

4 СЕКЦІЯ «КОМП'ЮТЕРНІ СИСТЕМИ ТА МЕРЕЖІ, БЕЗПЕКА ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ СИСТЕМ»

УДК 004.056

Nelasa Hanna¹, Al-Khamad Nauras²

¹ Candidate of technical sciences, assoc. professor at the NU “Zaporizhzhia Polytechnic”

² Phd student at the NU “Zaporizhzhia Polytechnic”, group A-172.13

ANALYSIS OF POST-QUANTUM MODULE-LATTICE BASED ALGORITHM SELECTED BY NIST

Existing public-key cryptographic schemes, such as the RSA cryptosystem, rely on the difficulty of factoring large numbers into prime factors – a problematic task as the numbers get larger. While computer scientists think that classical computers are practically incapable of factoring integers larger than 2048 bits, researchers have demonstrated that a cryptographically relevant quantum computer could break RSA-2048 in a matter of hours by using Shor's algorithm. If malicious actors were to get access to encrypted data, this could disrupt and harm customer and organizational trust in digital communication, online transactions in retail, digital signatures in finance, and critical infrastructure.

In a landmark announcement, the National Institute of Standards and Technology (NIST) has published its first set of post-quantum cryptography (PQC) standards.

NIST has finalized the following three PQC standards to strengthen modern public-key cryptography infrastructure for the quantum era:

- ML-KEM (derived from CRYSTALS-Kyber) – a key encapsulation mechanism selected for general encryption, such as for accessing secured websites;
- ML-DSA (derived from CRYSTALS-Dilithium) – a lattice-based algorithm chosen for general-purpose digital signature protocols;
- SLH-DSA (derived from SPHINCS+) – a stateless hash-based digital signature scheme.

The ML-KEM standard specifies the Module-Lattice-Based Key-Encapsulation Mechanism [1].

A KEM consists of three algorithms [1]. The three algorithms are:

- a probabilistic key generation algorithm;
- a probabilistic encapsulation algorithm;
- a deterministic decapsulation algorithm.

The purpose here is to review and analyze how this standard can be used for encrypting a message and establishing a shared private key between two parties.

At the start we have two parties: Alice and Bob who want to form a secure communication.

First, Alice generates public and private keys. In NIST's standard those are called encapsulation and decapsulation keys. This can be done by computing the following equation: $As + e$, where A is a matrix that consists of polynomials, s and e are vectors of polynomials as well. The result would be vector t (fig. 1). Matrix A and vector s are randomly generated and vector e is chosen from noise distribution. After previous computation (A, t) is considered to be the public key (or encapsulation key) of Alice and s is the private key (or decapsulation key).

$$\begin{bmatrix} A \end{bmatrix} \times \begin{bmatrix} s \end{bmatrix} + \begin{bmatrix} e \end{bmatrix} = \begin{bmatrix} t \end{bmatrix}$$

$(A, t) - \text{public key}$
 $s - \text{private key}$

Figure 1 – Alice generates public and private keys

Alice then sends the public key to Bob and he chooses his own random polynomial vector r for encryption of the message m . The encryption procedure also uses error vectors e_1 and e_2 . The polynomials within e_1 , e_2 and r are completely random and small, just like the ones in s .

The goal of this procedure is to encrypt message m and send it to Alice. The message m will be used to provide shared private key for secure communication.

Bob encrypts m using the public key (A, t) . The encryption procedure calculates two values (u, v) as follows:

$$u = A^T \cdot r + e_1 \quad (1.1)$$

$$v = t^T \cdot r + e_2 + m \quad (1.2)$$

Now the ciphertext consist of those two values: (u, v) . A polynomial vector u and the polynomial v . Those parameters will be sent to Alice. Now Alice can decrypt (u, v) to obtain the message m by using her private key s .

Given the private key s and a ciphertext (u, v) , the decryption for Alice is straightforward. Alice computes the following:

$$m = v - s^T \cdot u \quad (1.3)$$

Now Alice and Bob should have shared private key to securely communicate with each other.

The NIST's standard of course is more subtle and there are more things to consider. Mostly is the NTT form that polynomials need to be transformed to for

the sake of efficiency when multiplying them. But on the surface, this is kind of how this new algorithm works.

Figure 2 shows a simple view of key establishment between two parties.

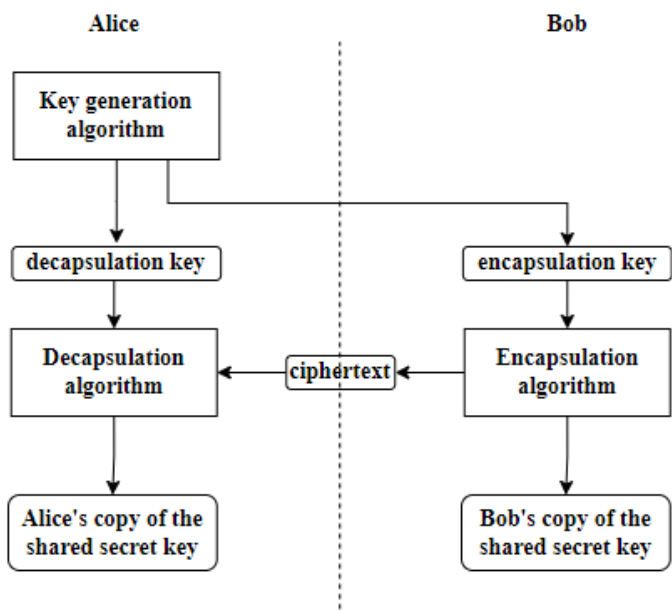


Figure 2 – A simple view of key establishment using KEM

As a result of this work, a new key exchange algorithm ML-KEM was reviewed and analyzed. For now, we can conclude that during the transition to post-quantum cryptography, there is considerable uncertainty regarding the robustness of both existing and new cryptographic algorithms. While we can no longer entirely trust traditional cryptography, we also cannot instantly place complete trust in post-quantum replacements until they have undergone significant scrutiny and real-world testing to uncover and correct potential implementation issues.

REFERENCES

1. National Institute of Standards and Technology (2024) Module-Lattice-Based Key Encapsulation Mechanism Standard. (Department of Commerce, Washington, D.C.), Federal Information Processing Standards Publication (FIPS) NIST FIPS 203. <https://doi.org/10.6028/NIST.FIPS.203>