, quite appropriate for this purpose, and is due to Wegman and Carter [22].

So, quantum key distribution is clearly an unconditionally secure means of establishing secret keys. Combined with unconditionally secure authentication, and an unconditionally secure cryptosystem (e.g. the one-time pad), a perfect cryptographic mechanism may be built. All of this is true *in principle,* of course: implementing the hardware for such a mechanism is far from simple.

Note that perfect security is not strictly necessary for most applications; it is merely a desideratum. However, BB84 and similar schemes *do* provide a highly secure solution to the key distribution problem, and in practice the keys produced in this way may well be used in imperfect cryptosystems, such as DES and AES. After all, these cryptosystems are, for all practical purposes, very secure.

The greatest bane for the implementer of quantum key distribution is the presence of noise in all practical communications channels, including optical fibers. It is possible to combat noise with so--called *quantum error correction* techniques [11, 18]. Many such practicalities have been resolved by experimental physicists: fully-working, commercial quantum key distribution systems are already available on the market, from companies such as MagiQ [15] and ID Quantique [12]. Several trials of quantum key distribution have succeeded, including recent experiments under Lake Geneva [10]. Finally, the Defense and Research Projects Agency (DARPA) is involved in the development of a point-to-point communications network based on quantum key distribution [7].

Relevant Research in Computer Science

BB84 is one of several **protocols** that implement the general principle of quantum key distribution. Charles Bennett proposed a simplified version of BB84, referred to as B92. Artur Ekert designed a protocol for the same purpose based on quantum entanglement [6], while numerous other variations can be found in the literature. Thus, a whole class of quantum communication protocols has been formed, and is worthy of careful study in its own right.

For computer scientists, these protocols pose an interesting challenge, since the phenomena involved are far more exotic than those we are all used to. Recent work in the field has included the application of modal logic to the analysis of these protocols [17], and the development of two quantum process algebras [9, 13].

Not unlike computer algorithms, protocols - and especially quantum protocols - are ideal targets for formal verification and analysis [8, 19]. My own work [19] has included the use of model-checking as a means of demonstrating formally the correctness and security of BB84.

Conclusions

There is much more to quantum cryptography than what can be said in the space of an introductory article; several accounts of the subject have appeared before in popular magazines, but not many computer scientists are aware of its implications. The fact that techniques from theoretical computer science may be applied effectively to quantum cryptographic protocols is a good indication that quantum cryptography as a whole is an exciting new area of work, and not just for physicists!

References

1

Bennett, C. H. & Brassard, G. (1984). Quantum cryptography: Public key distribution and coin tossing. In *Proceedings of International Conference on Computers, Systems and Signal Processing*, New York.

2

Bennett, C. H., Brassard, G., & Robert, J. M. (1998). Privacy amplification by public discussion. *SIAM Journal on Computing,* 17(2), pp. 210-229.

3

Brassard, G. & Salvail, L. (1993). Secret-key reconciliation by public discussion. In *Workshop on the Theory and Application of Cryptographic Techniques (EUROCRYPT 93)*, pp. 410-423, Springer-Verlag.

4

Bouwmeester, D., Ekert, A., & Zeilinger, A. (2000). *The Physics of Quantum Information.* Springer-Verlag.

5

Brassard, G. (1998). *Modern Cryptology: A Tutorial.* Volume 325 of Lecture Notes in Computer Science. Springer-Verlag.

6

Ekert, A. (1991). Quantum cryptography based on Bell's Theorem. *Physical Review Letters*, 67(6), pp. 661-663.

7

Elliot, C. (2004). Quantum Cryptography. *IEEE Security and Privacy Magazine* 2 (4), pp. 57-61.

8

Gay, S. J. & Nagarajan, R. (2002). Formal verification of quantum protocols. Available from e-print archive arXiv.org (record: `quant-ph/0203086`).

9

Gay, S. J. & Nagarajan, R. (2005). Communicating quantum processes. To appear in *POPL '05: Proceedings of the 32nd ACM Symposium on Principles of Programming Languages, Long Beach, California*.

10

Gisin, N., Ribordy, G., Tittel, W., & Zbinden, H. (2002). Quantum cryptography, *Reviews of Modern Physics* 74, pp. 145-195.

11

Gruska, J. (1999). *Quantum Computing.* McGraw-Hill.

12

ID Quantique. **http://www.idquantique.com/**

13

Lalire, M. & Jorrand, P. (2004). A process algebraic approach to concurrent and distributed quantum computation: operational semantics. In *Proceedings of the 2nd International Workshop on Quantum Programming Languages*. Turku Centre for Computer Science.

14

Mayers, D. (2001). Unconditional security in quantum cryptography. *Journal of the ACM* 48(3), pp. 351-406.

15

MagiQ Technologies. **http://www.magiqtech.com/**

16

Menezes, A., van Oorschot, P., & Vanstone, S. (1997). *Handbook of Applied Cryptography.* CRC Press.

17

van der Meyden, R. & Patra, M. (2003). Knowledge in quantum systems. In *Proceedings of the Conference on Theoretical Aspects of Rationality and Knowledge*, pp. 104-117. ACM Press.

18

Nielsen, M. & Chuang, I. (2000). *Quantum Computation and Quantum Information.* Cambridge University Press.

19

Papanikolaou, N. (2004). *Techniques for Design and Validation of Quantum Protocols*, M.Sc. thesis, University of Warwick.

20

Schneier, B. (1996). *Applied Cryptography.* Wiley.

21

Shor, P. (1997). Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, *SIAM Journal on Computing* 26, pp. 1484-1509.

22

Wegman, M. & Carter, J. (1981). New hash functions and their use in authentication and set equality. *Journal of Computer and System Sciences* 22(3), pp. 265-279.

23

Wiesner, S. Conjugate coding. *SIGACT News* 15 (1983), pp. 78-88. Original manuscript dated 1969.

24

Wootters, W. K. & Zurek, W. H. (1982). A Single Quantum Cannot be Cloned. *Nature*, 299, 802--803.

Biography

Nick Papanikolaou (**N.Papanikolaou@warwick.ac.uk**) received his B.Sc. in Computer Systems Engineering from the University of Warwick, UK in 2003. During the academic year 2003-4 he undertook an M.Sc. by Research (thesis title: *Techniques for Design and Validation of Quantum Protocols*) at the same institution; he is currently a doctoral student at Warwick and a Tutor for the Higher National Certificate/Diploma in Business IT with RDI Consultants Ltd.