~Graduation Thesis Presentation~

Qryptic Chat : a Simple Secure Chat Application based on Quantum Key Distribution(QKD)

Takihara Yudai, 4th year Bachelor student
The University of Aizu
Supervisor: Prof. Anh T. Pham

Feb. 17, 2025

Outline

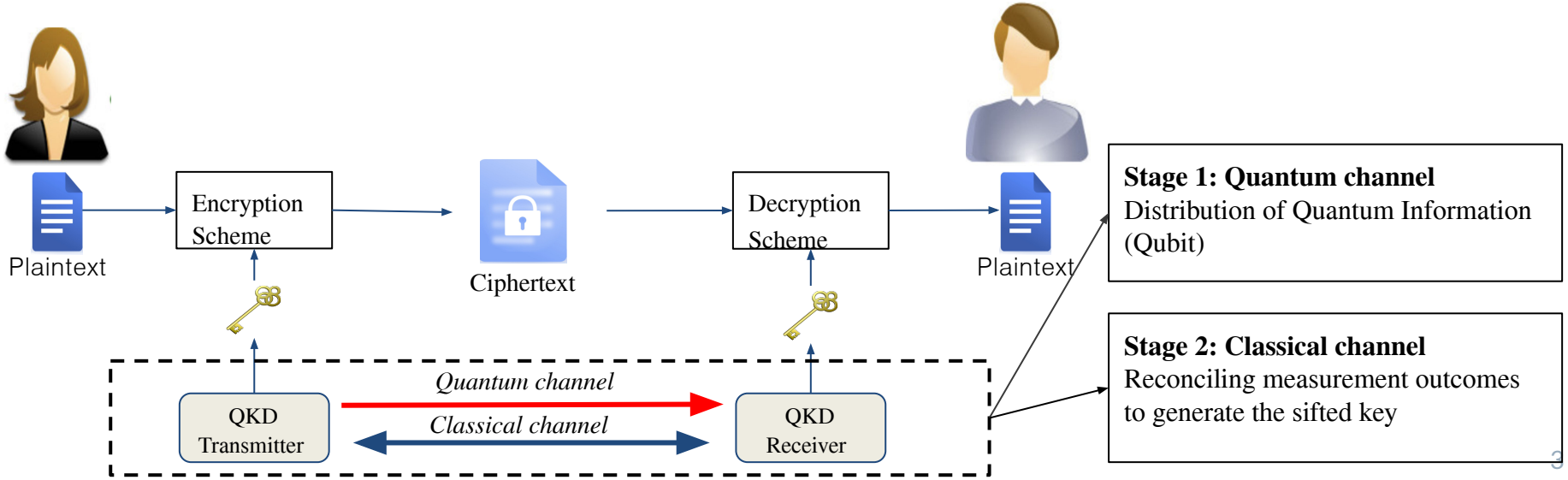
- **Research Background**
  - Quantum Key Distribution (QKD)
  - BB84 QKD Protocol
  - Why is QKD necessary?
  - My scope for the research

- **Simulation for QKD-based chat application**
  - Raw Key Rate
  - QBER
  - Final Key Rate

- **Conclusion**

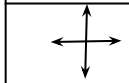
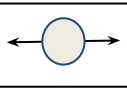
Quantum Key Distribution (QKD): Introduction

- QKD is a promising method to *distribute secure keys* secretly between legitimate users
  - It bases on the laws of quantum physics. → hard to attack for adversaries
  - First QKD protocol proposed by C. Bennett and G. Brassard in 1984, i.e., *BB84 Protocol*
  - Some of best-known Japanese companies have been working on various QKD projects, e.g., Toshiba, NEC, and NTT



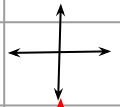
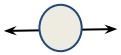
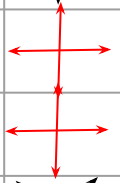
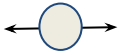
BB84 QKD Protocol

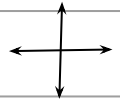
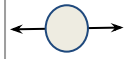
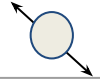
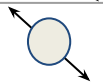
- BB84 uses photon polarization states to encode the bits of the key
- Each bit is encoded with a random polarization basis:  or 

Base \ Bit	0	1
		
		


Alice


Bob

Bit	Base	Qubit State
0		
0		
1		
0		

Base	Qubit State	outcome(bit)	Derived key
		0	0
		1	discard
		1	discard
		0	0

Why is QKD necessary?

- Problem of symmetric cryptography (The method that use secret key for encryption and decryption)
  - Key exchanging - How can share secret key between two legitimate users.

- Solution
  - **Asymmetric cryptography** (The method uses two types of keys : private key and public key)
    - Computational secure : guarantees the security with reasonable assumptions about an adversary's capabilities

But in the near future

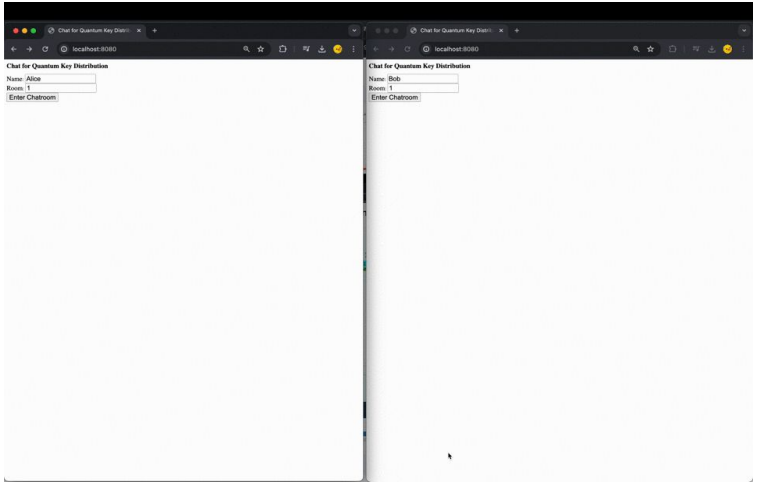
- Establishing Quantum computers (Computers with unusually fast processing speeds)
  - The algorithm of complex mathematical calculations (e.g., RSA) are easily deciphered.

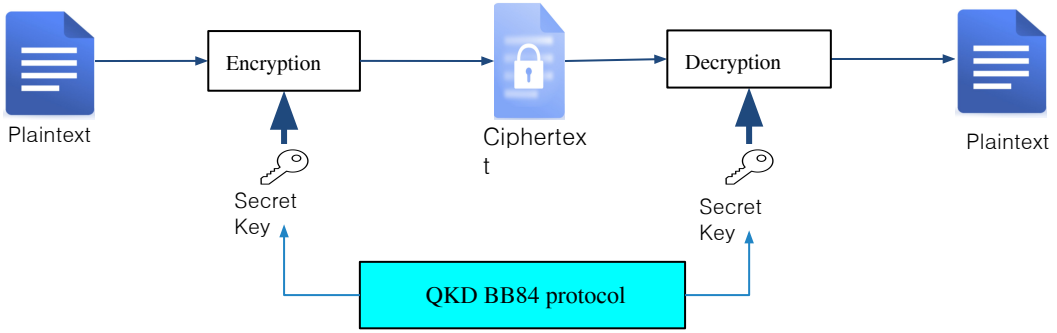
→ New key distribution method needed

My Scope for the research

My Scope: To develop and simulate the Qryptic Chat, a simple secure E2E Chat Application based on QKD

➢ ***The BB84 protocol*** can be applied for sharing secret keys between two legitimate users





Develop and simulate Secure Chat Application using IBM Quantum Experience(IQX)

- IQX: an open platform offered by IBM and available for quantum computing services
- ***Qiskit***, open source SDK for quantum computing and support to develop and simulate application
  - Generate Qubit (the basic unit information for quantum computing)

Outline

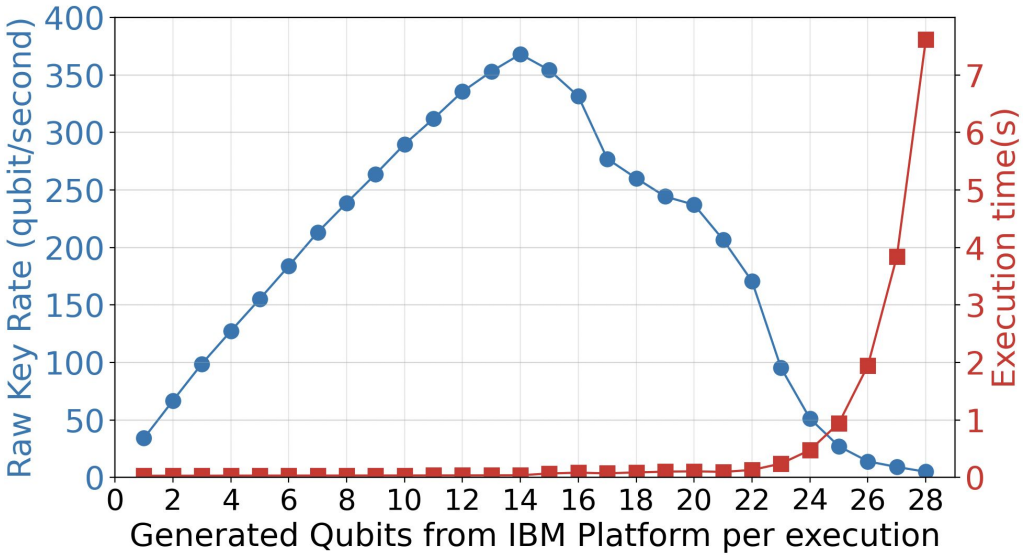
- **Research Background**
  - Quantum Key Distribution (QKD)
  - BB84 QKD Protocol
  - Why is QKD necessary?
  - My scope for the research

- **Simulation for QKD-based chat application**
  - Raw Key Rate
  - QBER
  - Final Key Rate

- **Conclusion**

Simulation QKD-based chat App: Raw key rate

Raw key rate: How many Qubits are provided per second from IQX.

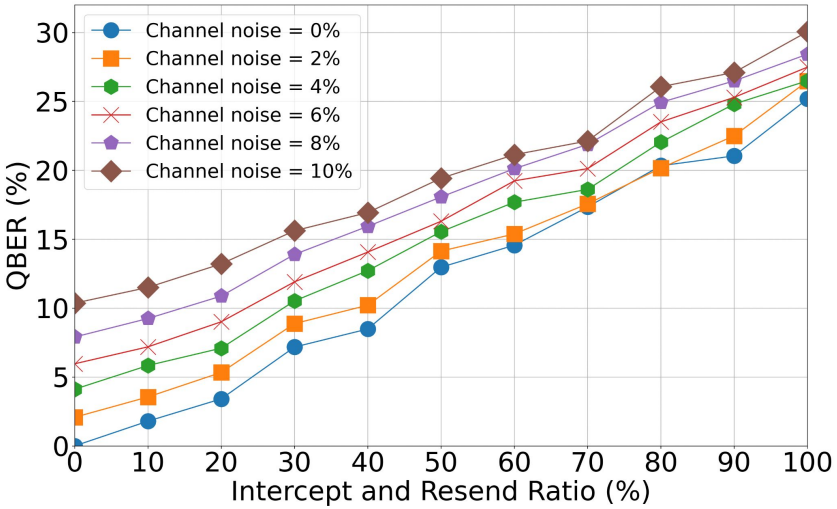


The highest raw key rate was found to be achieved with 14 qubits.

➢ Required key length can be generated in the shortest possible time.

Simulation QKD-based chat App: QBER

QBER: Calculated based on bit errors between Alice and Bob in the sifted key
>> Ratio of bit errors in sifted keys



- Increases with higher IRA ratios and channel noise frequencies, reflecting more bit errors in the sifted key between Alice and Bob.
- In an ideal BB84 QKD protocol without channel noise (0%), Alice and Bob measure a QBER of 25% under 100% eavesdropping. >> Same as theoretical rate.

Simulation QKD-based chat App: Final key rate

Final key rate [1]: $R = c \times s \times p[H(A|E) - fH(A|B)]$

- Raw key rate
- Sifting coefficient (= 0.5 in case of BB84)
- Parameter estimation coefficient (0.75 - 0.9)
- The amount of information that Eve is uncertain about Alice's key after key reconciliation (KR) step

Notations

- $[H(A|E)]$ denotes the amount of information that Eve is uncertain about Alice's key after the sifting step.
- $[H(A|B)]$ denotes the theoretical amount of information that Alice and Bob need to exchange for KR, which is also the information leaked to Eve during the KR step.
- f is the efficiency of the error correction algorithm.

Remarks

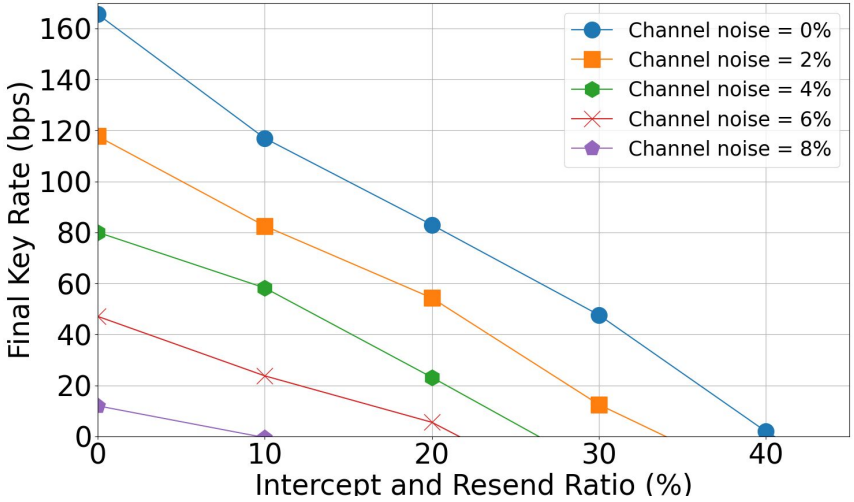
- In the case of BB84, $H(A|E) = 1 - h(QBER)$ and $H(A|B) = h(QBER)$, where h is the binary entropy function

[1]: G. Bebrov, "On the (relation between) efficiency and secret key rate of qkd," Scientific Reports, vol. 14, p. Article number: 3638, 2024

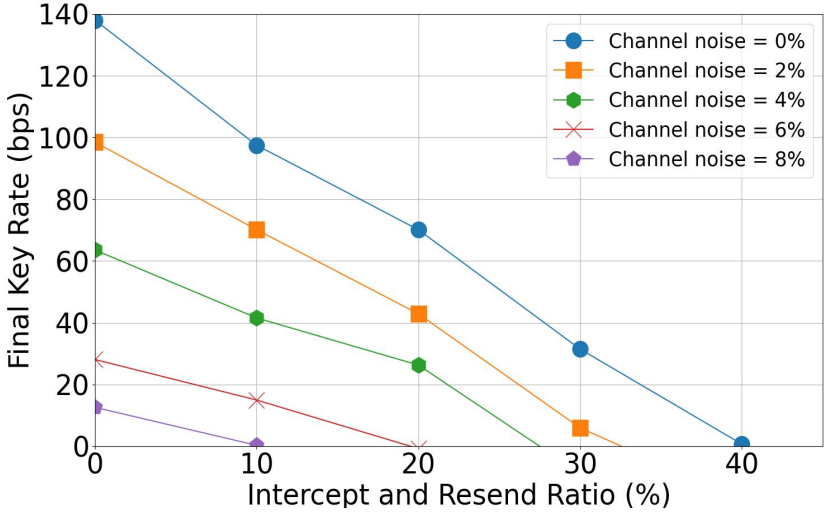
Simulation QKD-based chat App: Final key rate

Final key rate: $R = c \times s \times p[H(A|E) - fH(A|B)]$

Parameter	Value
c	368.0363931165903
p	0.9
s	0.5
f	1.22



Parameter	Value
c	368.0363931165903
p	0.75
s	0.5
f	1.22



The Final key rate falls as the Intercept and resend ratio and noise channel levels increase.

Outline

- **Research Background**
  - Quantum Key Distribution (QKD)
  - BB84 QKD Protocol
  - Why is QKD necessary?
  - My scope for the research

- **Simulation for QKD-based chat application**
  - Raw Key Rate
  - QBER
  - Final Key Rate

- **Conclusion**

Conclusion

- Development of the Qryptic Chat
  - BB84 protocol was applied for that application to communicate securely.
  - IBM Quantum Experience (IQX) support to develop and simulate QKD-based application.

- Remarkable observations from the simulation results
  - *Raw key rate*
    - Investigate the number of qubits to reach the required key length (e.g. 1000 bits) in the shortest possible time, using the number of qubits provided by IQX.
  - *QBER*
    - Reached approximately 25% at 0% channel noise and 100% IRA ratio, aligning closely with the expected theoretical value for BB84
  - *Final key rate*
    - Decreased with increasing channel noise and IRA ratio, demonstrating the impact of these factors on key generation efficiency.

- The results verify the reliability of Qryptic Chat as a secure communication system and emphasize its potential for further optimization and practical implementation in real-world QKD applications.

Thank you for your listening!