Engineering Science & Technology
http://ojs.wiserpub.com/index.php/EST/

UNIVERSAL WISER PUBLISHER

Review Article

A Comprehensive Survey of Post-Quantum Cryptography and Its Implications

Bharat S. Rawal[1,2], **Anjan Biswas**[2,3,4,5*]

[1]Department of Computer Science and Digital Technologies, Grambling State University, Grambling, LA, USA
[2]Department of Mathematics and Physics, Grambling State University, Grambling, LA, USA
[3]Mathematical Modeling and Applied Computation Research Group, Center of Modern Mathematical Sciences and their Applications, Department of Mathematics, King Abdulaziz University, Jeddah, Saudi Arabia
[4]Department of Applied Sciences, Cross-Border Faculty of Humanities, Economics and Engineering, Dunarea de Jos University of Galati, Galati, Romania
[5]Department of Mathematics and Applied Mathematics, Sefako Makgatho Health Sciences University, Medunsa, Pretoria, South Africa
Email: biswasa@gram.edu

Received: 27 December 2023; **Revised:** 11 March 2024; **Accepted:** 11 March 2024

Abstract: Cryptography has been in existence since before the advent of computers. It encompasses a variety of techniques designed to secure information, whether it is at rest or in transit. Symmetric key cryptography includes symmetric keys, which serve the dual purpose of encrypting and decoding communications. Descriptions have been provided for public/private key pairs, also known as asymmetric keys. Such pairs may be categorized as secret/public key pairs, where the private key remains confidential, while the public key is accessible to all relevant parties for communication and information exchange. The realm of private and public key cryptography is diverse. Recently, the National Institute of Standards and Technology (NIST) endorsed four post-quantum cryptography systems. This paper presents a comprehensive survey of cryptography, examining the implications of intricate cryptographic systems on our existing networks.

Keywords: symmetric cryptography, asymmetric cryptography, quantum computing, post-quantum cryptography, lattice-based cryptography, algorithm, Crystal-Dilithium, Crystal-SPHINCS+

1. Introduction

As quantum threats become more prevalent, Post-Quantum Cryptography (PQC) emerges as a promising solution in terms of both efficiency and effectiveness. Unlike conventional encryption methods susceptible to quantum attacks, PQC computations offer robust security without compromising computing performance. PQC strikes a balance between enhanced security and functional productivity by leveraging numerical problems, including cross-section-based or hash-based techniques. The implementation of PQC exhibits precise timing, making it a sensible option for data collection across diverse computing scenarios. PQC addresses the pressing demand for security while concurrently focusing on enhancing performance in the rapidly evolving field of digital communication [1].

In the realm of cryptography, the advent of quantum computing platforms presents a significant challenge to the security of conventional public-key cryptographic algorithms [2], such as Elliptic Curve Cryptography (ECC)

DOI: https://doi.org/10.37256/est.5220244169

and Rivest-Shamir-Adleman (RSA). These algorithms hinge on the complexity of computing discrete logarithms or managing large numbers-tasks at which quantum computers excel. Recognizing the imminent threat, Post-Quantum Cryptography (PQC) has arisen to devise new cryptographic computations that remain resilient in the face of quantum attacks [3].

The normalization and adoption of Post-Quantum Cryptography (PQC) represents a fundamental step in fortifying digital defense against emerging quantum threats. As cryptographic systems confront the impending risk of quantum attacks, global initiatives are underway to establish standardized PQC algorithms [4]. This presents an opportunity to explore novel mathematical concepts and ideas within post-quantum cryptography. A profound understanding of mathematical principles and their application in cryptography is essential for designing encryption algorithms resilient to quantum computer attacks. Consequently, new mathematical ideas and methods have been uncovered, extending their applications beyond PQC to computer science and mathematics [5]. One noteworthy field is lattice-based cryptography, a subclass of PQC. This cryptographic approach relies on the complexity of certain algebraic problems using lattices-geometric structures. Lattices can be employed to construct cryptographic algorithms that effectively counteract attacks from quantum computing systems. The examination of lattices and their characteristics plays a crucial role in this context [6]. Similar challenges arise as Post-Quantum Cryptography (PQC) aims to replace traditional cryptographic strategies, necessitating a delicate balance between current security objectives and the constraints of existing systems [7]. Overcoming this challenge demands strategic planning, retrofitting, or, in some instances, a gradual transition to more integrated systems. Effectively addressing legacy system compatibility is crucial to ensuring the widespread adoption and effectiveness of PQC in various industrial landscapes [7].

The following is how the rest of the paper is structured: The introduction is covered in section I, and the related work is covered in section II. Then, section III is about the types of cryptography. Section IV is about the need to consider post-quantum cryptography. Section V describes the post-quantum cryptography. Section VI describes the performance of the crypto-system. Finally, Section VII concludes the research paper.

2. Related works

In the domain of cryptography, the approach of quantum PCs represents a critical test of the security of conventional public-key cryptographic calculations, like RSA and ECC. These algorithms depend on the difficulty of considering substantial numbers or deciding discrete logarithms, issues that quantum PCs can tackle effectively. To address this approaching danger, the field of Post-Quantum Cryptography (PQC) has arisen, intending to foster new cryptographic calculations that are resistant to quantum assaults. PQC presents both challenges and opportunities. From one perspective, planning and executing PQC calculations is a complicated and progressing process. These calculations frequently have bigger key sizes and computational prerequisites contrasted with customary calculations, which can affect execution and adaptability. The execution of PQC shows cutthroat execution times, settling on it as a reasonable decision for getting information in different figuring conditions. As a quantum-safe worldview, PQC tends to the quick requirement for security and does so with an eye on upgrading execution in the consistently developing scene of digital communication [8]. The normalization and reception of PQC addresses a basic stage in strengthening computerized protection from arising quantum dangers. As cryptographic frameworks face the approaching gamble of quantum assaults, worldwide endeavors are in progress to lay out normalized PQC calculations [9]. Normalization endeavors, driven by associations like NIST, assume an urgent part in laying out an establishment for far-reaching reception. The drawn-out security of PQC depends on its capacity to endure current cryptographic difficulties as well as the capricious scene of future headways. Ceaseless refinement, normal updates, and an initiative-taking position against potential weaknesses are fundamental for guaranteeing perseverance through the flexibility of PQC despite developing dangers [10]. Heritage frameworks, profoundly imbued in numerous associations, represent a significant test for the consistent reconciliation of PQC. These obsolete frameworks frequently miss the mark on inborn adaptability to oblige the original calculations and conventions presented by PQC. Similar issues emerge as PQC looks to supplant traditional cryptographic strategies, requiring a sensitive harmony between current security goals and the limitations of the inheritance framework [11]. Overcoming this issue requires vital preparation, retrofitting, or, at times, a progressive change to additional coordinated frameworks. As PQC develops, its consistent consolidation into

an expansive range of utilizations guarantees a hearty safeguard against advancing digital dangers in our undeniably interconnected and information-driven world [12]-[13]. Overall, PQC presents both challenges and opportunities. While planning, conducting, and normalizing PQC algorithms is an overwhelming task, the possible advantages of improved security and insurance against future quantum assaults offset these difficulties. Rawal and Shah presented a Secure UDP (S-UDP) framework for faster and secure communication [14]. PQC offers the potential chance to shield our advanced framework, safeguard delicate information, and cultivate development in the area of cryptography. As quantum processing innovation develops, PQC will assume an undeniably pivotal part in guaranteeing the security and uprightness of our computerized world.

3. Types of cryptography

3.1 *Symmetric cryptography schemes*

(1) Advanced Encryption Standard (AES) is a widely used symmetric encryption algorithm with three key sizes: AES-128, AES-192, and AES-256.

Merits: Three key sizes (AES-128, AES-192, and AES-256) offer flexibility based on security requirements.

Drawbacks: Vulnerable to side-channel attacks if not implemented properly.

(2) Triple Data Encryption Algorithm (TDEA/Triple DES): The Data Encryption Standard (DES) algorithm is used three times in a row with distinct keys in a symmetric encryption process.

Merits: Increased security compared to DES due to triple iterations. Still used in certain legacy systems.

Drawbacks: Slower compared to newer algorithms like AES. Key management challenges due to multiple keys.

(3) Safe Hash Algorithm (SHA) The cryptographic hash algorithms SHA-1, SHA-224, SHA-256, SHA-384, and SHA-512 are part of the family.

Merits: Provides data integrity and authenticity through hashing. Different variants offer flexibility in security levels.

Drawbacks: Vulnerable to collision attacks in the case of weaker variants like SHA-1.

(4) The Message Digest algorithm (MD) is a family of cryptographic hash functions that includes the MD2, MD4, and MD5 hash functions (however, MD5 is regarded as weak and is not advised for new applications).

Merits: Offers hash functions for various applications.

MD5 is considered weak and not recommended for new applications due to vulnerabilities.

Drawbacks: Offers hash functions for various applications.

(5) Keyed-Hash Message Authentication Code (HMAC): A technique that enables message integrity and authentication through the integration of a cryptographic hash function (such as SHA) with a secret key.

Merits: Combines cryptographic hash function and secret key for integrity and authentication.

Drawbacks: Reliance on the strength of the underlying hash function.

(6) A symmetric block cipher called the International Data Encryption Algorithm (IDEA) was created to take the place of the Data Encryption Standard (DES).

Merits: Created as a replacement for DES, providing enhanced security.

Drawbacks: Less widely used compared to AES, potentially leading to fewer community evaluations.

(7) Skipjack: A symmetric block cipher created for the government's key escrow mechanism, the Clipper chip.

Merits: Created as a replacement for DES, providing enhanced security.

Drawbacks: Limited adoption outside of its intended use.

3.2 *Asymmetric cryptography schemes [2]*

An asymmetric cryptosystem is considered more secure because we do not have to share our private key or reveal it with a third party.

1. Digital Signature Algorithm (DSA).
2. Elliptic Curve Digital Signature Algorithm (ECDSA).
3. Rivest-Shamir-Adleman (RSA) is a public-key cryptosystem.

4. Menezes-Qu-Vanstone (MQV) is an authentication scheme for key agreements based on the Diffie-Hellman scheme.

3.3 *Hash-based digital signature schemes*

The primary goal is to construct a fixed-size hash value, usually a digest, from input data. This hash value uniquely identifies the input.

Integrity and authentication: Hash functions are used to ensure the integrity of data. Any changes to the supplied data generate a different hash value, making manipulation easier to detect. Furthermore, hash-based systems frequently use secret keys to authenticate the provenance of data.

No Reversibility: Hash functions, unlike encryption techniques, are supposed to be one-way, which means it should be computationally impossible to reverse the process and get the original input from the hash result. Hash functions are often computationally efficient, allowing for the rapid production and verification of hash values.

Rather than relying on extra cryptographic presumptions like hardness based on number theory, hash-based cryptosystems just use cryptographic hash functions. Thus, cryptanalysis's window of opportunity is closed. This lessens the system's overall complexity. In order to attain the desired performance, the hash-based scheme must be flexible in the hash function it chooses because it is intrinsically dependent on the application-specific environment. This technique protects the application from numerous assaults thanks to the collision resistance, pre-image resistance, and second-pre-image resistance properties of hash functions [15]-[16].

Numerous characteristics of the hash-based scheme that are advantageous to the Internet of Things (IoT) environment are identified in [17]-[18]. There are lightweight hash function variants that give IoT applications the choice of appropriate device parameters for resource-constrained devices, enhancing network performance. Since hash functions in hash-based schemes only work in one direction, they are secure with both backward and forward secrecy [16]. Johannes Buchmann, Erik Dahmen, and Michael Szydlo introduced various hash-based signature schemes [8].

3.4 *Code-based cryptography*

In this system, the one-way function employs error-correcting code C and computes a condition related to the parity check matrix of C [8], [9], [15], [17]. A Goppa code is an error-correcting code built around modular algebra, which is the process that occurs when a series of integers increases to a given number and then returns to zero once attained [19].

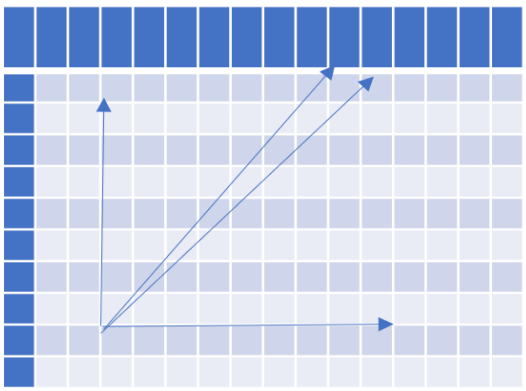


Figure 1. Two-dimensional lattice

3.5 *Lattice-based cryptography*

A lattice can be defined as the set of intersection points of an infinite, regular, but not necessarily orthogonal n-dimensional grid. For example, the set of integer vectors Z^n is a lattice. In computer science, lattices are usually

represented by a generating basis. A lattice L is a set of points in n-dimensional space with a cyclic arrangement, such as the one shown in Figure 1. A basis vector refers to a vector that generates the lattice by forming a basis for it. Basis vectors are essential in lattice theory as they help describe the structure of the lattice and provide a way to express any lattice point using integer linear combinations of these basis vectors.

The lattice vectors are those that span the entire lattice. These vectors define the periodic structure of the lattice. An excellent basis is one in which the base vectors are almost perpendicular. If the base vectors are almost parallel, they are referred to as bad bases. In lattice-based cryptography, the private key utilizes the good basis, and the public key uses the poor basis. In computer science, determining the shortest or closest vector is considered a challenging problem. The closest vector problem is more difficult to solve with a bad basis than with a good one.

Formally, given n linearly independent vectors $b_1, \ldots, b_n \in R^m$, the lattice generated by them is a set of vectors [8].

$$L(b_1, \ldots, b_n) \overset{def}{=} \left\{ \sum_{i=1}^{n} x_i b_i \mid x_i \in Z \right\}$$

It has a security-based worst-case scenario connection. Another desirable worst-case connection to achieve for classical (non-quantum), compared with number theoretical cryptography. The random key generation process requires hard problems [8]. The Learning With Errors (LWE)-based cryptosystem is thought to be the most effective lattice-based cryptosystem to date, and it is backed up by a theoretical demonstration of security [9].

4. Why do we need to consider PQC now?

For different applications, different block cipher modes are utilized. Depending on the type of application, security levels may vary and can be divided into high, medium, and low [1]. National security communication requires high security, chatroom and social networking software may work with medium security. For low security priority application speed of data is more important. We can increase security simply by increasing block size, key size, or increasing the number of rounds. In addition, by compressing data, we can minimize data loss [1]. Data compression is handled before the encryption process.

Shor discovered quantum algorithm factoring and discrete log functions in 1994. This algorithm has the capability of damaging the most widely used public key exchange cryptosystems. The development of the Grover method demonstrated that quantum computers outperform conventional computers in exploring databases by a square root factor [20]. If quantum computers become widely available, there will be a requirement to protect information recorded before the quantum era [21]. Grover's algorithm reduces the time required quadratically. Doubling the key size restores the security level.

Shor's algorithms can break the following cryptosystems:

RSA;

Diffie-Hellman key-exchange;

Elliptical Curve Cryptosystem;

Buchmann-William's Key exchange;

Algebraic Homomorphic.

When are we expecting the availability of large quantum computers? 10, 15, or 20 years, depending on the speed of technological advancements.

Why do we have to act now?

(1) Development and standardization take time.

(2) Improvement also takes time.

(3) Take time to build confidence in PQC.

(4) It takes time to improve the usability of PQC.

(5) Last year 2022 NIST announced the list of PQC.

5. Post-quantum cryptography

The ultimate goal of post-quantum cryptography is to produce cryptographic systems that can be resistant or secure in the presence of both quantum and conventional computers. As stated earlier, what makes quantum computers so powerful has to deal with the use of bits. With conventional computers a bit is either a one or zero, but dealing with quantum computing introduces something called a qubit. A qubit can be either a one or zero and everything in between at the same time making it infinite. Initializing a brute force attack while using a quantum computer would prove to be detrimental because it would be all combinations at the same time. And again, as stated earlier because of the unstable environment due to the presence of electrons qubits cannot be kept in the same state for a long time, deeming it a technology for the near future.

The four encryption methods are Fast-Fourier Lattice-based Compact Signatures over N-th degree Truncated polynomial Ring Units (NTRU) (Falcon), Stateless hash-based signatures scheme SPHINCS+, Crystals-Dilithium, and Crystal-Kyber were selected by NIST. Additionally, it uses its public key as a method of learning by making mistakes or learning with errors. Unlike typical lattice-based signature algorithms, which typically use a truncated Gaussian distribution to generate the coefficients in their error vectors, Dilithium has a unique error distribution. How this works is you have to find short vectors in lattices, the strength of the key is based on the size of the matrix polynomials. This method of Crystals-Dilithium key is the smallest public key and signature size. Moving forward to SPHINCS+ cryptography which utilizes Few-Time-Signature (FTS) schemes [22]. This method uses a so-called hypertree to authenticate a large number of key pairs with few time signatures. Signature schemes known as 'few-time signatures' enable a key pair to generate a limited quantity of signatures. A pseudo-random FTS key pair is selected to sign each new message.

5.1 *Crystal dilithium*

Crystal Dilithium is a strong post-quantum cryptographic signature system that the NIST is now considering standardizing. When used on a large-scale quantum computer, Shor's method can easily break discrete logarithm and integer factorization issues, which are the foundation of traditional public-key cryptography approaches. Dilithium, on the other hand, is a good option for safe post-quantum cryptographic signatures since it solves Learning With Errors (LWE) difficulties more easily than both conventional and quantum computers [10], [23]. Learning With Errors is a mathematical issue that underpins many lattice-based cryptography methods. In the LWE problem, the goal is to separate random noise from a linear relationship between secret and observed values.

Crystal Dilithium cryptography concept provides a defense against these impending threats. Motivated by the overwhelming power of fake Dilithium crystals, experts have looked into the possibility of using cross-section-based encryption, a subset of postquantum cryptography, to create cryptographic frameworks that are resistant to quantum attacks. The development of Dilithium signature schemes is one of the most promising uses of Crystal Dilithium in cryptography. These strategies aim to fortify defenses against conventional and quantum attacks. Unlike traditional cryptographic computations, which rely on intricate numerical jobs, Crystal Dilithium is sensitive to the difficulty of particular cross-section problems, which makes it applicable to quantum computations. rely on how challenging it is to solve lattice-based tasks [1], [4].

Following is a high-level illustration of the process of creating cryptographic keys using matrices, functions, and hash operations to produce the public.

KeyGen () Algorithm [24]

(1) The binary values in $\{0, 1\}$ are used to construct (ρ, ρ', K), which add up to a 512-bit value. For this, the input ζ is subjected to the hash function $H(\zeta)$.

(2) The function Expand$A(\rho)$ is utilized to generate q from a matrix A with dimensions of $k \times l$.

(3) There are two values (s_1, s_2) chosen from a set S of length l.

(4) Expand$S(\rho')$ is used to construct a new set of values η from the set S and ρ'.

(5) The Power2Round$q$ function is applied to the value t for a predetermined number of times (d iterations).

(6) Using function H, tr is obtained from the hash of ρ and t_1 concatenated.

Sign () Algorithm [24]

Input: sk (secret key) and M (message)

Output: $\sigma = (\tilde{c}, z, h)$ - a signature consisting of three components.

Steps:

(1) Generate matrix A with dimensions $k \times l$ and obtain q through Expand$A(\rho)$.

(2) Create μ, a 512-bit value derived from the hash of tr concatenated with M.

(3) Initialize κ to 0 and set (z, h) to undefined.

(4) Generate ρ' using the hash function with K concatenated with μ.

(5) While (z, h) is undefined:

Choose a value y from a set S with length l.

Calculate γ_1 using ExpandMask (ρ', κ).

Compute $w = Ay$.

Extract w_1, the higher bits of w using a specific length $(2\gamma_2)$.

Calculate $\tilde{c}$ using the hash of μ concatenated with w_1 (256 bits).

Sample c from a certain set $B\tau$ based on $\tilde{c}$.

Compute $z = y + cs_1$.

Calculate r_0 using the lower bits of $(w - cs_2)$ with a specific length $(2\gamma_2)$.

(6) Check conditions:

If the infinity norm of z is greater than or equal to $(\gamma_1 - \beta)$ or the norm of r_0 is greater than or equal to $(\gamma_2 - \beta)$:

Set (z, h) to undefined.

Otherwise:

$y \in S^{l}_{\gamma 1} \leftarrow$ ExpandMask (ρ', κ)

(7) Check conditions:

$w \leftarrow Ay$.

(8) $w_1 \leftarrow$ HighBits$q$

$(w, 2\gamma_2)$.

(9) $\tilde{c} \in \{0, 1\}$

$256 \leftarrow H(\mu // w_1)$

(10) $c \in B\tau \leftarrow$ SampleInBall $(\tilde{c})$

(11) $z \leftarrow y + cs_1$

(12) $r_0 \leftarrow$ LowBits$q$ $(w - cs_2, 2\gamma_2)$

(13) if $// z // \infty \geq \gamma_1 - \beta$ or $// r_0 // \geq \gamma_2 - \beta$ then

(14) $(z, h) \leftarrow \perp$

(15) else

(16) $h \leftarrow \text{MakeHint}_q$ $(-ct_0, w - cs_2 + ct_0, 2\gamma_2)$

(17) if

(18) $// ct_0 // \infty \geq \gamma_2$ or the number of 1's in h is greater than ω then.

(19) $(z, h) \leftarrow \perp$

(20) $\kappa \leftarrow \kappa + l$.

Verify() Algorithm [24]

Input: pk, M, σ

Output: v is valid or invalid

1 $A \in R_q^{k \times 1} \leftarrow$ Expand$A(\rho)$

2 $\mu \in \{0, 1\}^{512} \leftarrow H(H(\rho // t_1 // M))$

3 $c \leftarrow$ SampleInBall$(\tilde{c})$

4 $w'_1 \leftarrow \text{Usehint}_q$ $(h, Az - ct_1 \cdot 2^d, 2\gamma_2)$

5 $v \leftarrow (// z // \infty < \gamma_1 - \beta)$ && $(c == H(\mu // w'_1))$ && (number of 1's in h is not greater than ω).

Let us break it down:

Input: pk (public key), M (message), σ (signature)

Output: v (validity)

Generate matrix A with dimensions $R_q^{k\times 1}$ using Expand$A(\rho)$.

Compute μ as a 512-bit value obtained from hashing the concatenation of $H(\rho /\!/ t_1 /\!/ M)$.

Sample a value c from a certain set using SampleInBall ($c\tilde{}$).

Use a hint h to compute w_1' utilizing a function Usehintq. This involves using the hint h along with some computation involving $Az - ct_1 \cdot 2^d$ with parameters γ_2.

Validate the signature:

Check if the infinity norm of z is less than $\gamma_1 - \beta$.

Verify if c is equal to $H(\mu /\!/ w_1')$.

Ensure that the number of 1's in h is not greater than a certain threshold ω.

Dilithium is a form of digital signature (DS) technique that is highly secure against specific message attacks due to the complexity of lattice challenges over module lattices. Crystal Dilithium focuses on digital signatures, and here are some of the characteristics of Crystal Dilithium that make it unique and intriguing in a post. Quantum context:

1. Asymmetric Cryptography;
2. Lattice-Based Security;
3. Digital Signature Scheme;
4. Quantum Resistance;
5. Key Size and Efficiency;
6. Cryptographic Agility;
7. Standardization and Scrutiny;
8. Publicly Available.

5.2 *Crystal-kyber*

In Oded Regev's publication in 2005, Kyber was developed with contributions from North America and Europe, representing diverse organizations governments, and enterprises this continuous development for the future employment of supercomputers or quantum computers, despite recent updates to Kyber, such as higher noise levels and reduced compression for the level one perimeter set, the system remains post, quantum and designed to be secure. Even in the face of quantum computers hyper features three security levels. All keys are the same size. The private key of a pair consists of polynomials with a tiny Coefficient. The detailed implementation algorithm is explained in [23], [25].

Kyber Security is based on the difficulty of solving mathematical issues linked with lattes. Quantum computing research is ongoing and intends to defend against the potential future. Quantum attacks are cyber security attacks that use quantum computers to launch a brute force attack. Hackers will be able to quickly crack popular asymmetric encryption schemes like RSA and ECC with the help of quantum attacks. This can result in the disclosure of private information, including personal information, passwords, etc.

The term was introduced to describe Kyber resilience in post-quantum cryptography (PQC). This resilience ensures that it is impossible to create ciphertext lawfully decrypt using two different private keys. Fortunately, [26] demonstrated that a robust hybrid Public Key Encryption (PKE) scheme can be built by combining Key Encapsulation Mechanism (KEM) with a suitable robust Digital Elevation Model (DEM) as stated in [27]. In other words, combining Kyber with a one-time strong pseudo-random and robust DEM will result in a post-quantum strong anonymous and robust PKE scheme [28].

5.3 *Falcon*

A further post-quantum digital signature algorithm is Falcon Cryptography. Falcon cryptography is renowned for its security and efficiency. Falcon makes use of the same components that Crystals-Dilithium does: a mathematical component, a generating component, a verification component, a security component, and instructions on how to sign using Falcon Cryptography. Falcon Cryptography employs the Learning with Errors (Ring-LWE) issue. These math

issues deal with error distribution and modular arithmetic. Even quantum computers are unable to answer these puzzles. Creating a message with Falcon Cryptography is remarkably similar to creating one with Crystals-Dilithium. Both the relevant public key and the private key must be generated. It is always necessary to keep the private key confidential, even when sharing the public key. In Falcon Cryptography, a digital signature is created by utilizing your private key whenever you wish to sign a communication. The signature technique used by Falcon Cryptography is extremely resistant to quantum attacks since it makes use of polynomial rings, error distributions, and many mathematical computations. People who have received your signed communication can use the public key you supplied to validate the authenticity of the signatures in order to accurately authenticate Falcon Cryptography. Similar to Crystals-Dilithium, they can rely on the message being signed by the right private key holder if the verification process is successful. This is an important level of security against quantum computers. Falcon is an additional post-quantum, lattice-based digital signature technique. These are the procedures that explain how Falcon operates, per [29].

Figure 2 is Cyclotomic Ring in Lattice-based Cryptography: In lattice-based cryptography, mathematical objects like rings play a crucial role in constructing secure cryptographic schemes. Cyclotomic rings have desirable mathematical properties that make them suitable for lattice-based constructions, providing a foundation for Falcon's security.

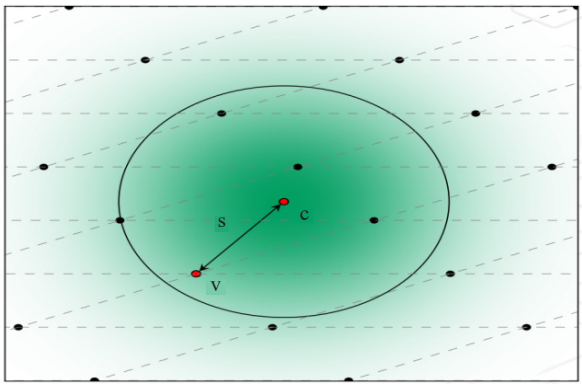


Figure 2. Vector *s*, *v* in in cyclometric ring *R* [30]

The work over the cyclotomic ring $R = Zq[x]/(x^n + 1)$ [30].

⇒ Keygen()

Start

1. Generate matrices A and B with coefficients in R such that:
   a. Set $BA = 0$.
   b. Ensure B has small coefficients.
2. Set pk (public key) ← A.
3. Set sk (secret key) ← B.

End.

⇒ Sign(m, sk)

Start

1. Compute c such that $cA = H(m)$.
2. Find a vector v in the lattice $\lambda(B)$, close to c.
3. Calculate $s \leftarrow c - v$.
4. Set the signature sig as $s = (s_1, s_2)$.

End.

⇒ Verify(m, pk, sig)
Start
1. Check if 's' is short.
   a. If 's' is short, proceed to step 2.
   b. If 's' is not short, reject the signature.
2. Calculate sA and $H(m)$.
3. Accept the signature if $sA = H(m)$.
   a. If $sA = H(m)$, accept the signature.
   b. If $sA \neq H(m)$, reject the signature.

End.

5.4 *SPHINCS+*

SPHINCS+ cryptography utilizes FTS schemes [22]. This method uses a so-called hyper-tree to authenticate a large number of key pairs with few-time signatures. Signature schemes known as 'few-time signatures' enable a key pair to generate a limited quantity of signatures. For every new communication, a pseudo-random FTS key pair is chosen to sign it. The FTS signature and the authentication data for that FTS key pair make up the signature. A hyper tree signature, or a signature using a certification tree of a Merkle tree signature, represents the authentication information.

6. Performance analysis

We have conducted a performance analysis on the total key size and number of cycles required to sign and vary the signature using the NIST data set of Zoo of cryptography [31]. Figure 3 compares the public key sizes and signature bytes of Falcon and SPHINICS+ at NIST security level 1.

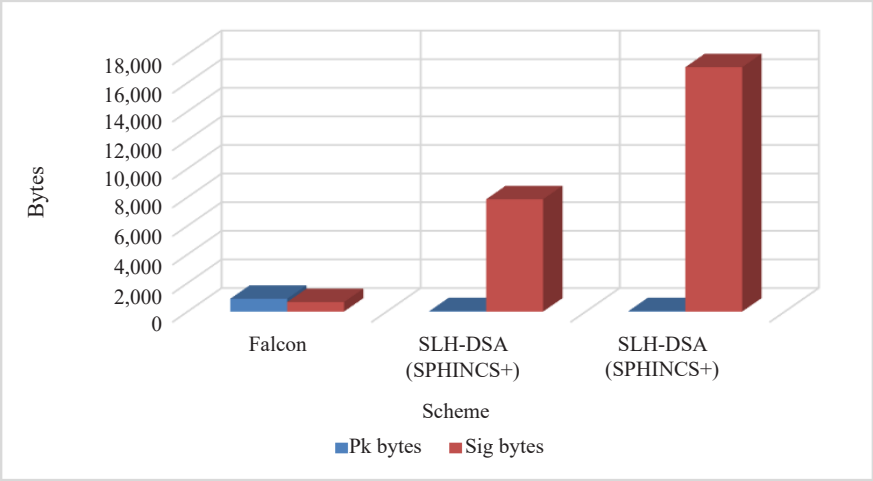


Figure 3. Key size analysis security level 1

Figure 4 compares the public key size, signature cycles, and verification cycles of Falcon and SPHINICS+. We can see that Falcon's Signature key size is substantially less than SPHINICS+. SPHINICS+ has a substantially smaller public key size than Falcon.

Dilithium at the NIST security level 5. We can see that SPHINICS+ has unusually large signature cycles, approximately 7 billion cycles. In comparison to the other two, SPHINICS+ also has a significant verification cycle.

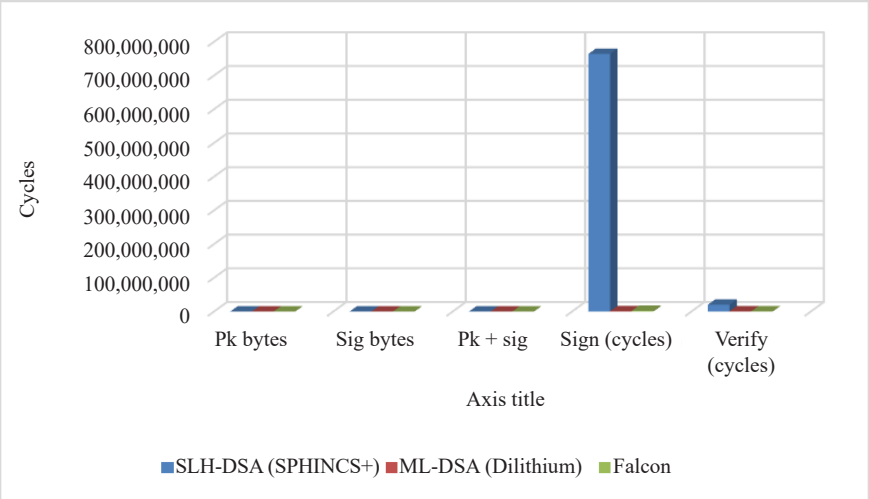


Figure 4. Signature analysis security level 5

Figure 5 compares the public key sizes and signature bytes of Falcon, SPHINICS+, and Dilithium at NIST security level 5. We can notice that SPHINICS+ offers the smallest public key size, and Dilithium has the largest size in comparison to the other two. Whereas, in terms of the signature key size, Falcon is the smallest and SPHINICS+ has the largest size.

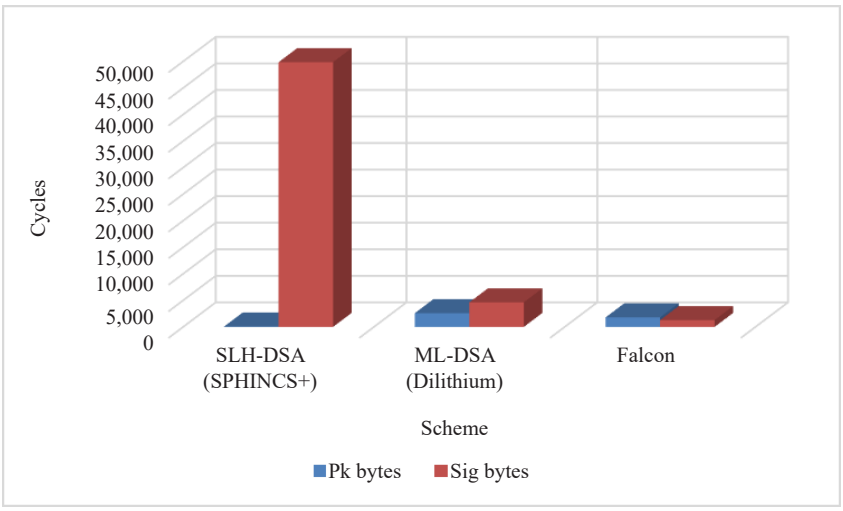
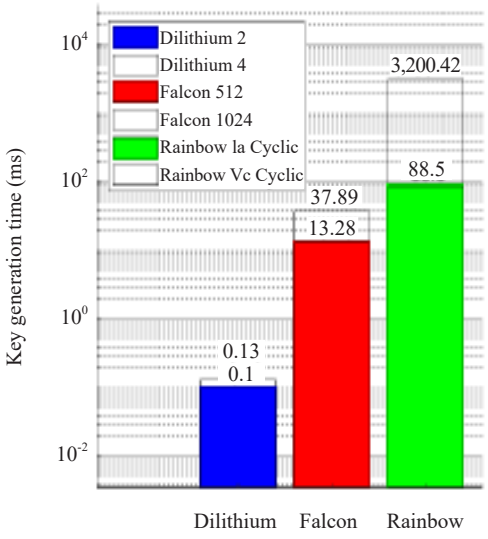
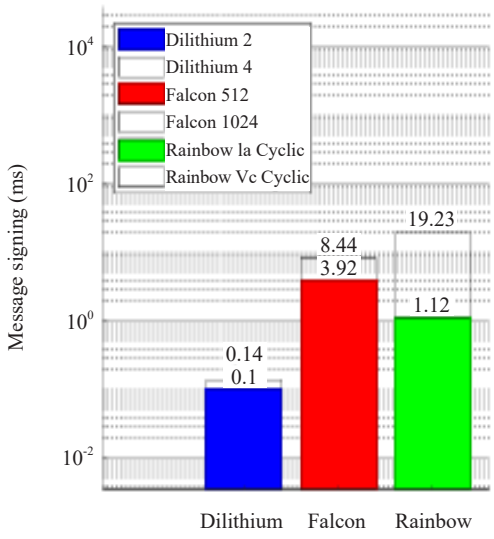


Figure 5. Key size analysis security level 5

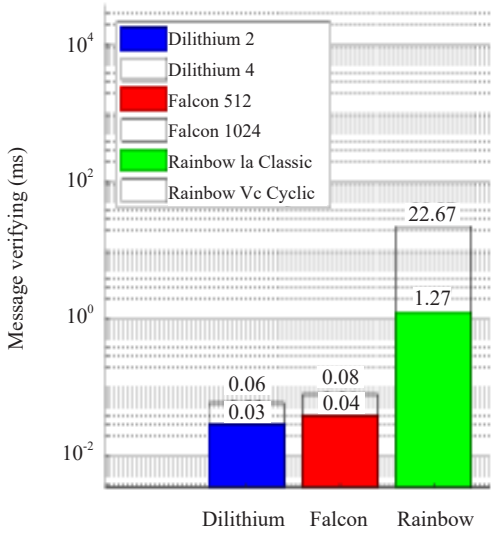
Figure 6 compares the key generation time, massage sinning time, and massage verification time for Dilithium, Falcon, and Rainbow. Dilithium offers the fastest key generation time around 0.1-0.13 milliseconds. The Rainbow has the slowest time, it takes around 88.5-3,200.42 milliseconds. If we look at the massage sinning time Dilithium is the fastest and it takes only 0.1-0.14 milliseconds. Similarly, Dilithium offers the fastest massage verification time of about 0.03-0.06 milliseconds.



(a) Key generation time



(b) Message signing time



(c) Message verifying time

Figure 6. The fastest (colored bar) and slowest (outlined bar) signature candidates from each family cross the three signature phases, with a message length of 100 Bytes [32]

7. Conclusion

Although the technical issues of PQC migration and cryptographic agility are crucial, many researchers and crypto analysts believe that a greater understanding of the problem's people, process, and policy factors is urgently required. While technological solutions are crucial, they also influence whether the PQC selected by NIST will be adopted. [29] describes several criteria for Conformity Assessment (CA) systems. The primary goal of most post-quantum cryptosystems is to demonstrate great security by failing the other criteria. Energy, latency, and resource consumption are the other factors and are critical to IoT networks. Many parameters should be included in an approach rather than just a few for it to be appropriate for real-time applications. The degree of protection that post-quantum cryptosystems provide against quantum assaults is not standardized. It is vital to understand the parameters and their

relative importance for Internet of Things applications. This can be achieved if major researchers in the field collaborate or if certain standards provide a set of protocols and methods. Future IoT networks with limited resources should see increasingly powerful devices. These devices are anticipated to continue being low-computational in the recent past. However, after ten to twenty years, computational capacity will rise. With the development of new quantum algorithms and high-performing quantum computers. The necessity for a transition to post-quantum cryptography stems from the potential threat that quantum computers represent to standard encryption techniques. Quantum computers are capable of solving some mathematical problems far quicker than classical computers, potentially making current widely used cryptography approaches vulnerable. The paper has introduced some important reasons for moving to post-quantum cryptography. To summarize, post-quantum cryptography is critical for addressing the possible vulnerabilities created by quantum computers. PQC offers high security with added complexity and large key sizes. PQC also suffers several forms of side-channel attacks.

Conflict of interest

The authors declare no conflict of interest.

References

[1] L. Ducas, T. Lepoint, V. Lyubashevsky, P. Schwabe, G. Seiler, and D. Stehlé, "Crystals-dilithium: Digital signatures from module lattices," *IACR Transactions on Symmetric Cryptology*, vol. 1, pp. 238-268, 2018.

[2] D. P. Joseph, M. Krishna, and K. Arun, "Cognitive analytics and comparison of symmetric and asymmetric cryptography algorithms," *International Journal of Advanced Research in Computer Science*, vol. 6, no. 3, pp. 51-56, 2015.

[3] M. Azouaoui, O. Bronchain, G. Cassiers, C. Hoffmann, Y. Kuzovkova, J. Renes, M. Schönauer, T. Schneider, F.-X. Standaert, and C. van Vredendaal, "Protecting dilithium against leakage: Revisited sensitivity analysis and improved implementations," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2023, no. 4, pp. 58-79, 2023.

[4] M. Ahmed, N. Moustafa, A. Barkat, and P. Haskell-Dowland, Eds., *Next-Generation Enterprise Security and Governance*. Boca Raton, Florida: CRC Press, 2022.

[5] W. Whyte, "Falcon: New Post-Quantum Cryptography Standard Advances Data Security," *qualcomm.com*, 2022. [Online], Available: https://www.qualcomm.com/news/onq/2022/07/falcon--how-this-new-u-s--adopted--qualcomm-backed-cryptography-. [Accessed Oct. 2, 2023].

[6] T. Pornin and T. Prest, "More efficient algorithms for the NTRU key generation using the IACR," 2019. [Online], Available: https://eprint.iacr.org/2019/015.pdf. [Accessed Oct. 2, 2023].

[7] National Security Agency | Frequently Asked Questions, 2021. Available: https://media.defense.gov/2021/Aug/04/2002821837/-1/-1/1/Quantum_FAQs_20210804.PDF. [Accessed Sept. 11, 2023].

[8] D. J. Bernstein, "Introduction to post-quantum cryptography," in: *Post-Quantum Cryptography*, D. J. Bernstein, J. Buchmann, and E. Dahemen, Eds. Berlin, Heidelberg: Springer, 2009, pp. 1-14.

[9] Peikert, C. "Public-key cryptosystems from the worst-case shortest vector problem," in Proceedings of the forty-first annual ACM symposium on Theory of computing, 2009, pp. 333-342.

[10] P. W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in Proceedings 35th Annual Symposium on Foundations of Computer Science. Santa Fe, NM, USA: IEEE, 1994, pp. 124-134.

[11] IBM Quantum Learning, "Explore gates and circuits with the Quantum Composer," Available: https://quantum-computing.ibm.com/composer/docs/iqx/guide/shors-algorithm. [Accessed Sept. 11, 2023].

[12] Information Technology/Cybersecurity-Overview, National Institute of Standards and Technology, Available: https://www.nist.gov/cryptography#:~:text=NIST%20continues%20to%20lead%20public,encrypting%20large%20amounts%20of%20data. [Accessed Sept. 11, 2023].

[13] M. Kumar and P. Pattnaik. "Post quantum cryptography (PQC)-An overview," in 2020 IEEE High Performance Extreme Computing Conference (HPEC), IEEE, 2020, pp. 1-9.

[14] B. S. Rawal and J. Shah, "SUDP: The frontier tool for security in 5G and beyond wired or wireless communication," in 2021 IEEE Globecom Workshops (GC Wkshps), Madrid, Spain, 2021, pp. 1-6.

[15] S. Suhail, R. Hussain, A. Khan, and C. S. Hong, "On the role of hash-based signatures in quantum-safe internet of things: Current solutions and future directions," *IEEE Internet of Things Journal*, vol. 8, no. 1, pp. 1-17, 2020.
[16] A. Kumar, C. Ottaviani, S. S. Gill, and R. Buyya, "Securing the future Internet of things with post-quantum cryptography," *Security and Privacy*, vol. 5, no. 2, pp. e200, 2022.
[17] J. Moon, I. Y. Jung, and J. H. Park, "IoT application protection against power analysis attack," *Computers & Electrical Engineering*, vol. 67, pp. 566-578, 2018.
[18] S. Balogh, O. Gallo, R. Ploszek, P. Špaček, and P. Zajac, "IOT security challenges: Cloud and blockchain, Postquantum cryptography, and evolutionary techniques," *Electronics*, vol. 10, no. 21, pp. 2647, 2021.
[19] G. C. Kessler, "An overview of cryptography," 2003.
[20] S. Renault, J. Geng, F. Dolhem, and P. Poizot, "Evaluation of polyketones with N-cyclic structure as electrode material for electrochemical energy storage: case of pyromellitic diimide dilithium salt," *Chemical Communications*, vol. 47, no. 8, pp. 2414-2416, 2011.
[21] S. S. Iqbal and A. Zafar, "A survey on post quantum cryptosystems: Concept, attacks, and challenges in IoT devices," in 2023 10th International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India: IEEE, 2023, pp. 460-465.
[22] D. J. Bernstein, A. Hülsing, S. Kölbl, R. Niederhagen, J. Rijneveld, and P. Schwabe, "The SPHINCS+ signature framework," in Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, 2019, pp. 2129-2146.
[23] T. X. Pham, P. Duong-Ngoc, and H. Lee, "An efficient unified polynomial arithmetic unit for CRYSTALS-dilithium," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 70, no. 12, pp. 4854-4864, 2023.
[24] S. Bai, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, P. Schwabe, G. Seiler, and D. Stehlé. "CRYSTALS-Dilithium: Algorithm Specifications and Supporting Documentation (Version 3.1)," 2021. Available: https://pq-crystals.org/dilithium/data/dilithium-specificationround3-20210208.pdf. [Accessed Sept. 11, 2023].
[25] R. Avanzi, J. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. M. Schanck, P. Schwabe, G. Seiler, and D. Stehlé, ''CRYSTAL-KYBER: Algorithm Specifications and Supporting Documentation; Version 3.02,'' 2021. Available: https://pq-crystals.org/kyber/data/kyber-specification-round3-20210804.pdf. [Accessed Sept. 11, 2023].
[26] P. Grubbs, V. Maram, and K. G. Paterson. "Anonymous, robust post-quantum public key encryption," in *Advances in Cryptology - EUROCRYPT 2022*, O. Dunkelman and S. Dziembowski, Eds. Lecture Notes in Computer Science, Springer, Cham. 2022.
[27] P. Farshim, C. Orlandi, and R. Roşie, "Security of symmetric primitives under incorrect usage of keys," *IACR Transactions on Symmetric Cryptology*, vol. 2017, no. 1, pp. 449-473, 2017.
[28] V. Maram and K. Xagawa. "Post-quantum anonymity of Kyber," in *IACR International Conference on Public-Key Cryptography*, Cham: Springer Nature Switzerland, 2023, pp. 3-35.
[29] https://www.pqshield.com. [Accessed Sept. 11, 2023].
[30] Post-Quantum signatures zoo. Available: https://pqshield.github.io/nist-sigs-zoo/wide.html. [Accessed Sept. 11, 2023].
[31] Falcon - An Update (nist.gov). https://csrc.nist.gov/CSRC/media/Presentations/falcon-round-2-presentation/images-media/falcon-prest.pdf. [Accessed Sept. 11, 2023].
[32] M. Raavi, S. Wuthier, P. Chandramouli, Y. Balytskyi, X. B. Zhou, and S.-Y. Chang, "Security comparisons and performance analyses of post-quantum signature algorithms," in International Conference on Applied Cryptography and Network Security, Cham: Springer International Publishing, 2021, pp. 424-447.