2026 Volume: 02 Issue: 11 www.ijrpa.com ISSN 2456-9995 Research Article

International Journal Research Publication Analysis

Page: 01-10

SECURE CHAT APPLICATION USING BB84 PROTOCOL

S. Sreshta*, B. Datta Sai, T. Ashwini, M. Srinija, P. Induvathi

Department of CSE (Artificial Intelligence & Data Science) GMRIT-Deemed to be University.

Article Received: 12 October 2026
Article Revised: 01 November 2026
Published on: 20 November 2026

***Corresponding Author: S. Sreshta**
Department of CSE (Artificial Intelligence & Data Science) GMRIT-Deemed to be University.

DOI: https://doi-doi.org/101555/ijrpa.3481

ABSTRACT

With the rapid progress of quantum computing, the public-key methods used in today's chat applications, such as RSA and Elliptic Curve Cryptography (ECC), may become unsafe, because Shor's algorithm can break them on a sufficiently powerful quantum computer. This project presents a web-based Post-Quantum Secure Chat Application (Chatify) that combines the BB84 Quantum Key Distribution (QKD) protocol with CRYSTALS-Kyber, a lattice-based key encapsulation mechanism. The BB84 key exchange is simulated in software: if an eavesdropper intercepts the simulated photons, the Quantum Bit Error Rate (QBER) rises above a configured threshold and the key exchange is aborted. Keys derived from the key exchange are used to encrypt messages with AES-256-GCM, and message integrity is checked with an HMAC. The application consists of a real-time web interface, a WebSocket server that relays only encrypted data, and a quantum simulation module. In the simulated attack shown in this paper, the measured QBER was 40.0% (6 of 15 sampled bits) against an abort threshold of 11.0%, and the key exchange was aborted; a tampered message was also rejected by the HMAC check. The system is a software prototype and is not a replacement for deployed quantum hardware or a formally audited security product.

KEYWORDS: Quantum computing, BB84 protocol, Quantum Key Distribution, post-quantum cryptography, CRYSTALS-Kyber, secure chat application, eavesdropping detection.

INTRODUCTION

Chat and messaging applications are now a central part of personal, professional and governmental communication, which makes the confidentiality, integrity and privacy of

transmitted messages very important. Most secure messaging platforms rely on classical public-key cryptography, mainly RSA and ECC. These algorithms are considered secure against classical computers, but their security rests on integer factorization and discrete logarithm problems, which Shor's algorithm can solve efficiently on a large quantum computer.

This creates the threat of "harvest now, decrypt later" (HNDL) attacks, in which an adversary stores encrypted traffic today in order to decrypt it when capable quantum hardware becomes available. Messages that must stay confidential for many years are therefore at risk even before such hardware exists. Developing communication systems that are prepared for this threat is an important area of research.

Two approaches address the problem. Quantum Key Distribution (QKD), of which BB84 is the earliest and best known protocol, encodes key bits into the polarization states of single photons. Because measuring a quantum state disturbs it, an eavesdropper introduces detectable errors, and the communicating parties can discard a compromised key. Post-Quantum Cryptography (PQC), such as the lattice-based CRYSTALS-Kyber key encapsulation mechanism, uses mathematical problems that are not known to be efficiently solvable by quantum computers and runs on ordinary hardware and networks.

The proposed project, Secure Chat Application Using BB84 Protocol, combines these ideas in a real-time web application named Chatify. Shared keys obtained from the BB84 simulation and from Kyber are used to encrypt messages end to end with AES-256-GCM before transmission. The central WebSocket server acts only as a relay for ciphertext and does not hold plaintext or private keys. The client interface also includes privacy features such as screenshot protection, chat-key verification and an attack simulation panel.

The system also has important limits. The BB84 protocol is simulated in software rather than run over optical hardware, so the results demonstrate the logic of eavesdropping detection and not the physical security of a real quantum channel. Any performance figures should come from actual measurements on the implemented system, and the application should be regarded as a research and demonstration prototype.

LITERATURE SURVEY

Several research studies have explored secure messaging, quantum key distribution and post-quantum cryptography. Existing secure messaging work includes real-time chat applications built with WebSockets and symmetric end-to-end encryption, multi-layer messaging schemes with forward secrecy, and privacy-preserving entry protocols for semi-open chat groups. A

systematic review of end-to-end encrypted messaging protocols also shows that major messaging systems are not yet fully prepared for quantum attacks. These works improve usability, performance or access control, but they still depend on classical cryptography.

Another line of research focuses on post-quantum cryptography. Benchmarking studies of algorithms such as CRYSTALS-Kyber and NTRU identify Kyber as a good candidate for low-latency key encapsulation. Hybrid schemes that pair a post-quantum key exchange with AES have been implemented for end-to-end encryption and wireless communication, and quantum-safe group key exchange and IoT security have also been studied. These approaches rely on computational hardness assumptions, so they cannot detect eavesdropping at the physical level, and some depend on dedicated hardware or are designed for constrained devices rather than chat applications.

A third line of research studies quantum key distribution. Researchers have described how BB84 and the uncertainty principle make interception detectable, proposed hybrid frameworks that combine QKD, post-quantum cryptography and quantum random number generators, and designed network-level protocols that combine classical, post-quantum and quantum key distribution. A chat prototype based on simulated quantum cryptography has also been built, but it does not include a standardized post-quantum key encapsulation mechanism. Overall, existing work either remains theoretical, works at network level, or does not combine QKD and post-quantum cryptography in a complete user-facing chat application.

The proposed Secure Chat Application Using BB84 Protocol aims to address some of the gaps left by this work by:

- Combining a simulated BB84 key exchange with CRYSTALS-Kyber inside one working chat application.
- Detecting simulated eavesdropping through the QBER and aborting the key exchange above a threshold.
- Encrypting messages end to end with AES-256-GCM through a relay server that does not see plaintext.
- Checking message integrity with an HMAC and rejecting tampered messages.
- Providing a web interface, an attack simulation panel and an administrator security log for demonstration and monitoring.

NOVELTY

- **Hybrid Quantum and Post-Quantum Design:** Integrates a simulated BB84 key distribution with CRYSTALS-Kyber key encapsulation in a single messaging application.
- **Eavesdropping Detection:** Estimates the QBER from a sacrificed sample of sifted bits and aborts the key exchange when it exceeds the configured threshold (11.0%).
- **Blind Relay End-to-End Encryption:** Messages are encrypted on the client with AES-256-GCM; the WebSocket server forwards ciphertext only.
- **Interactive Attack Simulation:** A built-in panel lets users run eavesdropping and message-modification scenarios and view the BB84 walkthrough step by step, which is useful for teaching and demonstration.
- **Application-Level Privacy and Monitoring:** Includes screenshot protection, chat-key viewing for verification, friend-request control, and an administrator dashboard with a security audit log.

WORKING

The Secure Chat Application is built as a web application with a real-time messaging interface, a backend server for user accounts and message delivery, and a quantum simulation module for key generation. The BB84 key exchange is simulated in software and combined with CRYSTALS-Kyber and AES-256-GCM to protect messages.

1. User Registration and Real-Time Messaging

Users register and log in through the web interface and connect with other users through friend requests. After a request is accepted, the two users can exchange text, voice and media messages in real time over WebSockets. Fig-1 shows two users chatting in the application.

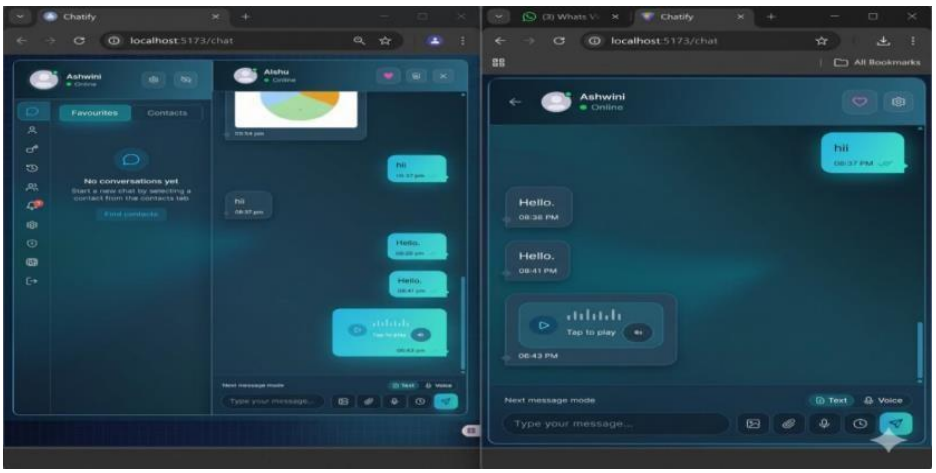


Fig-1: Real-time messaging and multimedia exchange interface of Chatify.

2. BB84 Quantum Key Distribution (Simulation)

BB84 is used to establish a shared secret key between two users, called Alice and Bob. Alice generates random bits and, for each bit, randomly chooses a rectilinear (+) or diagonal (×) basis to encode it as a photon polarization state. Bob measures each photon in a randomly chosen basis. Over a classical channel they then announce their bases, not their bits, and keep only the positions where the bases match; this step is called sifting. In the application, a run uses 64 simulated photons. The architecture is shown in Fig-2.

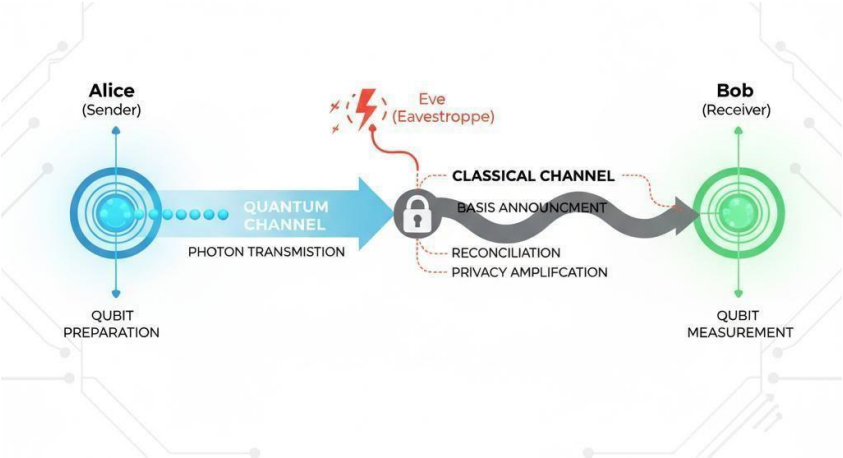


Fig-2: Architecture of the BB84 protocol.

3. Eavesdropping Detection Using QBER

A random sample of the sifted bits is sacrificed and compared to estimate the Quantum Bit Error Rate: QBER = (mismatched sampled bits / total sampled bits) × 100. If Eve intercepts and measures the photons in a wrong basis, she changes the polarization state and introduces errors that Bob can observe. If the QBER is above the threshold (11.0% in the application), the key is discarded and the session is aborted; otherwise the process continues. In the attack simulation shown in Fig-3 and Fig-4, 6 of 15 sampled bits were mismatched, giving a QBER of 40.0%, and the key exchange was aborted. This is a single simulated run with a small sample, so the value should not be read as a general error rate; it shows that interception raises the QBER well above the threshold.

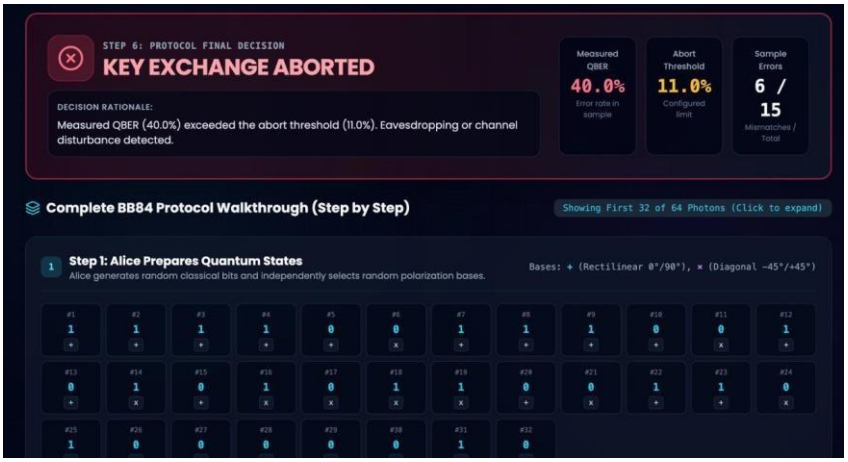


Fig-3: Key exchange aborted when the QBER crosses the threshold.



Fig-4: QBER calculation from the sacrificed sample.

4. Error Correction and Privacy Amplification

When the QBER is within the threshold, the remaining small differences between Alice's and Bob's sifted keys are removed using classical error correction. Privacy amplification then shortens the key to remove any information that may have leaked, producing the final shared secret key. The application reports when a 256-bit key has been established.

5. Post-Quantum Key Encapsulation Using CRYSTALS-Kyber

As a second layer, Bob generates a Kyber public and private key pair and shares the public key with Alice over the classical channel. Alice encapsulates a shared secret key using the public key and sends the resulting ciphertext through the server. Bob decapsulates it with his private key and recovers the same shared secret. The server sees only the public key and the ciphertext. The process is shown in Fig-5.

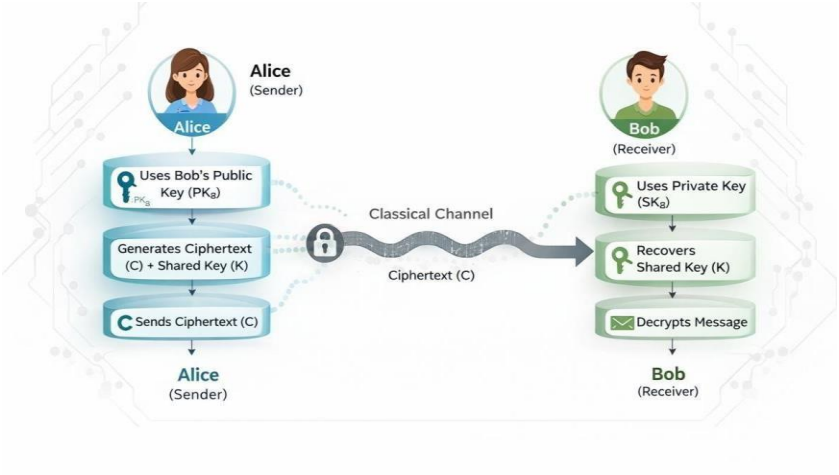


Fig-5: Architecture of the CRYSTALS-Kyber key encapsulation.

6. Message Encryption and Blind Relay

The keys obtained from the BB84 exchange and from Kyber are combined and used to encrypt each message with AES-256-GCM on the sender's device. The ciphertext is sent through the WebSocket server, which only relays it, and the receiver decrypts it with the shared key. The shared chat keys can be viewed in the application so that users can verify them.

7. Integrity Check and Tamper Simulation

An HMAC computed with the shared key is attached to each message. In the message-modification simulation in Fig-6, Eve changes the message in transit; when the receiver recomputes the HMAC, the digest does not match, and the forged message is discarded.

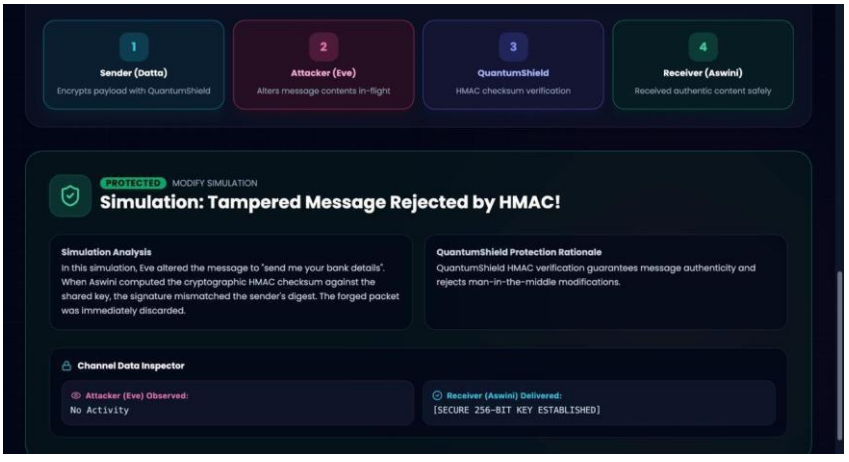


Fig-6: Tampered message rejected by the HMAC check.

8. Privacy Features and Administrator Monitoring

The client blocks screenshots in the chat window and shows a warning. An administrator page

lists registered users, banned accounts, feedback and a security audit log. In the demonstration deployment, the dashboard showed 21 registered users, 46 messages and 186 simulations run, with simulation events recorded in the audit log.

9. Testing and Performance Evaluation

The system was tested in a simulated quantum environment. Messages were exchanged between users, simulated interception raised the QBER above the threshold and aborted the key exchange, and tampered messages were rejected. Accuracy, precision and recall are classification metrics and do not describe a key-exchange protocol, so they are not used here. More suitable measurements are the QBER with and without an eavesdropper over many runs, final key length, key-exchange time and message delivery latency. Any numbers reported in the project should come from actual measurements of the implemented system.

FLOW CHART

Fig-7 shows the stages of the BB84 protocol used in the application, from preparation of the photons to use of the final key.

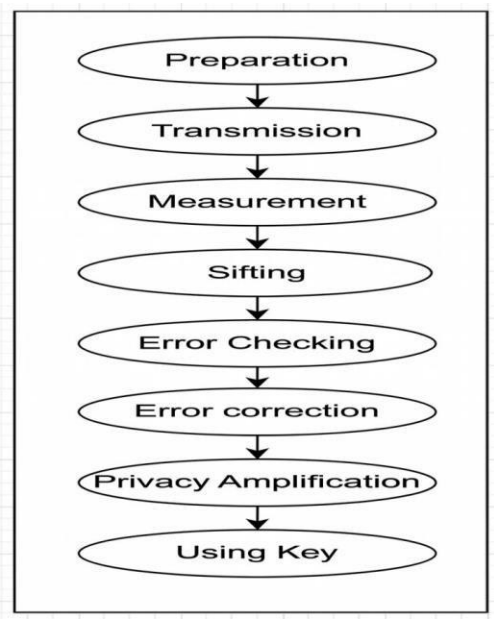


Fig-7: Flowchart of the BB84 protocol.

FUTURE SCOPE

The current system is a software prototype, and several improvements are possible. The most important is replacing the BB84 simulation with real quantum hardware, using optical fiber links and single-photon detectors, so that the physical security properties can be tested. The BB84 stage should also be studied under realistic noise, with the QBER threshold, sample size

and key length evaluated over many runs.

On the cryptography side, further NIST-standardized post-quantum algorithms can be added, and the key exchange can be extended to secure group chats and conference calls. The security of the implementation should be reviewed independently, including the handling of keys in the browser. Native Android and iOS applications would make the system easier to use.

Other directions include blockchain-based tamper-proof security logs, decentralized storage such as IPFS to avoid single points of failure, and machine-learning modules to detect anomalies, intrusion attempts and spam. Performance and scalability should be measured with many simultaneous users.

CONCLUSION

The Secure Chat Application Using BB84 Protocol demonstrates how quantum-based key exchange and post-quantum cryptography can be combined in a practical messaging application. The simulated BB84 protocol generates a shared key and detects eavesdropping through the QBER, CRYSTALS-Kyber provides a key exchange based on post-quantum assumptions, and AES-256-GCM with an HMAC protects message confidentiality and integrity through a relay server that sees only ciphertext.

In the demonstrated attack, the measured QBER of 40.0% exceeded the 11.0% threshold and the key exchange was aborted, and a tampered message was rejected. Because BB84 is simulated and the tests are limited, the results should be seen as a demonstration of the design and not as proof of security against real attackers. With real quantum hardware, broader testing and independent security review, the approach could contribute to communication systems that are better prepared for quantum-era threats.

REFERENCES

1. Davidson, L. Soezima and F. Virdia, “Practical semi-open chat groups for secure messaging applications,” in Proc. IEEE European Symposium on Security and Privacy (EuroS&P), IEEE, 2025, pp. 1–16.
2. S. Gore, S. Pagar, R. Arote, M. Shinde and P. Kanade, “Secure chat application using quantum cryptography simulation,” International Journal for Research Trends and Innovation (IJRTI), vol. 10, no. 11, pp. 198–202, 2025.
3. J. Kamalakumari, A. Kiran, G. Radha, Y. Chandini, M. Tiwari and V. Hemamalini, “Quantum cryptography protocols ensuring secure communication in the era of quantum computing,” ITM Web of Conferences, vol. 76, Art. no. 05009, 2025.

4. R. Abela, C. Colombo, P. Malo, P. Šys, T. Fabšič, O. Gallo, V. Hromada and M. Vella, “Secure implementation of a quantum-future GAKE protocol,” Lecture Notes in Computer Science, vol. 13075, pp. 103–121, Springer.
5. N. Krishnamoorthy, S. Subbaiah and J. Revathi, “Post-quantum cryptography: Securing future communication networks against quantum attacks,” Nanotechnology Perceptions, vol. 20, no. S14, pp. 264–278, 2025.
6. P. Bhatikare, P. Bansode, S. Gavade, A. Garkal, S. Karande and M. A. Pardesi, “Real time chat application,” International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE), vol. 13, no. 2, pp. 1111–1114, 2025.
7. Ojetunde, T. Kurihara, K. Yano, T. Sakano and H. Yokoyama, “A practical implementation of post-quantum cryptography for secure wireless communication,” Network, vol. 5, no. 2, Art. no. 20, 2025.
8. J. Sen, “Security and privacy management of IoT using quantum computing,” book chapter, 2025.
9. T. Manikumar, V. Kesavan, M. F. Antony, A. V. Raman and V. G. Ramsubramanyam, “Veilcomm: Next-generation secure messaging with custom encryption and key exchange,” in Proc. ICRDICCT 2025, pp. 447–460.