Electrical, Control and Communication Engineering
ISSN 2255-9159 (online)
ISSN 2255-9140 (print)
2022, vol. 18, no. 1, pp. 43–49
https://doi.org/10.2478/ecce-2022-0006
https://content.sciendo.com

Simulation and Application Purpose of a Randomized Secret Key with Quantum Key Distribution

Olaf Grote* (*PhD Student, Department of Telecommunications Systems & Engineering, University Politécnica de Madrid, Spain*),
Andreas Ahrens (*Professor, Hochschule Wismar, University of Applied Sciences: Technology, Business and Design, Wismar, Germany*)

***Abstract* – The Quantum Key Distribution (QKD) is a well-researched secure communication method for exchanging cryptographic keys only known by the shared participants. The vulnerable problem of a secret key distribution is the negotiation and the transfer over an insecure or untrusted channel. Novel further developments of the QKD communication method are part of in-field technologies and applications in communication devices, such as satellites. However, expensive physical test setups are necessary to improve new application possibilities of cryptographic protocol involving components of quantum mechanics and quantum laws of physics. Therefore, optical simulation software can play a part in essential QKD simulating and further developing quantum-based cryptosystems. In the paper, the authors consider a feasible QKD setup based on the BB84 protocol to create a symmetric key material based on achieving a linear key rate via optical simulation software. The paper still provides two experimental architecture designs to use the QKD for a cryptosystem.**

***Keywords* – BB84 protocol, Quantum key distribution, Quantum cryptography, Simulation.**

I. Introduction

Quantum Cryptography (QC) and Quantum Key Distribution (QKD) are generally synonymous in the literature. The primary focus is to enable secure key distribution based on quantum mechanics. The QKD differs from the classical key distribution. The classical key distribution based on schemes, such as integer factorisation or discrete logarithm, uses the unproven mathematical assumption, which cannot be solved in a definite time with classical computing resources. According to Shor's algorithm [1], the asymmetric encryption methods, which break in polynomial time cryptographic primitives and schemes, are considered broken [2]. The QKD works on the physical axioms of quantum mechanics with subatomic particles, such as photons, rather than computational complexity. The benefit of quantum is the randomly selected measurement method, which provides a correlation result for both participants without disclosing their measured values. Both participants, therefore, have a shared key, while an attacker has no information about the key. Any eavesdropping in the quantum key distribution process can be detected, guaranteed by the quantum No-Cloning Theorem. Reference [3] is the first research work that identified specifically prevented copies of an unknown quantum state. QKD is, consequently, a secure method for key exchange with unconditional security that also provides the ability to detect the presence of an eavesdropper attempting to learn the key. The first QKD protocol, BB84 protocol, was proposed and named by Bennett and Brassard in 1984 [4]. Since then, additional QKD protocols and adapting, subsequent QKD protocols have been invented, e.g. B92 protocol [5] and E91 protocol [6]. Other QKD setups e.g. the BBM92 protocol [7] operate with the entanglement-based function. However, all QKD protocols based on the essential properties of BB84 and is the most analysed and often implemented QKD protocol. It is a high-quality protocol for a noiseless quantum channel. Our experiment deals with the BB84 protocol to provide a secret key using quantum mechanical effects for research purposes. The unconditionally secure protocol, in theory, needs further experimental verification. This paper shows the complete process from generating a photon-polarised coded bitstream to a final binary key and concluding ASCII converted symmetric key for data encrypting and decrypting purposes by using, e.g. Java program. Furthermore, an architecture and design purpose describe a technical use case to demonstrate this technical approach.

In contrast to other works and QKD simulations, e.g., Buhari et al. [8] and Archana et al. [9], with the focus on the generic basic setup of the QKD protocol, our proposed simulation concentrates on practical work. Our research interest is the process of the quantum bit-stream generation, the binary bit-stream composite, and the formal encoding to a symmetric key for cryptographic purposes by adapting the initial QKD simulation. This research article is a follow-up work based partially on our short conference paper [10] and adapted the QKD BB84 protocol setup by optical simulation software with its sequential process steps. The main contributions of our presented study are the practical QKD simulation and theoretical architecture design to use the symmetric key material. We also provide an overview of how QKD could be integrated into a cryptosystem.

* Corresponding author.
E-mail: olaf.grote@alumnos.upm.es

The rest of the perspective paper is structured as follows. Section 2 describes the simulation and modelling of the QKD protocol BB84. A brief description of the QKD communication protocol shows the functional work of BB84. We introduce our related simulation and discussion results in Section 3. In Section 4, we consider the two theoretical architecture designs for this simulation, which is based on three generic phases. Finally, the paper concludes in Section 5.

II. Setup and Simulation of BB84 Protocol

The setup of our simulation is based on the original QKD protocol BB84 and modelled with the optical simulation framework OptSim^{TM} [11]. The modelling divided into three major stages: transmitter, channel, and receiver. Fig. 1 shows the schematic high-level experimental QKD system. On the transmitter side, the laser emits photons that pass through an amplitude (AM), phase modulator (PM), and multiplexer (MUX). The polarised photons are sent via the fibre channel (FC) to the receiver side (DE-MUX). The received polarised photons were detected (DT) and measured by an optical meter device. We illustrate

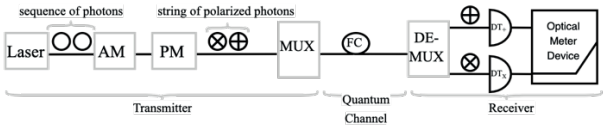

Fig. 1. Schematic setup of a generic QKD environment.

the experimental details as depicted in Fig. 2. The transmitter side initiates the quantum sequence and generates single photons. Photons are based on electromagnetic waves within an electric field and a magnetic field vertical to them. These electromagnetic waves can vibrate and align in some direction. This direction can be used for the polarization to pass the quantum device to give them an exact and measurable value, the quantum state of each photon. After passing the quantum device, the emitted photons are sent through a Quantum Channel (QC). The QC describes the communication channel where the polarised data stream is transmitted to the receiving end. This QC between the transmitter and receiver applies a depolarised channel (Pauli channel). On the receiver side, we detect the photons with a measured polarization. During the data comparison, the transmitter and recipient agree on whether errors have occurred during the transmission. The QKD process and protocol implementation need a lab or testbed with dedicated technical components and physical environments. Taking these factors into account, the physical optical components were simulated with an optical program. The block diagram shows a simplified procedure for the QKD protocol and is intended to serve as a basis for creating integration layers and interfaces. Furthermore, the diagram lacks the comparison between transmitter and receiver via another, classic channel. Since in our setup it can be assumed that a research team is working on a setup and emulating the transmitter and receiver. The block diagram shows a purely sequential process, coordinated with the optical software. This is automated for programmatic iteration control.

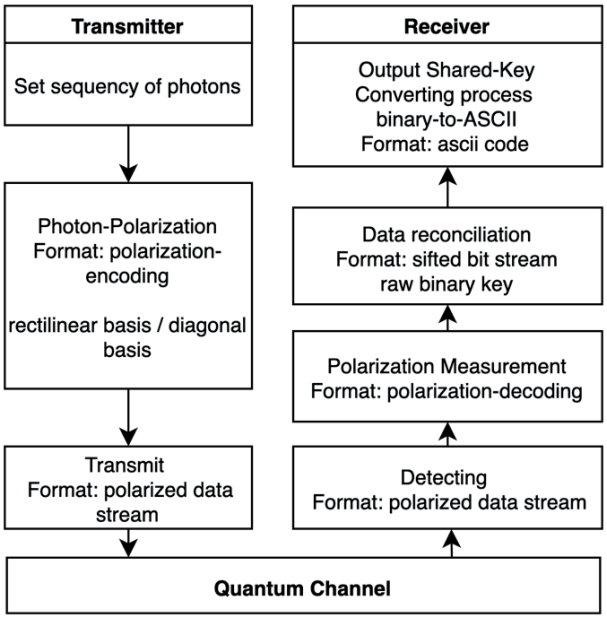

Fig. 2. QKD communication block diagram.

A. BB84 Protocol Notations and Preliminaries

Fig. 3 illustrates the experimental setup. The slightly modified operation steps of our protocol are as follows. A Coherent Wave (CW) Laser device is the quantum source that emits single photons with quantum properties of light for data transition. After passing a beam polarization, the photon contains a specified polarization angle. After transmitting the qubit through the QC, the qubits pass the Beam Splitter (BS), the Polarizing Beam Splitter (PBS), and are detected by a Single Photon Detector (SPD). Finally, the qubits are measured, verified, and QBER (Quantum Bit Error Rate) corrected, and the final binary key is obtained. In our last task, the binary key is encoded into ASCII format for symmetric encryption purposes. Table I shows the associated physical QKD devices with our software items. Then, the ideal BB84 protocol is initiated in our simulation and the procedure is adjusted.

TABLE I

Alignment of Physical and Software Components

Physical component	**Software component**	**Properties**
Coherent Wave (CW) Laser	CW Laser (single Mode)	Frequency
Beam Splitter (BS)	Pseudo Random Binary Sequence (PRBS)	50:50 division
Polarizing Beam Splitter (PBS)	Polarization Transformer	$\|0\rangle$, $\|45\rangle$, $\|90\rangle$, $\|135\rangle$
Quantum Channel (QC)	Optical Fibre Channel (FC)	Wavelength, Loss
Single Photon Detector (SPD)	Polarization Monitor	$\|0\rangle$, $\|45\rangle$, $\|90\rangle$, $\|135\rangle$

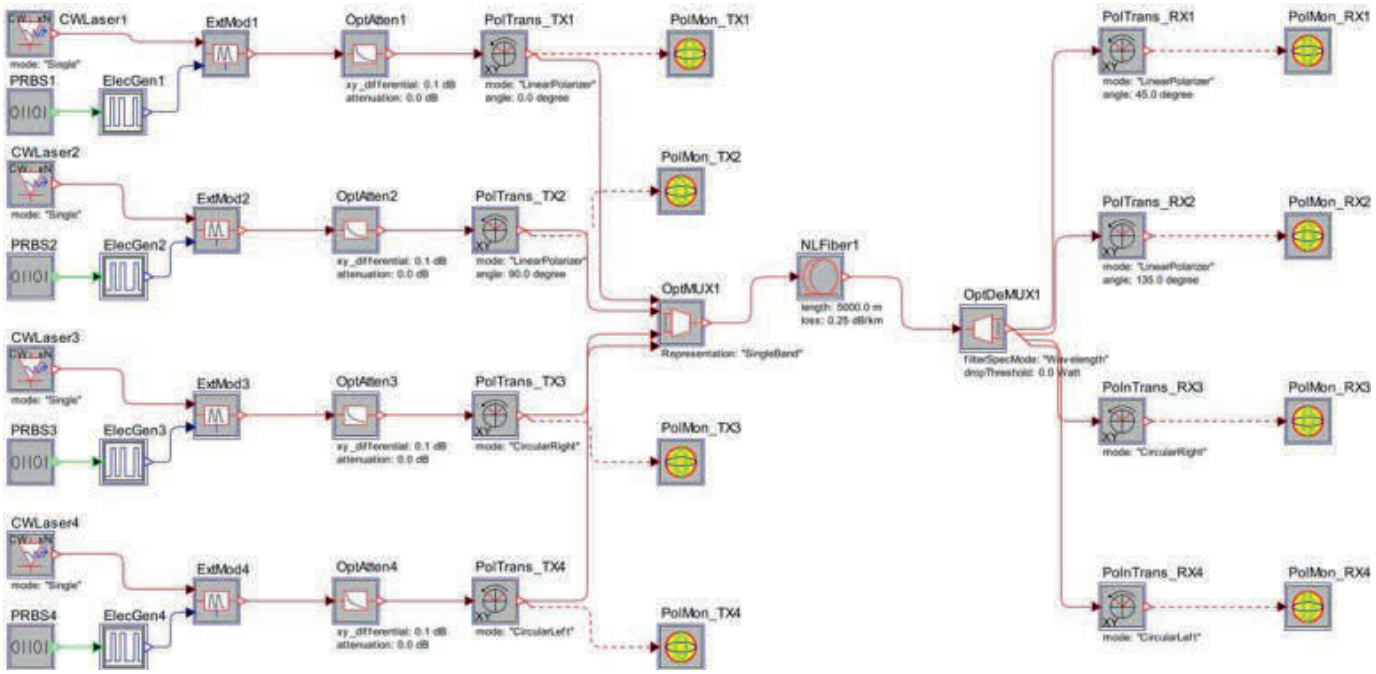

Fig. 3. BB84 protocol setup of the simulation.

- Transmitter: The simulation is started with the sequence initialising S of randomly generated photons p_i as:

$$S = \{p_1, \ldots, p_n\}. \quad (1)$$

- Transmitter: The quantum state is denoted as one of four polarization angles. Let $|\varphi\rangle$ be the quantum state of the photon p_i, where

$$p_i \in \{|0\rangle, |45\rangle, |90\rangle, |135\rangle\}, i = (1, \ldots, n), i \in \mathbb{N} \quad (2)$$

and the transformed photons p_i are sent via a quantum channel.

- Receiver: Two complementary bases can be measured, i.e., the rectilinear basis with the vector notation horizontal as $|H\rangle$ with 0 degree and the orthogonal vector $|V\rangle$ In opposite, the diagonal basis is defined with the diagonal vector $|D\rangle$ and 45 degree and the anti-diagonal vector $|A\rangle$ as 135 degree. On the receiver side, the polarized photons and randomly chosen measurement bases are detected, where

$$|H\rangle = |0\rangle, |V\rangle = |90\rangle, |D\rangle = |45\rangle, |A\rangle = |135\rangle \quad (3)$$

- Receiver: All data are captured as a string A, where

$$A = (a_1, \ldots, a_n), a_i \in \mathbb{R}. \quad (4)$$

It is assumed that a_1 is not mandatory for the correct measured polarization state $|\varphi\rangle_i$ corresponding (3). In this case, A stands for the captured data stream as string (a_1,…, a_n), where a_1 is a measured value of $\mathbb{R}$. Furthermore, the quantum bit error rate $QBER$ is computed, and the process is continued when the quantum error bit rate average $QBER_\varnothing$ condition is valid:

$$QBER \leq QBER_\varnothing \quad (5)$$

Hence, it is possible to detect a physical interference of the quantum channel or cheating by a Man-in-the-Middle-Attack (MITM) with a raised $QBER_\varnothing$.

- Transmitter-Receiver: Data reconciliation and negotiation via a public channel will be decided to trust the binary string A. Based on condition (4) and (5), the polarization data stream A is examined, cleaned up, and decoded. After negotiation, the shifted raw binary key will be transformed into a binary string K_b, where

$$K_b : A \rightarrow K_b(A), K_b \in 0,1. \quad (6)$$

- Converting process. The final binary key K_b is ready to convert in ASCII code for cryptographic purposes. Let k be a symmetric key based on ASCII code table. The binary key K_b is mapped on the ASCII Unicode number, where

$$k : K_b \rightarrow k. \quad (7)$$

B. Initializing Quantum Transmitter

For implementing the BB84 protocol, the polarization of singular photons is crucial to encode the binary information. The polarization encoding process aligns the binary bit-stream with the polarised photons. The photons use conjugate bases, either rectilinear or diagonal basis, so that orthogonal states can arise with the binary information. Therefore, on the one hand, a single photon assumes a binary state of '0' if it has an angle of 0° with a rectilinear base or 45° with a diagonal base. On the other hand, to create a binary state of '1', an angle of 90° is used with a rectilinear base or 135° with a diagonal base. For modelling a truly random state, Pseudo-Random Binary

Sequence (PRBS) component determines the reflection and transmission. Furthermore, the software-defined Polarizing Beam Splitter (PBS), named Polarization Transformer, splits the orthogonal quantum state of the photon, and becomes a qubit. In our simulation, a qubit is described with a (2×2) density matrix. According to [12], a density matrix represents the polarization states. In our simulation, we align the dedicated polarization to a binary quantum state, defined in condition (3) and represented as:

$$|H\rangle|0\rangle = 0, |V|90\rangle = 1, \quad (8)$$

$$|D\rangle|45\rangle = 0, |A|135\rangle = 1. \quad (9)$$

C. Quantum Channel

The photons with the quantum state pass the depolarised channel, called Pauli channel. A Pauli channel is a practical model for measurement and analysis of error correction, according to [13]. A single-photon through the noisy depolarised channel appears to maintain the native state with an error probability. Noise considers a big challenge in QKD. This problem can be overcome by modifying the simulation. In our setup, a noise immune QKD is simulated. By the way, the noise from eavesdropping is a dedicated research field. Noise can come from various optical components, such as fibre optic channel, and issues, such as birefringence, polarization dispersion, and free space, e.g., scattering, absorption, and diffraction. To summarise, noise triggering factors cause poor performance in QKD, especially for the secure key generation rate and long-distance. Our simulation framework implements a single-mode Fibre Optic Channel with default cable wavelength $\lambda = 1550$ nm and standard attenuation coefficient of 0.25 dB/km. Transmitter and receiver side are connected for Multiplexer/Demultiplexer over 5 km fibre.

D. Measurement and Data Reconciliation

Measurement and data reconciliation are the active processes on the receiver side. The detector function is to capture and convert the light into an electrical signal. Avalanche Photon Diode (APD) achieves a sufficient detection rate of 50 % quantum efficiency. Thus, the quantum position is measured with a polarization monitor in our simulation, all encoded information is captured, and each iteration is tagged. As a result, it is easier to identify and negotiate the bit error in the raw key during the data reconciliation process and finalize it. Based on the Heisenberg Uncertainty Principle, an incorrectly measured base on the receiver side leads to the quantum information suddenly lost on the other bases. Therefore, our measurement will occur with a random probability of 50 %. The assumption is that the transmitter side chooses the measurement basis incorrectly on average 50 % during the quantum key exchange. Nevertheless, only 25 % of measured bits will be different. The detected polarised data stream contains the measured basis and can be converted into a binary string to get the raw binary key. The length of the raw binary key is approximate about half of the initialized bit sequence. After capturing the whole key material with bit errors, the key sifting solves the invalid polarization states. Then, the participants of the key distribution align the information via a classical channel. The key distillation is simulated in our simulation and the iteration is aborted if the bit error rate is larger than the threshold. The last and final step is to encode the final binary key in ASCII code. The final bit key is encoded with ASCII based characters, where a character corresponds to a decimal number for cryptographic purposes.

III. Simulation Results and Discussion

Simulation results are essential for verifying the simulation experiment. We explain the principle of our simulation and give the results. The objective of the paper is to improve and set up an applied QKD simulation with optical software. Several test sequences and iterations were performed to simulate communication. Furthermore, we use pseudo-randomized numbers were used to emulate BS, which is not perfect but sufficient. Thus, a perfect random number is unnecessary for secure QKD, which was published by Wang [14]. In general, the focus is to set up a QKD simulation for evaluation and study QKD protocols efficiently. The comparison results for each QKD phase are presented in tabulated form in our paper [10]. Our test laboratory simulates a distance length of 5 km for measurement purposes and minimises inaccuracies such as imperfect quantum source and noise. Practical results on the secure QKD over much longer distances are well known [15]. From a simulation perspective, it is necessary to increase the distance to ensure a valid symmetric key material over a long distance as well. Furthermore, a java-based framework is created to convert the binary key into an ASCII-encoded 8-byte crypto key. Nevertheless, some correlations of simulation output statistics and published research results are still upcoming challenges.

IV. Architecture and Design Purpose

The experimental and theoretical architecture is based on the idea that QKD can be used to generate secret key material between two distant parties for a cryptographic purpose. Pursuing the experimental and theoretical architecture is based on the idea that QKD could generate secret key material between two distant participants for a cryptographic purpose and to integrate into a cryptosystem. Two designs of architecture are presented to model the QKD for additional usage. The on-premises key generator QKD model is designed for test-labs and playground purpose to test the functionality of new QKD protocols and properties. The remote key generator QKD model is an enterprise design. Both models use the QKD functionality and our generic three phase process.

A. Preliminaries and Notations

The following notations are used in this architecture design:

K_Q – the created symmetric key material via QKD;
K_{QA} – the created transmitter key material via QKD;
K_{QB} – the created receiver material key via QKD;
$t_n(z)$ – time n during a process or function z.

The system model is generally divided into three phases to ensure simple implementation. The three phases are the following: Initialisation Phase, Storing Phase and Cryptosystem Phase. Due to the sequential execution of the three phases, the following phases are initialised at a particular time *t*. Each phase splits into generic processes and executes a function *f* or an operation *x*. Thus, time *t* applies to the respective phases such as t_0(QKD), $t_1(x)$, $t_2(f)$. The Initialisation Phase t_0(QKD) describes the native QKD method with the three agnostic QKD processes: 1) raw key exchange, 2) sifting key and 3) secret key defined in our QKD simulation. The Initialisation Phase uses this method to generate a key for symmetric cryptographic purpose, which is defined as K_Q. The following Storage Phase $t_1(x)$ contains a method to save the symmetric key K_Q from the Initialisation Phase and keep it available for cryptographic purposes. The Storage Phase contains two central processes: the key storage process and the interface method for pull/push requests. The Cryptosystem Phase $t_2(f)$, the third and final phase, carries the cryptographic process. Let $f(e)$ be a function to encrypt a message *m* and $f^{-1}(d)$ – an inverse function to decrypt a ciphertext *c*. The key K_Q can be used as a Pre-shared key (PSK) for a symmetrical encryption and decryption process. The modular structure is particularly advantageous for further adaptations or changes. Furthermore, post-quantum cryptographic (PQC) encryption can be implemented in the cryptosystem phase without an adjustment of the quantum key generation Initialisation Phase. Fig. 4 visualises the structure of the three phases with the processes. The presented system model is divided into two application scenarios. The first scenario is a laboratory environment named "On-premise key generator" to simulate and test the implementation. The second application scenario, defined as "Remote key generator", represents a possible use-case to create a symmetric key material on two remote sides at the same time. Both system models use the introduced notations.

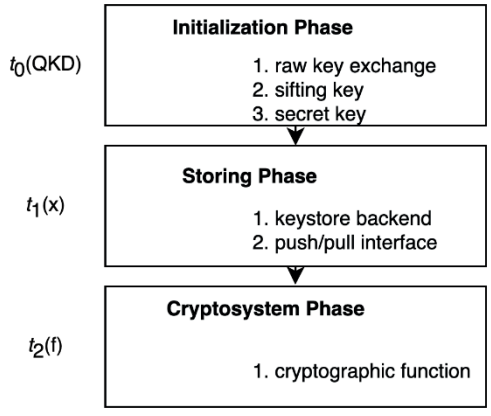

Fig. 4. View of generic three phases with the dedicated processes and time steps.

B. On-premise Key Generator

The Initialisation Phase begins with the native QKD at time t_0 and is initiated locally in a trustworthy environment. Thus, for t_0(QKD) the structure for two environments A and B, applies to instantiate a truly randomised key. Quantum Generator – Setup A initiates the QKD process, while Quantum Generator – Setup B is the recipient in this QKD environment. The three agnostic standardised QKD processes contain for both setups: 1) raw key exchange, 2) sifting key and 3) secret key. Furthermore, let K_Q be the truly randomised key and K_{QA}, K_{QB} –the keys of the participants; therefore, $|K_Q| \equiv |K_{QA}| \equiv |K_{QB}|$ applies. The time $t_1(x)$ describes the Storing Phase. Using a programmatic operation, the truly randomised key K_Q from the first phase can be saved and used for another push/pull cryptographic process via an interface request. The Cryptosystem Phase $t_2(f)$ uses the truly randomised key K_Q via pull request from the keystore backend for cryptographic function *f*. Fig. 5 shows the workflow as a block diagram:

- Initialisation Phase t_0: This phase is the native QKD protocol defined as t_0(QKD) for key generating. The three key creating processes of the QKD protocol deliver the raw key exchange between setup A and B on classical bit-stream material with the sifting key correlation processes and random sequence of bits. This process output is the truly randomised secret stream key for symmetric encryption. Output: Symmetric Key K_Q.
- Storing Phase t_1: Input: Symmetric Key K_Q. The secret key K_Q must remain trusted and safe from an adversary on-site and off-site. The secret key K_Q is stored with the function $t_1(x)$ in a key-ring backend. This operation trigger depends on the request (push/pull), which corresponds to the keystore backend interface to handle the symmetric key K_Q. The integrity of the key K_Q can be proofed with a quantum-safe hash-based function. Output: pull/push request, K_Q.
- Cryptosystem Phase t_3: Input: push/pull request, K_Q. The generic building block of the cryptosystem is asymmetrically triggered. Each algorithm for encryption $f(e)$ and/or decryption function $f^{-1}(d)$ with the same symmetric key K_Q may be disjoint from each site.

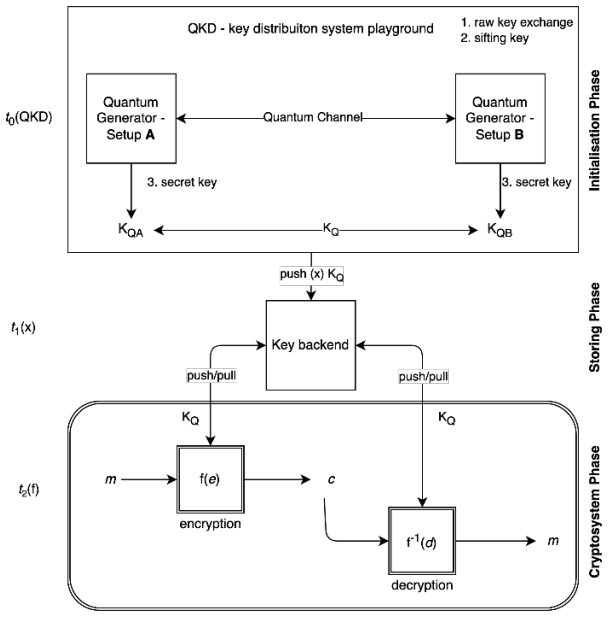

Fig. 5. Generic view of the QKD on-premise architecture.

C. Remote Key Generator

Identical three phases are illustrated in Fig. 6 but stretched over two locations. One location is on-site with the Quantum Generator – Setup A and initiates QKD properties. The Quantum Generator – Setup B environment operates on the off-site co-location. The red double arrow describes the geographical and physical distance between the locations. The quantum channel connects the two locations. This setup enables at once two participants and will be described as one system unit measurement. The single-photon measurement guarantees the verification of photon states and in the end the generation of symmetric key material for cryptographic purposes. The datacentre connections between the two geographical sites define the quantum communication setup. A very short distance and free space QKD communication can also be realised with a wavelength $\lambda = 800$ nm. Performing QKD setup for long-distance communication, such as regional or international datacentres, the fibre-network communication should have a wavelength λ value between 1300 nm and 1550 nm. The connection is implemented via Fibre Channel (FC). Furthermore, an Internet Protocol (IP) based network connects the two locations for data transfer. The generated key K_Q is available locally K_{QA} and remotely K_{QB} and defined as exactly $|K_Q| \equiv |K_{QA}| \equiv |K_{QB}|$. However, the quantum key can be used for a symmetric cryptographic purpose. The key K_Q will be stored as usual in a keystore backend. It can be used with common interfaces. In this case, a data vault is locally encrypted with a quantum-safe procedure and the quantum key K_Q, which already exists off-site. This vault will be synchronised with built-in methods over a classic secured network connection such as Internet Protocol Security (IPsec) or Transport Layer Security (TLS) to the off-site location. On the off-site location, the vault can be decrypted with the existing and stored K_Q with truly quantum mechanical properties based on physical fundamentals. The detailed workflow proceeds:

- Initialisation Phase t_0: The first step is to send photons from the transmitter on-site location via an established quantum channel to the receiver off-site location. It means that each photon is encoded with one of two orthogonal polarisation bases, and each polarisation may represent one of two states. This can be a linearly polarised photon. It may be either vertically or horizontally polarised. The second step is to measure the state on the off-site environment for each photon along a randomly chosen basis. The process ends after matching segments of the bit-stream key and the exceeded error rate is aligned with the expected error rate, and the used bases for each photon are compared. The result is the truly randomised key K_Q for a cryptographic purpose, stored simultaneously in two disjointed locations. Output: Symmetric Key K_Q for on-site and off-site environment.
- Storing Phase t_1: Input: Symmetric Key K_Q. The secret key K_Q must remain trusted and safe from an adversary on-site and off-site. The secret key K_Q must be stored with the function $t_1(x)$ in a saved keystore backend. The trigger for this operation depends on the request (push/pull), which corresponds to the keystore backend interface to save the symmetric key K_Q. The integrity of the key K_Q can be carried out with a quantum-safe proofed hash-based function. Output: pull/push request, K_Q.
- Cryptosystem Phase t_2: Input: push/pull request, K_Q. The generic building block of the cryptosystem is asymmetrically triggered. Each algorithm for encryption $f(e)$ and/or decryption function $f^{-1}(d)$ with the same symmetric key K_Q may be disjoined from each site. In a case of a proof of concept or for research, the cryptosystem uses a symmetric block cipher.

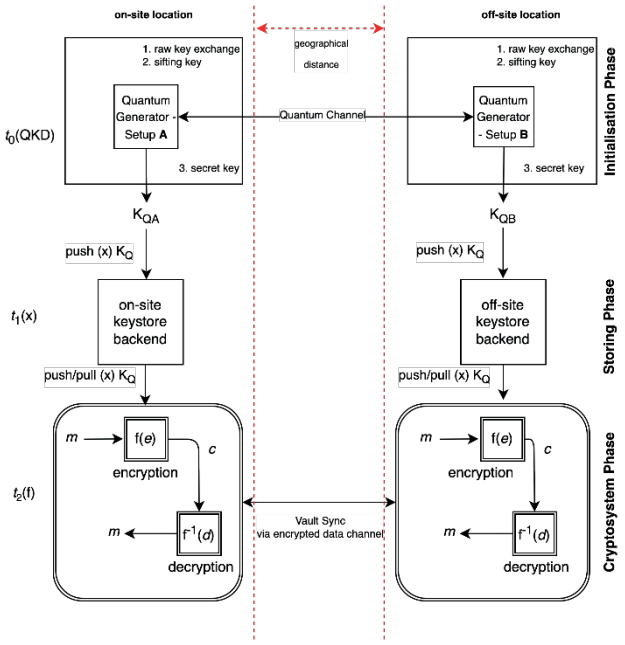

Fig. 6. Stretched architecture with two locations.

V. Conclusion

The paper has proposed a simulation framework to model the QKD BB84 protocol on a classical computer and realise a unique symmetric key for cryptographic purposes. The current setup simulation is based on the functionality of the BB84 protocol. The simulation has demonstrated that a QKD generated secure key can be used for encryption and decryption. Furthermore, two experimental architecture designs have been defined with standardized process steps for another QKD use-case. On the one hand, an example of future work is QKD protocol use for applied platforms with idempotent automated tests to ensure practically oriented implementation. Furthermore, an interface between QKD protocol and an application has to be created. On the other hand, the authors of the research have investigated and analyse the truly randomized entropies symmetric key. Another item of our further research plan is the implementation of other QKD protocols and methods in order to use this simulation for further QKD tests and change the underlayer cryptosystem with a post-quantum cryptography (PQC) scheme.

REFERENCES

[1] P. W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *Proceedings 35th Annual Symposium on Foundations of Computer Science*, Santa Fe, NM, USA, Nov. 1994, pp. 124–134. https://doi.org/10.1109/SFCS.1994.365700

[2] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Journal on Comput*ing, vol. 26, no. 5, pp. 1484–1509, 1997. https://doi.org/10.1137/S0097539795293172

[3] J. L. Park, "The concept of transition in quantum mechanics," *Found Phys*, vol. 1, pp. 23–33, Mar. 1970. https://doi.org/10.1007/BF00708652

[4] C. H. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," in *Proceedings of the IEEE International Conference on Computers, Systems and Signal Processing*, Bangalore, India, Dec. 1984, pp. 175–179.

[5] C. H. Bennett, "Quantum cryptography using any two nonorthogonal states", *Physical review letters*, vol. 68, pp. 3121–3124, 1992, https://10.1103/physrevlett.68.3121

[6] A. K. Ekert, "Quantum cryptography based on Bell's theorem," *Physical review letters*, vol. 67, pp. 661–663, 1991. https://10.1103/PhysRevLett.67.661

[7] *S*. Mishra *et al*., "BBM92 quantum key distribution over a free space dusty channel of 200 meters", *Journal of Optics*, vol. 24, no. 7, pp. 074002, 2022. https://doi.org/10.48550/arXiv.2112.11961

[8] A. Buhari, Z. A. Zukarnain, S. K. Subramaniam, H. Zainuddin, and S. Saharudin, "An efficient modeling and simulation of quantum key distribution protocols using OptiSystem," in *2012 IEEE Symposium on Industrial Electronics and Applications*, Bandung, Indonesia, Sep. 2012, pp. 84–89. https://doi.org/10.1109/ISIEA.2012.6496677

[9] B. Archana and S. Krithika, "Implementation of BB84 quantum key distribution using OptSim," in *2015 2nd International Conference on Electronics and Communication Systems (ICECS)*, Coimbatore, India, Feb. 2015, pp. 457–460. https://doi.org/10.1109/ECS.2015.7124946

[10] O. Grote, A. Ahrens, and C. Benavente-Peces, "Modelling and simulation of quantum key distribution using OptSim," in *2021 IEEE Microwave Theory and Techniques in Wireless Communications (MTTW)*, Riga, Latvia, Oct. 2021, pp. 160–164. https://doi.org/10.1109/MTTW53539.2021.9607165

[11] Synopsys. *OptSim, Software for Design and Simulation of Optical communication*. [Online]. Available: https://www.synopsys.com/photonic-solutions/optsim/single-mode-network.html. [Accessed on: Dec. 13, 2021].

[12] S. Pirandola *et al*., "Advances in quantum cryptography", *Advances in Optics and Photonics*, vol. 12, no. 4, pp. 1012–1236, 2020. https://doi.org/10.1364/AOP.361502

[13] S. Flammia and J. Wallman, "Efficient estimation of Pauli channels", *ACM Transactions on Quantum Computing*, vol.1, no. 1, Art. no. 3, Dec. 2020. https://doi.org/10.1145/3408039

[14] X. Wang, "Perfect random number generator is unnecessary for secure quantum key distribution", 2004, *arXiv*:quant-ph/0405182v2.

[15] A. Boaron *et al*., "Secure quantum key distribution over 421 km of optical fiber", *Physical Review Letters*, vol. 121, Art. no. 190502, Nov. 2018. https://doi.org/10.1103/PhysRevLett.121.190502

Olaf Grote received his Dipl.-Inform. (FH) and M. Eng. degree in IT Security and Forensics from University of Applied Sciences of Wilhelm Büchner Darmstadt, Germany and University of Applied Sciences of Wismar, Germany in 2012 and 2018, respectively. Since 2019, he has been a PhD student at the Higher Technical School of Telecommunication Engineering and Systems, Polytechnic University of Madrid, Spain. His research interests are Post-quantum Cryptography (PQC) and the application of quantum cryptography distribution of cryptography keys QKD.
Address: Department of Telecommunication Engineering and Systems, the Universidad Politécnica Madrid., 28031 Madrid, Spain.
E-mail: olaf.grote@alumnos.upm.es
ORCID iD: https://orcid.org/0000-0003-4158-5538

Andreas Ahrens received the Dr.-Ing. and Dr.-Ing. habil. degree from the University of Rostock in 2000 and 2003, respectively. In 2008, he became a Professor for Signal and System theory at the Hochschule Wismar, University of Technology, Business and Design, Germany. His main field of interest includes error correcting codes, multiple-input multiple-output systems, iterative detection for both wireline and wireless communication, cyber security as well as social computing. Address: Hochschule Wismar, University of Applied Sciences: Technology, Business and Design, Philipp-Müller-Straße 14, 23966 Wismar, Germany.
E-mail: andreas.ahrens@hs-wismar.de
ORCID iD: https://orcid.org/0000-0002-7664-9450