Lab – Quantum Cryptography

Jesus Lopez and Viviana Cadena

Prepared under the supervision of

Mohammad Saidur Rahman, Ph.D.
Department of Computer Science
University of Texas at El Paso

Contents

1 Introduction **2**

2 Quantum Key Distribution Protocols **2**

3 BB84 Protocol **3**
3.1 Steps of the Protocol 3

4 Activities **5**
4.1 Simulating BB84 Protocol 5
4.2 Introduction to Polarization and Bases 6
4.3 Simulating BB84 Protocol with Hardware 7
4.4 Eavesdropping Experiment 8

A 4-Bit Character Encoding Reference **9**

B Deliverables **10**

References **10**

1 Introduction

Cryptography, the practice of encrypting messages and data, has always been a fundamental topic in the field of secure communication [1]. However, classical public cryptographic algorithms share a major limitation — there is no theoretical guarantee that encryption keys will remain secure forever. Even if an adversary cannot currently break the encryption, advances in computing power, such as quantum computer, may eventually able to break the encryption.

Quantum cryptography offers a solution by leveraging the principles of quantum mechanics. One of the fundamental principles of quantum mechanics is that observing a quantum state—like the polarization of a photon—causes it to collapse into a specific value. This property enables the creation of encryption keys that are not only random but also theoretically secure — any eavesdropping attempt can be detected [1]. Consequently, quantum key distribution (QKD) allows both the sender and recipient to share a key with the assurance that any interception would be evident.

2 Quantum Key Distribution Protocols

Quantum Key Distribution (QKD) exploits the principles of quantum mechanics to generate a shared secret key between two parties through a quantum channel, providing information-theoretic security.[2] It relies on the no-cloning theorem, which states that unknown quantum states cannot be duplicated, thus ensuring that any attempt at eavesdropping introduces detectable disturbances. QKD protocols generally consist of two main phases:

- **Quantum transmission phase:** During this phase, quantum states (e.g., polarized photons) are transmitted and measured by the communicating parties (commonly referred to as Alice and Bob).
- **Post-processing phase:** In this phase, the raw bit strings obtained from the quantum transmission phase are processed using classical algorithms for error correction and privacy amplification, ultimately producing a pair of identical and secure cryptographic keys.

Polarization Bases in Quantum Key Distribution

The polarization of photons forms the foundation for encoding information in quantum key distribution (QKD). In this process, information is represented using the polarization states of individual photons. Two primary polarization bases are used:

- **Rectilinear basis (+)**: This basis includes horizontal (0°) and vertical (90°) polarizations.
- **Diagonal basis (X)**: This basis includes polarizations at 45° and 135° angles.

By assigning binary values (e.g., 0 and 1) to specific polarization directions, information can be securely encoded into individual photons. Polarization is achieved using optical components such as polarizing filters or birefringent crystals, which ensure that each photon is emitted with the intended polarization state. This controlled encoding plays a key role in ensuring the security and reliability of quantum communication systems.

A Quantum Key Distribution (QKD) protocol defines the procedures for securely establishing key material between remote users. QKD protocols can be broadly categorized into three main types [3]:

- **Discrete-variable protocols:** These include well-known schemes such as BB84 [4, 5], B92 [6], E91 [7], and SARG04 [8], where individual quantum states (e.g., single photons with specific polarizations) are used to encode information.
- **Continuous-variable protocols (CV-QKD):** These protocols use continuous properties of light, such as the quadratures of the electromagnetic field, to encode key information [9, 10].
- **Distributed-phase-reference protocols:** Examples include Coherent One-Way (COW) and Differential Phase Shift (DPS), which rely on the phase differences between pulses rather than individual quantum states [11, 12, 13].

3 BB84 Protocol

The BB84 protocol, introduced by Charles Bennett and Gilles Brassard in 1984, is the first and one of the most widely known Quantum Key Distribution (QKD) protocols [4]. It enables two parties to *establish a shared secret key over a quantum channel* with the ability to detect any eavesdropping attempts.

3.1 Steps of the Protocol

A. **Alice Sends Qubits**

- Alice randomly selects a bit value (0 or 1) for each qubit.
- She randomly chooses a basis for encoding each bit: rectilinear (+) or diagonal (X).
- Using these selections, she prepares and sends a stream of qubits to Bob over a quantum channel.

B. **Bob Measures the Qubits**

- Bob independently and randomly selects a basis (rectilinear or diagonal) to measure each incoming qubit.
  - If Bob's basis matches Alice's, he measures the correct bit value.
  - If the bases differ, the measurement result is random.

C. **Public Basis Comparison**

- After all qubits have been transmitted and measured, Alice and Bob publicly compare the bases they used (not the bit values).
- They discard all bits where their bases did not match.
- The remaining bits, for which the bases matched, form the *raw key*.

D. **Error Checking**

- Alice and Bob publicly compare a random subset of their raw key to check for discrepancies.
- If an eavesdropper (Eve) attempted to intercept and measure the qubits, her actions would introduce detectable errors due to the no-cloning theorem and measurement disturbance.
  - If the error rate is below a predefined *threshold*, they proceed [14, 15, 16].
  - If too many errors are detected, the key is discarded, and the protocol restarts.

E. **Post-Processing**

- **Error Correction:** Alice and Bob apply classical error correction techniques to resolve any remaining discrepancies.

- **Privacy Amplification:** They apply privacy amplification methods to distill a final, shorter secret key that is highly secure [17, 18, 11].

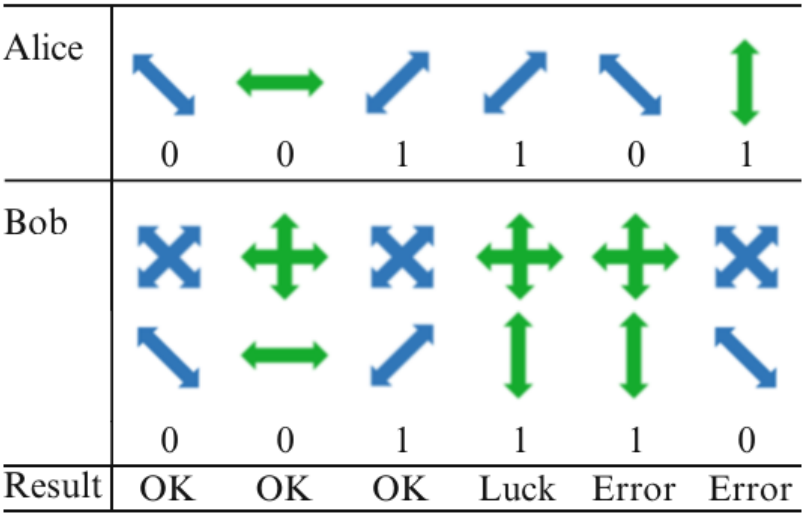

Figure 1: Illustration of key distribution using the BB84 protocol. Alice sends qubits prepared with random bit values and polarization bases (rectilinear or diagonal). Bob measures each qubit using randomly chosen bases. After public basis comparison, only the bits measured using matching bases are retained to form the raw key. *Adapted from* [3].

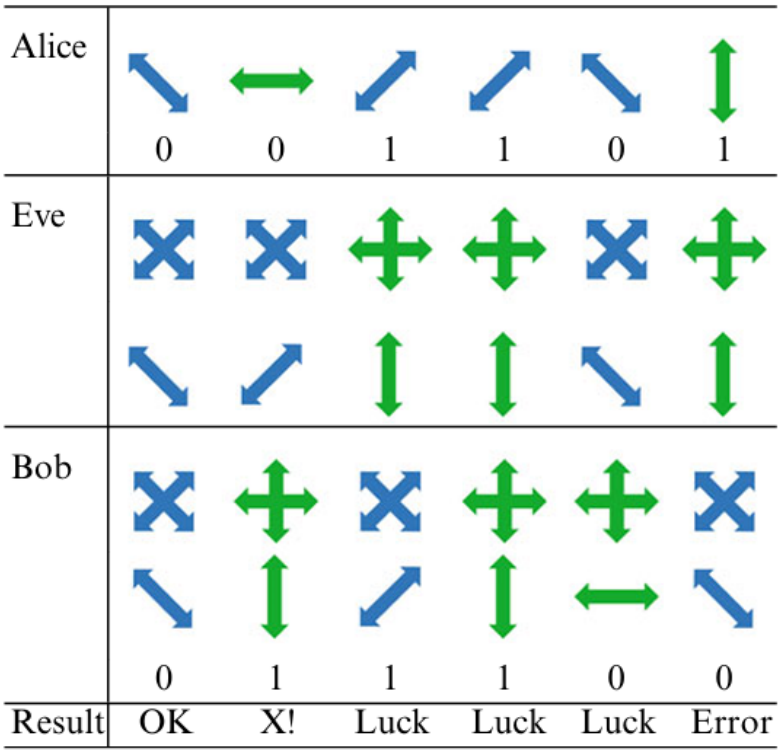

Figure 2: An example of eavesdropping in the BB84 protocol. Eve intercepts and measures the qubits, introducing errors when her basis does not match Alice's. As a result, Bob may receive incorrect values. These inconsistencies help Alice and Bob detect Eve's presence during the key verification stage. *Adapted from* [3].

4 Activities

Disclaimer: For this lab we are going to be working with the Quantum Cryptography Demonstration kit developed by Thorlabs [19]. **Please make sure to be wearing gloves all the time you are operating the lab equipment**.

4.1 Simulating BB84 Protocol

Objective: Simulate key generation using the BB84 quantum key distribution protocol and encrypt a message using bitwise XOR.

Instructions: We are going to trace the transmission, sifting, and encryption of two letters using bases, bits, and XOR logic.

1. Alice sends bits prepared in random bases. Bob randomly chooses his measurement basis.

Pulse	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**
Alice Basis	×	×	+	×	+	+	+	×	×	+	×	×	×	+	+	×	+	×
Alice Bit	1	0	0	1	1	0	0	1	1	1	0	1	0	0	0	1	0	1
Bob Basis	+	×	×	×	+	+	+	×	+	×	×	+	+	×	+	+	+	×
Bob Bit																		
Match?																		

Table 1: Alice Transmission and Bob Measurement

***Note**: When bases match, Bob records Alice's bit deterministically. When bases do not match, 0 or 1 is selected at random. (Pick them yourself)*

2. Alice and Bob exchange their bases over a public channel. Circle or check (✓) each pulse where the bases match.

3. Sifting yields the 10-bit key: `0110010001`.
Alice and Bob select the **first 8 bits** of this key ($K =$ `01100100`) for encryption.

Letter Definitions:	**H** $\rightarrow 0111$	**I** $\rightarrow 1000$	**J** $\rightarrow 1010$	**L** $\rightarrow 1100$
XOR ($\oplus$) Rules:	$0 \oplus 0 = 0$	$1 \oplus 1 = 0$	$0 \oplus 1 = 1$	$1 \oplus 0 = 1$

4. **Alice: Encryption ($C = M \oplus K$):** Select two letters from the box above to form an 8-bit plaintext message M, and calculate the encrypted ciphertext C using XOR:

Letter	Letter 1:				Letter 2:			
Data Bit (M)								
Encryption Key (K)	0	1	1	0	0	1	0	0
Encrypted Bit ($M \oplus K$)								

Table 2: Alice: Message Encryption

5. **Bob: Decryption ($M = C \oplus K$):** Bob receives the encrypted ciphertext bits C and computes bitwise XOR with key K to recover the message:

Received Bit (C)								
Encryption Key (K)	0	1	1	0	0	1	0	0
Data Bit ($C \oplus K$)								
Recovered Letter								

Table 3: Bob: Message Decryption

Questions:

- What is the 8-bit key used for encryption?
- What is your encrypted message?

4.2 Introduction to Polarization and Bases

Objective: Understand how polarization represents quantum states and how measurement in different bases affects results.

Instructions: For this activity we are going to be using the QKD kit. Please make sure to wear gloves for the activity, ask if not provided.

1. Set Alice lens to 0°.
2. Set Bob lens at:
   - 0°: Observe transmission.
   - 90°: Observe blockage.
   - 45°: Observe partial transmission.
3. Repeat, but now change Alice lens from 0° to 45°.

Basis	**Bit**	**Polarization**
+	0	0°
+	1	90°
×	0	-45°
×	1	45°

Table 4: Possible Combination of Basis

Questions:

- What do the transmission results tell you about the quantum measurement? What can you conclude?
- Why does partial light pass at 45°?

4.3 Simulating BB84 Protocol with Hardware

Objective: Perform physical key generation and One-Time Pad encryption using the optical kit.

Instructions:

1. Alice randomly sets her polarizer angles to send bits, and Bob randomly sets his analyzer to record the measurements:

Pulse	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**
Alice Basis (+/×)																		
Alice Bit																		
Bob Basis (+/×)																		
Bob Measured Bit																		
Match? (✓)																		

Table 5: Hardware Key Sifting Record

2. Identify the pulses where the bases matched. Take the first 8 matching bits to form your hardware key K.

3. Alice selects two letters, encrypts them via XOR ($M \oplus K$), and transmits the ciphertext bits through the optical channel using the + basis ($0° = 0$, $90° = 1$):

Letter	Letter 1:				Letter 2:			
Data Bit (M)								
Hardware Key (K)								
Encrypted Bit ($M \oplus K$)								

Table 6: Alice: Hardware Encryption

Transmitted Ciphertext (C)								
Received Ciphertext (C')								

Table 7: Ciphertext Optical Transmission (+ Basis)

4. Bob uses his copy of the hardware key K to decode the received ciphertext via XOR:

Received Bit (C')								
Hardware Key (K)								
Data Bit ($C' \oplus K$)								
Recovered Letter	Letter 1:				Letter 2:			

Table 8: Bob: Hardware Decryption

Questions:

- What is the final key generated by the hardware?
- How many total bits matched after base reconciliation?
- Did Bob successfully decode Alice's original letters?

4.4 Eavesdropping Experiment

Objective: Explore how an intercept-resend eavesdropper (Eve) introduces detectable errors into the shared key.

Instructions:

1. Alice, Eve, and Bob select their bases. Eve intercepts Alice's signal, measures it, and prepares a new state to send to Bob:

Pulse	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**
Alice Basis	×	×	+	×	+	+	+	×	×	+	×	×	×	+	+	×	+	×
Alice Bit	1	0	0	1	1	0	0	1	1	1	0	1	0	0	0	1	0	1
Eve Basis	×	×	+	+	×	+	+	+	×	×	×	×	+	×	+	×	+	+
Eve Bit																		
Bob Basis	+	×	×	×	+	+	+	×	+	×	×	+	+	×	+	+	+	×
Bob Bit																		

Table 9: Transmission with Eve Intercepting and Resending

Interception Rules:

- If Eve's basis matches Alice's, Eve measures Alice's bit accurately. If they differ, Eve measures a random bit (0 or 1).
- Bob measures Eve's re-transmitted state. If Bob's basis matches Eve's, Bob measures Eve's bit accurately. If they differ, Bob measures a random bit (0 or 1).

2. Alice and Bob publicly reconcile their bases. Only the **10 pulses** where Alice's basis matched Bob's basis are kept (Pulses 2, 4, 5, 6, 7, 8, 11, 15, 17, 18). Record and compare their bit values below:

Matched Pulse	**2**	**4**	**5**	**6**	**7**	**8**	**11**	**15**	**17**	**18**
Alice Bit	0	1	1	0	0	1	0	0	0	1
Bob Bit										
Error Detected? (×)										

Table 10: Sifted Key Comparison Showing Eve-Induced Errors

Because Eve measured in the incorrect basis approximately 50% of the time, she collapsed the quantum states and introduced an error rate close to the theoretical 25% into the sifted key.

Questions:

- If Bob measures a bit that matches Alice's despite Eve's interference, does that mean no eavesdropping happened? Explain.
- Why does Eve inevitably introduce errors into the communication when using non-orthogonal bases?
- What is the significance of the No-Cloning Theorem in preventing Eve from intercepting undetectable copies of quantum states?

A 4-Bit Character Encoding Reference

Use this table to convert letters to 4-bit binary values for encryption and decryption.

Letter	4-Bit Binary	Letter	4-Bit Binary
H	0111	**J**	1010
I	1000	**L**	1100

Table 11: Recommended Lab Letters

Letter	Decimal Value	Binary (4-bit)	Letter	Decimal Value	Binary (4-bit)
A	0	0000	I	8	1000
B	1	0001	J	9	1001
C	2	0010	K	10	1010
D	3	0011	L	11	1011
E	4	0100	M	12	1100
F	5	0101	N	13	1101
G	6	0110	O	14	1110
H	7	0111	P	15	1111

Table 12: Full 4-Bit Binary Alphabet

B Deliverables

Your task for this lab is to write a well-structured report that includes the following components:

- **Introduction**
  Provide a brief overview of the lab.
- **Methodology**
  Provide a step by step description of the tasks conducted in each activity.
- **Results**
  Report and reflect,the outcomes obtained from each activity and address the questions that are at the end of each one of the activities.
- **Supporting material**
  Include any relevant supporting materials such as Pictures of the experiments and/or math to show the proof that validate your results.
- **Conclusion**
  Summarize and reflect on the key findings from each of the activities. Mention any challenges you encountered while performing the experiments.

References

[1] Thorlabs, *Quantum Cryptography Demonstration Kit Manual*, December 2020, mTN005660-D02. [Online]. Available: https://www.thorlabs.com

[2] M. Sabani, I. Savvas, D. Poulakis, and G. Makris, "Quantum key distribution: Basic protocols and threats," in *Proceedings of the 26th Pan-Hellenic Conference on Informatics*, ser. PCI '22. New York, NY, USA: Association for Computing Machinery, 2023, p. 383–388. [Online]. Available: https://doi.org/10.1145/3575879.3576022

[3] M. Mehic, S. Rass, P. Fazio, and M. Voznak, *Quantum Key Distribution Networks: A Quality of Service Perspective*. Cham: Springer, 2022. [Online]. Available: https://doi.org/10.1007/978-3-031-06608-5

[4] C. H. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," *Theoretical Computer Science*, vol. 560, pp. 7–11, 2014, theoretical Aspects of Quantum Cryptography – celebrating 30 years of BB84. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0304397514004241

[5] P. W. Shor and J. Preskill, "Simple proof of security of the bb84 quantum key distribution protocol," *Physical review letters*, vol. 85, no. 2, p. 441, 2000.

[6] C. H. Bennett, "Quantum cryptography using any two nonorthogonal states," *Phys. Rev. Lett.*, vol. 68, pp. 3121–3124, May 1992. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.68.3121

[7] A. K. Ekert, "Quantum cryptography based on bell's theorem," *Phys. Rev. Lett.*, vol. 67, pp. 661–663, Aug 1991. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.67.661

[8] V. Scarani, A. Acin, G. Ribordy, and N. Gisin, "Quantum cryptography protocols robust against photon number splitting attacks¡? format?¿ for weak laser pulse implementations," *Physical review letters*, vol. 92, no. 5, p. 057901, 2004.

[9] P. Jouguet, S. Kunz-Jacques, A. Leverrier, P. Grangier, and E. Diamanti, "Experimental demonstration of long-distance continuous-variable quantum key distribution," *Nature photonics*, vol. 7, no. 5, pp. 378–381, 2013.

[10] N. Jain, H.-M. Chin, H. Mani, C. Lupo, D. S. Nikolic, A. Kordts, S. Pirandola, T. B. Pedersen, M. Kolb, B. Ömer *et al.*, "Practical continuous-variable quantum key distribution with composable security," *Nature communications*, vol. 13, no. 1, p. 4740, 2022.

[11] V. Scarani, H. Bechmann-Pasquinucci, N. J. Cerf, M. Dušek, N. Lütkenhaus, and M. Peev, "The security of practical quantum key distribution," *Reviews of modern physics*, vol. 81, no. 3, pp. 1301–1350, 2009.

[12] M. Mehic, M. Niemiec, S. Rass, J. Ma, M. Peev, A. Aguado, V. Martin, S. Schauer, A. Poppe, C. Pacher *et al.*, "Quantum key distribution: a networking perspective," *ACM Computing Surveys (CSUR)*, vol. 53, no. 5, pp. 1–41, 2020.

[13] D. Bacco, J. B. Christensen, M. A. U. Castaneda, Y. Ding, S. Forchhammer, K. Rottwitt, and L. K. Oxenløwe, "Two-dimensional distributed-phase-reference protocol for quantum key distribution," *Scientific reports*, vol. 6, no. 1, p. 36756, 2016.

[14] O. Amer, V. Garg, and W. O. Krawec, "An introduction to practical quantum key distribution," *IEEE Aerospace and Electronic Systems Magazine*, vol. 36, no. 3, pp. 30–55, 2021.

[15] V. Makarov, A. Anisimov, and J. Skaar, "Effects of detector efficiency mismatch on security of quantum cryptosystems," *Physical Review A—Atomic, Molecular, and Optical Physics*, vol. 74, no. 2, p. 022313, 2006.

[16] H. Shu, "Asymptotically optimal prepare-measure quantum key distribution protocol," *International Journal of Theoretical Physics*, vol. 62, no. 8, p. 191, 2023.

[17] C. H. Bennett, G. Brassard, and J.-M. Robert, "Privacy amplification by public discussion," *SIAM journal on Computing*, vol. 17, no. 2, pp. 210–229, 1988.

[18] R. Renner and R. König, "Universally composable privacy amplification against quantum adversaries," in *Theory of Cryptography Conference*. Springer, 2005, pp. 407–425.

[19] Thorlabs, Inc., "Quantum cryptography educational kit," https://www.thorlabs.com/newgrouppage9.cfm?objectgroup_id=9869, 2025, accessed: 2025-03-24.

Homework 1: Complete BB84 Protocol Simulation by Hand

Objective: Trace a full BB84 transmission, perform basis sifting, and encrypt a two-letter message using One-Time Pad XOR logic.

Reference Alphabet (4-bit binary):	**H** $\rightarrow 0111$	**I** $\rightarrow 1000$	**J** $\rightarrow 1010$	**L** $\rightarrow 1100$
XOR ($\oplus$) Rules:	$0 \oplus 0 = 0$	$1 \oplus 1 = 0$	$0 \oplus 1 = 1$	$1 \oplus 0 = 1$

Instructions:

1. Below is a sequence of 16 quantum pulses transmitted by Alice. Complete Bob's measured bits:
   - If Bob's basis **matches** Alice's basis, write Alice's bit deterministically.
   - If Bob's basis **differs**, write a random bit outcome (for grading consistency, assume Bob's random detector outcome is **0** whenever bases differ).

Pulse #	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Alice Basis	+	×	+	+	×	×	+	×	+	×	+	×	+	+	×	×
Alice Bit	1	0	0	1	1	0	1	1	0	1	1	0	0	1	0	1
Bob Basis	+	+	+	×	×	+	+	×	×	×	+	+	×	+	×	+
Bob Bit																
Match? (✓)																

Table 13: Pulse Transmission Table

2. In the **Match?** row, mark a checkmark (✓) on every column where Alice and Bob used the exact same basis.
3. List the indices of all matched pulses and write down the resulting sifted key K:
   - Matched Pulse Numbers: _____
   - Sifted Key (K): _____
4. Alice wants to transmit the word **"HI"** (**H** $= 0111$, **I** $= 1000$). Using the first 8 bits of your sifted key K, complete the encryption table:

Letter	Letter 1: **H**				Letter 2: **I**			
Plaintext Bit (M)	0	1	1	1	1	0	0	0
Sifted Key (K)								
Ciphertext ($C = M \oplus K$)								

Table 14: Alice: Encryption

5. Show that Bob decrypts the ciphertext back into "HI" by computing $M = C \oplus K$:

Received Ciphertext (C)								
Sifted Key (K)								
Recovered Bit (M)								
Recovered Letter								

Table 15: Bob: Decryption