UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE CIENCIAS FÍSICAS

DEPARTAMENTO DE ÓPTICA

TRABAJO DE FIN DE GRADO

Código de TFG: OPT14

CRIPTOGRAFÍA CUÁNTICA

Quantum Cryptography

Supervisor: Ángel S. Sanz Ortiz

Sara Varas Vicente

Grado en Física

Curso académico 2021-22

Convocatoria Junio

QUANTUM CRYPTOGRAPHY

Resumen:

A lo largo de la historia, la humanidad ha diseñado métodos de encriptación cada vez más sofisticados para guardar información; pero junto a ellos, la manera de romperlos también avanza. El desarrollo de la Mecánica Cuántica en la primera parte del siglo XX hizo tambalear la Criptografía Clásica y la publicación del Algoritmo de Shor puso definitivamente de manifiesto que el poder de los ordenadores cuánticos rompería fácilmente su seguridad. Por ello surge la necesidad de desarrollar nuevos algoritmos de encriptación y generación de claves resistentes: llega la Criptografía Cuántica. Este trabajo se centrará en estudiar sus bases para posteriormente aplicarlas en el protocolo más conocido de Distribución Cuántica de Claves, el BB84. Asimismo se analizarán los posibles ataques y cómo se lleva a cabo experimentalmente. Así, no debemos considerarla como algo futurista sino que es ya una realidad.

Abstract:

Throughout history, humanity has developed increasingly sophisticated encryption methods to keep information secure; but along with them, the way to break them is also advancing. The development of Quantum Mechanics in the early part of the XX century, made Classical Cryptography unstable and the publication of Shor's Algorithm made it definitively clear that the power of Quantum Computers would easily break its security. Hence, the need to develop new algorithms for encryption and resistant keys generation: Quantum Cryptography arrives. This project will focus on studying its foundations and then, applying them to the best-known Quantum Key Distribution protocol, BB84. It will also analyse the possible attacks and how it is carried out experimentally. Thus, we should not consider it as something futuristic because it is already a reality.

Contents

1 Introduction

History has taught us the importance of information. Those who have more, are said to be more powerful. This is why humans create the necessity of keeping information as a secret or even as a real treasure. Encrypted messages are not a new fashion, Julius Caesar the emperor created his own code which consisted of substituting every letter in the Latin alphabet with the letter three positions afterwards. Not that far ago, during the Nazi regime, a German engineer invented the Enigma Machine. Its encrypting settings were changed daily and they could encrypt information without their enemies knowledge. Alan Turing, with the help of the Polish Cipher Bureau, could finally crack the Enigma code in 1939 [1]. Once this happened, this cryptosystem was not longer useful: cryptographic systems are very powerful... until their decryption algorithm is broken.

But, what is the formal definition of a cryptosystem? A cryptosystem is *'a system consisting of an encryption algorithm, a decryption algorithm, and different text spaces: plaintexts (x), ciphertexts (y), and keytexts'* [2]. There are different types depending on the use or not of cryptographic keys:

- **Unkeyed cryptosystem**: a cryptographic system that uses no key.
- **Secret key cryptosystem**: a cryptographic system that uses secret keys that are shared between the participating entities.
- **Public key cryptosystem**: a cryptographic system that uses secret keys that are not only shared between the participating entities.

Within cryptosystems using key, we distinguish three methods:

- **Symmetric key-based encryption**: it uses the same key to encrypt and decrypt data. Some of them are DES and AES (see Fig. 1).

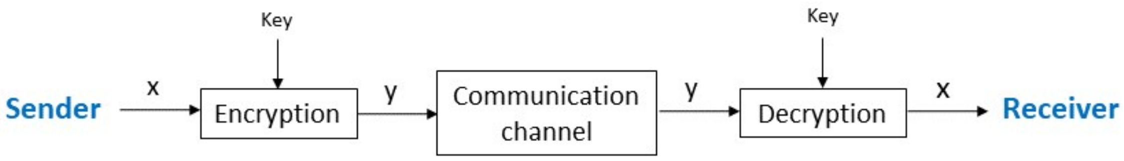


Figure 1: Symmetric encryption scheme.

- **Asymmetric key-based encryption**: it involves the use of different keys to encrypt and decrypt data. The most known, and the one we will explain later, is RSA (see Fig. 2).

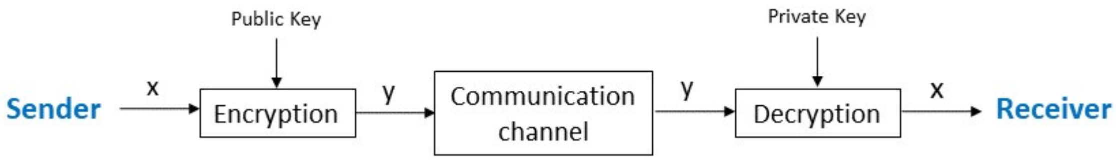


Figure 2: Asymmetric encryption scheme.

- **One-way hash functions**: we will not analyse this method during this work [3].

The methods mentioned before are the most used nowadays. Referring to some of them, their security is based on the current impossibility of factoring large integers, therefore, decrypting data without knowing the key becomes a computational resource challenge that, today, is unthinkable. However, with the arrival of quantum computers and their expected great computational power, the security of this algorithm would be easily compromised. The need to develop new algorithms resistant to the power of a quantum computer is created: quantum-based cryptography arrives [4]. Facing this need, the Quantum Key Distribution (QKD) method arises. It is a way of distributing secret keys to use them after in symmetric encryption methods. But what is truly characteristic of this method is that it follows the laws of Quantum Mechanics. This fact allows us to take advantage of the fundamental aspects of the quantum world, such as the no-cloning theorem or the perturbation of the state when we measure it, among others.

Henceforward, two classical methods will be analysed along with their vulnerabilities to finally reach the quantum section where Quantum Mechanics Principles and QKD will be explained, in particular, the BB84 protocol in depth.

2 Classical Cryptography

The concept of cryptography may not be familiar to most people because we are not aware of our daily contact with it. Almost every technological device or application nowadays uses cryptography methods to protect data: from doing a bank transaction to sending your friends messages in WhatsApp (which it is an example of an application using a public key cryptosystem).

The AES algorithm or the RSA Cryptosystem are two great examples of classical cryptography. In this section, we will analyse shortly the first one, while we will study more in depth RSA. Finally, we will try to determine if they could be post-quantum cryptography, i.e, if they would resist quantum computers attacks.

2.1 The Advanced Encryption Standard (AES)

This symmetric-encryption algorithm is a block cipher; so, to begin with, we will explain what is a block cipher: Given a block of plaintext (x), this type of cipher will encrypt all bits of x at the same time and with the same secret key, i.e, every plaintext bit depend on the rest of bits [5].

The process of the algorithm consists on several rounds of encryption where multiple transformations are applied to data. In the aim of understanding what happens during these transformations, we introduce two concepts:

- **Confusion**: encryption operation whose aim is to make the relation between plaintext, ciphertext and key as complex as possible. We achieve this with bit substitutions.
- **Diffusion**: encryption operation whose purpose is to spread the influence of a single plaintext bit over many ciphertext bits. Its goal is to hide statistical probabilities of the plaintext. We can implement this through bit permutations [6].

These operations are made repeatedly and combined through the rounds in order to achieve strong encryption.

In the case of the AES algorithm, the block length is limited to 128 bits but it allows three key lengths of 128, 192, and 256 bits. The different keys provide different levels of security with more number of rounds. Today, AES-256 is believed to be the most secure because there are no known analytical attacks; brute-force attacks would take several decades since there are 2^{256} different key combinations [7].

2.2 The Rivest–Shamir–Adleman Algorithm (RSA)

The RSA is an asymmetric key cryptosystem. As it is depicted in Figure 2, it works with two different keys: the public key that it is used to encrypt the message and everyone can use it; and the private key, essential to decrypt data and it is only known by one person. This provides an advantage over symmetric cryptosystems since if an eavesdropper gets the key in the AES, he could decrypt all the messages. However, with the RSA, an attacker will not be able to decrypt the message just by knowing the public key [8].
But, how does this algorithm work? We will describe what it is called 'schoolbook RSA' since this system is not secure but it holds the essence of the RSA algorithm.
Key generation is the first step. It consists of the following steps:

1. Find two random different large primes p and q
2. Set $n = p \cdot q$
3. Compute $\phi(n) = (p-1) \cdot (q-1)$
4. Pick a random e $\in 1, 2, \ldots, \phi(n) - 1$ such that gcd $[e, \phi(n)] = 1$
5. Compute d such that $d \cdot e = 1 \cdot \text{mod } \phi(n)$

The public key will be given by the values n and e: $K_{pub} = (n, e)$ while the private key will be given by n and d: $K_{priv} = (n, d)$.

The second step is encrypting and decrypting the message:
Imagine you want to send a message x to your friend. You encrypt x computing $y = x^e \cdot mod(n)$ using $K_{pub} = (n, e)$. Now, you send y to your friend and he proceeds to decrypt the message: he computes $x = y^d \cdot mod(n)$ using $K_{priv} = (n, d)$ only known by him [9]. This is the encryption and decryption procedure used by the RSA.

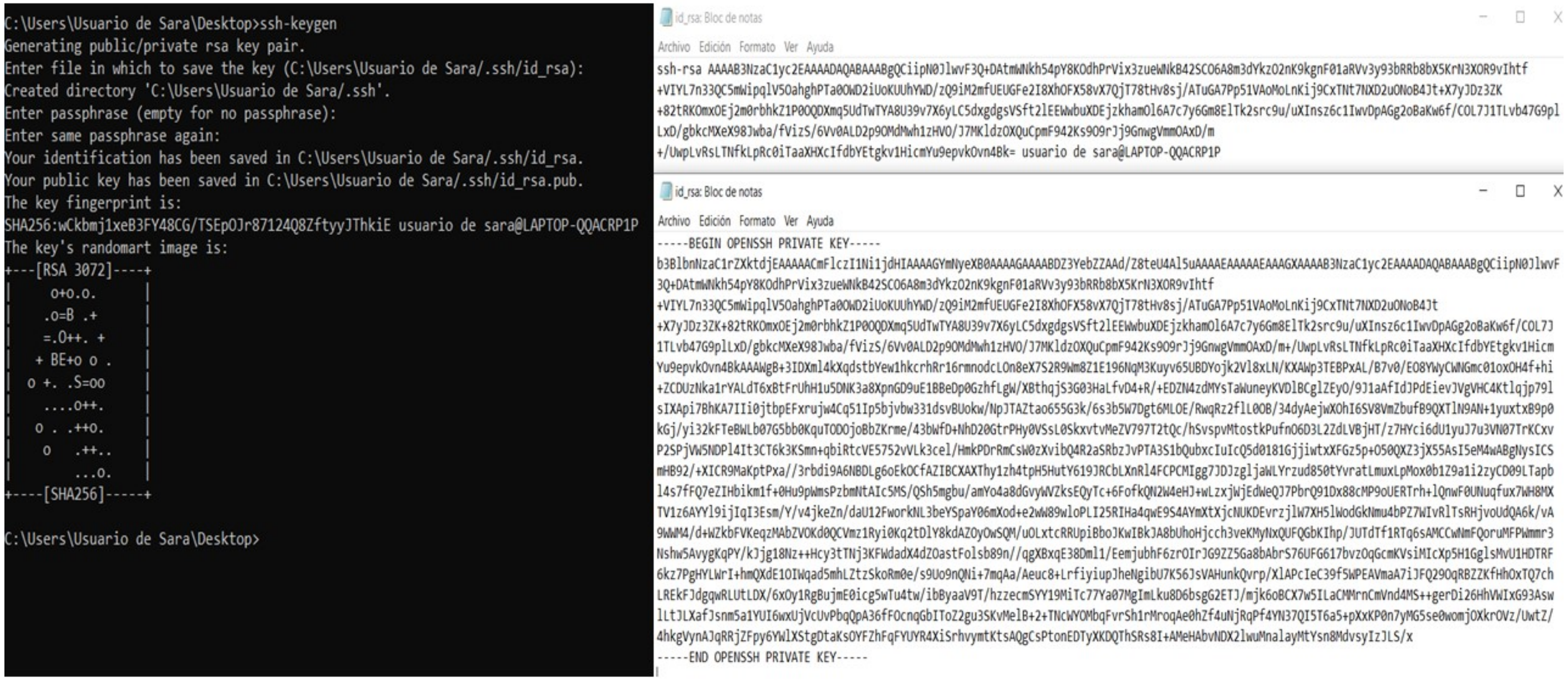
As it has been mentioned earlier, this is not secure enough since RSA is deterministic and malleable, among other defaults [5]. In order to overcome these weaknesses, we introduce *Optimal Asymmetric Encryption Padding* (OAEP). Briefly, this modified algorithm tries to remove the probabilistic trace of the plaintext (similar to the diffusion operation in AES algorithm) by adding random digits to the plaintext before the encryption [10].

Nevertheless, how could an eavesdropper try to steal data? One way is using the public key. He could try to factorise n in order to obtain d and then compute the private key. He would face factorization problem since the known classical algorithms are not able to do this operation in reasonable times. Then, if we use choose large prime numbers p and q, RSA will not be vulnerable to classical computers. There are, though, other attack strategies [10].

To conclude the explanation of the RSA algorithm, it is practical to explain an example. At first glance, the execution of this method can seem far from our computer abilities but actually, you can easily generate a RSA-key pair in your computer. Below it is shown the steps we have followed in a Windows' console in order to generate two documents corresponding to the private and public key, so you can encrypt and decrypt messages.

2.3 Classical Cryptosystems and Quantum Computers

The algorithms described before are two of the most used cryptosystems nowadays. They are widely used from account banks to NSA's documents. As we can infer from the previous reviews,


id_rsa: Bloc de notas
Archivo Edición Formato Ver Ayuda
id_rsa: Bloc de notas
Archivo Edición Formato Ver Ayuda

```
C:\Users\Usuario de Sara\Desktop>ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\Usuario de Sara/.ssh/id_rsa):
Created directory 'C:\Users\Usuario de Sara/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\Usuario de Sara/.ssh/id_rsa.
Your public key has been saved in C:\Users\Usuario de Sara/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:wCkbmj1xeB3FY48CG/TSEpOJr87124Q8ZftyyJThkiE usuario de sara@LAPTOP-QQACRP1P
The key's randomart image is:
+---[RSA 3072]----+
|     o+o.o.      |
|    .o=B .+      |
|    =.O++. +     |
|   + BE+o o .    |
|  o +. .S=oo     |
|    ....o++.     |
|   o . .++o.     |
|    o   .++..    |
|         ...o.   |
+----[SHA256]-----+

C:\Users\Usuario de Sara\Desktop>
```

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQCiipN0JlwvF3Q+DAtmWNkh54pY8KOdhPrVix3zueWNkB42SCO6A8m3dYkzO2nK9kgnF01aRVv3y93bRRb8bX5KrN3XOR9vIhtf
+VIYL7n33QC5mWipqlV5OahghPTa0OWD2iUoKUUhYWD/zQ9iM2mfUEUGFe2I8XhOFX58vX7QjT78tHv8sj/ATuGA7Pp51VAoMoLnKij9CxTNt7NXD2uONoB4Jt+X7yJDz3ZK
+82tRKOmxOEj2m0rbhkZ1P0OQDXmq5UdTwTYA8U39v7X6yLC5dxgdgsVSft2lEEWwbuXDEjzkhamOl6A7c7y6Gm8ElTk2src9u/uXInsz6c1IwvDpAGg2oBaKw6f/COL7J1TLvb47G9pl
LxD/gbkcMXeX98Jwba/fVizS/6Vv0ALD2p9OMdMwh1zHVO/J7MKldzOXQuCpmF942Ks9O9rJj9GnwgVmmOAxD/m
+/UwpLvRsLTNfkLpRc0iTaaXHXcIfdbYEtgkv1HicmYu9epvkOvn4Bk= usuario de sara@LAPTOP-QQACRP1P
```

```
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAACmFlczI1Ni1jdHIAAAAGYmNyeXB0AAAAGAAAABDZ3YebZZAAd/Z8teU4Al5uAAAAEAAAAAEAAAGXAAAAB3NzaC1yc2EAAAADAQABAAABgQCiipN0JlwvF
3Q+DAtmWNkh54pY8KOdhPrVix3zueWNkB42SCO6A8m3dYkzO2nK9kgnF01aRVv3y93bRRb8bX5KrN3XOR9vIhtf
+VIYL7n33QC5mWipqlV5OahghPTa0OWD2iUoKUUhYWD/zQ9iM2mfUEUGFe2I8XhOFX58vX7QjT78tHv8sj/ATuGA7Pp51VAoMoLnKij9CxTNt7NXD2uONoB4Jt
+X7yJDz3ZK+82tRKOmxOEj2m0rbhkZ1P0OQDXmq5UdTwTYA8U39v7X6yLC5dxgdgsVSft2lEEWwbuXDEjzkhamOl6A7c7y6Gm8ElTk2src9u/uXInsz6c1IwvDpAGg2oBaKw6f/COL7J
1TLvb47G9plLxD/gbkcMXeX98Jwba/fVizS/6Vv0ALD2p9OMdMwh1zHVO/J7MKldzOXQuCpmF942Ks9O9rJj9GnwgVmmOAxD/m+/UwpLvRsLTNfkLpRc0iTaaXHXcIfdbYEtgkv1Hicm
Yu9epvkOvn4BkAAAWgB+3IDXml4kXqdstbYew1hkcrhRr16rmnodcLOn8eX7S2R9Wm8Z1E196NqM3Kuyv65UBDYojk2Vl8xLN/KXAWp3TEBPxAL/B7v0/EO8YWyCWNGmc01oxOH4f+hi
+ZCDUzNka1rYALdT6xBtFrUhH1u5DNK3a8XpnGD9uE1BBeDp0GzhfLgW/XBthqjS3G03HaLfvD4+R/+EDZN4zdMYsTaWuneyKVDlBCglZEyO/9J1aAfIdJPdEievJVgVHC4Ktlqjp79l
sIXApi7BhKA7IIi0jtbpEFxrujw4Cq51Ip5bjvbw331dsvBUokw/NpJTAZtao655G3k/6s3b5W7Dgt6MLOE/RwqRz2flL0OB/34dyAejwXOhI6SV8VmZbufB9QXTlN9AN+1yuxtxB9p0
kGj/yi32kFTeBWLb07G5bb0KquTODOjoBbZKrme/43bWfD+NhD20GtrPHy0VSsL0SkxvtvMeZV797T2tQc/hSvspvMtostkPufnO6D3L2ZdLVBjHT/z7HYci6dU1yuJ7u3VN07TrKCxv
P2SPjVW5NDPl4It3CT6k3KSmn+qbiRtcVE5752vVLk3cel/HmkPDrRmCsW0zXvibQ4R2aSRbzJvPTA3S1bQubxcIuIcQ5d0181GjjiwtxXFGz5p+O50QXZ3jX55AsI5eM4wABgNysICS
mHB92/+XICR9MaKptPxa//3rbdi9A6NBDLg6oEkOCfAZIBCXAXThy1zh4tpH5HutY619JRCbLXnRl4FCPCMIgg7JDJzgljaWLYrzud850tYvratLmuxLpMox0b1Z9a1i2zyCD09LTapb
l4s7fFQ7eZIHbikm1f+0Hu9pWmsPzbmNtAIc5MS/QSh5mgbu/amYo4a8dGvyWVZksEQyTc+6FofkQN2W4eHJ+wLzxjWjEdWeQJ7PbrQ91Dx88cMP9oUERTrh+lQnwF0UNuqfux7WH8MX
TV1z6AYYl9ijIqI3Esm/Y/v4jkeZn/daU12FworkNL3beYSpaY06mXod+e2wW89wloPLI25RIHa4qwE9S4AYmXtXjcNUKDEvrzjlW7XH5lWodGkNmu4bPZ7WIvRlTsRHjvoUdQA6k/vA
9WWM4/d+WZkbFVKeqzMAbZVOKd0QCVmz1Ryi0Kq2tDlY8kdAZOyOwSQM/uOLxtcRRUpiBboJKwIBkJA8bUhoHjcch3veKMyNxQUFQGbKIhp/JUTdTf1RTq6sAMCCwNmFQoruMFPWmmr3
Nshw5AvygKqPY/kJjg18Nz++Hcy3tTNj3KFWdadX4dZOastFolsb89n//qgXBxqE38Dml1/EemjubhF6zrOIrJG9ZZ5Ga8bAbrS76UFG617bvzOqGcmKVsiMIcXp5H1GglsMvU1HDTRF
6kz7PgHYLWrI+hmQXdE1OIWqad5mhLZtzSkoRm0e/s9Uo9nQNi+7mqAa/Aeuc8+LrfiyiupJheNgibU7K56JsVAHunkQvrp/XlAPcIeC39f5WPEAVmaA7iJFQ29OqRBZZKfHhOxTQ7ch
LREkFJdgqwRLUtLDX/6xOy1RgBujmE0icg5wTu4tw/ibByaaV9T/hzzecmSYY19MiTc77Ya07MgImLku8D6bsgG2ETJ/mjk6oBCX7w5ILaCMMrnCmVnd4MS++gerDi26HhVWIxG93Asw
lLtJLXafJsnm5a1YUI6wxUjVcUvPbqQpA36fFOcnqGbIToZ2gu3SKvMelB+2+TNcWYOMbqFvrSh1rMroqAe0hZf4uNjRqPf4YN37QI5T6a5+pXxKP0n7yMG5se0womjOXkrOVz/UwtZ/
4hkgVynAJqRRjZFpy6YWlXStgDtaKsOYFZhFqFYUYR4XiSrhvymtKtsAQgCsPtonEDTyXKDQThSRs8I+AMeHAbvNDX2lwuMnalayMtYsn8MdvsyIzJLS/x
-----END OPENSSH PRIVATE KEY-----
```

Figure 3: Generating public-private key pair with Windows' console.

their security is based on different mechanisms. AES's security consists in large key sizes while RSA's security is based on the lack of fast factoring algorithms. But, this takes a turn with the appearance of quantum algorithms and quantum computers. Two important quantum algorithms are introduced briefly now:

- **Shor's algorithm.** Peter Shor proposed in 1994 this algorithm which could break RSA's security. It consists on transforming our factorization problem into an order finding problem, i.e. we have to find the smallest r that satisfies $e^r = 1 \cdot mod(n)$ being e and n the integers of the public key. Using Quantum Principles such as superposition and Quantum Fourier Transform, a Quantum Computer would be able to factorise n in less than a day [11].

- **Grover's algorithm.** This is a quantum search algorithm. Given N items, there is one that satisfies a given condition (our goal is to find the key with which the message was encrypted). In order to find this item, a classical algorithm will take at average O(N/2) steps. Using quantum parallelism, among other quantum operations, we can evaluate more than one item at once. This allows us to compute the item in O($\sqrt{N}$) steps [12]. However, symmetric cryptosystems are not vulnerable to this algorithm if we use long keys. Just by duplicating the key's length (if we use AES-128 in a classical computer, we will have to use AES-256 in a quantum one), this algorithm will not be efficient enough to break them.

As a conclusion, apparently, asymmetric cryptosystems will not be longer useful during the Quantum Computer Era but symmetric ones are believed to be Post Quantum Cryptography, i.e. cryptography algorithms that can not be broken with classical nor quantum computers [13]. The following table, published by National Institute of Standards and Technology (NIST) [14], shows the impact of quantum computers on classical algorithms:

Cryptographic Algorithm	**Type**	**Purpose**	**Impact from large-scale quantum computer**
AES	Symmetric key	Encryption	Larger key sizes needed
SHA-2, SHA,3	-	Hash functions	Larger output needed
RSA	Public key	Signatures, key establishment	No longer secure
ECDSA, ECDH	Public key	Signatures, key exchange	No longer secure
DSA	Public key	Signatures, key exchange	No longer secure

Table 1: Impact of Quantum Computing on Common Cryptographic Algorithms. ECDSA: Elliptic Curve Cryptography Signature Algorithm; ECDH: Elliptic Curve Diffie–Hellman; DSA: Digital Signature Algorithm; SHA-2: Secure Hash Algorithm-2; RSA: Rivest–Shamir–Adleman Algorithm; AES: Advanced Encryption Standard.

Nowadays, it has been proposed several Post Quantum algorithms but they are not the real aim of this work. From now on, we will review Quantum Cryptography: cryptography which bases its security on the Quantum Mechanics Principles [14].

3 Quantum Cryptography

We enter in the Quantum Physics world. Before, we must leave aside our intuition because the laws that govern this world are nothing compared to our day-to-day. In this section, it will be explained some of the basics Quantum Mechanics Principles in order to take advantage of them in Quantum Cryptography. Finally, it will be described Quantum Key Distribution: an innovative and secure way of key exchange.

3.1 Fundamental Aspects

Just as in Classical Computing we use bits, in Quantum Computing the basic unit of information is the qubit (quantum bit). They can take a value of one or zero but, as they are governed by Quantum Physics, they have very special properties that will explain in this section.

- **Superposition.** Nearly everyone have ever heard about Schrödinger's cat being neither dead nor alive. This is an unreal but illustrative example of superposition. A quantum state can be represented by the sum of two different and contrary states with a given probability. For example, imagine a die representing a quantum object; we have it in a cube so we can't see it. We will represent the die's state as a superposition of the six possible results with a probability 1/6:

  $$|State\rangle = \sqrt{\frac{1}{6}}\,|1\rangle + \sqrt{\frac{1}{6}}\,|2\rangle + \sqrt{\frac{1}{6}}\,|3\rangle + \sqrt{\frac{1}{6}}\,|4\rangle + \sqrt{\frac{1}{6}}\,|5\rangle + \sqrt{\frac{1}{6}}\,|6\rangle\,. \tag{1}$$

  This is true until we do a measurement of the state, then, as we will see, the state collapses [15].

  Superposition can seem a little strange but it is experimentally checked, for example, in the Young's double-slit experiment. But, how can we build this superposition states? Qubits can be physically represent by photons or particles such as electrons. For this explanation, we are going to focus on the latter. Electrons have an intrinsic property: the spin, which can be ±1/2. This is an ideal characteristic to represent the qubit's value. Pauli's matrices are used to represent spin so, we will denote $|0\rangle$ and $|1\rangle$ as the eigenvectors of σ_z (identifying them with qubit's states) and $|+\rangle$ and $|-\rangle$ for σ_x so that

  $$|0\rangle = \sqrt{\frac{1}{2}}(|+\rangle + |-\rangle). \tag{2}$$

  $$|1\rangle = \sqrt{\frac{1}{2}}(|+\rangle - |-\rangle). \tag{3}$$

  We now introduce Stern–Gerlach Apparatus (SGA) in order to achieve superposition: We establish an inhomogeneous magnetic field in the z-axis (having the gradient perpendicular to the direction of the particle). Sending classical dipoles, the device will deflect them along a vertical line depending on the orientation of their poles. However, when we send electrons, the deflection will only be upwards or downwards, manifesting the existence of the spin (see Fig. 4a). But the z-axis is nothing special, the same happen with the other directions.

  Now we block the downwards electrons and put another SGA (with the magnetic field in x-axis) after the first one. We could expect a similar pattern to the classical magnet oriented in the z-axis: it does not suffer deflection showing in the screen a pattern in the center of it but we must always bear in mind that this is not the classical world. The pattern will be a distribution of leftwards and rightwards electrons, how can be this possible if the incoming electron had its spin upwards? As we have seen before, the pattern of a quantum particle consist of two big peaks, but nothing in the center. Thus, Quantum Mechanics creates a superposition state of the electron using σ_x eigenvectors, i.e. it changes the basis in which the state is expressed as (2) (see Fig. 4b) [4, 16].

- **Measuring.** Taking up the die's example: after rolling the die and checking for the result, the superposition state collapses into a single state, for example $|State'\rangle = |2\rangle$ with probability

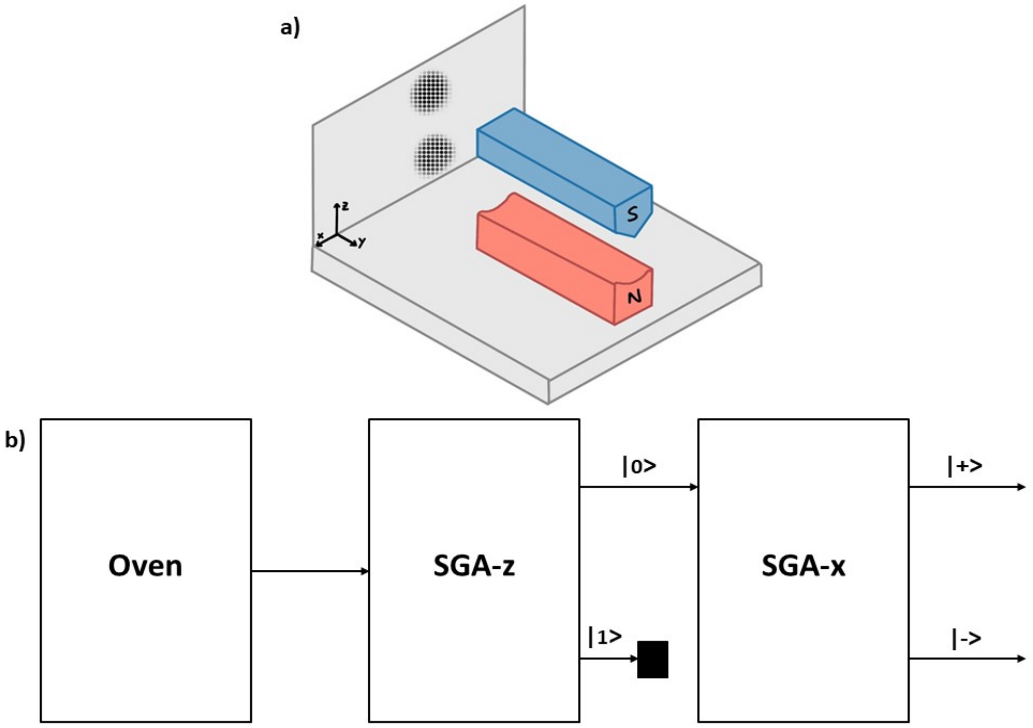


Figure 4: a) Pattern for a z-axis Stern-Gerlach. b) Stern-Gerlach cascade's scheme.

1/6. This means we perturb the state after measuring it [4]. Beside this, an important property is the impossibility to perfectly distinguish between two non-orthogonal states: imagine having $|\psi\rangle$ that can be either $|\psi_1\rangle$ or $|\psi_2\rangle$ (they are non-orthogonal states). The probability, P, of guessing $|\psi\rangle$ is given by the Helstrom Bound and is $P \leq \frac{1}{2}(1-\sin\theta)$. This property will be extremely useful to detect eavesdropping in BB84 protocol [16, 17].

- **Entanglement.** Given the states $|\psi_i\rangle$ and $|\chi_i\rangle$ for i=1,2, that represent two particles of systems A and B. The interaction of the two systems could lead to the physical state

$$|\Psi\rangle = \frac{1}{\sqrt{2}}(|\psi_1\rangle \otimes |\chi_1\rangle - |\psi_2\rangle \otimes |\chi_2\rangle). \tag{4}$$

If it is not possible to factorise this state as a multiplication of two superposition states, this is what we called an entangled state. Now, we consider that these states represent the z-component of the spin so, following the notation used before we have:

$$|\Psi\rangle = \frac{1}{\sqrt{2}}(|1\rangle \otimes |0\rangle - |0\rangle \otimes |1\rangle). \tag{5}$$

The two particles are far apart. We measure S_{Az} resulting $|1\rangle$, the second particle will instantly collapse to the state $|0\rangle$: the measurement of A affects to B and makes it to collapse instantly no matter how far are the particles apart. It may look like the second particle already knew its z-component or the particles are communicating faster than light, how can be this possible? [18]. In 1935, Einstein, Podolski and Rosen [19] proposed that the Quantum Mechanic's Theory was incomplete:

> “Every element of the physical reality must have a counterpart in the physical theory [. . .] If, without in any way disturbing a system, we can predict with certainty the value of a physical quantity, then there exists an element of physical reality corresponding to this physical quantity.”

So there must be some ’hidden variables’ we are missing because if not, the example before violates locality’s principle [19]. However, in 1964, Bell proved that, if Quantum Mechanics was a Hidden Variable Theory, it must satisfy the following inequality:

$$|E(\vec{a},\vec{b}) - E(\vec{a},\vec{b'})| + |E(\vec{a'},\vec{b'}) + E(\vec{a'},\vec{b})| \leq 2. \tag{6}$$

where $\vec{a}$ and $\vec{a'}$ are two different unitary directions for SGA(A) and $\vec{b}$, $\vec{b'}$ are two different unitary directions for SGA(B) and E is the correlation between the measurements. Using the state $|\Psi\rangle$ written before, we could think of $\vec{a'} = \vec{b}$ and the angle θ between $\vec{a}$ and $\vec{a'}$ is the same as the one between $\vec{b}$ and $\vec{b'}$ so the correlations are C($\vec{a}$,$\vec{b}$)=$\cos\theta$; C($\vec{a}$,$\vec{b'}$)=$\cos 2\theta$; C($\vec{a'}$,$\vec{b'}$)=$\cos\theta$ and C($\vec{a'}$,$\vec{b}$)=1. Introducing these values in the inequality and assuming $\theta \leq \pi/2$ we obtain $2\cos\theta - 2\cos 2\theta \leq 1$. We can easily see this inequality is violated except for $\theta = 0, \pi/2$, i.e. we can state that Quantum Mechanics is not a Hidden Variable Theory [20, 18]. In the following properties, it will be explained why instantaneous communication with quantum entanglement is not allowed.

- **Quantum Teleportation.** Process of transferring quantum information using entangled states. To carry it out, we need two parties, Alice and Bob. They share an entangled state called ’Bell State’: written in z-basis it takes the form

$$|\Phi_{ij}\rangle = \frac{|0j\rangle + (-1)^i |1\bar{j}\rangle}{\sqrt{2}}, \tag{7}$$

for i,j=1,0 and being $\bar{j}$ the negation of j [16]. For example, they share the state

$$|\Phi_{00}\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}. \tag{8}$$

Each one takes a qubit and gets far away (we will name the first qubit as Alice’s one (A) and the second one as Bob’s (B)). Alice wants to send an unknown qubit $|C\rangle = \alpha|0\rangle + \beta|1\rangle$ to Bob. She can not measure it because of the quantum collapse so she interacts the third qubit with the entangled state obtaining:

$$|CAB\rangle = |\psi\rangle \otimes |\Phi_{00}\rangle = \frac{\alpha|000\rangle + \alpha|011\rangle + \beta|100\rangle + \beta|111\rangle}{\sqrt{2}}. \tag{9}$$

This can be written in terms of Bell States being the entangled qubits A and C:

$$|CAB\rangle = \frac{1}{2}[\Phi_{00}(\alpha|0\rangle+\beta|1\rangle)+\Phi_{10}(\alpha|0\rangle-\beta|1\rangle)+\Phi_{01}(\alpha|1\rangle+\beta|0\rangle)+\Phi_{11}(\alpha|1\rangle-\beta|0\rangle)]. \tag{10}$$

As we can notice, Bob’s qubit has ’adopted’ the structure of the original $|C\rangle$. Now, Alice measures $|CAB\rangle$ so that she will obtain $|\Phi_{ij}\rangle$ with probability 1/4. She sends classically the two bits that denote the obtained Bell State to Bob. In our example, if she gets $|\Phi_{00}\rangle$, Bob will know that his qubit is the original C; else, he will have to perform an unitary transform to recover the original C [21].

We must highlight that quantum teleportation does not violate the no-cloning theorem, which we will see below, because we are not copying the quantum state, instead, it will be 'teleported' to another state so the initial one will be transformed into another one: as we have seen, the original C ends up taking part of the entangled state while Bob's one is 'transformed' into C.

- **No-cloning Theorem.** It establishes that an arbitrary quantum state can not be exactly copied. What would be the consequences if we could clone states? First of all, Heisenberg's uncertainty principle would be violated as we could measure, for example, the particle's location and measure the momentum's clone. Moreover, speed limit would also be violated as the information would be transmitted instantly: Alice and Bob share the previous entangled state

$$|\Phi_{00}\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}, \tag{11}$$

where the two bits are the same for each one. Alice measures the state in z-basis obtaining $|0\rangle$ so the state collapses to $|\Phi'_{00}\rangle = |00\rangle$. Now Bob could measure the state in z-basis obtaining the desired result $|0\rangle$ but he could also measure in x-basis obtaining $|+\rangle$ or $|-\rangle$ so he will not know which was the bit measured by Alice. If he could make many copies of the state $|\Phi'_{00}\rangle$, he could measure the copies in z-basis obtaining always the same result so that will be the right basis. However, when he measures in x-basis he will obtain half of the times $|+\rangle$ and the other half $|-\rangle$ knowing he is not measuring in the correct basis. In this case, the information would be transmitted instantly because Bob does not need to know in what basis Alice has done her measurement [22].

Now we show the mathematical proof of the theorem: given two arbitrary states $|\phi_i\rangle$ for i=1, 2, we want to copy one of them in another state $|\chi\rangle$. We could achieve this through an universal unitary operator U, i.e. $U(|\phi_i\rangle \otimes |\chi\rangle) = |\phi_i\rangle \otimes |\phi_i\rangle$. Taking the inner product of the left side for both i:

$$(\langle\phi_1| \otimes |\chi\rangle\, U^\dagger)(U\,|\phi_2\rangle \otimes |\chi\rangle) = \langle\phi_1|\phi_2\rangle\,, \tag{12}$$

where we have used that U is unitary $U^\dagger U = 1$. Now, taking the inner product of the right side

$$(\langle\phi_1| \otimes \langle\phi_1|)(|\phi_2\rangle \otimes |\phi_2\rangle) = \langle\phi_1|\phi_2\rangle^2\,. \tag{13}$$

Combining the results, we obtain the following equation

$$\langle\phi_1|\phi_2\rangle = \langle\phi_1|\phi_2\rangle^2\,, \tag{14}$$

which only has solutions $\langle\phi_1|\phi_2\rangle = 0$ or $\langle\phi_1|\phi_2\rangle = 1$ but we hypothesized that $|\phi_1\rangle$ and $|\phi_2\rangle$ were arbitrary states so we can not achieve these conditions at all [22, 23] (See Fig. 5).

3.2 Quantum Key Distribution (QKD)

What are the biggest drawbacks of Classical Cryptography? Many of them are based on mathematical methods that might be destroyed by quantum computers and it is difficult to ensure privacy when sharing the private key. The need of searching for post-quantum cryptography arises. This section was wrongly named 'Quantum Cryptography' because actually, we are not going to describe algorithms to encrypt and decrypt data, as we did in classical cryptography... Instead, we should talk about Quantum Key Distribution since it will be described a provably secure method to distribute keys involving two communication channels, a classical and a quantum one, and using the Quantum Principles explained before. There are plenty of QKD protocols but the main

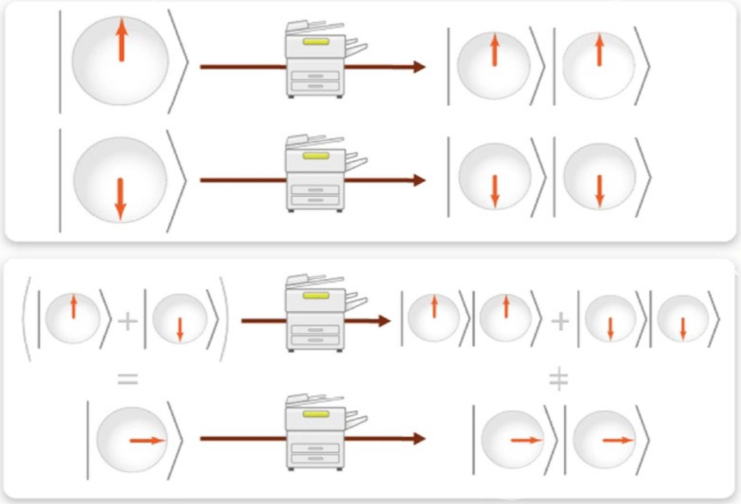


Figure 5: A 'cloning machine' could clone any of the two elements of the basis $\{|0\rangle, |1\rangle\}$, but not a superposition state [24].

focus will be on the BB84 protocol. Building on the broad explanation of SGA in the previous section, we will explain this protocol with electrons and SGAs. Nevertheless in the experimental part, qubits's physical representation are photons. Furthermore, we will test the protocol's security in the presence of an eavesdropper: how can someone steal information without being able to copy states (No-cloning Theorem) and proving that Quantum Mechanics can detect eavesdropping (Measurement Perturbation). Finally, we will study how can we make this happen and how is the situation now [21, 25].

3.2.1 The BB84 protocol

This protocol owes its name to Bennett and Brassard who published the work in 1984. This protocol can be described with any 2-level quantum system but as it has been said before, our explanation will be based on electrons. The protocol requires a quantum channel and a classical one. We will start with the quantum one. The methodology is as follows:

Alice and Bob want to ensure their communication. To achieve this, they establish a protocol to exchange a provable secure secret key. Each one has a SGA with the magnetic field in z-axis (we will call it SGAz) and one with the magnetic field in x-axis (we will call it SGAx). Before starting, they agree in giving the bit value 0 to the quantum states $|0\rangle$ and $|+\rangle$ and they will associate the bit value 1 with $|1\rangle$ and $|-\rangle$. Alice randomly chooses to use SGAz or SGAx. Then, she prepares an electron in a superposition state in the chosen basis, i.e., she can either prepare

$$|\psi\rangle = \sqrt{\frac{1}{2}}\,(|0\rangle - |1\rangle)\,, \tag{15}$$

if she chooses Z, or

$$|\psi\rangle = \sqrt{\frac{1}{2}}\,(|+\rangle - |-\rangle)\,, \tag{16}$$

if her choice was X. She sends $|\psi\rangle$ through the chosen SGA, i.e., she 'measures' the state, causing the collapse of $|\psi\rangle$. Knowing the result, Alice writes the corresponding bit and she sends the electron to Bob. He repeats the process: he chooses a random basis between Z and X and then,

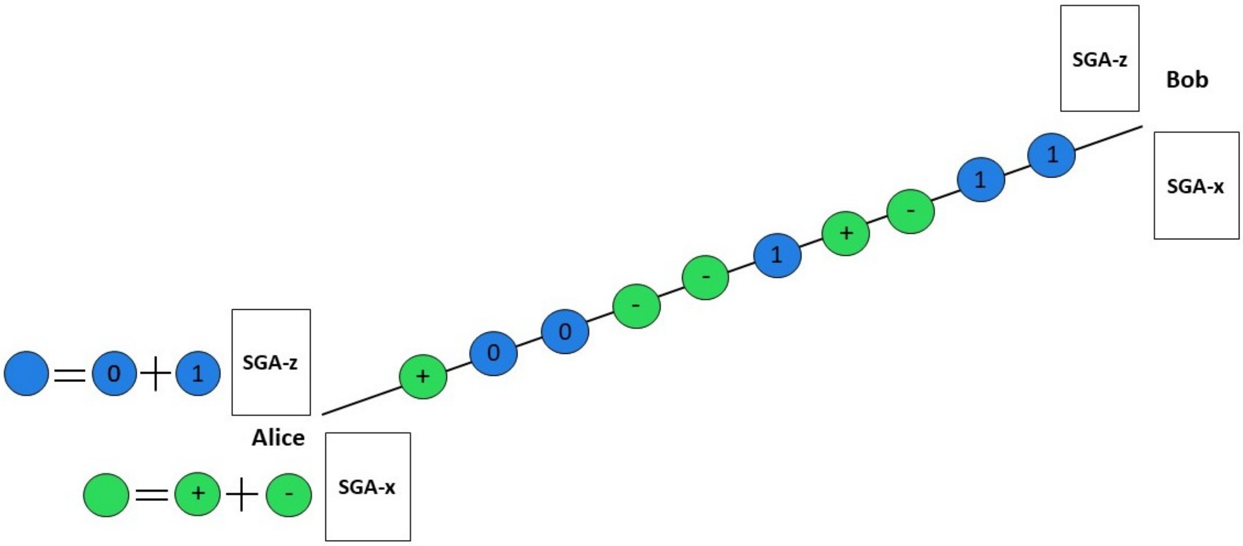


Figure 6: BB84 scheme using electrons and SGA.

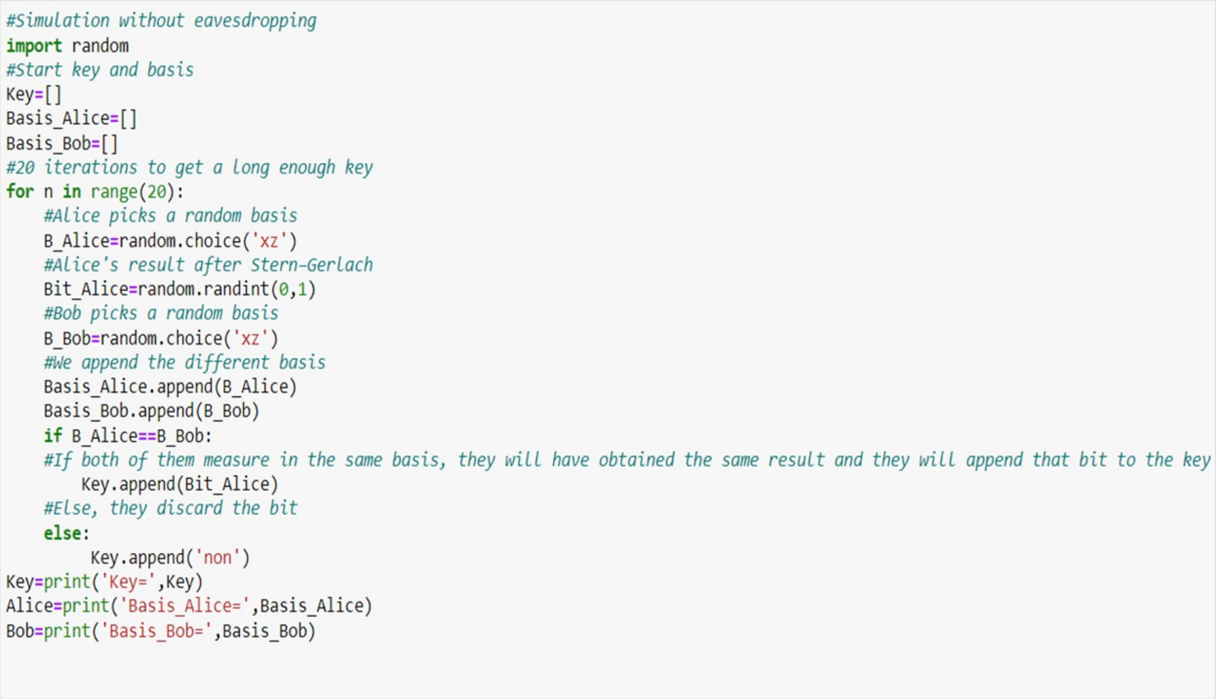
he sends the electron through the corresponding SGA. The state collapses and he writes down the corresponding bit result.

They repeat this process many times to improve the security of the protocol until each one has a long list of the used basis and the corresponding bit in each measurement. This is the end of the use of the quantum channel. In Fig. 6, it is shown a scheme of the BB84 protocol where the electrons are represented by spheres [4].

Now, the classical part begins. They need an authentic channel which means that Alice and Bob must identify themselves and anyone can listen to the communication but without modifying the information. Alice and Bob share their list of the basis used each time but not the obtained bit. In the case they made the same choice, the obtained bit will be the same because Alice already sent to Bob the 'collapsed' state so measuring in the same basis will not affect it. However, if their basis do not match they can not be sure if the obtained bit is the same. This was explained in the previous section when we showed the SGA's cascade: the 'collapsed' state will be 'written' in the other basis being in a superposition state. Hence, when Bob sends the state through the SGA, he will have 50% probability of obtaining the same bit as Alice [4].

Then, Bob and Alice will obtain two strings of bits which differ by 25% on average. This strings are called the *raw key*. However, this high error rate would make the malfunctioning of the protocol so they discard the bits from the measurements where the basis chosen by each do not coincide. This shorter string is called the *sifted key* and now, they are sure it is the same for both. As this discard method shortens a lot the string (by 50% on average) we will need to repeat the protocol many times [26].

Now, in order to numerically simulate the BB84 protocol, a Python script has been prepared on purpose. Specifically, this script makes random choices of the basis and the corresponding obtained bits. Then, the lists of basis is compared to discard those bits corresponding to the cases where the bases do not coincide. The script is shown in Fig. 7 and the outcomes rendered are shown in the two tables below, which correspond to two different cases of the protocol to corroborate the



```
#Simulation without eavesdropping
import random
#Start key and basis
Key=[]
Basis_Alice=[]
Basis_Bob=[]
#20 iterations to get a long enough key
for n in range(20):
    #Alice picks a random basis
    B_Alice=random.choice('xz')
    #Alice's result after Stern-Gerlach
    Bit_Alice=random.randint(0,1)
    #Bob picks a random basis
    B_Bob=random.choice('xz')
    #We append the different basis
    Basis_Alice.append(B_Alice)
    Basis_Bob.append(B_Bob)
    if B_Alice==B_Bob:
    #If both of them measure in the same basis, they will have obtained the same result and they will append that bit to the key
        Key.append(Bit_Alice)
    #Else, they discard the bit
    else:
         Key.append('non')
Key=print('Key=',Key)
Alice=print('Basis_Alice=',Basis_Alice)
Bob=print('Basis_Bob=',Basis_Bob)
```

Figure 7: Python script simulation without Eve.

randomness of the process.

As we can observe in these tables, in every case where Alice's basis and Bob's is not the same, the bit is discarded and this is marked with a '-'. However, in those cases where the basis do coincide, the measurement will be the same and they associate the corresponding bit following the agreement established before beginning the protocol. These bits will form the sifted key.
It is also interesting to study if the expected probabilities mentioned before are fulfilled: in both the first and second tables, Alice and Bob have chosen the same basis 10 times out of 20, fulfilling the expected probability. However, we can not check the others probabilities as the simulation do not 'create' a key for Bob since if the basis do not coincide the bit is discarded, even if he would have got the same bit as Alice.

Using an ideal quantum channel, the sifted key obtained by Alice and Bob would be the same but, actually, the channel has quantum noise that could make bit flips. Because of this, Alice and Bob must estimate quantum bit error rate (QBER) comparing some bits of their strings. If this rate is greater than a threshold, it indicates the presence of an eavesdropper or imperfections in the quantum channel. Bennett and Brassard also described a second part of the protocol in order to apply error correction: Alice and Bob divide their strings in n blocks which are not likely to contain more than one error. Now they compare the bit parity of one block (how many bits with value 1 their block have). If the results are not the same, there is an odd number of errors in that

Alice's basis	x	x	z	x	z	z	x	z	x	x	z	z	z	z	x	z	x	x	x	z
Bob's basis	x	x	x	x	x	x	x	z	x	z	x	z	z	z	x	x	z	z	z	x
Key	1	1	-	0	-	-	1	0	1	-	-	0	0	0	1	-	-	-	-	-

Table 2: BB84 first simulation.

Alice's basis	x	z	z	x	z	z	x	z	x	x	z	x	x	x	z	x	x	x	z	z
Bob's basis	x	x	z	x	z	x	x	x	x	z	z	x	x	z	x	x	z	z	x	x
Key	1	-	0	1	1	-	0	-	0	-	1	0	1	-	-	0	-	-	-	-

Table 3: BB84 second simulation.

block so they re-split it until they find the error. Repeating this procedure with all blocks, their keys will contain no errors or an even number of them. Then, they exchange the position of several bits and repeat the procedure. Now they are sure that their key strings are exactly the same [27]. In the previous simulations, we have used 20 iterations, i.e. Alice would have sent 20 different electrons to Bob. As we can see, the obtained sifted key is too short and will not be secure enough due to environmental errors or the presence of Eve, an eavesdropper. We can conclude that we need many more repetitions of the protocol.

In the next section, we will study the presence of an eavesdropper during the BB84 protocol and how can we avoid this attack.

3.2.2 Eavesdropping

It is useful to remember the reason of the appearance of QKD: we need provable secure methods to exchange private keys. Thus, in this section we will be testing the BB84 protocol with the presence of Eve, an eavesdropper.

Eve has two SGA as Alice and Bob. She intercepts some electrons from Alice and then she resends them to Bob. We have learnt that measuring causes the collapse of the state, i.e. it is 50% probable she measures in the same basis as Alice so the electron's state will not be disturbed and they can not detect her presence (See Fig. 8). However, if Alice and Bob chose the same basis but Eve measures the electron in the other one, the bit corresponding to that measure will be the same 50% of the times in contrast to 100% probability in the ideal case without eavesdropping. This intercept-resend strategy will not be successful for Eve in the last case, so Alice and Bob can realise (if that bit is in the list of bits they compare) there is an eavesdropper if their basis are the same but not their results (See Fig. 9) . They would begin the protocol again. The impossibility of distinguish between non-orthogonal states, as are the ones used in this protocol, makes this protocol secure as Eve will not know if she has measured in the correct basis or not. Moreover, the No-cloning theorem makes impossible for Eve to intercept the state, clone it so she can measure the clone but not the original one [26].

As we did in the ideal BB84 protocol, now we will simulate the protocol but in the presence of Eve. Tables 4, 5 are the results obtained with another Python script that is attached below (Fig. 10). We can observe again '-' when Alice's basis does not coincide with Bob's and this time, we see a

Alice's basis	z	x	z	x	z	z	x	z	x	x	x	x	z	z	z	x	x	z	x	z
Eve's basis	x	x	z	x	x	z	z	x	x	z	z	z	x	x	x	x	x	x	x	x
Bob's basis	x	x	z	z	z	z	z	z	z	x	z	x	x	x	x	z	x	x	x	x
Key	-	1	1	-	W	1	-	W	-	W	-	0	-	-	-	-	1	-	1	-

Table 4: BB84 with eavesdropper first simulation.

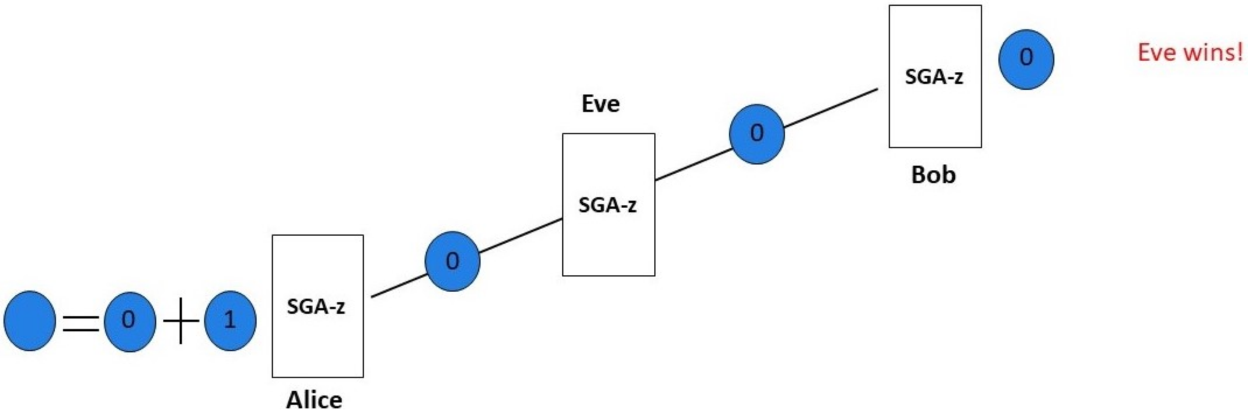


Figure 8: Eve measures in the same basis as Alice and Bob so she does not disturb the state.

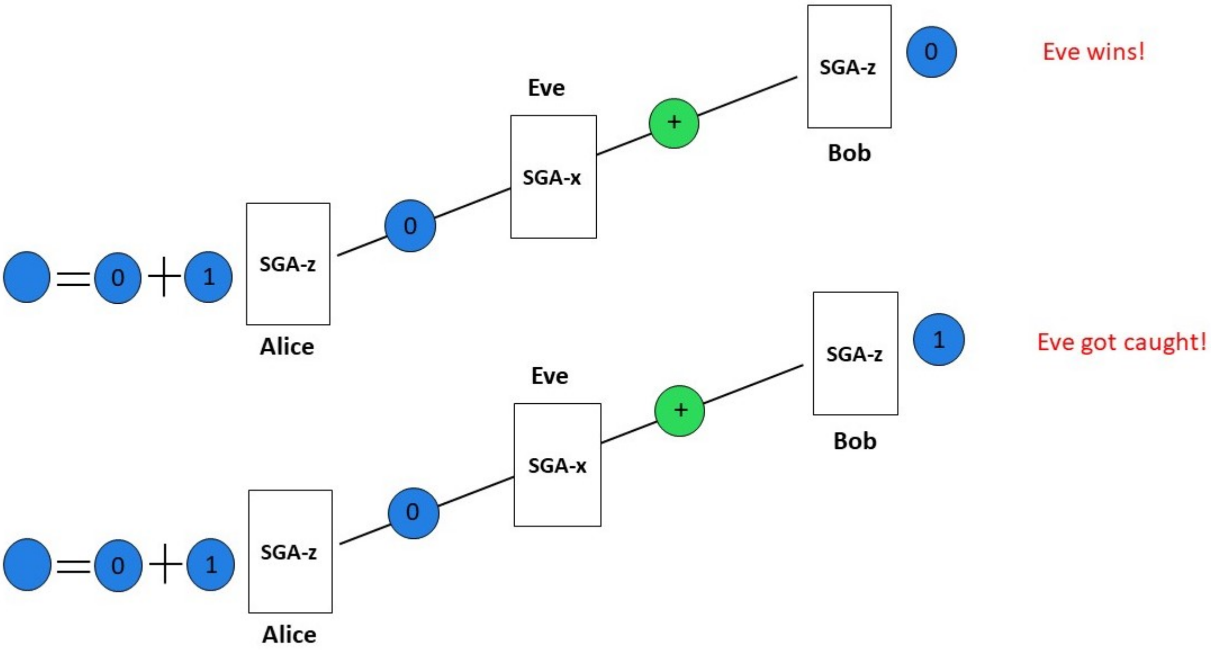
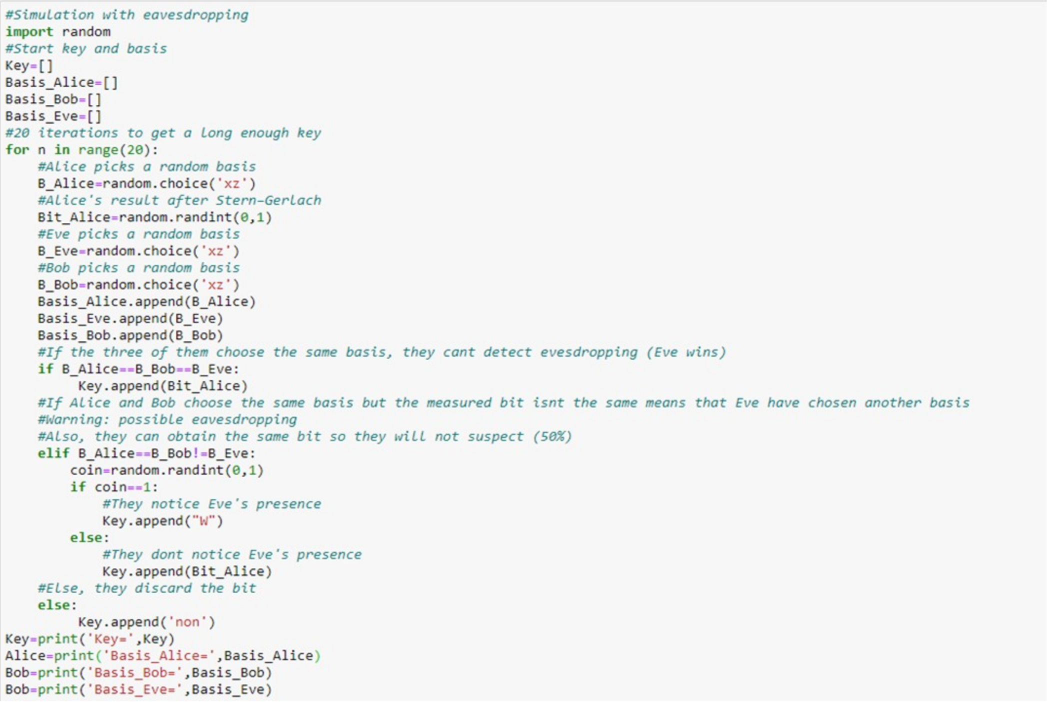


Figure 9: Alice and Bob measure in the same basis but not Eve. In the upper image, even having disturb the state, Bob measures the same bit as Alice. In the picture below, she is not so lucky and she gets caught.

letter 'W' which means *'Warning there can be an eavesdropper'* in those cases where Alice's basis and Bob's coincide but not Eve's. They can not detect this case always as we have said before that if Eve's basis does not coincide with their, there is a 50% chance of Bob measuring the same bit as Alice so they will not be on alert. A good example of this is the first column of Table 5 where Alice and Bob measure in X-basis but Eve measures in Z-basis but, even so, Bob measures the same bit as Alice and they do not suspect.

Certainly, this is an improbable event since there is a 25% chance Alice and Bob choose the same basis while Eve in the contrary one. Moreover, there is a 50% chance that Eve will be discovered, i.e. the total probability of this event 12.5%.



```
#Simulation with eavesdropping
import random
#Start key and basis
Key=[]
Basis_Alice=[]
Basis_Bob=[]
Basis_Eve=[]
#20 iterations to get a long enough key
for n in range(20):
    #Alice picks a random basis
    B_Alice=random.choice('xz')
    #Alice's result after Stern-Gerlach
    Bit_Alice=random.randint(0,1)
    #Eve picks a random basis
    B_Eve=random.choice('xz')
    #Bob picks a random basis
    B_Bob=random.choice('xz')
    Basis_Alice.append(B_Alice)
    Basis_Eve.append(B_Eve)
    Basis_Bob.append(B_Bob)
    #If the three of them choose the same basis, they cant detect evesdropping (Eve wins)
    if B_Alice==B_Bob==B_Eve:
         Key.append(Bit_Alice)
    #If Alice and Bob choose the same basis but the measured bit isnt the same means that Eve have chosen another basis
    #Warning: possible eavesdropping
    #Also, they can obtain the same bit so they will not suspect (50%)
    elif B_Alice==B_Bob!=B_Eve:
        coin=random.randint(0,1)
        if coin==1:
            #They notice Eve's presence
            Key.append("W")
        else:
            #They dont notice Eve's presence
            Key.append(Bit_Alice)
    #Else, they discard the bit
    else:
         Key.append('non')
Key=print('Key=',Key)
Alice=print('Basis_Alice=',Basis_Alice)
Bob=print('Basis_Bob=',Basis_Bob)
Bob=print('Basis_Eve=',Basis_Eve)
```

Figure 10: Python script simulation with Eve.

Alice's basis	x	x	x	z	x	z	z	x	z	x	x	z	x	z	x	x	x	z	x	x
Eve's basis	z	z	x	x	x	x	z	x	x	x	x	z	z	z	z	x	z	z	x	x
Bob's basis	x	z	x	z	z	z	x	x	x	x	x	z	z	z	z	z	z	z	z	x
Key	1	-	1	W	-	1	-	0	-	0	1	1	-	0	-	-	-	1	-	0

Table 5: BB84 with eavesdropper second simulation.

As we have analysed before, implementing error correction methods delete the noise caused by the environment or Eve. Even so, we have to bear in mind that the use of a classical channel (non confidential) can leak some information to Eve so she can learn some information and reproduce a part of the key. To avoid this, Alice and Bob should apply privacy amplification protocols to minimize Eve's information. The methodology is as follows:

> Alice choose two bits and computes their XOR value, i.e. their sum modulo 2. Now she tells Bob the position of those bits but not the obtained result. Remember their current key is exactly the same so their XOR value will be the same. They replace the two bits with their XOR value and continue repeating this procedure. Thus, Eve's information is considerably reduced but also is the key length. Because of this we need to exchange long raw keys [26]. Here is a small example to illustrate this process: Alice and Bob share now the same bit string 1011001011.They want to apply privacy amplification so Alice choose bits in position 4 and 7, i.e. she has chosen 1 and 1, and tells Bob their position. They compute the XOR value $1 \oplus 1 = 0$ and replace it. The new key will be 101000011 which is shorter than the original one.

3.3 Experimental realization

So far, it has been explained the theoretical basis of QKD, in particular the BB84 algorithm. During the whole project, we have worked with electrons playing the role of qubits because the explanations seemed easier; but actually, in experimental realization, qubits are represented by photons. We can establish analogies to understand this new representation: electron's spin is now the photon's polarization and the SGA is replaced with polarizers. The four new possible states will be vertical and horizontal polarization, analogous to the computational basis $|0\rangle$, $|1\rangle$ and $\pm 45^{\underline{o}}$ polarization corresponding to the superposition states $|+\rangle$, $|-\rangle$.

Each electron represented a qubit, so each photon should do the same. We did not have problems with throwing a single electron each time through the SGA. Nevertheless, we do not have enough technology to create single photon sources. In response to this issue, attenuated light coherent sources are used. These will carry a number of photons given by Poissonian statistics which means that we will always have a probability of having more than a photon in each pulse.

But this solution brings other setbacks. Our eavesdropper Eve could intercept those multi-photon pulses, keeping one photon and resending the rest to Bob through a lossless channel. When the pulse only carries one photon, she blocks it. During the classical part, she will easily get information about the key just by measuring them in the same basis as Alice. This attack is known as Photon Number Splitting (PNS).

Fortunately, Hwang came up with a possible solution in 2003: introducing decoy states with another source. Alice will interchange these two sources while sending the pulses but making the one used to transmit the key most likely to carry only one photon each pulse and the decoy one, most likely to carry multi-photon. Eve can not distinguish between them so she will continue intercepting the multi-photon pulses believing they come from both sources in the same proportion on average. However, Alice and Bob will then analyse the yield of both sources. Knowing that Bob uses number-photon blind detectors, the same yield (on average) of both sources will be expected as they are sent though the same imperfect channel. If they notice that the yield of the decoy source is much greater than the other one, they will know Eve is stealing information so they interrupt the protocol. How? Remember that when Eve intercepts the multi-photon pulses, she resends the photons through a **lossless** channel, so in those cases, the yield will be greater than if they were to continue in the original one [28].

This is basically how we are implementing BB84 nowadays. There are many ongoing experiments (using decoy states or other methods) that keep breaking records for the longest distance over which the protocol could be established or the fastest key transmission realization. These all are extremely interesting but I think (and as I have tried along this project) it is better to talk about things that we find more tangible so we can realise Quantum Cryptography is not a utopian or futuristic idea but is already with us. We all know Toshiba. Apart from making computers, it is one of the most pioneering companies in the experimental realisation of QKD. We mentioned it because they use the modified BB84 protocol explained above with decoy states. In 2020, they achieved establishing secure data transmission between the National Composites Centre (NCC) and the Centre for Modelling and Simulation (CFMS) in Bristol (See Fig.11). The last update made was that it had been running for six months with an absolute success. Indeed, they claimed that the most difficult part of the installation was the fiber links, not the quantum part. This is only a little example of Toshiba's achievements, as they also connected two hospitals using this protocol in order to transmit medical data or achieving 600km distance long quantum communication

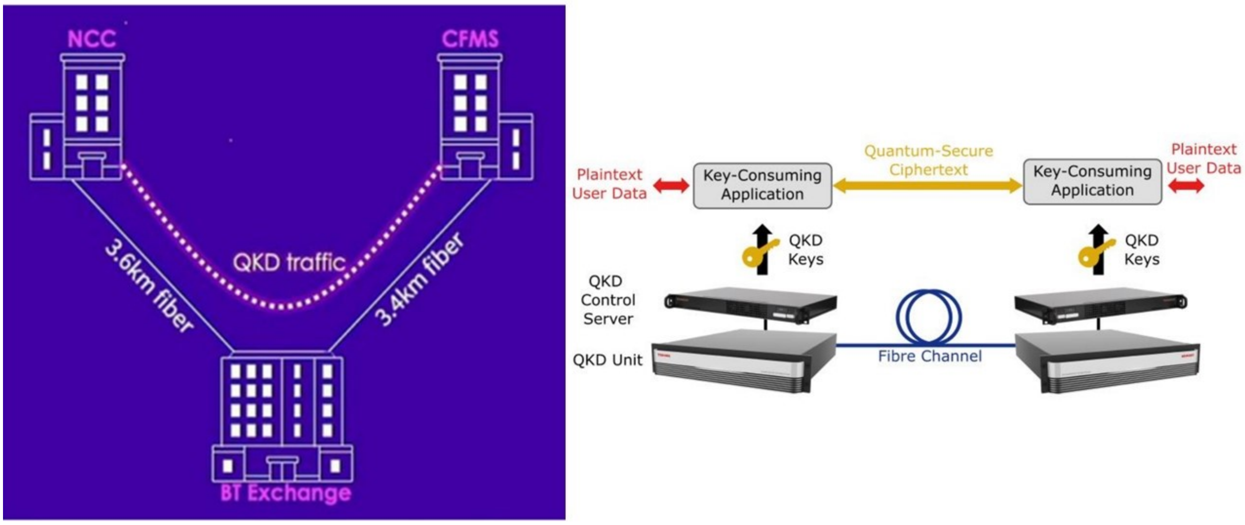


Figure 11: **Left:** Scheme of the transmission between NCC and CFMS. **Right:** Simple scheme of the QKD system.

overcoming the environment faults. It is noticeable to mention that many big companies around the world, as Telefónica, PSNS or ADVA, are buying their QKD systems and trying to implement them.

4 Concluding/Final remarks

The first objective of this project was to answer why we need Quantum Cryptography. Analysing two of the most known and used classical algorithms nowadays, we reach the conclusion that algorithms based on the incapability of classical computers will not be valid with Quantum Computers' arrival. Shor demonstrated with his quantum algorithm how easy it was to break RSA with Quantum Mechanics. Once this section was closed, we focused on Quantum World, studying its special rules such as superposition states or the No-Cloning Theorem. Then, we were ready to explain Quantum Cryptography that actually it is a misleading name because we research methods to exchange key pairs securely. So we will call it Quantum Key Distribution instead. BB84 is the most known protocol. Selecting randomly the electron's state with SGAs and associating each of them with a bit value will give us a key. To complete the study, we supposed the presence of an eavesdropper and its impact was analysed. In the last section, we exchanged the role of electrons with photons as they are used as qubits in experiments and a modified BB84 protocol was explained to avoid the information leak. Finally we highlighted the main role played by Toshiba in experimental QKD.

In conclusion, although Quantum Mechanics goes against our intuition, its curious rules allow us to develop new technologies such as QKD and we should not see it as something far away since, as we have been able to study, it is already a reality.

References

[1] W. Buchanan, *Cryptography.* Edinburgh: River Publishers, 2017.

[2] L. F. Bauer, *Encyclopedia of Cryptography and Security.* Boston: Springer US, 2005.

[3] R. Oppliger, *Contemporary Cryptography.* Norwood: Artech House, 2005.

[4] C. Hughes, J. Isaacson, A. Perry, R. F. Sun, and J. Turner, *Quantum Computing for the Quantum Curious.* Switzerland: Springer, 2021.

[5] C. Paar and J. Pelzl, *Understanding Cryptography.* Berlin: Springer, 2009.

[6] C. Cid, S. Murphy, and M. Robshaw, *Algebraic Aspects of Advanced Encryption Standard.* Boston: Springer, 2006.

[7] J. R. Vacca, *Computer and Information Security Handbook.* Cambridge: Morgan Kaufmann Publishers, 2017.

[8] S. Katzenbeisser, *Recent Advances in RSA Cryptography.* Boston: Springer, 2001.

[9] S. Vaudenay, *A Classical Introduction to Cryptography.* New York: Springer, 2006.

[10] S. Y. Yan, *Cryptanalytic Attacks on RSA.* New York: Springer, 2008.

[11] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Journal on Computing*, vol. 26, pp. 1484–1509, 1997.

[12] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Annual ACM Sumposium on Theory of Computing*, pp. 212–219, ACM, 1996.

[13] C. Balamurugan, K. Singh, G. Ganesan, and M. Rajarajan, "Post-quantum and code-based cryptography—some prospective research directions," *Cryptography*, vol. 5, p. 38, 2021.

[14] L. Chen, S. Jordan, Y.-K. Liu, D. Moody, R. Peralta, R. Perlner, and D. Smith-Tone, "Nist, report on post-quantum cryptography," 2016-04-28 2016.

[15] D. Z. Albert, *Quantum Mechanics and Experience.* United States: Harvard University Press, 1992.

[16] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information.* Cambridge: Cambridge University Press, 2010.

[17] B. K. Meister, "The Helstrom bound," arXiv:1603.04774, 2016.

[18] M. Razavy, *Heisenberg's Quantum Mechanics.* United States: World Scientific Publishing, 2011.

[19] A. Einstein, B. Podolski, and N. Rosen, "Can quantum-mechanical description of physical reality be considered complete?," *Phys. Rev.*, vol. 47, pp. 777–780, 1935.

[20] J. S. Bell, "On the Einstein-Podolski-Rosen paradox," *Physical Review*, vol. 1, pp. 195–200, 1964.

[21] D. McMahon, *Quantum Computing Explained.* New Jersey: John Willey and Sons Inc., 2008.

[22] M. L. Bellac, *A Short Introduction to Quantum Information and Quantum Computation.* New York: Cambridge University Press, 2006.

[23] W. K. Wootters and W. H. Zurek, "The no-cloning theorem," *Phys. Today*, vol. 62, pp. 76–77, 2009.

[24] W. Dür, R. Lamprecht, and S. Heusler, “Towards a quantum internet,” *Eur. J. Phys.*, vol. 38, p. 043001, 2017.

[25] E. Dervisevic and M. Mehic, “Overview of quantum key distribution technique within IPsec architecture,” in *ISCRAM 2021 Conference Proceedings – 18th International Conference on Information Systems for Crisis Response and Management* (A. Adrot, R. Grace, K. Moore, and C. W. Zobel, eds.), pp. 391–403, Blacksburg, VA (USA), Virginia Tech.

[26] N. Gisin, G. Ribordy, W. Tittel, and H. Zbinden, “Quantum cryptography,” *Rev. Mod. Phys.*, vol. 74, pp. 145–202, 2002.

[27] M. Niemiec, “Error correction in quantum cryptography based on artificial neural networks,” *Quantum Information Processing*, vol. 18, pp. 164–192, 2019.

[28] W. Y. Hwang, “Quantum key distribution with high loss: Toward global secure communication,” *Phys. Rev. Lett.*, vol. 91, p. 057901, 2003.