Karishma Dobhal , 2026, 14:4
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

A Hybrid RSA and ML-KEM Framework for Secure and Quantum-Resilient Communication

Karishma Dobhal, Y.P. Raiwani
Hemvati Nandan Bahuguna Garhwal University (A Central University), Srinagar Garhwal-246174, Uttarakhand, India

Abstract- Classical public-key cryptography, mainly RSA, is at risk from the emergence of quantum computers. A complete shift to Post-Quantum Cryptography (PQC) faces vulnerabilities due to implementation and trust issues, despite NIST's approval of standards such as ML-KEM (Kyber). This paper presented a performance and bandwidth trade-off study of a proposed Hybrid Key Encapsulation Mechanism that fused the NIST-standardized ML-KEM-768 (FIPS 203) with the traditional RSA-3072. Two separate secret keys S1 and S2 were produced and encapsulated utilizing both classical and post-quantum public keys in our dual-layered model. In order to create a strong session key that ensures trust transition and backward compatibility with current network infrastructures, these secrets were then combined using a Hybrid Key Derivation Function (HKDF). In comparison to standalone systems, the study assessed the computational cost and communication latency brought about by this hybrid overhead. According to experimental data, the hybrid architecture offers a crucial security fail-safe against both classical flaws and future quantum threats without prohibitively high performance, even though it increases ciphertext size.

Keywords: Post-Quantum Cryptography (PQC), ML-KEM, RSA.

I. INTRODUCTION

For many years classical cryptography including RSA, has protected our digital communications. It relies on very hard-to-solve mathematical problems. Although the threat of Shor's algorithm [1] is theoretical, quantum computers are advancing rapidly with qubit counts increasing year by year. A computer like that might compromise government communications, financial systems, and the privacy of historical data, a situation that is sometimes referred to as the Years To Quantum (Y2Q). The need for Post-Quantum Cryptography (PQC) is urgent due to the future risk. Currently, attackers are silently storing encrypted data and waiting to decrypt it later when quantum computers become available. This process is called 'Harvest Now, Decrypt Later' (HNDL) [8]. It is currently necessary to secure data for a long time using quantum-resistant cryptographic algorithms.

To address this problem, the National Institute of Standards and Technology (NIST) has finalized the first set of PQC standards [20], including ML-KEM (Module Lattice-based Key Encapsulation Mechanism) [3,6] and ML-DSA (Module Lattice-based Digital Signature) [4,7]. Lattice-based cryptography algorithms are built on hard mathematical problem over lattices [9,10]. PQC protects us from future quantum attacks but it has not been tested in real-world use as much as RSA. Switching to the direct PQC domains is not reliable as there are various causes, such as networking, real-world testing, side-channel vulnerabilities, and compatibility issues. The basic principle of a hybrid system [12] is Dual Security [12,13].

Mathematically, there are two layers. If one layer fails, the other still protects our data. To address these challenges, this paper proposed a hybrid Cryptographic framework. In this scheme, we designed a Dual-key encapsulation system. This system combined RSA and ML-KEM and presented how to integrate it into the existing model. Existing studies focus on performance benchmarking, and there is limited work on RSA and ML-KEM that prioritizes backward compatibility and long-term data integrity. This paper addressed security and compatibility, ensuring organizations don't have to replace their existing RSA systems to become quantum-safe.

The rest of the paper is organized as follows: Section 2 defines related work, Section 3 presents our proposed framework, Section 4 discusses results, and Section 5 presents the conclusion. Section 6 define declaration part.

© July 2026 Karishma Dobhal, This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly credited.

II. RELATED WORK

The transition to quantum cryptography has become one of the important domains of cryptographic research for PQC and hybrid cryptographic systems. Shor [1] was the first to mathematically prove that theoretically a quantum computer can break the RSA algorithm. The author solved factorization in polynomial time with O (n3) operations. Since then, various researcher aimed on calculating the number of physical qubits required to break RSA-2048. Fowler et al. [17] introduced an architecture using surface code error correction and estimates approximately one billion qubits would be required to break RSA-2048. Subsequently, Gidney et al. [2] presented that 20 million noisy qubits would be enough to break RSA-2048 that could be broken in just 8 hours using a surface-code quantum computer.

Regarding a good security balance and key sizes, NIST chose ML-KEM (Kyber) for the main key encapsulation mechanism [3,6]. Liu et al. [5] presented a survey on lattice-based digital signature methods. Analyzed different security methods such as hash-and-sign and Fiat-Shamir.The Open Quantum Safe (OQS) project [19] released liboqs for OpenSSL hybrid key exchange prototyping. Their main focus was performance benchmarks. Stebila et al. [11] introduced alterations to the TLS 1.3 [15] handshake to provide two set of keys.

Recent research focuses on Elliptic Curve (ECDH) hybrids rather than the RSA-KEM methods. In parallel, Google used Chrome to test the CECPQ1 (X25519 + Kyber) hybrid in a real-world setting [18], their findings demonstrated that when latency increases slightly the connection stability stays strong. Most previous literature [14] focuses mainly on PQC algorithms performance (speed) or latency (Time). Various researchers have found the concept of Trust Transition and shown how different organizations can keep using their RSA systems while adding new quantum protection. Our work presented a modular architectural logic that uses a Dual-Key approach to give priority to backward compatibility and long-term data integrity.

III. PROPOSED METHODOLOGY

Our proposed hybrid framework combined RSA and ML-KEM (Kyber) [3]. This framework ensured backward compatibility with quantum resistance. Fig 1. illustrates a hybrid key exchange process that combines classical (RSA) and post quantum cryptography (KEM). This architecture is categorized into 3 basic modules.

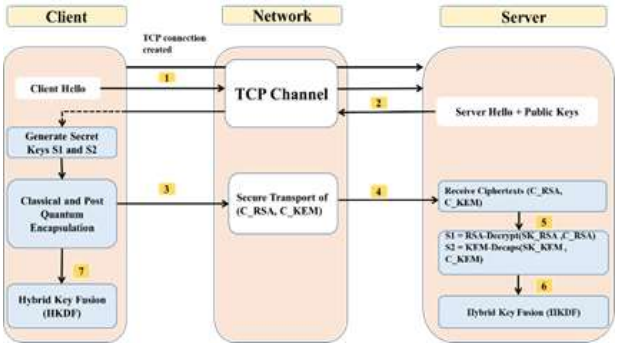

Fig. 1. Hybrid Key Encapsulation Mechanism Implementation and Security Analysis

This architecture uses two keys: one for classical algorithms and the other for post-quantum algorithms. Both algorithms offered a comparable 128-bit security level, RSA-3072 and ML-KEM-768 were chosen to ensure that neither component constituted the weaker link in the hybrid structure. The architecture uses a layered security model, where both keys are combined into a session key. The dual-key process further breaks down into the following phases:

Key Generation: The receiver used their respective standard key generation methods to create two independent key pairs prior to the encapsulation process- a post-quantum ML-KEM-768 key pair (SKKEM, PKKEM) and a conventional RSA-3072 key pair (SKRSA, PKRSA). During the initial handshake (Server Hello), the sender received both public keys, PKRSA and PKKEM, while the recipient safely stored the corresponding private keys for the decapsulation stage.

Classical Key Encapsulation: Using the receiver's RSA-3072 public key PKRSA to encrypt a randomly generated secret S1.

$$C_{RSA} = \text{RSA-Enc}(S1, PK_{RSA}) \quad (1)$$

Quantum-resistant Key Encapsulation: Using the receiver's ML-KEM-768 public key (PKKEM) to encapsulate a secret S2

$$(C_{KEM}, S_2) = \text{ML-KEM-Encaps}(PK_{KEM}) \quad (2)$$

Hybrid Key Fusion: Integrating both keys S1 and S2 using a Hash-based Key Derivation Function (HKDF) [16].

$$K_{Session} = \text{HKDF-SHA256}(\text{Salt}, S_1 \| S_2, \text{Context}) \quad (3)$$

Where S1, S2 ∈ {0, 1}256 are independent secrets and Salt is a random 256-bit value per session. By ensuring that the session key's security depends on both classical and post-quantum components, the hybrid architecture increases resistance against single-point cryptographic failures.

We implemented the hybrid architecture using Python 3.10. For the implementation and testing phase, we utilized a Windows 11 Pro environment, running on hardware equipped with an Intel Core i5 processor and 8GB of RAM. Jupyter Notebook was used for implementation and testing, which has been quite helpful for prototyping and data visualization. For this framework, we chose RSA-3072 from the PyCryptodome library to handle classical security, ensuring our encryption meets current standards. We integrated ML-KEM-768 (Kyber) using the liboqs Python wrapper to create a quantum-resistant layer to defend against future threats and developed our 'Key Fusion' logic using the hashlib package to bring everything together. This created a single secure session key by combining secrets from both classical and quantum sources using the HKDF-SHA256 algorithm.

IV. ANALYSIS AND RESULTS

We have designed this system in such a way that it can handle the extra data from hybrid cryptography. By integrating RSA-3072 and ML-KEM-768, the total ciphertext comes to 1472 bytes. It is suitable choice to stay under the standard 1500-byte Ethernet limit (MTU). The main advantage of the hybrid framework is that, the entire hybrid handshake is transmitted within a single TCP segment, preventing data fragmentation, and ensures low network latency without compromising the security levels established in the earlier sections.

Table 1: Key and Ciphertext Sizes

Component	RSA-3072	ML-KEM-768	Hybrid
Public Key Size	624bytes	1184 bytes	1568 bytes
Ciphertext Size	384 bytes	1088 bytes	1472 bytes
Shared Secret Size	32 bytes	32 bytes	32 bytes
Security Level	128-bit	128-bit	128-bit

Table 1 shows that the hybrid approach maintains manageable key and ciphertext sizes while ensuring dual-layer security.

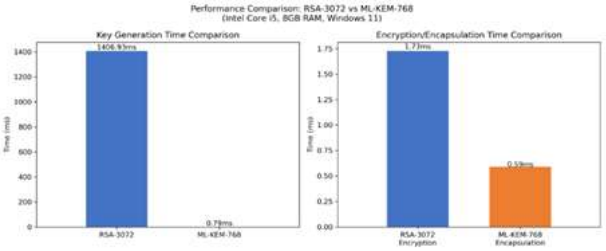

Figure 2: Comparison of performance

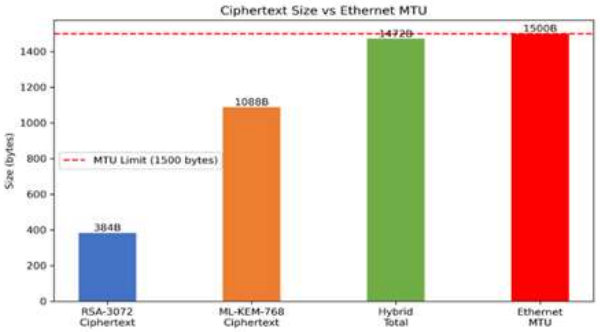

Figure 3: Ciphertext and Ethernet

However, Figure 2 shows that for RSA-3072 key generation took significantly more time than ML-KEM-768 because RSA-3072 had to rely on the complicated task of finding large prime numbers, while ML-KEM-768 used a simple method of lattice sampling. Moreover, the encapsulation time taken

by ML-KEM-768 (0.59 ms) was almost three times lower than the encryption/encapsulation time of RSA-3072 (1.73 ms). Although the ciphertext generated by ML-KEM-768 was larger in size, this experiment showed that ML-KEM-768 performed better than RSA-3072 in both of these processes (Figure 3). The extra layer of security was not an obstacle for networking because the size of the hybrid ciphertext was still within the 1500-byte limit set by the Ethernet MTU.

V. CONCLUSION

In this paper, we present the result of hybrid cryptography in a real network without compromising the performance. A combination of classical and quantum algorithms is not only theoretical but also used in the real world within 1472 bytes. This framework also ensures practical deployment feasibility. While the majority of current work is focused on speed and performance, our work focuses on how to help organizations to become quantum-safe without replacing their RSA systems. However, this framework was tested only through Python simulation. In the future, this framework can be tested on real hardware and integrated into TLS 1.3. testing on IoT and embedded devices would also be an interesting direction.

REFERENCES

1. P. W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," Proc. 35th Annual Symp. on Foundations of Computer Science, 1994.
2. C. Gidney and M. Ekera, "How to factor 2048-bit RSA integers in 8 hours using 20 million noisy qubits," Quantum, vol. 5, 2021.
3. NIST FIPS 203, Module-Lattice-Based Key-Encapsulation Mechanism Standard, 13 August, 2024.
4. NIST FIPS 204, Module-Lattice-Based Digital Signature Standard.13 August, 2024.
5. Liu, F., Zheng, Z., Gong, Z. et al. A survey on lattice-based digital signature. Cybersecurity 7, https://doi.org/10.1186/s42400-023-00198-1,Springer, 2024.
6. Bos, J. et al., "CRYSTALS-Kyber: a CCA-secure module-lattice-based KEM." In 2018 IEEE European Symposium on Security and Privacy (EuroS&P) (pp. 353-367), IEEE 2018.
7. Ducas, L. et al., "CRYSTALS-Dilithium: A lattice-based digital signature scheme." IACR Transactions on Cryptographic Hardware and Embedded Systems, 238-268, 2018.
8. M. Mosca, "Cybersecurity in an era with quantum computers: will we be ready?" IEEE Security & Privacy, vol. 16, no. 5, pp. 38-41, 2018.
9. O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," Journal of the ACM, vol. 56, no. 6, pp. 1-40, 2009.
10. V. Lyubashevsky, C. Peikert, and O. Regev, "On ideal lattices and learning with errors over rings," Journal of the ACM, vol. 60, no. 6, pp. 1-35, 2013.
11. P. Schwabe, D. Stebila, and T. Wiggers, "Post-quantum TLS without handshake signatures," in Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS), 2020, pp. 1461–1480.
12. N. Bindel, et al., "Hybrid key encapsulation mechanisms and authenticated key exchange," in Proc. Int. Conf. Post-Quantum Cryptography (PQCrypto), 2019, pp. 206–226.
13. F. Giacon, F. Heuer, and B. Poettering, "KEM combiners," Proc. Public Key Cryptography (PKC), 2018, pp. 190-218.
14. Paquin, C., Stebila, D., Tamvada, G. (2020). Benchmarking Post-quantum Cryptography in TLS. In: Ding, J., Tillich, JP. (eds) Post-Quantum Cryptography. PQCrypto 2020. Lecture Notes in Computer Science (), vol 12100. Springer, Cham. https://doi.org/10.1007/978-3-030-44223-1_5.
15. E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.3," RFC 8446, 2018.
16. H. Krawczyk and P. Eronen, "HMAC-based Extract-and-Expand Key Derivation Function (HKDF)," RFC 5869, 2010.
17. A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, "Surface codes: Towards practical large-scale quantum computation," Physical Review A, vol. 86, no. 3, p. 032324, 2012.
18. D. O'Brien, D. Adrian, D. Benjamin, and B. Beck, "Protecting Chrome Traffic with Hybrid Kyber KEM," Chromium Blog, Google, Aug. 2023.
19. Open Quantum Safe Project, "liboqs: C library for quantum-resistant cryptographic algorithms,"

2025. [Online]. Available: https://openquantumsafe.org/liboqs/. Accessed: Oct. 6, 2025.

20. G. Alagic et al.,"Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process,"NISTIR 8413, July 2022. DOI: 10.6028/NIST.IR.8413.