Quantum Key Distribution Protocols: BB84 & E91

Damiane Kapanadze

Introduction:

This paper will discuss two quantum key distribution protocols. BB84 uses the property of superposition to ensure safe communication, and E91 uses entanglement to achieve the same goal. The paper also discusses the necessity of these technologies and explains the quantum properties needed to understand why it works.

Key distribution:

Some of the encryption systems are secure from quantum computer attacks. Quantum computers will need more than a trillion years to break symmetric encryption (AES-GCM). But it takes a matter of hours to break the asymmetric one. See the table below for more information. But does having quantum computer-proof encryption mean secure communication? No, because if someone was to find out the key used for encryption, they would be able to decrypt without the need for a quantum computer. So the weak point in the system is key distribution. If key distribution is compromised, so is the whole communication. The commonly used key distribution algorithm is Diffie–Hellman key exchange, but it can be broken with quantum computers. That is the main threat to secure communication in the quantum era. So to ensure secure communication, researchers and experts came up with quantum key distribution. We will discuss two protocols in this paper.

TABLE 4.1 Literature-Reported Estimates of Quantum Resilience for Current Cryptosystems, under Various Assumptions of Error Rates and Error-Correcting Codes

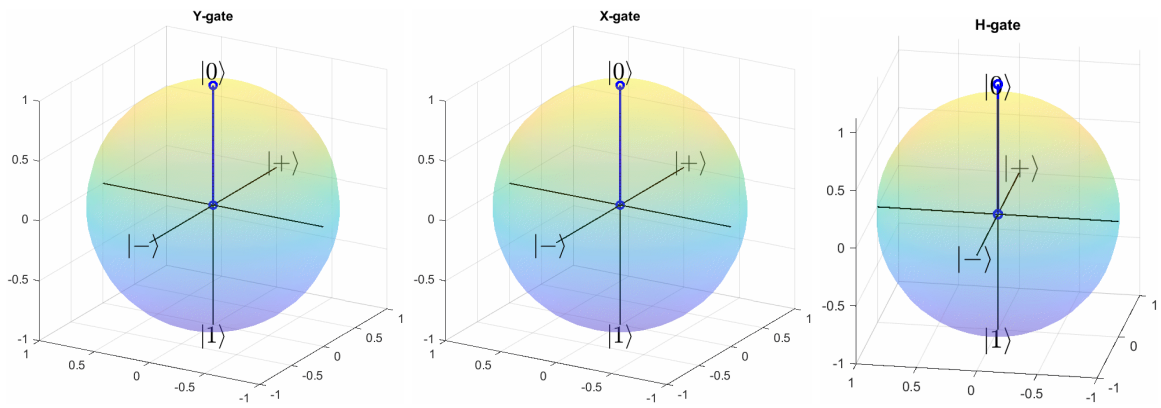
Cryptosystem	Category	Key Size	Security Parameter	Quantum Algorithm Expected to Defeat Cryptosystem	# Logical Qubits Required	# Physical Qubits Required[a]	Time Required to Break System[b]	Quantum-Resilient Replacement Strategies
AES-GCM[c]	Symmetric encryption	128 192 256	128 192 256	Grover's algorithm	2,953 4,449 6,681	4.61×10^6 1.68×10^7 3.36×10^7	2.61×10^{12} years 1.97×10^{22} years 2.29×10^{32} years	
RSA[d]	Asymmetric encryption	1024 2048 4096	80 112 128	Shor's algorithm	2,050 4,098 8,194	8.05×10^6 8.56×10^6 1.12×10^7	3.58 hours 28.63 hours 229 hours	Move to NIST-selected PQC algorithm when available
ECC Discrete-log problem[e-g]	Asymmetric encryption	256 384 521	128 192 256	Shor's algorithm	2,330 3,484 4,719	8.56×10^6 9.05×10^6 1.13×10^6	10.5 hours 37.67 hours 55 hours	Move to NIST-selected PQC algorithm when available
SHA256[h]	Bitcoin mining	N/A	72	Grover's Algorithm	2,403	2.23×10^6	1.8×10^4 years	
PBKDF2 with 10,000 iterations[i]	Password hashing	N/A	66	Grover's algorithm	2,403	2.23×10^6	2.3×10^7 years	Move away from password-based authentication

Superposition:

In Quantum mechanics, superposition refers to the idea that a particle, like an electron or photon, can exist in multiple states at the same time until we measure it. Once measured, it collapses to one of the states. Let's define superposition in quantum bits by comparing it to regular bits. Regular bits have two states: 0 and 1. And the bit can be either in one or the other. For the quantum bit, let's imagine a sphere with $|0\rangle$ state on the north pole and $|1\rangle$ state on the south. The qubit can be represented as a point on the surface of the sphere and is a combination of both states. Once measured, it collapses in one depending on the probabilities, so the measurement always gives $|1\rangle$ or $|0\rangle$. Quantum computing takes advantage of this behavior.

Gates:

Now let's talk about gates. On a single regular bit, there is only one gate, NOT gate that can be applied. And the NOT gate flips the bit from $|0\rangle$ to $|1\rangle$ or vice versa. The equivalent of NOT gates in qubits are X-gate and Y-gate. Both of them turn qubit form $|0\rangle$ to $|1\rangle$ or vice versa, but the difference is how they do it. If we imagine the qubit as a vector in a Bloch sphere, the X-gate rotates the vector 180 degrees over the x-axis, and the y-gate rotates the vector 180 degrees on the y-axis. There is also a Z-gate, which you can probably guess, rotates the vector 180 degrees over the z-axis. And the important point to keep in mind is that these gates if applied twice, return the vector to the initial position. There is a Hadamard gate, which, applied once, creates a superposition, applied twice - returns to the initial position. It rotates the vector around the x=z or y=z line. (see the figure below)



BB84:

The BB84 protocol is named after its creators, Charles Bennes and Gilles Brassard, and the year of its creation, 1984. The protocol was developed to enable two parties to securely exchange cryptographic keys over an insecure quantum communication channel. To achieve this,

the protocol uses the principles of quantum mechanics, specifically the superposition of quantum states and the no-cloning theorem.

For the sake of the example, let's imagine we have Alice, Bob, and Eve. Alice wants to communicate with Bob without Eve's eavesdropping. Alice and Bob need to agree on a key but also make sure that Eve doesn't know it.

To achieve this, Alice chooses a key of classical bits with a length of 4n, where n is a big number to ensure more secure communication. The key is all $|0\rangle$ states a t first, but she applies X-gate randomly, in the end getting 2n of $|0\rangle$ and 2n of $|1\rangle$ states. Let's imagine this on a small scale, until now, Alice has something like this: $|1\rangle$ $|0\rangle$ $|1\rangle$ $|1\rangle$ $|0\rangle$ $|1\rangle$ $|0\rangle$ $|0\rangle$. Now she randomly chooses bases for the qubits: rectilinear or diagonal. If she chooses rectilinear, she doesn't change the qubit, but if she chooses diagonal, she applies the Hadamard gate to the qubit. So if for the bases she gets: R R D R R D D D (R for recliner and D for diagonal), she modifies her key and gets: $|1\rangle$ $|0\rangle$ $1/\sqrt{2}(|0\rangle-|1\rangle)$ $|1\rangle$ $|0\rangle$ $1/\sqrt{2}(|0\rangle-|1\rangle)$ $1/\sqrt{2}(|0\rangle+|1\rangle)$ $1/\sqrt{2}(|0\rangle+|1\rangle)$. So she has two stings: one consisting of $|1\rangle$s and $|0\rangle$s, and another of Rs and Ds. And now she sends the qubits on the quantum channel to Bob. The quantum channel in practice is usually fiber optics cables, and the qubits are photons.

Once Bob receives the key, he randomly chooses bases to measure each qubit, with equal probability. For example, he chose this combination: D R D R D R R D. He then applies H-gate to the qubits he chose to measure as diagonal states. In this case, position 2,3,4,8 definitely matches:

R R D R R D D D

D R D R D R R D

And then he measures the qubits, so he ends up with two stings of 4n length as well: the measurement resulting in |1⟩s and |0⟩s, and another of Rs and Ds.

Since both of them are choosing their bases at random with equal probability, half of the time they will end up using the same bases and the other half different. Since Hadamard gate is reversible, when both of them have the same base, Bob will end up with the same qubit that Alice started with. If they choose different bases, then Bob will get the right qubit only half of the time.

Now they compare their strings of Rs and Ds on an unencrypted line and only keep the qubits where they both used the same base and discard the rest. So, in the example, they would be left with RDRD (position 2,3,4,8). If Eve hasn't intercepted the communication, the length of this string will be about 2n.

If Eve intercepted the line, she would like to clone it and send the clones to Bob while measuring the original herself. But it is impossible because of the non-cloning theorem. So the best she can do is to choose the base at random, measure the qubits, and send the already measured qubits to Bob.

Let's look at the qubits at which Alice and Bob agree with the base. Then half of the time, Eve would also agree. Then all of them would get the same bit as the original provided by Alice. But the other half of the time, when Eve chooses the wrong basis, then she will send a qubit that is in a superposition of Bob's basis states. Giving him the right bit only half of the time.

So if Eve didn't eavesdrop, the bit strings of the length of 2n would be identical, but if she did, Bob's string of |1⟩s |0⟩s would not be identical to Alice's. So Bob and Alice compare half of their bits - n - over an unencrypted communication. If they are identical, then Eve didn't

eavesdrop, and they can use the rest of n bits as an encryption key. If they disagree on a quarter of the bits, they know that Eve is intercepting their communication, and it's not safe.

Here is a table demonstrating what we just discussed:

Alice's bit	$\|1\rangle$	$\|0\rangle$	$\|1\rangle$	$\|1\rangle$	$\|0\rangle$	$\|1\rangle$	$\|0\rangle$	$\|0\rangle$
Alice's basis	X	X	H	X	X	H	H	H
Bob's basis	H	X	H	X	H	X	X	H
Bob's bits	R	$\|0\rangle$	$\|1\rangle$	$\|1\rangle$	R	R	R	$\|0\rangle$
Alice approves			✓					✓
Private Key		$\|0\rangle$		$\|1\rangle$				

X - X-gate, H - Hadamard gate, R - random, ✓ - approved

Entanglement:

Entanglement means that at least two particles are entangled, correlated in such a way that the state of one cannot be described independently of the state of the other. When particles are entangled, their properties such as spin, polarization, or angular momentum, are intertwined and influence one another. So if one is measured, the other also behaves like it was measured and collapses to one of the states. For example, let's imagine entangled qubits where, when one is

measured, the second one also collapses to the same state. That would be denoted like this:

$\frac{1}{\sqrt{2}}|\uparrow\rangle|\uparrow\rangle + \frac{1}{\sqrt{2}}|\downarrow\rangle|\downarrow\rangle$

E91:

The name of this protocol comes from its creator Arthur Ekret, and the year of its creation 1991. The protocol uses the quantum mechanical property of entanglement and ensures secure communication by detecting eavesdroppers if there is one. Let's again discuss the protocol on Alice and Bob's example.

The qubits need to be prepared beforehand, by Alice or a third party and then distributed to Alice and Bob. The qubits need to be entangled in a way that when one is measured, the other collapses in the same state. So like this: $\frac{1}{\sqrt{2}}|\uparrow\rangle|\uparrow\rangle + \frac{1}{\sqrt{2}}|\downarrow\rangle|\downarrow\rangle$.

When Alice and Bob have distributed the entangled qubits, Alice chooses a basis to measure it using 3 different bases chosen at random. So ⅓ of the time, They end up using the same bases, but ⅔ of the time - different. Since the qubits are entangled, when they use the same basis, they get the same bit as well. But when they use different bases, they have a ¼ chance of measuring the same bit. Here is the detailed explanation:

Let's imagine Alice chooses ($|\searrow\rangle$,$|\nwarrow\rangle$), and Bob chooses ($|\swarrow\rangle$,$|\nearrow\rangle$). So when Alice makes measurement, the qubits jump to $|\searrow\rangle|\searrow\rangle$ or $|\nwarrow\rangle|\nwarrow\rangle$, and she will write down 0 or 1 respectively. Now Bob has to make a measurements. Let's suppose that Alice's measurement was $|\searrow\rangle$, then Bob's qubit will also be in state $|\searrow\rangle$. We have to rewrite this on Bob's basis. Orthonormal basis in direction Θ is: $\left(\begin{bmatrix} cos(\frac{\Theta}{2}) \\ -sin(\frac{\Theta}{2}) \end{bmatrix}, \begin{bmatrix} -sin(\frac{\Theta}{2}) \\ cos(\frac{\Theta}{2}) \end{bmatrix}\right)$. So Alice's basis is $\left(\begin{bmatrix} \frac{1}{2} \\ \frac{\sqrt{3}}{2} \end{bmatrix}, \begin{bmatrix} \frac{\sqrt{3}}{2} \\ -\frac{1}{2} \end{bmatrix}\right)$ and Bob's is

$\left(\begin{bmatrix} \frac{1}{2} \\ -\frac{\sqrt{3}}{2} \end{bmatrix}, \begin{bmatrix} \frac{\sqrt{3}}{2} \\ \frac{1}{2} \end{bmatrix}\right)$. If Alice gets 0 - $\begin{bmatrix} \frac{1}{2} \\ \frac{\sqrt{3}}{2} \end{bmatrix}$, then Bob's measurement will be $\begin{bmatrix} \frac{1}{2} & -\frac{\sqrt{3}}{2} \\ \frac{\sqrt{3}}{2} & \frac{1}{2} \end{bmatrix} \times \begin{bmatrix} \frac{1}{2} \\ \frac{\sqrt{3}}{2} \end{bmatrix} = \begin{bmatrix} -\frac{1}{2} \\ \frac{\sqrt{3}}{2} \end{bmatrix}$. And $\begin{bmatrix} -\frac{1}{2} \\ \frac{\sqrt{3}}{2} \end{bmatrix} = -\frac{1}{2}\begin{bmatrix} \frac{1}{2} \\ -\frac{\sqrt{3}}{2} \end{bmatrix} + \frac{\sqrt{3}}{2}\begin{bmatrix} \frac{\sqrt{3}}{2} \\ \frac{1}{2} \end{bmatrix}$, so there is $(-½)^2 = ¼$ chance of Bob getting 0 as well. he same goes for if Alice chooses 1, or they measure on another basis. As long as they are choosing their basis from 0°, 120°, and 240°.

To go back to Alice and Bob's communication. After measuring the bits, they share the basis they used for the measurement on public communication service. And then they share the bits for which they used different bases. If ¼ of the bits match, then they know that the communication was secure and no one has intercepted it. So they use the bits they didn't share, the pens where they had the same bases, as a private key for encryption.

Now what happens if the communication is intercepted? If Eve tries to intercept the communication, then she needs to randomly choose the basis as well. That changes the probability of Alice's and Bob's bits matching when the bases don't match from ¼ to ⅜. So when they compare their bits, if instead of ¼, ⅜ of them match, they know that the communication wasn't secure, and they will not use the private key.

Here is a table demonstrating what we just discussed:

Alice's basis	0°	120°	0°	240°	120°	240°
Alice's bits	$\|1\rangle$	$\|0\rangle$	$\|1\rangle$	$\|1\rangle$	$\|0\rangle$	$\|1\rangle$
Bob's basis	240°	0°	120°	240°	120°	0°
Bob's bits	R	R	R	$\|1\rangle$	$\|0\rangle$	R

Alice approves	X	✓	X			X
Private Key				$\|1\rangle$	$\|0\rangle$	

[0°, 120°, 240°] - degree of basis, R - random, ✓ - approved, X - denied

Bibliography:

Gheorghieș, A.-Ș., Lăzăroi, D.-M., & Simion, E. (n.d.). *A Comparative Study of Cryptographic Key Distribution Protocols*. https://eprint.iacr.org/2021/031.pdf

Lecture 12: Quantum key distribution. Secret key. BB84, E91 and B92 protocols. Continuous-variable protocols. (n.d.). https://mpl.mpg.de/fileadmin/user_upload/Chekhova_Research_Group/Lecture_4_12.pdf

Board, T. (2019). Quantum Computing. In *National Academies Press eBooks*. https://nap.nationalacademies.org/read/25196/chapter/6#98

Bernhardt, C. (2020). Quantum Computing for Everyone.