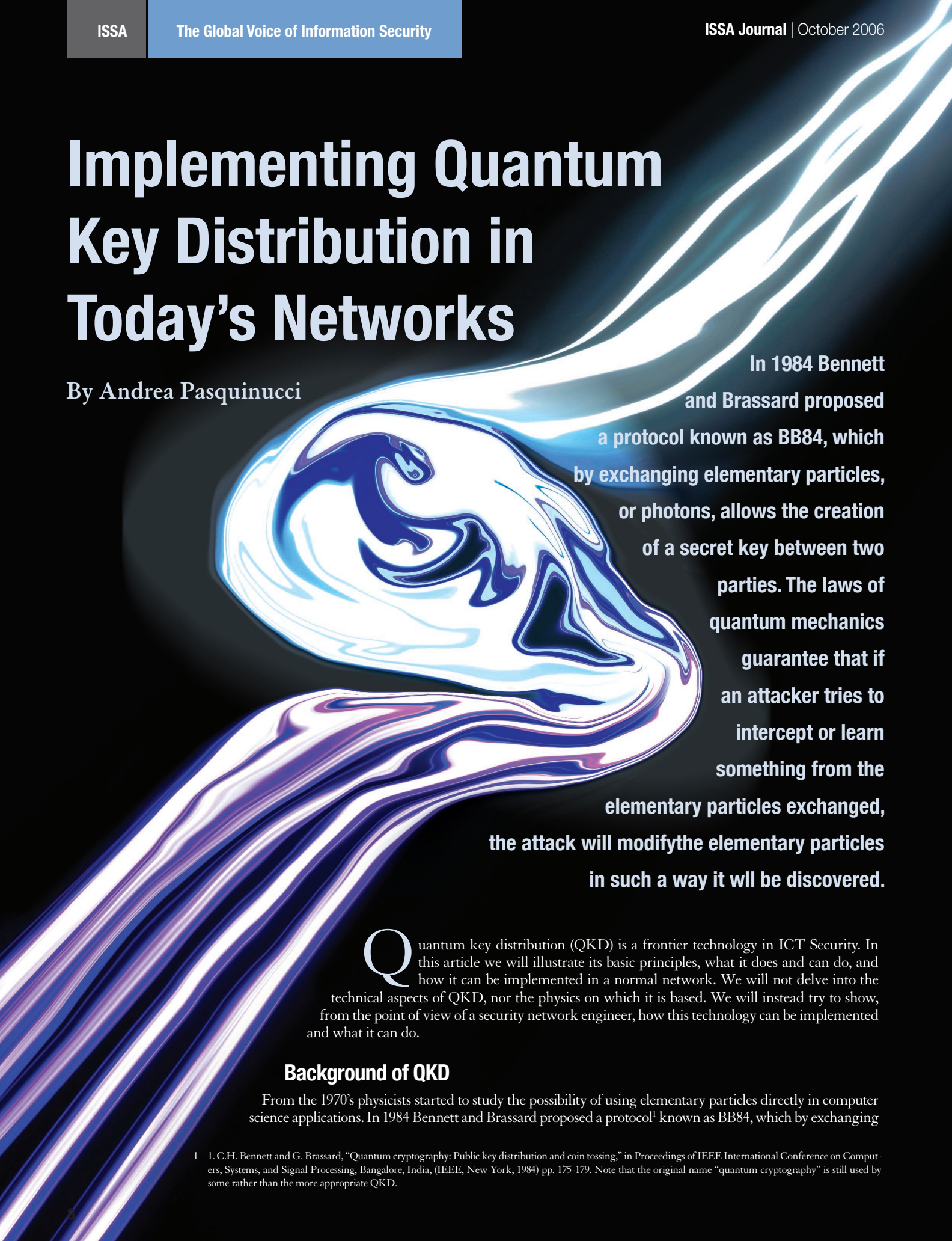
Implementing Quantum Key Distribution in Today's Networks

By Andrea Pasquinucci



In 1984 Bennett and Brassard proposed a protocol known as BB84, which by exchanging elementary particles, or photons, allows the creation of a secret key between two parties. The laws of quantum mechanics guarantee that if an attacker tries to intercept or learn something from the elementary particles exchanged, the attack will modifythe elementary particles in such a way it wll be discovered.

Quantum key distribution (QKD) is a frontier technology in ICT Security. In this article we will illustrate its basic principles, what it does and can do, and how it can be implemented in a normal network. We will not delve into the technical aspects of QKD, nor the physics on which it is based. We will instead try to show, from the point of view of a security network engineer, how this technology can be implemented and what it can do.

Background of QKD

From the 1970's physicists started to study the possibility of using elementary particles directly in computer science applications. In 1984 Bennett and Brassard proposed a protocol[1] known as BB84, which by exchanging

1 1. C.H. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," in Proceedings of IEEE International Conference on Computers, Systems, and Signal Processing, Bangalore, India, (IEEE, New York, 1984) pp. 175-179. Note that the original name "quantum cryptography" is still used by some rather than the more appropriate QKD.

elementary particles, or photons, allows the creation of a secret key between two parties (Alice and Bob). The laws of quantum mechanics guarantee that if an attacker (Eve) tries to intercept or learn something from the elementary particles exchanged, she will modify the elementary particles in such a way that Alice and Bob will discover it. It does not matter which tool or attack Eve may use, she can do whatever she is allowed by nature – but quantum mechanics guarantees she will be discovered.

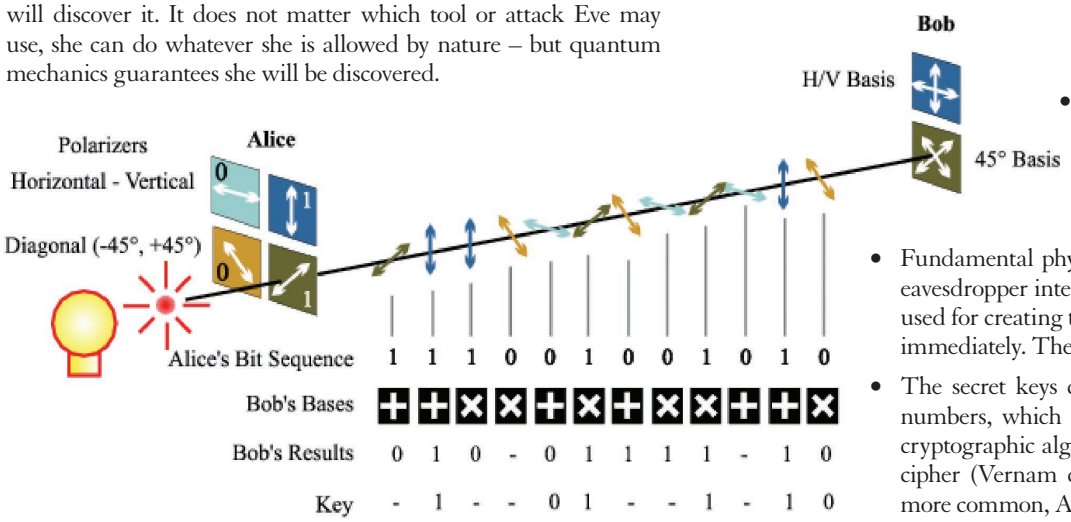


Figure 1. The BB84 protocol[2]

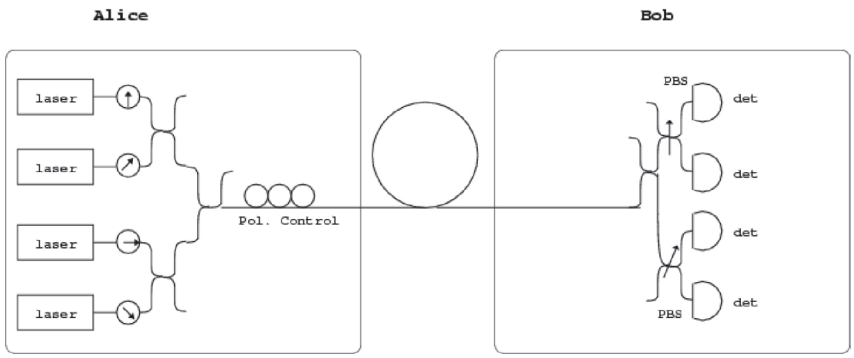


Figure 2. Scheme of the experimental setup of BB84 (PBS = Polarization Beam Splitter)[3]

Figures 1 and 2 give a schematic presentation of how the BB84 protocol works[2] [3]. An *idealized* BB84 key exchange runs as follows[4] (see also Figure 4).

Alice randomly chooses one photon out of a set of four possible photons with special different polarizations, each polarization representing the value 0 or 1 of one bit. Bob randomly chooses one of two detecting devices and measures the polarization of the photon he has received. Each device can measure faithfully only two of the four polarizations, and gives a random result for the other two. Alice and Bob repeat this procedure for many photons. Then for each photon exchanged, Bob tells Alice which device he has used and Alice tells Bob if he has used the correct one. Alice and Bob discard all bits for which Bob has used the wrong device and they check, using a classical error correcting algorithm, if there are errors in the remaining bits. If there are errors, it means that Eve has eavesdropped on the photons. n this case the key must be discarded.

Basic principles of QKD

From this very brief description of the process of the protocol, it follows that:

- QKD is a technology which, by using elementary particles, allows the creation in real-time of a secret key between two parties.
- Fundamental physics laws guarantee that if an eavesdropper intercepts the elementary particles used for creating the secret key, she is discovered immediately. The key is then discarded.
- The secret keys created by QKD are random numbers, which can be safely used as keys in cryptographic algorithms like the one-time pad cipher (Vernam cipher) or the less secure, but more common, AES, Triple DES, etc.

A typical illustration of the use of quantum key distribution is given in Figure 3 below. Two parties exchange encrypted data through a communication channel, which can also be the Internet. The devices which encrypt/decrypt the data receive the secret keys from QKD. Both devices receive the same keys, which can be used, for example, in symmetric cryptographic algorithms.

As we have seen, QKD does not encrypt or decrypt data, nor does it provide *a priori* security on the key exchange. Indeed, contrary to what happens in "classical," as opposed to quantum cryptography, QKD detects an eavesdropper obtaining information upon the elementary particles exchanged, which could enable her to obtain (part of) the secret key. After the secret key has been created, or *a posteriori,* QKD validates the secret key and vouches for the fact that nobody has eavesdropped on its creation.

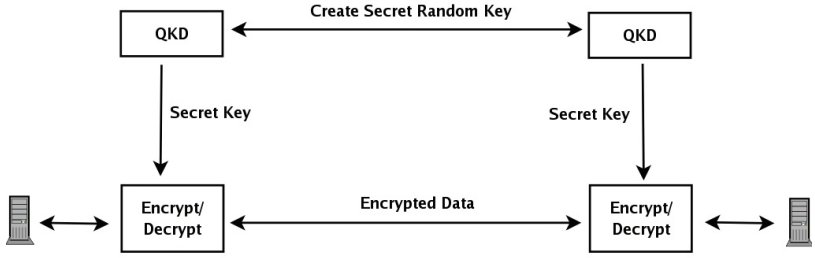


Figure 3. Typical use of QKD to create secret keys

QKD and asymmetric cryptography

A cryptographic system which uses quantum key distribution does not need to use any asymmetric algorithm. Asymmetric algorithms are usually used for creating or exchanging secret keys or making digital signatures. For example, the Diffie-Hellman asymmetric algorithm creates a secret key shared by two parties. (Notice that

2 H. Bechmann-Pasquinucci, Eavesdropping in Quantum Cryptography, Geneva University, 2002

3 A. Pasquinucci, "A First Glimpse at Quantum Cryptography," 2004, http://www.ucci.it/it/qc/whitepapers/index.html

4 N. Gisin, G. Ribordy, W. Tittel, H. Zbinden, "Quantum Cryptography," Reviews of Modern Physics, Vol. 74, p. 145, 2002, http://arXiv.org/abs/quant-ph/0101098

QKD guarantees that if an attack is performed after the generation of the key, then even if the classical algorithm is broken, the secret key cannot be obtained.

this algorithm alone does not allow for authentication between the parties.) The RSA algorithm can be used to send a secret key to a corresponding party whose public key is known. A digital signature, or encryption with the private key, is used as a means to assert the validity of the message's source. This allows the two parties to authenticate each other.

As we have seen, Alice and Bob need to exchange some classical information in running the BB84 protocol, and for this they need to authenticate themselves and the BB84 messages they exchange – for example, to prevent man-in-the-middle attacks. After having created the secret key, Alice and Bob exchange some encrypted data which also needs to be authenticated. To authenticate their communications they can use message authentication code (MAC) algorithms instead of asymmetric algorithms. MAC algorithms require the two parties to share a secret key beforehand, which is then used as the authentication item. Digital signatures, on the other hand, rely on the fact that each public key is uniquely related to the corresponding private key, which identifies the maker of the signature.

Thus a cryptographic system that adopts QKD can use MAC algorithms for authentication, hash algorithms for integrity and symmetric algorithms for confidentiality – avoiding completely the use of asymmetric algorithms. But is this important?

The answer at present must be, not really. Asymmetric algorithms are considered secure, and for sure are much more portable, common and inexpensive than QKD. For today's security, QKD is so much more expensive and difficult to implement (as we will see) that we may well wonder if there could be any reason to consider it. But if we look to the future, the perspective changes. There are many doubts about the security of the mathematical problems on which asymmetric cryptography is based. Furthermore, it is already known that when quantum computers arrive, they will render all asymmetric algorithms broken. In this case QKD could come to our rescue, at least for the tasks QKD is able to accomplish.

QKD is probably the first of new developments coming from the field of quantum information, which deals with managing information directly within elementary particles. Quantum computers are the other well-known result of quantum information, but this field of research is very young, and we expect it will result in new surprises.

Principles of QKD

Before discussing how to implement a quantum key distribution system in an existing network, we need to discuss more precisely some of its main features. A QKD system has two main parts (see Figure 4):

1. In the first part of the QKD protocol, the *quantum* part, Alice chooses randomly between 0 and 1 and encodes the chosen value in an elementary particle, which in practice is a single photon. There are more ways to encode the chosen value in a photon; again, Alice has to choose one of them at random. Alice then sends the photon to Bob, who measures it and records the measured value. This procedure is repeated many times, generating a so-called "raw key" before passing to the next step. In practice, obviously, this all is automatic, and the random choices are driven by certain optical phenomena yielding physical, truly random results. Moreover, this first part runs continuously, and every N photons the second part is activated on the last N bits.
2. In the second part of the QKD protocol, the *classical* part, Alice and Bob communicate over a usual communication channel, typically a digital channel, and exchange some information allowing them to establish whether Eve has eavesdropped on the exchange of the elementary particles. Alice and Bob also exchange some other information on this channel which allows each of them, separately but following exactly the same procedure, to distill the secret key. (These steps are usually called "error correction" and "privacy amplification.")
3. Finally, Alice and Bob have created the secret key and are certain nobody has eavesdropped or has any knowledge of the key. (If somebody has, then Step 4 is to go out and look for Eve!) Notice that the final key is usually much shorter, even by order of magnitude, than the initial raw key (the random photons/bits sent by Alice to Bob) because many bits are used for the steps in the second part.

A few comments are in order. The first part of QKD requires the exchange of single photons, and this currently is accomplished by use of a (dark) fiber dedicated to the purpose. Obviously this part of the protocol runs continuously: The rates of generation of the bits of the raw keys are of the order of MHz to GHz. For every group of photons exchanged, the second part is run on the just-created raw key.

Though seldom discussed in the QKD community – probably because it adopts simple and reasonably secure classical algorithms – the second part is quite important for a practical implementation of QKD. First of all, this part requires a classical communication channel (even a telephone would do). This channel is often a TCP/IP connection between the two parties. The information exchanged is public, that is, not encrypted. The protocol does not require these communications to be kept secret, and actually assumes Eve is listening in on them. But the data exchanged between Alice and Bob must be authenticated and integral. For this Alice and Bob must use a MAC and a hash to guarantee authentication and authenticity. Which requires, of course, that Alice and Bob share a secret key beforehand!

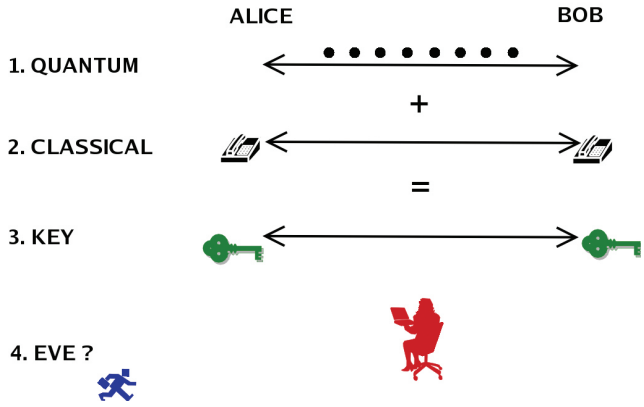


Figure 4. The principal phases of a QKD protocol

That is exactly what happens. Alice and Bob share a secret key before running QKD, in order to secure the classical communications. Every time a new secret key is created, some of its bits can be used as the new secret key to secure the classical communications. From this point of view, QKD is a "key extension" protocol which, given an initially shared, short secret key, generates random shared secret keys of arbitrary length. Notice that the newly created secret key has no dependence on the initial shared key; indeed, it is truly random.

Thus QKD does not solve the problem of the initial distribution of secret keys, as asymmetric cryptography plus PKI (public key infrastructure) or similar infrastructures aim to do. But with QKD the key distribution problem is limited to the first key, since after that the system generates all needed keys on its own. From some points of view, this may be seen as a weakness of QKD. It is certainly a feature that limits deployability. On the other hand, the fact that Alice and Bob are forced to meet in person to first start the system represents a way of obtaining a higher level of security.

The classical part of the protocol has a quite interesting feature to add to its security. We have said that the communication on the classical channel is public, and that the kind of data exchanged does not leak any information about the secret key to an eavesdropper. Thus an attack on the classical channel must be done in real time, for example as a man-in-the-middle attack. This implies that if one day the classical algorithms adopted in this phase are broken, the security of all secret keys created before that moment will not be compromised. The use of a broken classical algorithm in the classical part of QKD implementation could allow an attacker clandestinely to obtain some information on the secret key, but *only if* she is able to mount her attack during the running of the protocol and before creation of the secret key.

This is a strong form of forward security (or future-proof security) since no attack on the classical channel can endanger the security of QKD if it is performed after the moment in which the secret key is created. Current classical protocols for key establishment usually do not offer such a forward security. If today an exchange for key establishment is recorded, and tomorrow the algorithm is broken, the created key may be obtained.

What is usually meant by the "perfect forward secrecy" of a classical key-establishment protocol is the fact that knowledge of any key material used to encrypt data does not give information on previous keys. In other words, an independent attack must be performed to recover every key used to encrypt data at different moments. But the typical attack in this case must record all data exchanged for the key establishment and then crack the key at a later time. QKD instead guarantees that if an attack is performed after the generation of the key, then, even if the classical algorithm is broken, the secret key cannot be obtained.

Hopefully these improvements will allow the key-generation rates to match the speed of the networks, thus allowing use of OTP on any QKD-encrypted link.

Summary

Quantum key distribution offers a very high level of security as a key-establishment protocol. The quantum part is provable as secure[5] [6] based on the laws of quantum physics (physicists usually call this "unconditional security"), whereas the classical part guarantees a strong form of forward security.

Thus from the security point of view, the main difference between using an asymmetric algorithm for key establishment and using QKD is as follows. If today an attacker records the data exchanged in the running of a key-establishment protocol, and in ten years the asymmetric algorithm of today is broken, then the attacker will be able to obtain all the keys that have been created in the interim. This will not be possible using QKD. The security offered by an asymmetric algorithm lasts until the algorithm is broken, whereas the security of QKD, if not broken during the running of the protocol[7], lasts indefinitely.

5 D. Mayers, "Unconditional Security in Quantum Cryptography," J.Assoc.Comp.Mach. 48, 2001, 351, http://arxiv.org/abs/quant-ph/9802025

6 D. Gottesman, H.-K. Lo, N. Lutkenhaus, J. Preskill, "Security of Quantum Key Distribution with Imperfect Devices," Quantum Information and Computation 4, 2004, 325, http://arxiv.org/abs/quant-ph/0212066

7 Technically, this is not exactly true. An attacker could use "quantum memories" to store information obtained during an attack on the quantum part of the process, so to be able to crack the key exchange during the running of the classical part or at a later moment. In any case, QKD guarantees the attacker will be discovered during the running of the protocol even if she uses quantum memories, which, by the way, nobody today really knows how to build.

Implementing QKD today

Quantum key distribution can be used today as a key-establishment protocol for networking. That is, it provides the secret keys for encrypting virtual private network (VPN) tunnels. To implement QKD we need a point-to-point connection between the two ends of the tunnel with a single piece of dark fiber. This is a costly requirement. A single fiber should be dedicated to exchange the photons of the first part of the QKD protocol, and there must not be repeaters or any device along the fiber – it must really be a single fiber. Any repeater or device in the fiber modifies the photons exchanged and makes it impossible to run the protocol: If we were to try this, the QKD implementation would tell us there is constantly an attacker intercepting the photons, and it would not produce any secret key.

The requirement for the absence of repeaters makes the total length of a QKD system limited: The maximum distance yet reached in the laboratories is 150 kilometers. In real implementation the usual distances run up to 50 or 100 kilometers.

Another important parameter for the QKD system of today is the key-generation rate. As already mentioned, even if the rate of photon exchange in the first part of the QKD protocol is in the MHz to GHz, the final key-generation rate is of the order of a few tenths of kilobits per second. This rate depends not only on the need to use many bits for the classical part of the protocol, but also on the loss of photons in the fibers and in the detection devices. In particular, the final rate depends on the quality and, most importantly, the length of the fiber. It should be noted that both the protocols and technologies adopted by QKD are developing very fast, and the key-generation rate of QKD systems is increasing practically every day.

The key-generation rate is an important factor for current implementations of QKD. Since QKD offers a very secure way to generate secret keys, it is natural to use it together (see Figure 3) with the only encrypting algorithm that has been mathematically proved to be secure, the one-time pad (OTP) cipher, or Vernam cipher. But the OTP requires a truly random secret key, used only once and only for as long as the encryption of the message. For QKD, the second and third requirements pose a real constraint; indeed, they imply that the key-generation rate must be equal to the speed of the data transfer. In other words, if we transfer data at 10 mbps, we need QKD

to generate secret keys at the same speed to be able to use OTP to encrypt the data. Since today's QKD key-generation rate is of the order of kbps, what can we do?

There still are communications which require only a few kbps –for example telephone, fax, and in particular VoIP (Voice over Internet Protocol). So we can imagine that in situations where very high security is required, such as the military, a telephone line can be secured with QKD. Moreover, we need only be sure that the average (daily) rate of data transfer is less than the rate of QKD key creation, since QKD can cache for some time the secret keys created (but not used) within the secure QKD devices. It can then use the cached secret keys when instantaneous data transfer is faster than the key-generation rate.

For more commercial and common applications and networks, the speed of QKD plus OTP is not enough. The current solution is to use common symmetric algorithms for data encryption, usually AES, which requires keys of 128, 192 or 256 bits. Adopting, for example, the combination QKD plus AES, one can change the AES secret key 20 or 30 times per second. In this way each secret key is used only for the data transmitted in 3 to 5 hundredths of a second. Since all secret keys are independent, a successful attack on network traffic encrypted with AES, which is theoretically possible by brute force but practically impossible, will require obtaining all keys. This means repeating the attack in an independent way on each block of data transmitted in those few hundredths of a second. This of course makes an attack more difficult to implement due to the amount of resources needed –but not impossible in theory.

QKD on the market

Quantum key distribution is already on the market and a few companies are selling QKD devices. Besides the dedicated QKD devices we have described up to now (those that create secret keys and pass them on to encryption devices), these companies also offer all-in-one devices, which can be plugged directly into existing networks. There are two kinds of all-in-one devices, those working at Layer 2 of the ISO/OSI stack and those working at Layer 3, that is, the IP layer.

The Layer 2 devices (see Figure 5) behave as if they were transceivers or bridges, which also provide security to the channel connecting them. Typically they have Ethernet interfaces toward the LAN and encrypt/decrypt the full Layer 2 frames which they receive/send to the LAN. The two QKD devices are connected by a standard telecom optical fiber (for example G.652 single-mode fiber) which is used for the exchange of the single photons of the quantum part of QKD. The exchange of the classical communications in the second part of QKD, and of the encrypted frames, can be done either on another dedicated fiber (or multiplexed on the same fiber), or through another Ethernet link on any network which reaches the second device. Frames are usually encrypted with AES and the secret keys are changed 100 times per second.

The Layer 3 devices (see Figure 6) behave like VPN gateways/concentrators or IP routers/firewalls which encrypt data communications, as is commonly done with IPSec, for example. These devices can be dedicated or can be similar to usual IP routers/firewalls, with all kinds of interfaces and functionalities, and with an extra optical fiber connection that exchanges the single photons in the quantum part of QKD. All other communications, the classical QKD part and the encrypted data, are exchanged on the usual IP links, which can even be Internet. Within these devices the QKD hardware provides the secret keys to the cryptographic engines, which encrypt/decrypt the IP packets.

Future developments: Networking and free space

Quantum key distribution is a young and rapidly developing field. In the next future there will be improvements in the optical technologies adopted, which will allow an increase in QKD key-generation rates. Hopefully these improvements will allow the rates to match the speed of the networks, thus allowing use of OTP on any QKD-encrypted link.

As far as distances are concerned, it is difficult to imagine the use of single fibers in practice longer than 100 kilometers. There are also physical limits that prevent single photons from traveling much farther than the same distance in optical fibers. Scientists are studying the possibility of building "quantum repeaters" which can lengthen the distance traveled by photons without disturbing them. The technology required is very similar to the one needed to build quantum computers. It is believed that at least five or ten more years will pass before quantum repeaters are built. In the meantime, the development is going in two different directions: networking and free space.

Networking

QKD networking refers to the realization of networks of QKD devices generating secret keys between endpoints connected through intermediate relays[8] [9]. The single photons will run only between one relay station and the next, but the overall protocol will allow creation of a secret key between the endpoints. Networking will also allow a single QKD device to exchange photons and create keys with many other QKD devices. That is, one device will be able to create different keys at one time, shared with different remote endpoints, and these keys will be used to encrypt different tunnels. The research in this field is merely beginning, though a few proposals have already been made.

8 H. Bechmann-Pasquinucci, A. Pasquinucci, "Quantum key distribution with trusted quantum relay," 2005, http://arxiv.org/abs/quant-ph/0505089

9 A. Pasquinucci, "Authentication and routing in simple Quantum Key Distribution networks," 2005, http://arxiv.org/abs/cs.NI/0506003

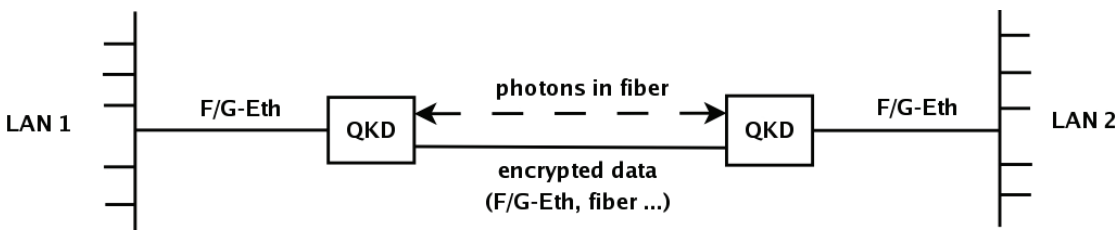


Figure 5. Layer 2 QKD network

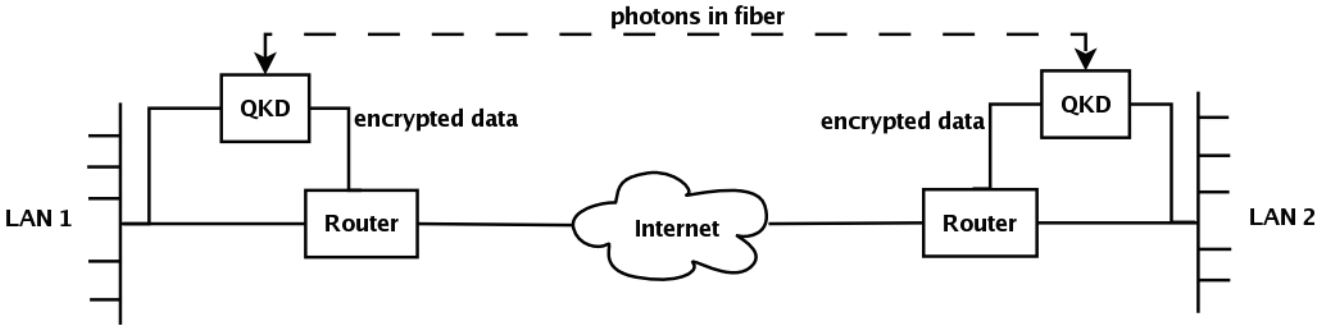


Figure 6. Very simple example of a Layer 3 QKD network

Free space

Another research direction is into the exchange of single photons in free space. Of course in this case atmospheric factors, like fog or heavy rain, can stop the communication almost completely. However, the military is very interested in free-space QKD –for example, for satellite communications. At the opposite end, free-space QKD can have interesting application at short ranges, from a few centimeters to a couple of kilometers. For example, in urban areas free-space QKD could be used between buildings, which should be tall since there must be a clear line of sight between the two telescopes used by the QKD devices. Free-space QKD devices are in the testing phase in various laboratories around the world, and we should soon see the first commercial products appear on the market.

What will QKD be good for?

We have seen that quantum key distribution is a protocol to create secret random keys using elementary particles. QKD offers a very high level of security, and can be used to secure network connections. As with many other new technologies, at the moment QKD is expensive and has requirements for implementation that are not always easy to satisfy. But there is obviously an important point to discuss besides the technology: What might QKD be good for?

Very often today, key establishment is not the weakest link in networking security infrastructure, and is far from the weakest link in the overall information security system of an organization. Nonetheless, there are situations in which management would be happy to consider a technology as promising as QKD for the security of communications.

Implementation of QKD should be considered today if very sensitive information must be protected for a very long time. This may be the case for financial institutions, governments, and the military. In a financial institution, for example, QKD could be employed to secure the connection between the main and the business continuity or disaster recovery elaboration centers.

The most important reason to adopt QKD at present is to protect network communications for a long time. Today's classical cryptography will protect our information for only a limited period. Both unexpected developments in mathematics and improvements in the power of computers could further shorten this time. As of today, QKD could be the only sure method of protecting communications data very long into the future – and in principle, forever.

About the Author

Andrea Pasquinucci, PhD, CISA, CISSP (andrea.pasquinucci@ucci.it), after an academic career in the '90s, is now a senior freelance consultant in ICT Security. His fields of expertise are network and operating-system security, and classical and quantum cryptography. He is a co-founder of the ISSA-Italy chapter.

References

Bechmann-Pasquinucci, H., *Eavesdropping in Quantum Cryptography,* Geneva University, 2002

Bechmann-Pasquinucci, H. and A. Pasquinucci, "Quantum key distribution with trusted quantum relay," 2005, http://arxiv.org/abs/quant-ph/0505089

Bennett, C.H. and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," in *Proceedings of IEEE International Conference on Computers, Systems, and Signal Processing,* Bangalore, India, (IEEE, New York, 1984) pp. 175-179

Gisin, N., G. Ribordy, W. Tittel and H. Zbinden, "Quantum Cryptography," *Reviews of Modern Physics*, Vol. 74, p. 145, 2002, http://arXiv.org/abs/quant-ph/0101098

Gottesman, D., H.-K. Lo, N. Lutkenhaus and J. Preskill, "Security of Quantum Key Distribution with Imperfect Devices," *Quantum Information and Computation* 4, 2004, 325, http://arxiv.org/abs/quant-ph/0212066

Mayers, D., "Unconditional Security in Quantum Cryptography," J.Assoc.Comp.Mach. 48, 2001, 351, http://arxiv.org/abs/quant-ph/9802025

Pasquinucci, A., "Authentication and routing in simple Quantum Key Distribution networks," 2005, http://arxiv.org/abs/cs.NI/0506003

Pasquinucci, A., "A First Glimpse at Quantum Cryptography," 2004, http://www.ucci.it/it/qc/whitepapers/index.html