Quantum Cryptography Review

Amir Hossein Jabbari

July 30, 2002

1 Introduction

The Quantum Cryptography, or more exactly, quantum key distribution, instead of depending on the usual complexity-theoretic assumptions such as the difficulty of factoring, depends on the uncertainty principle of quantum physics.

The Uncertainty Principle of quantum physics implies: *The more precisely the position of an object is determined, the less precisely the momentum is known in this instant, and vice versa.*[1]

It means that if you observe a moving object, you cannot both decide at what speed it is moving, and the precise location which it is at. If you measure one of them, you can't measure the other correctly. Thus, there is an *uncertainty* about one of the two properties. For example if you know that an electron is moving at a definite speed, the uncertainty principle says that you can't possibly know where it is. And since you don't know where it is, it may possibly be anywhere. There is a small possibility that the electron may exists at any given point in the universe at the moment you are trying to observe it.

Considering uncertainty principle of photons traveling from one node to another and use of polarizing filters[2] in both sender and receiver nodes are main elements of quantum key distribution method to generate a secret key between two nodes. This key can then be used with conventional cryptographic algorithms.

2 Quantum Key Distribution Protocol

1. Alice sends a random sequence of photons polarized horizontal ($\leftrightarrow$), vertical ($\updownarrow$), right-diagonal ($\nearrow$) and left-diagonal ($\nwarrow$).To generate a n bits key, Alice should send $2n$ bits photon pulses.

   Example: Alice and Bob want to generate an 8 bits secret key. Therefore Alice is sending 16 bits of random photon pulses.

[1] Heisenberg's Uncertainty Principle

[2] Traveling photons vibrate in some direction. When a large group of photons vibrate in the same direction they are *polarized*. Polarization filters allow only photons that are polarized in a certain direction through; the rest are blocked.

↗ ↕ ↖ ↖ ↕ ↔ ↔ ↗ ↔ ↖ ↗ ↕ ↗ ↕ ↖ ↕

2. Bob measures the photons' polarization in a random sequence of bases, either rectilinear (+) or diagonal (×) by a polarization detector. Bob cannot measures in both bases at the same time since measuring in on base will destroy the possibility of measuring another.

   Example:

   Bob sets his detector randomly on either diagonal or rectilinear bases.

   × × × × + + × × × × + + + × + ×

   If Bob sets his detector correctly he will record the correct polarization. If not, he will record some random measurement. However Bob cannot tell if he has recorded the correct measurement or not.

3. Bob tells Alice which bases he used for each photon he received over an insecure channel

4. Alice tells him which bases were correct.

   Example:

   √ ~ √ √ √ √ ~ √ ~ √ ~ √ ~ ~ ~ ~

5. Alice and Bob keep only the data from these correctly-measured photons, discarding all the rest.

   Example:

   ↗ ↖ ↖ ↕ ↔ ↗ ↖ ↕

6. This data is interpreted as a binary sequence according to the coding scheme that has been published publicly. The data can be interpreted according the following table scheme.

Binary Value	*Base*	
1	↕	↗
0	↔	↖

   Example:

   1 0 0 1 0 1 0 1

   The advantage of this method is that in an *ideal machine* Eve cannot eavesdrop. Like Bob, Eve can only guess the correct polarization and her guess could only be half of the times right. Since wrong guesses changes the polarization of the photons, Alice and Bob will end up with different bit strings in the presence of Eve. Enhancement to this protocol allow Alice and Bob to use their bits even in the presence of Eve, by comparing the *parity* of subsets of the bits.

3 Inadequate Transmission in Practice

The protocol described, as mentioned before works perfect for an *ideal quantum cryptography machine*. However, building such machine in practice is hard. First, detectors usually have noise thus even in absence of Eve, the data Bob receives is different from what Alice has sent. Second, producing a pulse containing only one photon is difficult. If instead of having one photon in each pulse, Alice produce a coherent pulse[3] or even incoherent pulse[4] then there is a possibility that eve may be able to split a pulse into two or more photons, reading one and let the others go to Bob. This probability (P) can be calculated as:

$$P = \mu^2/2 \quad \begin{array}{l}\text{where, } \mu \text{ is the expected number of photons per pulse} \\ \text{and, } \mu \ll 1\end{array}$$

4 Parity Check for Eavesdropping

The procedure described in [1] for Alice and Bob to reconcile their bits takes place over a public channel. Since Eve presumably listens to all public transmissions, Alice and Bob must reveal as little information as possible while still ensuring that they end up with identical keys. They can do this by agreeing upon a random permutation of the bits in their strings (to randomize the locations of errors), and then splitting the resulting string into blocks of size k. The constant k is chosen so that each block is unlikely to contain more than one error. This k is usually chosen by experiment rather than theory. Alice and Bob then compare the parity of each block.

Parity function X:

$$\oplus X = \begin{cases} 1 & \text{if X has odd number of 1's in it's binary form} \\ 0 & \text{otherwise} \end{cases}$$

Example:

$$\oplus(X \oplus X) = 0$$

If Alice and Bob find a pair of blocks with mismatched parities, they continually bisect the block into smaller and smaller blocks, comparing parities each time, until the error is found. To ensure that Eve learns nothing from this process, Alice and Bob discard the last bit of each block whose parity they disclose.

After completing this process once, there will still be mismatches in those blocks which happened to contain an even number of errors. So Alice and Bob repeat the process several more times with increasing block sizes until they believe the total number of errors to be low. At this point, the above strategy becomes inefficient because Alice and Bob must discard a bit for each block they compare, and the probability of finding an error in each block is low. So Alice and Bob switch to a

[3] superposition of quantum states with several photons

[4] statistical mixture of coherent states

new strategy, which they again perform multiple times. Each time, they choose a random subset of the bit positions in their complete strings, and compare parities. The probability of disagreement if the subset strings are not identical is exactly 1/2. If a disagreement occurs, a bisective search for the error is performed, this time using random subsets rather than blocks. The last bit of each subset is discarded. Eventually, all the errors will have been removed, and Alice and Bob will go through enough parity checks without discovering any errors that they may assume their strings are identical.

At this point, Alice and Bob posses identical strings, but those strings are not completely private. Eve may have gained some information about them. Although this second strategy may cause some errors in Bob's string, if Eve uses it on only a small number of bits, the induced errors will be lost among the errors caused by noise in the detectors and other physical problems. During the reconciliation phase, Eve did not gain any information, since the last bit of each parity check set was discarded. However, some of her original information about specific bits may have been converted to information about parity bits. For instance, if she knew the value of a bit x in string y, and Alice and Bob revealed the parity of y and discarded x, Eve would then know the parity of the remaining bits of y. If we say that Eve knows a parity bit about a string if she knows the parity of a non-empty subset of that string, then if Eve started out knowing at most k physical bits of the key, she will know at most k parity bits of the key after reconciliation [1].

In any case, Alice and Bob share an n-bit string S, and we will suppose that Eve knows at most k deterministic (i.e. parity or physical) bits of S. Alice and Bob wish to compute an r-bit key K, where $r < n$, such that Eve's expected information about K is below some specified bound. To do so, they will choose a compression function g:

$$g : \{0,1\}n \rightarrow \{0,1\}r$$

and compute $K = g_{(S)}$. The question is, what kinds of functions are appropriate for this purpose? That is, which functions, when applied to S, will yield a K about which Eve knows almost nothing?

Definition: A class G of functions $A \rightarrow B$ is universal if for any distinct x_1 and x_2 in A, the probability that $g_{(x_1)} = g_{(x_2)}$ is at most $\frac{1}{|B|}$ when g is chosen at random from G according to the uniform distribution.

An example of a universal class is the set of permutations of A onto itself, since for any g in the set, the probability that $g_{(x_1)} = g_{(x_2)}$ is zero, which is less than $\frac{1}{|A|}$. It is shown that if Eve knows k deterministic bits of S, and Alice and Bob choose their compression function g at random from a universal class of hash functions $\{0,1\}n \rightarrow \{0,1\}r$ where $r = n - k - s$ for some safety parameter $0 < s < n - k$, then Eve's expected information about $K = g_{(S)}$ is less than or equal to $\frac{2-s}{\ln 2}$ bits. One such hash function to generate K is for Alice and Bob to compute an additional r random subset parities of S, this time keeping the results secret. The r results of the parities will be the final r-bit key.

Given this result, one might ask how Alice and Bob are to determine the value of k, i.e. how much information has been leaked to Eve. As a conservative estimate, they can simply assume that all transmission errors were caused by eavesdropping (although most likely some came from detection errors). Alice and Bob can use the beam intensity m and the bit error rate to calculate the expected fraction of S that Eve has learned. If they are conservative in their assumptions and add several standard deviations to their results, they will have a safe upper bound on the number of bits leaked to Eve.

The above discussion assumes that Eve knows only deterministic bits, so another issue is whether it might be more useful to her to obtain probabilistic information about S instead. In other words, rather than measuring photons in the same bases as Alice and Bob, she could pick a basis halfway in between them. This will give her a result that matches Alice's with probability approximately 85%, regardless of which basis Alice uses. She will not gain any information when Bob reveals his measurement choices, so with this strategy all of her information is probabilistic rather than deterministic. Conceivably, this probabilistic information could be more resistant to privacy amplification than deterministic information. However, it turns out that this is not the case, so if Eve wishes to optimize her expected information on the final key, she should use the same bases as Alice and Bob, obtaining only deterministic bits.

5 New development, accomplishments and weaknesses

One of the first practical models of *Quantum Cryptography Machines* was made by *Charles H. Bennett, Francois Bessette* and *Gilles Brassard* [1]. In that model, the quantum channel itself was free air optical path of approximately 32 centimeters. The problem with implementing their system over long distances is that fiber optic cables ruin the polarization of the light, so the photons need to be sent through a vacuum in a straight tube. A 200 yard quantum channel was built in 1992 using interference rather than polarization, because interference patterns survive somewhat better in fiber optic cables [3]. Today using fiber-optical quantum channel has increased this path to 67km.[2] Optical fibers, in spite of their very high quality, are not perfectly transparent. When propagating, a photon will sometimes get absorbed and thus not reach the end of the fiber. Using repeaters is impossible since they corrupt the key according to uncertainty principle of quantum physics. The other disadvantage of quantum key distribution is that its key exchange rate, is still low compared to the bit rates common in conventional telecommunication. Using Bennett's protocol, which has been described previously, typically exchanges a thousand bits per second.[1] This low bit rate is the price to pay for absolute secrecy. The bits exchanged using quantum cryptography constitute a key, which is then used to encrypt data. These data can then be exchanged over a conventional channel at a high rate.

6 conclusion

Initially, quantum cryptography was thought of by everyone mostly as a work of science-fiction because the technology required to implement it was out of reach. The main breakthrough came when Bennett and Brassard realized that photons were never meant to *store* information but to *transmit* it. Later on they came up with their practical protocol. Yet building such machine was expensive. A decade later we can see the quantum cryptography machine is made and available in market. However, computing speed of current computers and complexity of factoring for regular computers, do not bring up the necessity of having a quantum cryptography machine. Today, growth in computer speed and new methods of using parallelism in computing, (ie. DNA computing methods) are increasing the demand for more secure channels, and quantum cryptography could be a solution for this need.

References

[1] C. H. Bennett, F. Bessette, G. Brassard, L. Salvail, and J. Smolin. Experimental quantum cryptography. *J. Cryptography*, 1992.

[2] D. Stucki, N. Gisin, and O. Guinnard. Quantum key distribustion over 67 km with a plug and play system. Presented in Switzerland March 2002.

[3] C. Zimmer. Perfect gibberish. *Discover*, pages 92–99, September 1992.