Quantum Cryptography: How Bill and Tanisha Outsmart Nathan

Maria Florenica Nardone
Department of Physics and Astronomy, The University of North Carolina at Chapel Hill
(Physics 521, in collaboration with Carter Chapman)
(Dated: November 21, 2025)

Quantum cryptography represents a fundamental shift in secure data communication, leveraging basic principles of quantum mechanics to achieve unparalleled security. This paper explores classical forms of cryptography and their downfalls, leading to the emergence and ultimate need for quantum cryptography. This paper presents a detailed analysis of how quantum cryptography differs from modern cryptographic methods and what advantages it provides. Specifically, this paper examines two primary quantum key distribution (QKD) protocols: BB84 and E91. Through analysis of eavesdropping methods and understanding of the quantum bit error rate (QBER), it is evident that QKD makes eavesdropping physically detectable rather than computationally difficult. QKD presents itself as a promising solution to the key distribution problem faced by classical cryptography and will likely be the future for information security in the post-quantum computing era.

I. INTRODUCTION

Quantum cryptography emerges from the limitations of classical systems. To understand why QKD is revolutionary, we must first understand the classical key distribution problem it solves and why existing solutions fail against the imminent threat of quantum computers

II. BACKGROUND: CLASSICAL CRYPTOGRAPHY

Classical cryptography primarily takes two forms: symmetric and asymmetric cryptography.

A. Symmetric Cryptography

Symmetric cryptography is the simpler and older of the two encryption methods. In symmetric encryption, there is one shared secret key between two individuals. Take for example Bill and Tanisha, two undercover spies. Bill wants to send a secret message to Tanisha but wants to make sure that anyone who intercepts the message will not understand it. Say he wants to send the message "outside" ("outside" is considered the plaintext). Bill and Tanisha met up previously and decided that their shared secret key was: 5. This means that for every letter in their plaintext, they would move 5 letters up the alphabet. Therefore "outside" becomes "tzyxnij" ("tzyxnij" is considered the ciphertext). Now Bill can send Tanisha "tzyxnij" without anyone else knowing what he said, and Tanisha knows that she has to reverse 5 letters for each letter to decrypt Bill's message. In this way, Bill and Tanisha use the same key to encrypt and decrypt the message. This simple example uses a Caesar cipher for illustration, but modern symmetric encryption employs far more complex mathematical operations.

This form of encryption can be very safe when using modern algorithms. AES (Advanced Encryption Standard) uses multiple rounds of encryption (composed of substitutions, permutations, and mixing) to create ciphertext that is computationally infeasible to decrypt without knowledge of the secret key. For example, trying to decrypt an encrypted message that was encrypted using AES-128 (a key with 128 bits or 16 bytes) via brute force is estimated to take a classical computer one quintillion (10^{18}) years [1], which is 7 orders of magnitude larger than the age of the universe. This is not to mention AES-192 and AES-256, which provide even stronger security with 192-bit and 256-bit keys respectively, making them effectively immune to brute-force attacks for the foreseeable future.

Quantum computers using Grover's algorithm would be able to perform brute-force key attacks using quadratically fewer steps than a classical computer, effectively reducing the security strength of symmetric keys by half [2]. However, even with this quantum advantage, the timescale required for a quantum computer to brute-force decrypt AES-128, AES-192, and especially AES-256 still remains computationally infeasible. In this way, symmetric encryption is considered quantum-resistant, and AES-256 in particular is recommended for long-term security in the post-quantum era.

However, symmetric encryption faces a critical challenge known as the key distribution problem. Before Bill and Tanisha can communicate securely using symmetric encryption, they must both have the same secret key. This then begs the question: how do they securely share this key in the first place? Note that if the key is intercepted during transmission, an attacker can decrypt all communications encrypted with that key, completely compromising the security of the system.

B. Asymmetric (Public-Key) Cryptography

Asymmetric cryptography, also known as public-key cryptography, was developed to solve the key distribution problem inherent in symmetric systems. Unlike symmetric encryption, asymmetric encryption does not rely on

this shared secret key. Instead, asymmetric encryption uses a mathematically related pair of keys (a public key and a private key). The public key can be freely distributed to anyone and is used to encrypt messages, while the private key is kept secret by the owner and is used to decrypt messages. The mathematical relationship between these keys ensures that data encrypted with the public key can only be decrypted with the corresponding private key, and vice versa.

Say Bill heard rumors that secret foreign agents were trying to overhear his conversations with Tanisha. If they meet at a park to exchange the secret key in person, someone could be there listening in on their conversation. Since they cannot safely meet in person without risk of being observed, they need another solution. To receive the secret message, Tanisha creates a set of key pairs using a mathematical algorithm. She then freely shares her public key online. Bill then uses Tanisha's public key to encrypt his message. Once encrypted, not even Bill himself can decrypt the message. Only Tanisha, with her secret private key, will be able to decrypt Bill's message. This can all be done without Bill having any knowledge of Tanisha's private key. This means that even if someone intercepts the message, they must have Tanisha's secret key to decrypt it.

This can also be visualized as a mailbox with a mail slot. Anyone can deposit a letter in the mailbox because everyone knows where the mailbox is and can send letters to them. But only the person with the key to the mailbox can open and read its content.

There are many mathematical algorithms used to make these related pairs of keys, all of which are based on "one-way functions." This means that they are easy to solve in one direction but extremely hard to reverse without specific information (the private key). The most widely used asymmetric encryption algorithm is RSA (Rivest-Shamir-Adleman). RSA's security relies on integer factorization. When creating an RSA key pair, two large prime numbers are multiplied. The multiplication is easy to do, but reversing the process and attempting to find the two prime numbers that were used to create the product is very difficult for a classical computer to solve. Modern day implementations of RSA, considered to be secure, use prime number products of 2048 or 4096 bits. Longer keys are more secure but take more time to generate and require more processing power for encrypting and decrypting. Another asymmetric algorithm commonly used is ECC (Elliptic Curve Cryptography), which bases its security on the difficulty of the elliptic curve discrete logarithm problem.

However, asymmetric encryption faces a severe threat from quantum computers. Unlike symmetric encryption, which is quantum-resistant, asymmetric encryption algorithms like RSA and ECC are fundamentally vulnerable to quantum attacks. This is because asymmetric encryption relies on the existence of a "really hard math problem," while symmetric encryption is based on the existence of a "random" shared secret key. Peter Shor proved the vulnerability of asymmetric encryption with the creation of an algorithm (now called Shor's algorithm) which can efficiently solve both the integer factorization problem and the discrete logarithm problem when run on a sufficiently powerful quantum computer. What may be infeasible to solve with a classical computer, a quantum computer running Shor's algorithm (and with sufficient qubits) could potentially break in hours [3].

The reason for this dramatic difference in solving capabilities is based on how quantum computers process information. Classical computers run on bits which either represent a binary 1 or 0. Quantum computers use quantum bits (qubits) which can exist in a superposition of states, meaning they can simultaneously represent multiple possible combinations of 1 and 0. This means quantum computers can evaluate many possible factors at the same time rather than checking them sequentially. Shor's algorithm takes advantage of such quantum phenomena (such as superposition and entanglement) to find the period of a mathematical function (such as the ones that govern RSA and ECC) exponentially faster than classical algorithms. Using Shor's algorithm and sufficient qubits, what would take a classical computer millions of years to crack is estimated to take a quantum computer a couple of hours [4]. Currently, quantum computers are not powerful enough to decrypt modern RSA and ECC algorithms. However, many researchers and governments are preparing for the day quantum computers do become powerful enough to break asymmetric cryptography. In fact, many cyber criminals are employing the "harvest now, decrypt later" strategy, in which they collect encrypted data in the hope that once quantum computers become available, they will be able to decrypt it.

C. Asymmetric Encryption and Post-Quantum Cryptography

This vulnerability in modern day key exchange has accelerated the move toward post-quantum cryptography (PQC), also known as quantum-resistant cryptography. PQC are advanced algorithms that remain secure against both classical and quantum computers. These new algorithms are based on mathematical problems that remain difficult even for quantum computers such as lattice-based, isogeny-based and multivariate cryptography. Unlike symmetric encryption, which remains secure (especially with increased key sizes), asymmetric encryption requires fundamentally new algorithms to survive in a post-quantum world.

While PQC represents one approach to living in a quantum computing world, it still relies on mathematical assumptions that could later be broken. An alternative to PQC and asymmetric cryptography as a whole is Quantum Key Distribution (QKD), a method that uses the fundamental laws of quantum mechanics to solve the key distribution problem that has plagued symmetric encryption from the beginning.

III. QUANTUM KEY DISTRIBUTION (QKD)

As mentioned previously, the main problem with symmetric encryption is the fact that classically there is no assured way Bill can securely share the secret key with Tanisha. With classical cryptography there is no guarantee that an eavesdropper couldn't simply intercept the message containing the secret key. It is important to note that symmetric cryptography is only as secure as the method by which the secret key is shared and later stored. Quantum key distribution (QKD) solves the problem of sharing the secret key securely.

Unlike PQC, which relies on a different "hard math problem," QKD relies solely on the fundamental laws that govern quantum mechanics which are inherent to nature. This means that (as per our current knowledge), QKD is not vulnerable to future algorithmic breakthroughs or advances in computing power. The security of QKD is guaranteed by basic quantum mechanical principles.

QKD is a method by which a secret key can be established between two parties without anyone else even having the capability to figure out what the secret key is. This can be done using a stream of polarized photons or other quantum particles. The protocol for this will be described shortly but the idea is that after this secret key is established, classical symmetric encryption will follow.

IV. FUNDAMENTAL QUANTUM PRINCIPLES BEHIND QKD

QKD's security is based on the fact that quantum information cannot be measured or copied without leaving a detectable trace. This is a manifestation of several quantum properties:

A. Wave Function Collapse

The act of measuring a quantum state causes the wave function of the particle to collapse, fundamentally changing it. This wave function collapse destroys any original information the particle held. This happens because the act of measuring projects the state onto an eigenstate of the measurement operator, irreversibly destroying the original superposition and yielding a probabilistic outcome of the new basis state.

Cryptographic implication: Say Nathan, an evil secret spy trying to eavesdrop on Bill and Tanisha's conversation, was trying to crack their secret key. If he does this by intercepting and measuring the quantum particles, he would inevitably disturb them. Specifically, if Nathan chooses a measurement basis that differs from the preparation basis (that Bill prepared and sent to Tanisha), the measurements yield random results which detectably disturb the state.

B. No-Cloning

The no-cloning theorem states that it is impossible to create an exact copy of an unknown quantum state. Mathematically there exists no unitary operator U that satisfies:

$$U|\psi\rangle|0\rangle = |\psi\rangle|\psi\rangle$$

for all quantum states $|\psi\rangle$, where $|0\rangle$ represents a blank state [5].

Cryptographic implication: It is impossible for Nathan to intercept quantum information, create a perfect copy, and forward the original state without disturbing it. The act of "information extraction" will inevitably alter the quantum state.

C. Heisenberg Uncertainty Principle

The Heisenberg uncertainty principle states that certain pairs of physical properties cannot be simultaneously known with precision. Specifically, one cannot know the exact position and exact momentum of a particle at the exact same time. For Nathan, this means that he cannot determine both the encoded basis and spin state at the same time.

In this way, quantum information is fundamentally different from classical information and the difference makes eavesdropping detectable with security guaranteed by rudimentary quantum mechanical laws.

V. THE BB84 PROTOCOL: HOW DO BILL AND TANISHA MAKE THE SECRET KEY?

The BB84 protocol, proposed by Charles Bennett and Gilles Brassard in 1984, was the first quantum key distribution protocol [6]. This protocol uses the quantum properties previously described (specifically with respect to photons) to allow Bill and Tanisha to share a secret key while detecting eavesdropping.

A. Summarizing the BB84 Protocol Using Spin-1/2 Particles for Ease of Understanding:

Assume that Bill and Tanisha have previously established that spin up in the X or Z basis is equivalent to a logical bit value of 1 and spin down in the X or Z basis has a logical bit value of 0.

1. Step 1: Preparation and Transmission

Say Bill is sending Tanisha a stream of spin-1/2 particles. He randomly prepares each particle to be in the X or Z basis with a Stern-Gerlach machine so as to collapse

the wave function into one of the two desired bases. He notes down which basis he prepared in and the spin of each particle. He then sends Tanisha this stream through a quantum channel.

2. Step 2: Measurement

Tanisha receives each spin-1/2 particle and measures them one by one. She randomly selects either the X or Z basis to measure in. Note that if she happens to measure in the same basis Bill prepared it in, she will measure the correct spin (thus they will have the same bit value for that particle). If she chooses the wrong basis she will get a random result with a 50% chance of error in the spin value.

3. Step 3: Sifting (Basis Reconciliation)

After all the particles are transmitted and measured, Bill and Tanisha will publicly state (over a regular, non-secure channel) which basis they measured each particle in (not the bit values). They will then each discard all the bits where their basis states did not match (this will likely be approximately 50% of the bits). The remaining bits form what is called the "sifted key." It is crucial to notice that if we are assuming no eavesdropping or loss of bits, Bill and Tanisha will now have a key that is known to them and no one else, even though they may have never talked to each other about the key.

4. Step 4: Error Testing

In order to test for eavesdropping, or any loss of bits not due to imperfect equipment (if not assuming ideal/perfect quantum channel), Bill and Tanisha must publicly compare a random sample of their sifted key. Bill and Tanisha will share a random subset of sifted bits through public channels. They then calculate the Quantum Bit Error Rate (QBER) which is the percentage of bits that do not match.

If there is no eavesdropping and the equipment is working properly, they should expect a very low error rate under 11% due to normal noise and detector imperfections. However, if the error rate is higher than the 11% threshold for BB84 then they know that Nathan has been eavesdropping on their conversation. This is because any measurement by Nathan in a basis different from Bill's preparation basis introduces random error and thus increases the QBER beyond the normal levels. If they notice this, then they will abort the whole protocol and start all over again.

Once confirmed that there has been no eavesdropping, the bits they publicly compared are discarded from their sifted key. The remaining bits are now the secret key. Bill and Tanisha now share a secret key born from nothing more than quantum mechanical principles, with the security of no one else knowing their key and without the need to use asymmetric encryption methods threatened by the emergence of quantum computers. They can then go on to use this secret key in conjunction with classical symmetrical encryption methods (like AES).

VI. THE E91 PROTOCOL

The E91 protocol, proposed by Artur Ekert in 1991, is more complex and is based on a source that generates a pair of entangled photons [7].One of each pair is then sent to Bill and Tanisha respectively. In this method Bill and Tanisha measure at 3 different measurement angles. Bill chooses from 0°, 45°, or 90°, while Tanisha chooses from 45°, 90° or 135°. After basis reconciliation they divide their measurements into two groups: one for key generation (values measured at 45° and 90°) and the other for security verification (values measured at 0° and 135°). The security verification values then use principles from Bell's Inequality (specifically its violation with respect to quantum mechanics), calculating the CHSH parameter:

$$S = |E(0°, 45°)+E(0°, 135°)+E(90°, 45°)-E(90°, 135°)|$$

With optimally chosen angles, the value for S is predicted to be $S = 2\sqrt{2} \approx 2.828$. Thus, security criteria is that if S is found to be larger than 2, then the key is secure. On the other hand, if S is found to be less than or equal to 2 then eavesdropping is suspected and the protocol must be aborted.

The advantages of using this protocol over BB84 include device and source independence. The violation of Bell's inequality would prove that a system is secure even if Bill and Tanisha don't trust their measurement devices or the source of the entangled pairs. Thus E91 is not just a simple distribution of keys, rather it performs a fundamental test to check for security.

VII. SOURCES OF ERROR

Errors in QKD come from two main sources:

A. Natural Error

These are forms of error due to equipment whether that be detector noise, particle/photon loss, misalignment of optical components, or multi-photon/particle events.

1. Distance

In the real-world, QKD systems are limited by distance. This is primarily due to photon loss in optical

fibers. As distance increases, more photons are lost, reducing the signal-to-noise ratio and increasing the QBER value. Currently, the record for using quantum encryption via fiber optic is 100 km. [8]

2. Multi-photon/particle Events

QKD assumes the ability of a laser to perfectly and consistently emit one photon at a time, but this is not assured. When multiple photons are accidentally sent in a single pulse, it creates security vulnerabilities that an eavesdropper could exploit.

B. Eavesdropping

When eavesdroppers (Nathan) interfere with the quantum channel, they introduce additional errors due to measurement disturbance.

1. Intercept-Resend Attack

This is the simplest attack method conducted by Nathan. Here, Nathan will attempt to intercept each particle Bill sends Tanisha. He will then measure the particle randomly (for he has no knowledge of what basis Bill used). Based on the measurement he will then prepare a new particle in the state he measured and send it to Tanisha. After basis reconciliation by Bill and Tanisha they will have their sifted bits. For these sifted bits the error introduced by Nathan is 25%.

This is because there is an inherent 50% chance that Nathan will choose the wrong basis to measure in. Say Tanisha then goes on to measure in the correct basis (Bill's original basis), she now has a 50% chance of error in the bit value. Yielding a QBER $= 0.5 \times 0.5 = 0.25 =$ 25%, surpassing the threshold for BB84 and alerting Bill and Tanisha of eavesdropping.

2. Other Eavesdropping Methods

There exists a multitude of eavesdropping methods including taking partial measurements and the entanglement-probe method. In either case, the more information Nathan tries to gain, the higher the QBER becomes, and the more likely it is that Bill and Tanisha will notice something is wrong and abort the protocol.

VIII. CONCLUSION

As quantum computers continue to advance, much of the public-key infrastructure that secures today's digital world will eventually be at risk. QKD provides the solution. Although it currently faces technical challenges including distance limitations and the need for reliable source and measurement devices, QKD holds the promise of a new level of security; one that is unconditional, guaranteed by fundamental quantum mechanical principles rather than mathematical assumptions.

Unlike classical cryptography, which depends on the assumed difficulty of certain mathematical problems, QKD's security is grounded in physics itself. Principles like the no-cloning theorem, measurement disturbance, and the Heisenberg uncertainty principle ensure that any attempt to eavesdrop on the key exchange will leave detectable traces. Using protocols such as BB84 and E91, two parties can generate a shared secret key while monitoring the quantum bit error rate (QBER) to verify whether an interception attempt has occurred.

As the technology matures, QKD will likely become an important part of secure communication systems, helping protect information in the post-quantum era using the fundamental laws of physics. This shift, from security based on hard math problems and computational difficulty, to security rooted in physical principles, represents a fundamental change in how we think about cryptography.

IX. ACKNOWLEDGMENTS

This research was completed by Maria Florencia Nardone but in collaboration with Carter Chapman. Guidance regarding the project was provided by professor Christian Iliadis.

[1] Schneier, B. (1996). Applied Cryptography: Protocols, Algorithms, and Source Code in C. John Wiley & Sons.

[2] National Institute of Standards and Technology (NIST). Post-Quantum Cryptography FAQs. https://csrc.nist.gov/projects/post-quantum-cryptography/faqs

[3] Shor, P. W. (1994). Algorithms for quantum computation: discrete logarithms and factoring.

[4] Kudelski Security Research. Quantum Attack Resource Estimate using Shor's Algorithm to break RSA vs DH/DSA vs ECC. https://kudelskisecurity.com/research/quantum-attack-resource-estimate-using-shors-algorithm-to-break-rsa-vs-dh-dsa-vs-ecc

[5] Wootters, W. K., & Zurek, W. H. (1982). A single quantum cannot be cloned. Nature, 299(5886), 802-803.

[6] Bennett, C. H., & Brassard, G. (1984). Quantum cryptography: Public key distribution and coin tossing. https://doi.org/10.1016/j.tcs.2014.05.025

[7] Ekert, A. K. (1991). Quantum cryptography based on Bell's theorem. Physical Review Letters, 67, 661 https://doi.org/10.1103/PhysRevLett.67.661

[8] Johansson, U. J. (2024). 100 kilometres of quantum-encrypted transfer. DTU News https://www.dtu.dk/english/newsarchive/2024/04/100-kilometres-of-quantum-encrypted-transfer