OXFORD JOURNAL
OF STUDENT
SCHOLARSHIP

Quantum-Resistant Cryptography and Its Implications for Blockchain and Cryptocurrency: A Comprehensive Mathematical Analysis

Tehzeeb Ali Gows Basha
Independent Researcher, Dubai, United Arab Emirates
bashatehzeebali@gmail.com

ABSTRACT

Modern public key cryptosystems rely on two assumptions regarding computational difficulty: the integer factorization problem (RSA) and the discrete logarithm problem (elliptic curve cryptography). These problems have stood the test of time for four decades of cryptanalysis. Yet, their algebraic and periodic nature makes them vulnerable to quantum algorithms, most notably Shor's algorithm (1994). This paper presents a detailed comparison between classical and quantum-resistant cryptographic systems with an emphasis on lattice-based systems. The Learning With Errors (LWE) problem and its variants (Ring-LWE, Module-LWE) do not have the same periodicity that makes problems like integer factorization and discrete logarithms susceptible to quantum algorithms. Furthermore, LWE problems have reductions from worst-case lattice problems and have mathematical proofs of quantum resistance. Due to the vulnerability of ECDSA algorithms for cryptocurrencies, it is estimated that current cryptocurrencies will no longer be secure once quantum computer technology matures within the next 10–30 years, potentially compromising holdings estimated at between $260 billion and $390 billion as of August 2026. The implications of cryptanalysis on public key cryptography systems motivate mathematical recommendations regarding the security of current cryptosystems and the need for a shift to post-quantum cryptographic standards.

Keywords: Post-Quantum Cryptography, Lattice-Based Cryptography, Modular Arithmetic, Elliptic Curves, Discrete Logarithms, Learning With Errors, Blockchain Security

I. INTRODUCTION

1.1 - Mathematical Foundations of Classical Cryptography

Modern cryptography developed out of three major discoveries in mathematics. The discovery of methods of testing and creating large prime numbers was made in the 1970s. Upon the discovery of large primes, Diffie and Hellman published a paper in 1976 describing a method of securely exchanging keys between

two parties over a public network (Diffie & Hellman, 1976). Finally, in 1977, Rivest, Shamir, and Adleman developed a system that used the properties of large prime numbers to allow for the encryption and secure transmission of data between two parties (Rivest et al., 1978).

Each of these systems depend upon a type of mathematical problem that is considered to be difficult to solve, but easy to perform by those who are supposed to be able to solve such problems. RSA encryption, for instance, is based upon the fact that it is difficult to factor large semiprimes (semiprimes are numbers that are the products of two primes). Elliptic curve cryptography is based upon the idea that it is difficult to solve the discrete logarithm problem on elliptic curves. Both of these types of cryptographic systems perform their operations within finite fields. RSA functions within the ring of integers modulo n, represented as $\mathbb{Z}_n$, and its unit group $\mathbb{Z}_n^*$. Elliptic curve cryptography functions within finite fields of points that lie on an elliptic curve.

1.2 - The Quantum Threat

Within the development of quantum computing, it has become evident that many of the cryptographic protocols used today may soon be exposed to an existential threat if the cryptographic protocols are not soon modified to account for the power of quantum computers. The algorithm known as Shor's algorithm, for instance, can be used to solve the discrete logarithm problem in polynomial time on a quantum computer (Shor, 1994). Thus, any cryptographic system whose security is based upon the idea of the intractability of the discrete logarithm problem (such as RSA encryption and elliptic curve cryptography) may be broken by a sufficiently advanced quantum computer employing Shor's algorithm.

1.3 - Research Objectives and Scope

The goal of this research project is to provide a thorough comparison between classical and quantum-resistant cryptographic systems through the performance of mathematical derivations and analyses of each system. As a result of these analyses, it will be evident to the readers of this report that some of these classical cryptographic systems are secure within the quantum world, but others are not. Furthermore, blockchain technology will be covered within this report as well, which will enable the readers of this report to become aware of the types of cryptographic systems based upon mathematical proofs that should be utilized in the future. Thus, the audience for this report consists of those who wish to gain a thorough understanding of the mathematical foundations of both classical and quantum-resistant cryptographic systems.

II. CLASSICAL CRYPTOGRAPHIC MATHEMATICS

2.1 - Modular Arithmetic

The majority of cryptographic systems are based upon operations that occur within modular arithmetic within finite algebraic structures. More specifically, the operations of cryptographic systems are often defined within mathematical groups. For integers a, b, n with $n > 0$, a is congruent to b modulo n, written $a \equiv b \pmod{n}$, if n divides the difference between a and b, that is, if n divides the quantity $a - b$.

The multiplicative group modulo n consists of all integers a within the range of 1 to $n-1$ that are relatively

prime to n:

$$\mathbb{Z}_n^* = \{\, a \in \mathbb{Z}_n : \gcd(a,n) = 1 \,\}$$

This is a group that defines an operation as the multiplication of each element of the group taken modulo n.

2.2 - RSA Cryptosystem

2.2.1. Mathematical Foundation

The RSA cryptosystem is based upon Euler's theorem:

Theorem 2.1. Let a and n be integers such that $\gcd(a,n) = 1$. Then

$$a^{\varphi(n)} \equiv 1 \pmod{n}$$

where $\varphi(n)$ is Euler's totient function.

2.2.2. Key Generation

To generate the keys for the RSA system, the following steps are performed:

1. Choose two different prime numbers p and q.
2. Calculate the value of n as the product $p \cdot q$, and calculate the totient function as $\varphi(n) = (p-1)(q-1)$.
3. Choose an integer e such that $1 < e < \varphi(n)$ and such that $\gcd(e, \varphi(n)) = 1$.
4. Calculate the integer d such that $d \equiv e^{-1} \pmod{\varphi(n)}$.

The public key for the RSA system is the key pair consisting of n and e. The private key for the RSA system is the key pair consisting of n and d.

2.2.3. Encryption and Decryption

To encrypt data within the RSA cryptosystem, the message m is raised to the power of e, with all operations performed modulo n:

$$c \equiv m^e \pmod{n}$$

To decrypt data within the RSA cryptosystem, the ciphertext c is raised to the power of d, with all operations performed modulo n:

$$m \equiv c^d \pmod{n}$$

To prove that RSA is correct, the definition of d must be utilized. Since d is the multiplicative inverse of e modulo $\varphi(n)$, there exists an integer k such that

$$ed = 1 + k\varphi(n)$$

Raising the ciphertext to the power d therefore yields m^{ed}, which may be rewritten as

$$m^{ed} = m^{1+k\varphi(n)} = m \cdot \left(m^{\varphi(n)}\right)^k$$

By Euler's theorem, $m^{\varphi(n)} \equiv 1 \pmod{n}$, so this reduces to $m \cdot 1^k \equiv m \pmod{n}$. Decryption therefore returns the original message m.

2.3 - Elliptic Curve Cryptography

2.3.1. Elliptic Curve Definition

An elliptic curve is represented by the following equation:

$$y^2 = x^3 + ax + b$$

where a, b are elements within a finite field $\mathbb{F}_p$ with p being a prime number greater than 3. Furthermore, the non-singularity condition requires that

$$4a^3 + 27b^2 \not\equiv 0 \pmod{p}$$

2.3.2. Point Addition

The set of points that lie along an elliptic curve, together with a point at infinity, creates an abelian group whose operation is point addition. For two distinct points (x_1, y_1) and (x_2, y_2) on the curve, the point that results from adding them has coordinates defined according to the following equations:

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1} \pmod{p} \tag{1}$$

$$x_3 = \lambda^2 - x_1 - x_2 \pmod{p} \tag{2}$$

$$y_3 = \lambda(x_1 - x_3) - y_1 \pmod{p} \tag{3}$$

The point that is created is represented by the coordinates (x_3, y_3). If the two points that are being added together are identical to one another, the value of λ is instead given by

$$\lambda = \frac{3x_1^2 + a}{2y_1} \pmod{p}$$

2.3.3. The Elliptic Curve Discrete Logarithm Problem

In elliptic curve cryptography, one of the problems that must be solved is the elliptic curve discrete logarithm problem (ECDLP):

Definition 2.1. Let P and Q be two points that are defined within an elliptic curve. The elliptic curve discrete logarithm problem is the problem of determining the integer k such that $Q = kP$.

The elliptic curve discrete logarithm problem is considered to be a difficult problem to solve within classical computing algorithms; the best-known algorithms require on the order of $\sqrt{n}$ operations to solve the discrete logarithm problem in elliptic curves, where n is the order of the elliptic curve group.

2.4 - The Discrete Logarithm Problem

In the finite field $\mathbb{F}_p$ where p is a prime number, let G be the multiplicative group whose elements have order n, where n is a divisor of $p-1$. Additionally, let g be a generator for that group.

Definition 2.2. Let G be a finite cyclic group of order n with a generator g. The discrete logarithm problem is, given an element h within G, to determine the integer x such that $g^x = h$.

Discrete logarithm problems are utilized within cryptographic systems such as the Diffie-Hellman and El-Gamal systems.

III. QUANTUM COMPUTING FUNDAMENTALS AND CRYPTOGRAPHIC VULNERABILITY

3.1 - Quantum Computing Basics

3.1.1. Quantum Bits (qubits) and Superposition

A qubit is a unit of information that can exist in a state of superposition. Thus, a qubit can be in a state of 0, 1, or both at the same time. A qubit can be represented as a linear combination of its basis states, written as $\alpha|0\rangle + \beta|1\rangle$, where α and β are complex amplitudes satisfying the condition that the sum of their squared moduli equals one:

$$|\alpha|^2 + |\beta|^2 = 1$$

3.1.2. Quantum Gates

Quantum gates are the operators that act upon the quantum bits within a quantum computer. A few of the more common quantum gates include the Hadamard gate, the Pauli-X gate (also known as the NOT gate), the CNOT gate, and the SWAP gate.

$$\text{Hadamard gate: } H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

$$\text{Pauli-X gate: } X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

$$\text{CNOT gate: CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

3.2 - Shor's Algorithm

3.2.1. Overview

The most common algorithm for factoring integers, known as Shor's algorithm, can factor any integer in time $O\left((\log N)^3\right)$. The algorithm is based upon the use of quantum Fourier transforms to perform the factorization, and is performed according to the following steps:

1. Choose a random integer a within the range of integers between 1 and $N-1$ inclusive such that $\gcd(a,N)=1$.
2. Determine the period r of the function $f(x)=a^x \pmod N$ with the use of quantum Fourier transforms.
3. If the calculated value for r is even, and if $a^{r/2} \not\equiv -1 \pmod N$, calculate the integers $\gcd(a^{r/2}+1,N)$ and $\gcd(a^{r/2}-1,N)$. Each of these integers will be a non-trivial factor of the original number N.

3.2.2. Mathematical Foundation

The algorithm relies upon the fact that factoring integers can be reduced to the problem of determining the period of a function. Thus, within the quantum computer, quantum Fourier transforms are utilized to determine that period. For integer N, choose a value for Q as a power of two, $Q=2^q$, such that

$$N^2 \leq Q < 2N^2$$

The initial state of the quantum computer is the state:

$$|\psi\rangle = \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x\rangle|0\rangle$$

Apply the function $f(x)=a^x \pmod N$ to both registers of the quantum computer. Thus, the new state of the quantum computer is:

$$\frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x\rangle \, |a^x \bmod N\rangle$$

Measure the second register of the quantum computer. The state of the first register will collapse to a superposition of values consistent with the period r. Apply a quantum Fourier transform to the first register and measure the first register again. The value of the measurement permits the period r to be recovered.

3.2.3. Complexity

The complexity of classical integer factoring algorithms is generally of the order

$$\exp\left((\log n)^{1/3}(\log\log n)^{2/3}\right)$$

The complexity of Shor's quantum factoring algorithm is $O\left((\log n)^3\right)$.

3.3 - Elliptic Curve Cryptography

In addition to the factoring problem, Shor's algorithm can also be used to solve the discrete logarithm problem on elliptic curves. Thus, the elliptic curve cryptosystems that rely upon the difficulty of solving such a discrete logarithm problem can also be broken by a quantum computer that utilizes Shor's algorithm to solve the problem (Shor, 1994). The elliptic curve discrete logarithm problem can be solved in time $O\left((\log n)^3\right)$, where n is the order of the elliptic curve group.

IV. QUANTUM-RESISTANT CRYPTOGRAPHIC MATHEMATICS

4.1 - Lattice Theory Foundations

Definition 4.1 (Lattice). A lattice within $\mathbb{R}^n$ is a discrete subgroup that is generated by a set of n linearly independent vectors within the same space.

Thus, a lattice L can be represented as the set of all integer linear combinations of the n linearly independent basis vectors $b_1, b_2, \ldots, b_n$:

$$L = \left\{ \sum_{i=1}^{n} z_i b_i \;\middle|\; z_i \in \mathbb{Z} \right\}$$

The matrix that is constructed from the basis vectors within the lattice is referred to as the basis matrix of that lattice.

4.1.1. Fundamental Lattice Problems

Definition 4.2 (Shortest Vector Problem (SVP)). Given a lattice within $\mathbb{R}^n$, find the non-zero vector within that lattice whose length is the shortest.

Definition 4.3 (Closest Vector Problem (CVP)). Given a lattice within $\mathbb{R}^n$, and a target vector in that same space, find the vector within the lattice whose distance to the target vector is the shortest.

The Shortest Vector Problem has been proven to be NP-hard under randomized reductions in classical computation [1]. Furthermore, no known quantum algorithm solves the SVP in polynomial time.

4.2 - Learning With Errors (LWE) Problem

4.2.1. Problem Definition

Definition 4.4. Let n, q be positive integers, and let χ be some distribution of values over the integers modulo q. Furthermore, let s be a secret vector. The Learning With Errors problem concerns a distribution of pairs (a, b) where a is a vector of n dimensions whose entries are elements of $\mathbb{Z}_q$, and where

$$b = \langle a, s \rangle + e \pmod{q}$$

with the error e drawn from the distribution χ. The Search-LWE problem is the problem of finding s given enough samples of the distribution of pairs (a, b).

The Decision-LWE problem is the problem of distinguishing samples generated by the Search-LWE distribution from pairs (a, b) sampled uniformly at random from $\mathbb{Z}_q^n \times \mathbb{Z}_q$.

4.2.2. Error Distribution

One of the most common choices of distribution for the error e in the LWE problem is the discrete Gaussian distribution $D_{\mathbb{Z},\sigma}$, which has a probability distribution function of:

$$D_{\mathbb{Z},\sigma}(x) = \frac{\exp\left(-\pi x^2/\sigma^2\right)}{\sum_{z \in \mathbb{Z}} \exp\left(-\pi z^2/\sigma^2\right)}$$

4.3 - Security Reduction: LWE to Lattice Problems

Theorem 4.1. Let n, q be positive integers such that q is prime and $q \geq 2\sqrt{n}$. Furthermore, let α be a positive real number less than 1 such that $\alpha q \geq 2\sqrt{n}$. Then, if there is an algorithm that can solve the Decision-LWE problem with non-negligible probability relative to the size of its inputs, there is an algorithm that can solve worst-case instances of the lattice problems $\text{GapSVP}_{\tilde{O}(n/\alpha)}$ and $\text{SIVP}_{\tilde{O}(n/\alpha)}$ [9].

4.4 - Ring-LWE

Ring-LWE (Learning With Errors over Rings) was developed as a means of improving the efficiency of the LWE problem. Thus, Ring-LWE shares many of the same features as the LWE problem, both in form and function, but has a structure that permits it to be implemented more efficiently than the original LWE problem.

Definition 4.5 (Ring-LWE). Let R be the ring of polynomials of degree less than n with integer coefficients, denoted $R = \mathbb{Z}[x]/(x^n + 1)$. Furthermore, let R_q be the set of polynomials in R whose coefficients are elements of $\mathbb{Z}_q$, that is, $R_q = R/qR$. Ring-LWE is defined in the same way as the LWE problem, but within this ring R_q: samples take the form (a, b) where a is within R_q, b is the product of a and a secret element s within R_q plus an error term, and the error is distributed according to the error distribution over the ring R.

Ring-LWE offers significant improvements to the efficiency of the algorithm relative to LWE while maintaining comparable security guarantees.

4.5 - Module-LWE

Module-LWE is a generalization of both the LWE and the Ring-LWE algorithms.

Definition 4.6 (Module-LWE). Let R_q be the ring of polynomials of degree less than n with coefficients in $\mathbb{Z}_q$, that is, $R_q = \mathbb{Z}_q[x]/(x^n + 1)$. Furthermore, let d be any positive integer greater than 1. Then Module-LWE is defined as a distribution of samples of the form (a, b) wherein a is within R_q^d, the set of d-dimensional vectors whose entries are elements of R_q; s is a secret vector within R_q^d; and

$$b = \langle a, s \rangle + e$$

with the error e distributed according to the error distribution over the ring R.

Module-LWE provides a generalization of both the LWE and Ring-LWE algorithms that permits a balance to be struck between security and efficiency.

V. MATHEMATICAL COMPARISON: CLASSICAL VS. QUANTUM-RESISTANT

5.1 - Algebraic Structure Analysis

Table 1 presents a comparison of the classical and quantum-resistant cryptographic systems. The classical systems, such as RSA and ECDSA, rely upon the structure of cyclic groups and the difficulty of the discrete logarithm problem within those groups. The quantum-resistant systems, such as the lattice-based algorithms, utilize n-dimensional lattices instead.

5.2 - Why Quantum Algorithms Fail on Lattices

Table 1: Comparison of classical and quantum-resistant cryptographic systems

Property	Classical	Lattice-Based
Mathematical structure	Group theory	Lattice theory
Hard problem	Factoring / DLP	LWE / SVP
Best classical attack	Subexponential	Exponential
Best quantum attack	Polynomial (Shor)	Exponential (Grover)
Key size	256 bits	1–3 KB

5.2.1. Absence of Periodicity

Shor's algorithm finds the period of the function:

$$f(x) = a^x \pmod{N}$$

where $a^r \equiv 1 \pmod{N}$ for some value of r. No such period exists for problems like the Learning With Errors problem:

$$f(a) = \langle a, s \rangle + e \pmod{q}$$

5.2.2. Worst-Case to Average-Case Reduction

Lattice-based cryptography rests on a connection that classical cryptography lacks: there is a proof that an efficient algorithm for solving randomly chosen average-case instances of the problem can be converted into an algorithm that solves worst-case instances of the corresponding lattice problem for any lattice.

Theorem 5.1 (Worst-Case Hardness). Suppose an attacker can solve instances of the average-case Learning With Errors problem with probability greater than $\frac{1}{2} + \varepsilon$ for some constant $\varepsilon > 0$. Then the same algorithm can solve instances of the worst-case GapSVP problem with approximation factor $\gamma = \tilde{O}(n/\alpha)$. The function $\tilde{O}(\cdot)$ hides factors that are polynomial in n.

5.3 - Security Parameter Analysis

The security parameters of the two families of systems are best compared against the classical security levels they are intended to replace. RSA at a 2048-bit modulus provides approximately 112 bits of classical security, and the secp256k1 curve used by Bitcoin provides approximately 128 bits. Both figures collapse under Shor's algorithm, since the underlying problems move from subexponential or exponential difficulty to polynomial time.

The lattice-based systems are parameterised against the NIST security levels instead. ML-KEM-512, ML-KEM-768 and ML-KEM-1024 target levels 1, 3 and 5 respectively, while the Dilithium parameter sets target levels 2, 3 and 5. The best known attacks against Module-LWE proceed by lattice reduction, typically the BKZ algorithm, whose cost grows exponentially in the block size required to recover the secret. Grover's algorithm offers only a quadratic speedup against unstructured search and therefore reduces the effective security by at most half the exponent, which is accommodated by the parameter choices rather than defeating them.

The cost of this security is borne in the size of the objects involved. The ML-KEM-768 encapsulation key occupies 1,184 bytes and its ciphertext 1,088 bytes, against the 33-byte compressed public key of secp256k1. The Dilithium signature figures given in section 8.1.1 show the same pattern. This size penalty, rather than any weakness in the underlying hardness assumption, is the principal obstacle to deployment.

VI. POST-QUANTUM CRYPTOGRAPHY STANDARDS

6.1 - NIST Standardization Process

The National Institute of Standards and Technology (NIST) is creating cryptographic standards that are secure from attacks by quantum computers. The process began in 2016 with the goal of having the standards published by 2024 [7].

6.2 - Selected Algorithms

6.2.1. CRYSTALS-Kyber (FIPS 203)

CRYSTALS-Kyber, standardized by NIST as ML-KEM, is a key encapsulation mechanism whose security is based upon the Module-LWE problem. This system uses the following parameters:

- $n = 256$
- $q = 3329$
- $k \in \{2,3,4\}$

The process for key generation is as follows:

1. Sample a matrix A of size $k \times k$ with elements uniformly distributed in R_q.
2. Sample a secret vector s with entries sampled from a centered binomial distribution.
3. Sample an error vector e with entries sampled from a centered binomial distribution.
4. Calculate $t = As + e \pmod{q}$.

The public key is (A,t) and the secret key is s.

6.2.2. CRYSTALS-Dilithium (FIPS 204)

CRYSTALS-Dilithium is a digital signature system whose security is based upon the Module-LWE problem. For each signature:

1. Sample a random value y.
2. Calculate $w = Ay$.
3. Calculate the challenge value $c = H(w,M)$ where H is a cryptographic hash function and M is the message being signed.
4. Calculate the signature $z = y + cs$.
5. If the sizes of z or $w - ct$ are outside some pre-defined bounds, repeat steps 1 to 4 until the signature

is accepted.

6.2.3. FALCON (draft FIPS 206)

FALCON is another digital signature system, selected by NIST for standardization as FN-DSA and covered by draft FIPS 206. FIPS 205 covers a different scheme, the hash-based SLH-DSA derived from SPHINCS+. Unlike the schemes based upon the Module-LWE problem, FALCON uses the NTRU lattice problem and the GPV framework for digital signatures. Fast Fourier sampling is used in its creation of the digital signatures.

6.3 - Security Levels

The standards provide for three security levels, each of which is roughly equivalent to a certain level of security in classical cryptography:

- Level 1: comparable to AES-128
- Level 3: comparable to AES-192
- Level 5: comparable to AES-256

VII. IMPLICATIONS FOR BLOCKCHAIN AND CRYPTOCURRENCY

7.1 - Current Cryptographic Infrastructure

7.1.1. Bitcoin

Bitcoin uses the ECDSA algorithm with the secp256k1 elliptic curve for digital signatures. Given the public key P and the private key d, the signature for a message m is created with the following equations:

$$r = (kG)_x \pmod{n}$$

$$s = k^{-1}(H(m) + rd) \pmod{n}$$

where k is a random value, G is the generator point of the elliptic curve, m is the message being signed, and n is the order of the group generated by G. Should an attacker obtain the public key for a Bitcoin address, they could use Shor's algorithm to mathematically recover the private key associated with that address.

7.1.2. Ethereum

Like Bitcoin, Ethereum utilizes the elliptic curve and ECDSA to create digital signatures for its transactions. Instead of using the public key of an address as its address on the Ethereum blockchain, Ethereum calculates the Keccak-256 hash of the public key to form the address of the account.

7.2 - Threat Timeline

Table 2 presents estimated technological developments in quantum computers. Beyond some threshold number of qubits in a quantum processor, those quantum computers will be able to pose a threat to the public key cryptography currently in use. There is no exact date for when cryptographically relevant quantum computers will be available, but various estimates exist.

Table 2: Quantum computing development timeline

Year	Technology
2019–2023	Google announced that it had achieved quantum supremacy with a 53-qubit processor. IBM announced the Condor processor with 1,121 qubits.
2025–2030	Estimates for the emergence of cryptographically relevant quantum computers (CRQCs).
2030–2035	Estimates for when such quantum computers will pose a widespread threat to the public key cryptography currently in use.

7.3 - Attack Scenarios

7.3.1. Store-Now-Decrypt-Later (SNDL)

In this type of attack, an adversary records public keys and ciphertexts that are exposed on the blockchain today and stores them, with the intention of recovering the corresponding private keys once a cryptographically relevant quantum computer becomes available. While such recovery is not currently possible, the exposure of public keys on the blockchain creates the conditions for the attack to succeed later.

7.3.2. Direct Key Recovery

Once an attacker gains access to a quantum computer, those public keys stored on the blockchain can be directly attacked in order to recover the private keys associated with those public keys. Estimates suggest that this type of attack will take between hours and days to execute for cryptocurrencies with 256-bit keys. Thus, those addresses associated with exposed public keys will be at risk of having their holdings compromised.

7.4 - Economic Impact Assessment

All figures in this section are stated as of 14 August 2026 and should be read as a dated snapshot rather than as current values, since market capitalisations move continuously.

As of that date, the market capitalisation of Bitcoin stands at approximately $1.3 trillion and that of Ethereum at approximately $233 billion, with Bitcoin representing roughly 57% of an aggregate cryptocurrency market of approximately $2.3 trillion. Public keys already exposed on the Bitcoin blockchain correspond to an estimated 20–30% of the total supply. A further estimated 15–20% of the supply is held at addresses whose private keys are lost or inaccessible, and those holdings cannot be migrated to a new signature scheme by their owners under any circumstances.

Applying the exposure estimate to Bitcoin's capitalisation on the stated date places the value directly at risk of quantum key recovery at approximately $260 billion to $390 billion. The proportion matters more than the absolute figure: because the exposure is a fraction of supply rather than a fixed sum, the quantity at risk scales with the market and the estimate above should be recomputed against the capitalisation prevailing at the time of reading.

7.5 - Migration Strategies

7.5.1. Hard Fork

A hard fork of the blockchain will require the creation of a new version of the protocol. Such a protocol will require:

- A new cryptographic algorithm
- A new protocol version
- A defined migration period for moving holdings from the old protocol to the new one
- Freezing of the addresses of those who do not participate in the protocol change

One of the challenges to implementing such a protocol will be the decentralized nature of the blockchain. Furthermore, many of the public keys in use on the blockchain do not have their associated private keys available. Other challenges will be the need to create the new protocol in a way that is backwards compatible with the existing protocol, as well as to account for the increased sizes of transactions that will need to be created under the new cryptographic algorithm.

7.5.2. Hybrid Cryptography

An alternative to creating a hard fork in the blockchain is the implementation of a hybrid cryptographic system. Such a system will utilize both classical and post-quantum cryptographic algorithms for the creation of digital signatures, so that a transaction carries both an ECDSA signature and a signature based upon the Module-LWE problem. The advantage of such a system is that even if the classical cryptographic system is broken, the transactions remain secure. Furthermore, retaining the existing signature workflow limits the changes required elsewhere in the protocol. The disadvantages include the roughly doubled size of digital signatures and the increased time required to verify each signature and transaction.

VIII. IMPLEMENTATION CASE STUDY

8.1 - Post-Quantum Bitcoin Prototype

8.1.1. Design Specifications

The design of a post-quantum version of Bitcoin will utilize the CRYSTALS-Dilithium digital signature system at NIST security level 2. The public keys will be of size 1,312 bytes, compared to 33 bytes in ECDSA; the digital signatures will be of size 2,420 bytes, compared to 71 bytes in ECDSA; and the time required for signature verification will be around 200 microseconds, compared to 50 microseconds for ECDSA.

8.1.2. Performance Analysis

Currently, the size of a typical Bitcoin transaction is around 250 bytes, of which the 33-byte compressed public key and the 71-byte ECDSA signature account for approximately 104 bytes. Substituting the CRYSTALS-Dilithium public key of 1,312 bytes and signature of 2,420 bytes replaces those 104 bytes with 3,732 bytes, a net increase of approximately 3,628 bytes. The resulting transaction is therefore approximately 3,900 bytes, or roughly 15 times its present size.

The consequence for throughput follows directly. A block that currently accommodates between 2,000 and 2,500 transactions would accommodate only approximately 130 to 160 transactions under the post-quantum scheme, assuming no change to the block size limit and no use of signature aggregation.

8.1.3. Scalability Solutions

Some of the possible solutions to the scalability issue of Bitcoin will be to:

- Increase the size of the block
- Implement signature aggregation algorithms
- Develop Layer-2 solutions such as the Lightning Network
- Implement optimized implementations of the post-quantum protocol

8.2 - Ethereum Post-Quantum Considerations

8.2.1. Smart Contract Implications

The implications of the shift to digital signatures based upon the Module-LWE problem will include changes to the way that smart contracts verify digital signatures of transactions. In addition, the construction of multi-signature wallets will change, as will the gas costs for smart contracts to verify digital signatures. Currently, smart contracts require around 3,000 units of gas to verify a digital signature. In contrast, the CRYSTALS-Dilithium digital signature system will require between 100,000 and 500,000 units of gas for signature verification.

8.2.2. Protocol Upgrades

The protocol for Ethereum will have to be changed to include support for the new digital signatures. For instance, the protocol can introduce a transaction type that uses these new signatures. Furthermore, the account abstraction system (EIP-4337) can be adapted to handle them. Finally, new precompiles can be created to speed the verification of these signatures.

IX. DISCUSSION

9.1 - Mathematical Foundations

The main goal of this research was to investigate some of the differences between classical cryptographic systems and post-quantum cryptographic systems. As discussed above, classical cryptographic systems mainly work within the context of cyclic groups, both finite fields and elliptic curves, and the discrete logarithm problem that exists within those groups. In contrast, the mathematical framework for lattice-based cryptographic systems is based upon n-dimensional Euclidean spaces within which there is no periodicity to exploit.

Another of the main differences between classical cryptographic systems and lattice-based cryptographic systems is the link between worst-case and average-case problems for those systems. When using lattice-based cryptographic systems, an algorithm that is efficient at solving average-case instances will also be

efficient at solving the worst-case instance of the same problem. However, classical cryptographic systems rely on the assumption that randomly chosen instances are hard, without any proof connecting that average-case hardness to worst-case hardness.

Finally, the reason that no efficient quantum algorithms have yet been created for lattice-based cryptographic systems lies in these mathematical differences. The cyclic groups within which classical cryptographic systems operate have an inherent periodicity, which allows quantum algorithms like Shor's algorithm to efficiently find the period and solve the problem. The lack of periodicity within lattice problems explains the inability of current quantum algorithms to solve those problems efficiently.

9.2 - Performance

One of the main findings of this research is that post-quantum cryptographic systems are significantly less performant than classical cryptographic systems. For instance, public keys based upon quantum-resistant cryptographic systems will be 40–50 times larger than those created by the ECDSA algorithm for Bitcoin. Additionally, digital signatures will be 30–40 times larger in size. Furthermore, the computational costs of using these new systems will be 2–10 times greater than those of classical cryptographic systems. Despite these disadvantages, these systems remain acceptable for use given the existential threat that quantum computers pose to current cryptographic systems.

9.2.1. Implementation Challenges

Beyond the performance differences between the two types of cryptographic systems, there are other challenges to the implementation of these quantum-resistant cryptographic systems in cryptocurrencies like Bitcoin and Ethereum. Some of these challenges include:

- Bandwidth requirements for the larger public keys and digital signatures
- Storage requirements for the blockchain to store the larger transactions
- Increased computational requirements for the blockchain nodes to verify digital signatures
- Backwards compatibility with existing cryptocurrencies

9.3 - Timeline Urgency

Given the current rate of development of quantum computers, estimates suggest it will take between 10 and 30 years for those quantum computers to reach the number of qubits that will allow them to pose a threat to current cryptographic systems [7]. Thus, while it may appear that no action is required in the present, such action is imperative. Should quantum computers develop at a faster rate than these current estimates, the urgency will be greater still.

9.4 - Future Research

Beyond the research that has already been performed on quantum-resistant cryptography, there are a few suggestions for future research in this area:

- Optimized implementations of the proposed cryptographic systems

- Development of other types of hybrid cryptographic schemes
- Development of techniques to aggregate digital signatures to reduce the overhead of using the proposed post-quantum cryptographic algorithms
- Exploration of other types of cryptographically hard problems
- Development of quantum-resistant smart contracts

X. CONCLUSION

This research presents a detailed and comprehensive analysis of the mathematical reasons that classical cryptographic systems fail against quantum attacks, while also demonstrating the resilience of lattice problems against such attacks. The main mathematical difference between classical and quantum-resistant systems is the structure of the problems that are utilized in those systems. While classical cryptography makes use of cyclic groups that have a periodic structure targeted by quantum algorithms, lattice problems do not have this same structure.

Furthermore, the fact that breaking the average case of a lattice problem implies breaking the worst case of that problem provides a solid theoretical foundation for the security of these cryptographic systems.

For cryptocurrencies like Bitcoin and Ethereum, such a threat poses a serious and approaching danger. As digital signatures that are based on elliptic curves are expected to be broken by quantum computers within 10 to 30 years, the holdings of affected users will be at risk of compromise. Thus, though at the cost of increased sizes of keys and signatures and higher computational costs, it is necessary to implement cryptographic systems that are secure against quantum computers.

It is therefore imperative that each of the cryptocurrencies create plans to implement post-quantum cryptographic systems such as Kyber, Dilithium, and FALCON. Such plans may include the creation of hybrid cryptographic systems to provide security under both the classical and the post-quantum settings. Furthermore, scalability solutions will need to be created to combat any decrease in the throughput of the blockchains. Finally, it will also be necessary to create timelines for such an upgrade, as well as to perform security audits on the proposed systems prior to their implementation.

Overall, the mathematical foundations of these systems are solid, the algorithms are standardized, and the threat timeline is reasonably well understood. The blockchain and cryptocurrency communities must take action to ensure the long-term security of their systems against quantum computers.

ACKNOWLEDGMENTS

The author acknowledges the foundational work of researchers in post-quantum cryptography, particularly the contributions of Oded Regev, Chris Peikert, and the NIST Post-Quantum Cryptography project team.

REFERENCES

[1] Ajtai, M. (1998). The shortest vector problem in L_2 is NP-hard for randomized reductions. *Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, 10–19.

[2] Bernstein, D. J., & Lange, T. (2017). Post-quantum cryptography. *Nature*, 549(7671), 188–194.

[3] BTQ Technologies. (2025). *BTQ Technologies announces quantum-safe Bitcoin using NIST-standardized post-quantum cryptography*.

[4] Chainalysis. (2023). *Quantum computing and crypto security*.

[5] Chen, Y. (2024). *Polynomial-time quantum algorithm for the principal ideal problem and applications to cryptanalysis*. Preprint.

[6] Diffie, W., & Hellman, M. (1976). New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6), 644–654.

[7] National Institute of Standards and Technology. (2024). *Federal Information Processing Standards Publications 203, 204 and 205: Post-Quantum Cryptography Standards*.

[8] Peikert, C. (2014). Lattice cryptography for the internet. *IACR Cryptology ePrint Archive*.

[9] Regev, O. (2005). On lattices, learning with errors, random linear codes, and cryptography. *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, 84–93.

[10] Rivest, R. L., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2), 120–126.

[11] Shor, P. W. (1994). Algorithms for quantum computation: Discrete logarithms and factoring. *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, 124–134.

[12] The Quantum Insider. (2024). *Blockchain and quantum computing are on a collision course*.

[13] TII Insights. (2024). *Navigating the quantum frontier: Arrival of NIST's first post-quantum cryptography standards*.