Post-Quantum Cryptography: Benchmarking ML-KEM Against RSA

Authors: Dominika Pacek
Nina Łabęcka

The rise of quantum computing presents a significant threat to classical cryptographic systems, particularly those relying on hard mathematical problems such as integer factorization and discrete logarithms. We explore the impact of quantum computing on traditional cryptographic algorithms and the necessity of transitioning to quantum-resistant cryptography. We provide an overview of post-quantum cryptographic (PQC) algorithms newly standardized by the National Institute of Standards and Technology (NIST) in the Federal Information Processing Standards (FIPS) 203, 204, and 205, which define quantum-secure key encapsulation and digital signature schemes. We examine existing implementations of these algorithms and evaluate the performance of ML-KEM against its classical alternative -- RSA.

Introduction

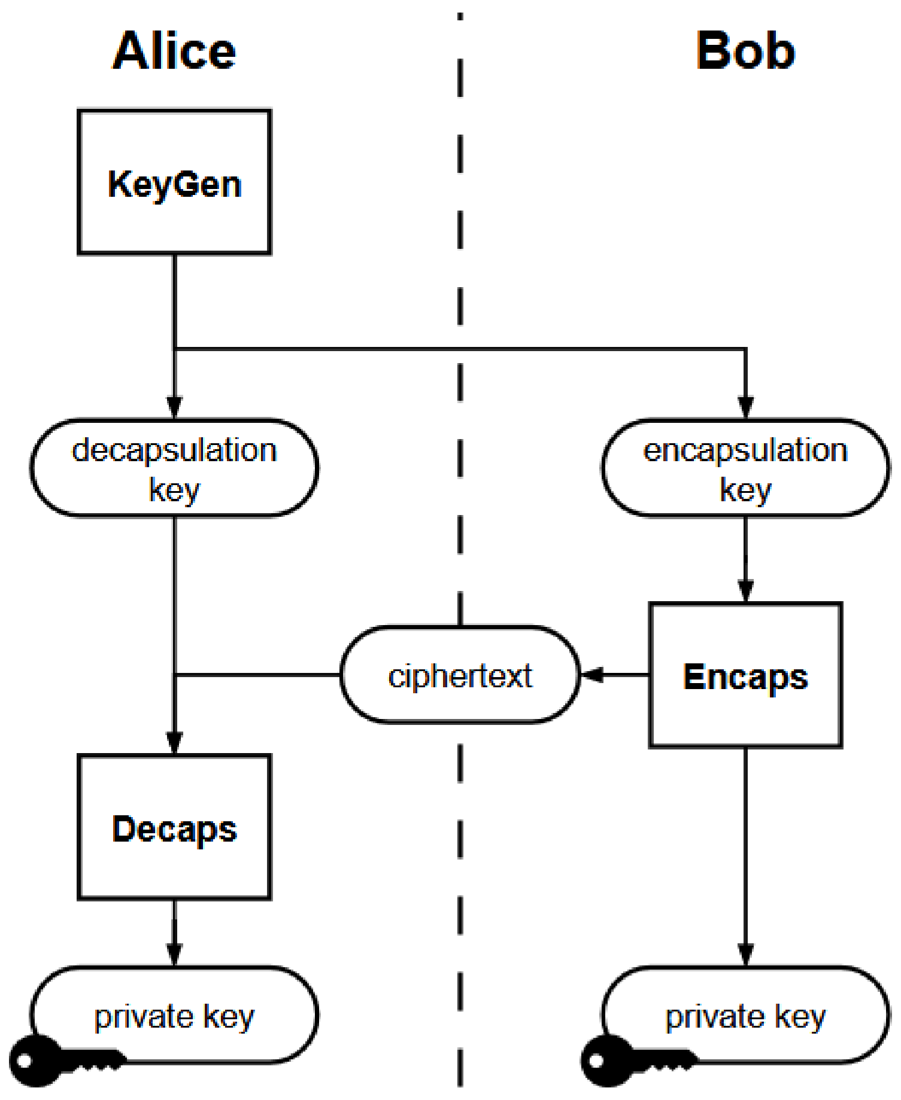
Cryptographic techniques form the foundation of secure communication in modern digital systems. They are essential for protecting data confidentiality, especially concerning messages in transit. At its core, cryptography relies on mathematical problems that are computationally difficult to solve, making it infeasible for an attacker to break the system within a reasonable timeframe. For example, the security of the RSA algorithm is based on the difficulty of factoring large composite numbers.
Quantum computers have the potential to solve problems that were previously intractable for classical computers, posing a significant threat to existing cryptographic methods. A prime example of this risk is Shor's algorithm, which enables integer factorization in polynomial time, faster than any known classical method, rendering traditional schemes like RSA vulnerable once sufficiently advanced quantum computers are developed.
Post-quantum cryptography (PQC) is a field of cryptographic research focused on developing algorithms that will remain secure against attacks from both classical and quantum computers. Unlike traditional public-key cryptosystems, which rely on number-theoretic problems vulnerable to quantum algorithms such as Shor's, post-quantum schemes are based on problems believed to be resistant to quantum computation, including lattice-based, code-based, multivariate, and isogeny-based approaches.

Post-quantum cryptography standardization

The American National Institute of Standards and Technology (NIST) started the move towards PQC standardization in 2016 by publicly asking cryptographic researchers to propose quantum-resistant public-key cryptographic algorithms.
In August 2024 NIST officially released three finalized PQC standards: Federal Information Processing Standards (FIPS) 203, 204 and 205. They are based on three best algorithms selected from those submitted and provide implementation instructions, pseudo-code and the intended uses for each of them. We decided to conduct benchmarks of ML-KEM, specified in FIPS 203.

Module-Lattice-Based Key-Encapsulation Mechanism

ML-KEM belongs to the group of key-encapsulation mechanism (KEM) algorithms, which allow two parties communicating over a public channel to establish a shared secret key. ML-KEM key exchange begins with Alice generating a public encapsulation key and a private decapsulation key. She can then share the encapsulation key with Bob, who uses it to generate a secret key, and a ciphertext which he sends back to Alice. Lastly, Alice uses the ciphertext and decapsulation key to generate her own copy of the secret key.



Methodology

This study evaluates the performance of ML-KEM against the commonly used classical cryptographic algorithm RSA. Benchmarks were conducted on a Linux (WLS/Ubuntu) system using Open Quantum Safe (liboqs) and OpenSSL with OQS integration. The evaluation focused on key generation, encapsulation, decapsulation, key sizes, and computational overhead, measured via speed_kem, openssl speed, and system profiling tools. All tests were performed on a Lenovo L14 laptop with the following specifications:

- CPU: AMD Ryzen Pro 7735U
- RAM: 64GB DDR5 RAM
- System: WLS (Ubuntu 24.04)
- Software: OpenSSL 3.0.13, Liboqs 0.12.0

We conducted performance benchmarks for ML-KEM and RSA using the openssl speed command for RSA and the speed_kem tool from the OQS library for ML-KEM. These tools measure the number of operations per second by repeated execution within a fixed time window. This approach reduces variability due to system overhead and provides a reliable estimate of algorithm efficiency.
We aligned the operations based on their functional roles in each cryptographic scheme, comparing encapsulation and decapsulation in ML-KEM to signing and verification in RSA, respectively.

Comparisons have been done between the three standardized parameter sets of ML-KEM (ML-KEM-512, ML-KEM-768 and ML-KEM-1024) and three popular RSA variants (RSA-1024, RSA-2048 and RSA-4096).

Results

First we gathered the theoretical attributes of keys generated for the tested algorithms, which provide a baseline for comparing their expected security and efficiency. These attributes include public and private key sizes, as well as bit security levels.

COMPARISON OF THEORETICAL ATTRIBUTES OF ALGORITHMS

Alg	Pub key size (B)	Priv key size (B)	Bit security (NIST)
ML-KEM512	800	1632	at least 128
ML-KEM768	1184	2400	at least 192
ML-KEM1024	1568	3168	at least 256
RSA 1024	not specified	128	80
RSA 2048	not specified	256	112
RSA 4096	not specified	512	140

Public key size is not specified for RSA as it depends on the random prime numbers chosen. Notably, ML-KEM variants require larger key sizes than RSA but offer significantly higher bit security, making them more resilient against attacks.
Bit security is a measure of the computational hardness of breaking a cryptographic algorithm, expressed in terms of the number of operations required to perform a brute-force attack. It quantifies security by comparing the difficulty of breaking an algorithm to that of a symmetric cipher with an equivalent key length. For instance, an algorithm with 128-bit security is considered as hard to break as a symmetric encryption scheme with a 128-bit key, assuming optimal attack strategies. In the context of post-quantum cryptography, bit security estimates account for both classical and quantum adversaries.
As the next step we generated pairs of public and private keys using the algorithms. The actual resulting key sizes reveal differences between theoretical expectations and real-world implementations, particularly due to encoding formats, metadata, and structural overhead. Compared to RSA, ML-KEM keys are significantly larger, reflecting the trade-offs in post-quantum cryptography where increased key sizes are necessary to achieve higher security levels.

COMPARISON OF MEASURED KEY SIZES FOR ML-KEM AND RSA

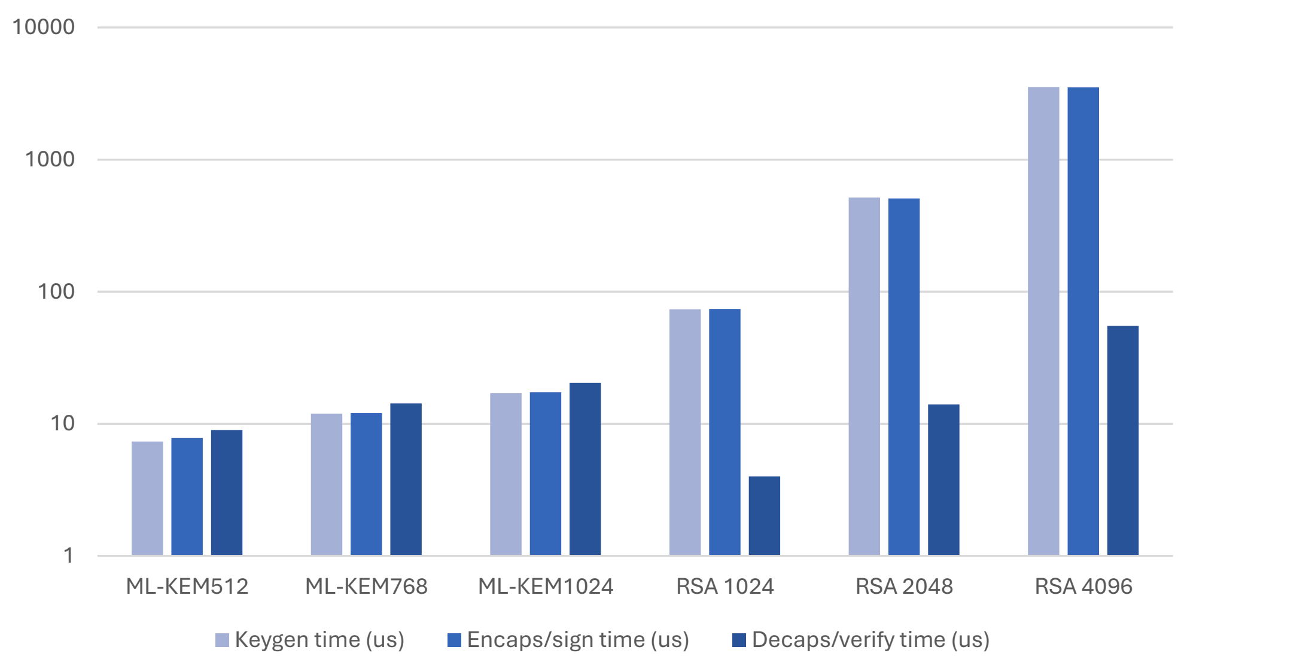
Algorithm	Pub key size on disk (B)	Priv key size on disk (B)
ML-KEM512	1166	3386
ML-KEM768	1686	4946
ML-KEM1024	2206	6506
RSA 1024	272	916
RSA 2048	451	1704
RSA 4096	800	3268

We then conducted performance benchmarks for ML-KEM and RSA, measuring the speed of key generation, encapsulation (or signing), and decapsulation (or verification) operations.

PERFORMANCE BENCHMARKING OF ML-KEM AND RSA OPERATIONS

Algorithm	Keygens /s	Encaps/sign /s
ML-KEM512	136480	127989
ML-KEM768	83667	83624
ML-KEM1024	58646	57768
RSA 1024	13549	13549
RSA 2048	1971	1971
RSA 4096	283	283

PERFORMANCE BENCHMARKING OF ML-KEM AND RSA OPERATIONS



The results highlight the contrast in efficiency between ML-KEM and RSA. ML-KEM exhibits significantly faster key generation and encapsulation rates compared to RSA, especially at higher security levels. In fact, ML-KEM's key generation and encapsulation speeds outperform its own decapsulation speed. Conversely, RSA's verification rate is much higher than its key generation and signing rates, further emphasizing the differences in performance between the algorithms. RSA demonstrates a higher rate of verification than ML-KEM at smaller key sizes, however at the highest security levels tested ML-KEM surpasses RSA in decapsulation efficiency as well.

207 Times more key generations per second were achieved by ML-KEM-1024 compared to RSA-4096, highlighting the efficiency of post-quantum key establishment at the highest tested security level.

As quantum computing continues to advance, algorithms like ML-KEM may become essential in securing communications, with its ability to offer robust security alongside remarkable performance. Continued research and optimization of these algorithms will help ensure their effective and efficient deployment in post-quantum environments.