Report: Performance Analysis of Kyber for Secure Communication

Shek Lun Leung

July 9, 2025

Contents

1 Executive Summary

This report details the implementation and performance testing of a secure communication prototype. The prototype leverages the **Kyber** post-quantum key encapsulation mechanism (KEM) to establish a shared secret, which is then used by the **Advanced Encryption Standard (AES)** for encrypting messages.

Simulations were conducted for the three NIST-standardized Kyber security levels: **ML-KEM-512, ML-KEM-768, and ML-KEM-1024**. The key findings confirm a direct and measurable trade-off between security, performance, and data size. As the security level increases, the computational time for key generation and exchange rises, and the size of public keys and ciphertexts grows significantly. This analysis validates the effectiveness of hybrid cryptosystems and provides a clear framework for selecting the appropriate Kyber variant based on specific application requirements.

2 Secure Chat Prototype Architecture

The prototype simulates a secure channel between two parties, "Alice" and "Bob," using a hybrid encryption model. This model combines the strengths of asymmetric and symmetric cryptography.

2.1 Workflow Description

The communication flow is based on the Key Encapsulation Mechanism (KEM) model:

1. **Key Generation (Bob):** Bob, the intended recipient of the secret, generates a public/private Kyber key pair (`ek`, `dk`). The public key (`ek`) can be shared openly.
2. **Encapsulation (Alice):** Alice, wanting to establish a secure channel with Bob, uses his public key (`ek`) to generate two items:
   - A 32-byte shared secret (`key`).
   - An encapsulation ciphertext (`ct`) that securely wraps the secret.
3. **Transmission:** Alice sends the ciphertext (`ct`) to Bob over an insecure public channel.
4. **Decapsulation (Bob):** Bob uses his private key (`dk`) to decapsulate the received ciphertext (`ct`), which yields the exact same 32-byte shared secret (`key`).
5. **Secure Messaging:** With both parties now possessing an identical secret key, they use it to encrypt and decrypt messages using a fast symmetric cipher (AES-GCM).

2.2 Core Technologies Used

Kyber (ML-KEM via `kyber-py`) The cornerstone of the key exchange. Kyber is a post-quantum cryptographic algorithm designed to be secure against attacks from both classical and quantum computers. Its role is not to encrypt user data directly, but to perform the crucial task of establishing a shared secret between two parties who have no prior trust. It is an **asymmetric** algorithm.

AES-GCM (via `cryptography` library) The Advanced Encryption Standard (AES) is the globally trusted standard for **symmetric** encryption. The GCM (Galois/Counter Mode) variant was used, which provides both **confidentiality** (preventing eavesdropping) and **authenticity/integrity** (ensuring the message was not tampered with). AES is extremely fast, making it ideal for encrypting the actual message content (bulk data).

3 Key Findings and Performance Comparison

The simulation yielded clear data on the performance and size characteristics of each Kyber variant.

3.1 Size Comparison: Keys and Ciphertext

Higher security levels in Kyber are achieved by using larger mathematical structures, which results in larger keys and ciphertexts. This has a direct impact on the network bandwidth and storage required for the key exchange.

Table 1: Kyber Variant Size Comparison

Parameter	ML-KEM-512	ML-KEM-768	ML-KEM-1024	Growth (512 to 1024)
Public Key (ek)	800 bytes	1184 bytes	1568 bytes	**+96%**
Private Key (dk)	1632 bytes	2400 bytes	3168 bytes	**+94%**
Ciphertext (ct)	768 bytes	1088 bytes	1568 bytes	**+104%**
Shared Key Size	32 bytes	32 bytes	32 bytes	**0%**

Finding: Moving from the lowest to the highest security level nearly **doubles** the amount of data that must be transmitted for the key exchange (public key and ciphertext). The resulting shared secret key, however, remains a constant 32 bytes (256 bits).

3.2 Performance Comparison: Execution Time

The increased complexity of higher security levels translates directly to higher computational cost.

Table 2: Kyber Variant Performance Comparison (Execution Time)

Operation (ms)	ML-KEM-512	ML-KEM-768	ML-KEM-1024	Growth (512 to 1024)
Key Generation	2.90 ms	4.37 ms	6.96 ms	**+140%**
Encapsulation	3.91 ms	6.33 ms	8.52 ms	**+118%**
Decapsulation	5.58 ms	8.37 ms	11.37 ms	**+104%**

Finding: The time required for all cryptographic operations more than **doubles** when moving from ML-KEM-512 to ML-KEM-1024. While these times are small for a single operation, they can become significant for servers handling many thousands of connections per second.

3.3 Security-Performance Trade-offs

The results clearly demonstrate a fundamental principle of applied cryptography: **Greater security assurance comes at the cost of increased computational resources and data overhead.**

- **ML-KEM-512 (Level 1):** Offers the fastest performance and smallest footprint. Ideal for environments where performance is critical and a 128-bit post-quantum security level is sufficient.
- **ML-KEM-768 (Level 3):** Represents a balanced "sweet spot." It provides a significant security boost over Level 1 for a moderate increase in cost. It is often recommended as a default choice.
- **ML-KEM-1024 (Level 5):** Provides the highest security guarantee against future attacks. It is best suited for applications protecting highly sensitive, long-term data where the performance and size overhead is an acceptable price for maximum security.

4 Benefits of Combining Post-Quantum and Symmetric Encryption

The prototype's architecture, which uses Kyber for key exchange and AES for data encryption, is known as a **hybrid cryptosystem** (specifically, a KEM-DEM model). This approach is the industry standard for a crucial reason: it leverages the best qualities of both cryptographic families.

Asymmetric Cryptography (Kyber) Its magic is the ability to create a shared secret over an open, insecure channel. However, it is computationally intensive and slow. It is not designed or suited for encrypting large amounts of data.

Symmetric Cryptography (AES) It is incredibly fast and efficient, capable of encrypting gigabytes of data in seconds. Its one limitation is that it requires both parties to already have the same secret key.

By combining them, we get the best of both worlds: *We use the slow, powerful tool (Kyber) for the one-time task of establishing the key, and then use the fast, efficient tool (AES) for the ongoing task of encrypting the actual communication.* This provides quantum-resistant key establishment with high-performance data protection.