10 Post Quantum Security

Mrs. Sonali Amol Parkhi*
Associate Professor, Information Technology Wing, College of Military Engineering, Dapodi, Pune.
*Corresponding Author: sonaliparkhi21@gmail.com

Abstract

The field of cryptography supports the security of global digital infrastructures, from securing government communications to protecting individual privacy in online transactions. For decades, classical cryptographic systems such as RSA, Elliptic Curve Cryptography (ECC), and Advanced Encryption Standard (AES) have provided reliable protection based on the computational complexity of certain mathematical problems. However, the rapid advancement of quantum computing threatens to undermine these foundational assumptions. Quantum algorithms, most notably Shor's and Grover's, promise to break widely deployed public-key systems and weaken symmetric-key security. This emerging threat has given rise to the field of post-quantum cryptography (PQC), which seeks to develop cryptographic algorithms resilient to attacks from quantum adversaries.

Keywords: Quantum Security, AES, RSA, ECC, PQC.

Introduction

A cryptographically relevant quantum computer could decrypt previously intercepted data, forge digital signatures, and compromise secure channels in critical sectors like finance, defense, and healthcare. Any entity storing encrypted data for the long term is at risk. [1] For instance, governments retain diplomatic cables, military communications, and intelligence briefings. Financial institutions archive transaction logs and client records, while healthcare providers store decades of medical histories. This could result in long term espionage, identity theft, and loss of competitive advantage.

The quantum threat extends to critical infrastructure, where compromised cryptographic protocols could disrupt energy grids, water systems, transportation networks, and financial platforms. These are not just data breaches; they are operational failures that could trigger cascading crises. Military systems are particularly vulnerable, as quantum-enabled adversaries could decrypt battlefield communications, expose defense strategies, and undermine national security. Cyber warfare will evolve as quantum computing accelerates. A nation with quantum superiority could bypass traditional cybersecurity defenses, escalate cyber conflicts, and gain an overwhelming strategic advantage. [2] To prevent this, governments and defense contractors must urgently deploy quantum-resistant cryptographic solutions for mission-critical systems. Enterprises and multinational corporations also face severe consequences. Intellectual property blueprints, formulas, source code, and strategic documents becomes accessible to competitors and cybercriminals. A quantum-enabled breach could erode market dominance, expose trade secrets, and cause irreparable financial damage.

- **Limitations of Classical Cryptography**

Classical cryptographic algorithms derive their security from problems such as integer factorization (RSA) and the discrete logarithm (ECC, Diffie–Hellman). While these problems are hard for classical computers, quantum computers can solve them efficiently in polynomial time using Shor's algorithm. Symmetric cryptography, while more robust, also sees its effective security reduced by Grover's algorithm, which offers a quadratic speedup for brute-force key search. [3]

- **Impact of Quantum Computing on Cybersecurity**

Once cryptographically relevant quantum computers (CRQCs) become practical, adversaries could decrypt historical and current communications posing a risk exacerbated by "store-now-decrypt-later" (SNDL) attacks. The transition to post-quantum security is thus not only a technical challenge but also an imperative for long-term data confidentiality, integrity, and trust in digital systems.

- **Objectives**

This chapter systematically explores the landscape of post-quantum security. It reviews quantum computing threat models, analyzes the vulnerabilities of current cryptographic systems, elucidates the foundations and classes of PQC algorithms, examines digital signatures and key exchange mechanisms, surveys standardization and deployment efforts, and highlights implementation challenges.

What is a quantum computer?

In 1981, Richard Feynman proposed a new way to model quantum interactions in complex systems. In that we need to represent each linked particle as a set of probabilities. As we add particles, these arrays grow exponentially. For any

sufficiently large system, we can no longer handle the storage and time requirements using existing computers.[4]

Feynman's suggestion is simple: Build a computer using entangled quantum objects. Such a computer could efficiently handle a number of tasks with which we could figure out how to take advantage of changing entangled quantum states.

- **What is a Qubit?**

The idea behind a quantum computer is to replace our classical bits with "qubits". Classical bits can be either 0 or 1, while a qubit takes on a *probability* of being 1 or 0, usually represented by a unit vector in three-dimensional space. The power of the qubit isn't a single bit, but multiple bits which are entangled with each other. If you can devise an algorithm in which these qubits interfere with each other in the solution to your problem, you can force these bits to take on the state of your solution instantly.

Fig. 1: Comparison of Classical Bit and Qubit Representation

- **What do quantum computers have to do with cryptography?**

In 1994, Peter Shor identified an algorithm that could use a quantum computer to break the RSA and Diffie Hellman cryptographic systems. Shor's algorithm was then extended to break ECC as well.

- **Why should you care about post-quantum cryptography?**

When you enter your credit card number on the web, that communication is protected by an encrypted channel which depends on both digital signing (to make

sure you are giving the credit card to the correct vendor), and public key exchange (to agree on a set of keys used between client and server to encrypt your communication). [5] If a sufficiently large quantum computer were to be built, they can easily be able to guess the credit card number and decrypt the communication.

- **When do you need to care?**

  This question can be answered using Mosca's Theorem:

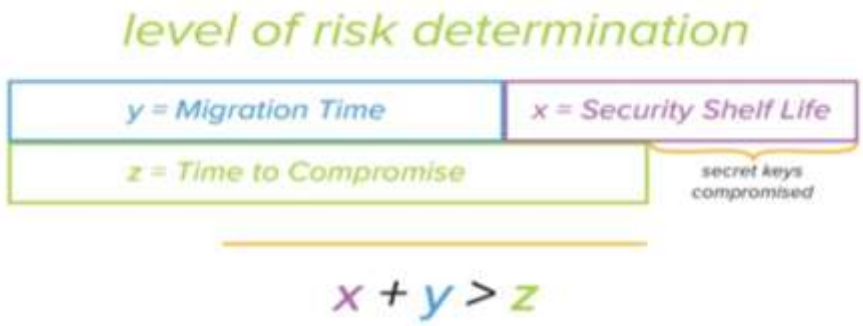

Fig. 2: Details of variables in Mosca's Theorem

If the sum of the time to migrate to the new algorithm (y) and the time you need the secret to be kept (x) is greater than the time left before we have a quantum computer that can break our public key algorithm (z) then your data will be compromised before its usefulness expires. The time you need to keep the secret (x) is usually known based on the application. For your credit card on the internet, for example, this would be maybe two or three years depending on your card's expiration date. For medical data, on the other hand, it could be decades.

Quantum Computing Threat Model

- **Shor's Algorithm and Its Cryptographic Implications**

Shor's algorithm, introduced in 1994, allows efficient factoring of large integers and computation of discrete logarithms. Shor's factoring algorithm finds one of two unknown variables that are crucial for efficiently factoring an integer. With two unknowns in one equation, finding both values quickly becomes classically intractable as the target integer gets larger. There are classical algorithms to find one of those values, but they become increasingly inefficient as the target integer gets larger.

Specifically, an unknown integer g when multiplied by itself p times and modulo divided by the integer N we want to factor equals one, or (g*p)%N=1. We start off knowing the number N we want to factor. Shor's Algorithm estimates p, the period of N, so we only need to guess g. Using the smallest practical number N, which is N=15, Shor's Algorithm returns period p=4. We can see that with (g*4)%15=1 we can guess g=2, which results in 24%15=1, or 16%15=1, being true. Guessing g gets harder as N grows larger, but not as hard as guessing both g and p.

- **Grover's Algorithm and Symmetric Key Security**

Grover's algorithm provides a quadratic speedup for unstructured search problems, including brute-force attacks on symmetric-key cryptography and hash functions. Proposed by Lov K. Grover in 1996, the algorithm addresses the problem of searching an unstructured database, where no prior ordering or heuristic information is available. In a classical computational model, searching such a database of N elements requires, on average, O(N) queries to locate a desired item. Grover's algorithm reduces this complexity to O(\sqrt{N}).

The core idea behind Grover's algorithm lies in the principles of quantum superposition and interference. By applying Hadamard transformations to an n-qubit register initialized in the zero state, the algorithm prepares an equal superposition of all 2^n possible database indices. Central to Grover's algorithm is the concept of an oracle. [6] The oracle does not reveal the solution directly; instead, it marks the desired state by applying a phase inversion. Following the oracle operation, the algorithm applies the diffusion operator.

Vulnerability of Current Cryptographic Systems

- **Quantum threat to Symmetric Key Cryptography**

Symmetric key cryptography is a cryptographic technique in which the same secret key is used by both the sender and the receiver for encryption and decryption of data. Its security depends on keeping this shared key confidential, making secure key distribution a central challenge. Symmetric algorithms are computationally efficient and well suited for encrypting large volumes of data, which is why they are widely used in applications such as disk encryption, secure communications, and data protection systems. Common symmetric key algorithms include the Advanced Encryption Standard (AES), Data Encryption Standard (DES), and ChaCha20. The emergence of quantum computing poses a measurable but manageable threat to symmetric key cryptography, primarily due to Grover's algorithm, which provides a quadratic speedup for brute-force key search.

- **Public-Key Cryptography Under Quantum Attacks**

Public-key systems are especially vulnerable to quantum attacks. RSA, ECC, and Diffie–Hellman are widely deployed in Internet protocols, software updates, digital signatures, and secure communications. Shor's algorithm renders all these schemes insecure, necessitating their replacement with quantum-resistant alternatives. This transition is a massive logistical and financial undertaking.

Grover's algorithm reduces the brute-force search space for symmetric ciphers and hash functions. While doubling key sizes restores pre-quantum security levels, this requires protocol changes and careful cryptographic engineering.

Algorithm	Purpose	Best Classical Attacks	Recommended Parameter Sizes (2025)
Diffie-Hellman	Key Exchange	▪ Finite Fields: GNFS (subexponential) ▪ Elliptic Curves: Pollard's Rho (exponential)	▪ Finite Fields: ≥ 3072-bit prime p ▪ Elliptic Curves: ≥ 256-bit prime
RSA Encryption	Message Encryption	GNFS (subexponential)	≥ 3072-bit modulus N
Digital Signatures (DH/RSA-based)	Authentication, integrity, non-repudiation	▪ RSA: GNFS (subexponential) ▪ Elliptic Curve: Pollard's Rho (exponential)	▪ RSA: ≥ 3072-bit modulus N ▪ Elliptic Curves: ≥ 256-bit prime

Fig. 3: Details showing Attacks and Improvement in Security Parameters for Classical Algorithms

- **Store/Harvest-Now-Decrypt-Later Attacks**

SNDL attacks involve adversaries recording encrypted communications now and decrypting them when quantum capabilities become available. This threat is particularly acute for sensitive or long-lived data, reinforcing the urgency of transitioning to quantum-resistant cryptography.

The three major types of public-key protocols are key exchange protocols, which establish shared, secret encryption keys based on exchanged public-key material, encryption protocols, in which publicly available key material is directly used to encrypt messages, and digital signature protocols, which are used to verify the authenticity of messages and their origins. In contrast to secret-key cryptosystems, the effect of quantum computers on the presently used public-key cryptosystems is devastating. This vulnerability makes public-key cryptography a primary focus in the development of post-quantum cryptography.

Foundations of Post-Quantum Cryptography (PQC)

- **Definition and Design Principles**

Post-quantum cryptography encompasses cryptographic primitives designed to withstand attacks from both classical and quantum adversaries. These schemes are typically based on mathematical problems for which no efficient quantum algorithms are known—such as lattice problems, code-based problems, hash functions, and multivariate polynomial equations. PQC algorithms rely on hardness assumptions that remain robust in the face of quantum computational capabilities.

- **Performance and Implementation Constraints**

A key challenge in PQC is balancing security with performance and implementation feasibility. Many PQC schemes require larger key sizes, increased memory, and more computational resources than their classical counterparts. These constraints impact deployment in resource-constrained environments such as IoT devices and embedded systems.

Classes of PQC Algorithms

- **Lattice-Based Cryptography (LBC)**

Lattice-based schemes, such as Kyber (encryption/KEM) and Dilithium (signatures), are among the most promising PQC techniques. Their security is based on the presumed hardness of problems like Short Integer Solution (SIS) and LWE. It supports efficient key exchange, encryption, signatures, and even advanced constructs like fully homomorphic encryption. A lattice is a geometric structure formed by an infinite set of points in a multi-dimensional space arranged in a periodic pattern. [7] Following figure shows the example calculation for a encryption of 1 bit using Lattice method.

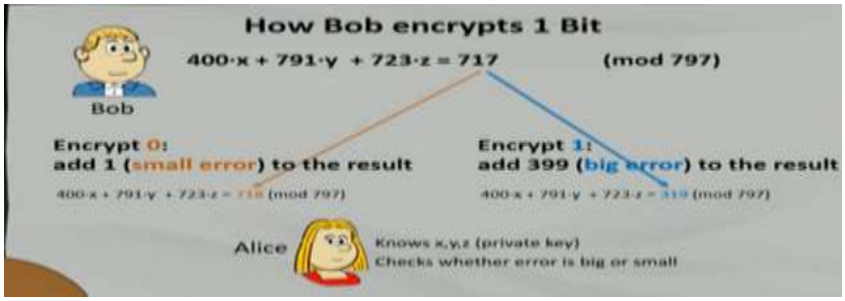

Fig. 4: Encryption of a Bit using Lattice Method

- **Code-Based Cryptography (CBC)**

Code-based systems, most notably the McEliece cryptosystem, derive their security from the hardness of decoding random linear codes. These schemes suffer from large key sizes, which can delay deployment. It is a form of public key cryptography based on error-correcting codes. In CBC, the public key is derived from an error-correcting code, and the private key is the knowledge of the decoding algorithm for that code. The scheme's security relies on the computational difficulty of decoding the code without knowing the private key.

- **Hash-Based Cryptography (HBC)**

Hash-based signature schemes, such as SPHINCS+, use the security of cryptographic hash functions to construct digital signatures. These schemes are particularly attractive for their simplicity and minimal reliance on unproven algebraic assumptions, but they are generally limited to signature applications.

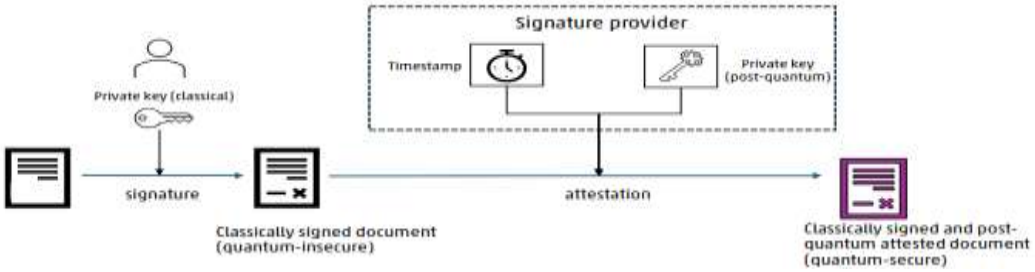

Fig. 5: Quantum state attestation of Digital Signature

HBC utilizes the collision resistance and one-way properties of hash functions to provide security guarantees. The signing process involves hashing the message with a secret key to create a digest and then applying a one-way function to the digest to produce the signature. The signature is appended to the message and can be verified by anyone with the corresponding public key. A collision occurs when two different inputs produce the same hash output. In hash-based cryptography, if an attacker can find a crash for the hash function, they can forge signatures and impersonate the signer. Therefore, the strength of HBC depends on the chosen hash function. The signing and verification processes are relatively fast compared to other digital signature schemes and they are typically small, making them suitable for low-resource devices and applications.

- **Multivariate Cryptography (MVC)**

Multivariate cryptography leverages the difficulty of solving systems of multivariate quadratic equations. It is a form of public key cryptography based on the difficulty of solving systems of multivariate polynomial equations. The public key is derived from a system of multivariate polynomial equations, and the private key is the secret knowledge of how to solve these equations efficiently. [8]

- **Isogeny-Based Cryptography (IBC)**

Supersingular elliptic curve isogeny cryptography is a post-quantum cryptographic scheme that is based on the mathematics of elliptic curves and isogenies. An isogeny is a function between two elliptic curves that preserves specific algebraic properties.[9]

Table 1: PQC Algorithm use cases

Algorithm Class	Mathematical Basis	Use Case
Lattice-Based	Shortest Vector Problem (SVP)	General Encryption & Signatures (e.g., ML-KEM, ML-DSA)
Code-Based	Decoding general linear codes	Key Exchange (e.g., HQC - selected by NIST in March 2025)
Hash-Based	Security of hash functions	Digital Signatures (e.g., SLH-DSA)

Multivariate	Solving systems of quadratic equations	Digital Signatures (e.g., MAYO under evaluation)
Isogeny-Based	Supersingular isogeny graphs	Historically used for key exchange; currently viewed as slower

Standardization and Global Initiatives

- **NIST Post-Quantum Cryptography Standardization Process**

NIST's PQC competition, launched in 2017, has driven the evaluation and selection of quantum-resistant algorithms based on security, performance, and implementation criteria. In July 2022, the first standards were selected: Kyber for encryption/KEM, and Dilithium, Falcon, and SPHINCS+ for digital signatures. International standards bodies, such as ISO and the Internet Engineering Task Force (IETF), are collaborating to harmonize PQC adoption. Major technology providers are piloting PQC deployments, particularly in cloud, IoT, and critical infrastructure sectors.

- **Challenges in Standard Deployment**

Deployment faces numerous challenges: cost (estimated in billions for large governments), complexity of migration, backward compatibility, and the need for global coordination. The rapid breakage of previously promising schemes during the standardization process underscores the necessity of ongoing cryptanalytic research.

Quantum-Resistant Security Beyond PQC

- **Quantum Key Distribution (QKD)**

QKD protocols, such as BB84, leverage quantum mechanics to enable information-theoretic secure key exchange. [10] While QKD offers unique security guarantees, it requires specialized hardware and is not a direct replacement for public-key cryptography. Quantum Key Distribution (QKD) is a secure communication method that uses the principles of quantum mechanics to produce and distribute a shared, random secret key known only to the communicating parties.

  - **How it Works:** The sender (Alice) transmits photons encoded with quantum states (qubits) to the receiver (Bob) over a quantum channel. Due to the Heisenberg Uncertainty Principle and the no-cloning theorem, any attempt by a third party (Eve) to measure or intercept these photons disturbs their quantum state, introducing detectable errors.[11] Alice and Bob then compare a subset of their measurements over a classical channel; if the error rate is below a certain threshold, they can be assured that no eavesdropping occurred and can use the remaining shared bits as a secure encryption key.
  - **Security Basis:** QKD offers information-theoretic security, meaning its unbreakability is guaranteed by the fundamental laws of physics, not by

computational complexity assumptions that might be challenged by future algorithms or computing power.

- **Comparison Between PQC and QKD**

PQC offers broad applicability and can be deployed on existing digital infrastructures, whereas QKD is suitable for niche high-security applications with specific physical and operational requirements.

Table 2: Comparison of PQC and QKD

Feature	Post-Quantum Cryptography (PQC)	Quantum Key Distribution (QKD)
Security Basis	Relies on mathematical problems believed to be hard for both classical and quantum computers.	Relies on the laws of quantum physics (e.g., no-cloning theorem, measurement disturbance principle).
Implementation	Software-based; can be integrated into existing network infrastructure without specialized hardware upgrades.	Hardware-based; requires specialized quantum equipment (e.g., single-photon sources/detectors) and dedicated fiber optic or free-space channels.
Scalability	Highly scalable and works over unlimited distances using existing internet protocols (e.g., TLS, SSH).	Limited by distance due to signal loss (typically a few hundred kilometers); long distances require trusted nodes or quantum repeaters.
Eavesdropping	Difficult to detect in real-time, which enables "harvest now, decrypt later" attacks.	Eavesdropping attempts physically disturb the quantum channel and are immediately detectable by the communicating parties.
Authentication	Provides full authentication and integrity services as part of the cryptographic scheme.	QKD itself only distributes keys and requires a separate, pre-authenticated classical channel to prevent man-in-the-middle attacks.

- **Hybrid Quantum-Classical Security Models**

A hybrid quantum–classical security model integrates classical cryptographic mechanisms with quantum-resistant and quantum-based techniques to ensure secure communication during the transition. In this model, conventional cryptographic algorithms such as symmetric encryption and classical public key schemes continue to operate alongside post-quantum cryptographic algorithms or quantum key distribution protocols. Hybrid approaches are commonly used in key exchange and authentication, where a session key is derived by combining outputs from both

classical and post-quantum methods, ensuring that security is preserved even if one component is later compromised. This layered design provides backward compatibility with existing infrastructures while offering resilience against future quantum adversaries, making hybrid quantum–classical security models a practical and strategic solution for maintaining long-term cryptographic security.

Implementation Challenges and Performance Trade-offs

- **Computational and Memory Overheads**

Most PQC schemes incur higher computational and memory costs than classical algorithms, particularly in key generation, encryption, and signature operations. Table 1 summarizes performance metrics for leading KEMs:

Table 3. Average Execution Time (ms) for Leading PQC KEMs

Algorithm	Key Generation	Encryption	Decryption
Kyber512	0.0095 ms	0.0114 ms	0.0081 ms
FrodoKEM	0.2301 ms	0.3181 ms	0.2989 ms
sntrup761	0.1968 ms	0.0145 ms	0.0137 ms

Migration requires inventorying vulnerable assets, deploying hybrid schemes, and ensuring backward compatibility with legacy systems. The financial and logistical burden of PQC migration is substantial, with government estimates running into billions of dollars and timelines extending over a decade. [12] The transition to PQC presents several challenges:

- **Performance Overhead:** PQC algorithms often require more computational resources (CPU, memory, bandwidth) and have larger key sizes than current encryption methods, which can impact performance, especially in resource-constrained environments like IoT devices.
- **Legacy System Compatibility:** A major hurdle is the need to update or replace a vast number of existing, outdated, IT infrastructures, protocols (like TLS), and devices, which can be expensive and time-consuming. [13]
- **Interoperability and Crypto-Agility:** Ensuring that new PQC systems can interoperate with existing ones during a multi-year transition, and building systems with "crypto-agility" (the ability to quickly switch algorithms if a weakness is found), adds complexity.
- **Implementation Security:** PQC algorithms are more complex and require careful implementation to avoid new vulnerabilities, such as side-channel attacks.
- **Lack of Awareness and Skills:** Despite high-level awareness, there is a gap in practical knowledge and skilled personnel within many organizations to manage the transition effectively.

Conclusion

The quantum threat to classical cryptography is real and imminent. Post-quantum cryptography offers a viable path forward, with lattice-based, code-based, hash-based, and multivariate schemes providing diverse approaches to quantum resistance. Standardization and deployment are well underway but face significant technical, financial, and organizational challenges.

Table 4: Comparison of Classical and Quantum Cryptography Algorithms

Key Distribution Method	Example Key Size	Current Use	Quantum Security Level	Breakable/ Not Breakable	Why?
RSA-2048	2048-bit	Widely used (TLS, VPN, Certificates)	Not secure	Breakable	Shor's algorithm factors 2048-bit RSA efficiently
RSA-4096	4096-bit	High-security systems	Not secure	Breakable	Larger but still polynomial-time breakable
Diffie–Hellman (DH)	2048-bit	Traditional key exchange	Not secure	Breakable	Uses discrete log; Shor's algorithm solves it
ECDH P-256	256-bit	TLS 1.3, mobile apps	Completely insecure	Breakable	Shor solves ECC discrete logs extremely fast
ECDH P-384	384-bit	High-security TLS	Not secure	Breakable	Only linearly harder for Shor
AES-128 (symmetric key)	128-bit	Common symmetric cipher	Partially secure	Nearly 7 months using Grover to break	Grover reduces search to $2^{64} \approx 1.8\times10^{19}$ ops
AES-256 (symmetric key)	256-bit	Quantum-safe symmetric encryption	Secure	10^{19} years to break	Grover reduces to 2^{128}
CRYSTALS-Kyber 512 (PQC key exchange)	512-bit security parameters	Future TLS, 5G, VPN	Secure	No known quantum attack	Lattice problems have no quantum speedups
CRYSTALS-Kyber 1024	High security PQC	Military-grade	Highly secure	Infeasible (beyond 10^{30} years)	Hard lattice problem; extremely large space
QKD (Quantum Key Distribution)	Quantum states	Experimental networks	Information-theoretic secure	Impossible to break (infinite time)	Attacker cannot read key without detection

- **Strategic Recommendations**
  - **Accelerate migration**: Organizations must inventory cryptographic assets, deploy hybrid schemes, and prioritize migration of high-value and long-lived data.
  - **Invest in research**: Ongoing cryptanalysis and implementation research are essential to ensure the robustness of PQC standards.
  - **Foster global coordination**: International standards and collaboration among governments, industry, and academia are vital for interoperability and security.
  - **Prepare for side-channel and implementation threats**: Secure coding, hardware protections, and best-practice engineering are as crucial as cryptographic primitives.

References

1. Peter W. Shor, Algorithms for Quantum Computation: Discrete Logarithms and Factoring, Proceedings of the 35th Annual Symposium on Foundations of Computer Science (FOCS), IEEE, 1994.
2. National Institute of Standards and Technology (NIST), Post-Quantum Cryptography: NISTIR 8105 and PQC Standardization Project, U.S. Department of Commerce, 2016–present.
3. Michele Mosca, Cybersecurity in an Era with Quantum Computers: Will We Be Ready?, IEEE Security & Privacy, vol. 16, no. 5, 2018.
4. Richard P. Feynman, Simulating Physics with Computers, International Journal of Theoretical Physics, vol. 21, nos. 6–7, pp. 467–488, 1982.
5. Michael A. Nielsen and Isaac L. Chuang, Quantum Computation and Quantum Information, Cambridge University Press, 2000.
6. Daniel J. Bernstein, Johannes Buchmann, and Erik Dahmen (eds.), Post-Quantum Cryptography, Springer, 2009.
7. Chris Peikert, A Decade of Lattice Cryptography, Foundations and Trends in Theoretical Computer Science, vol. 10, nos. 4–5, 2016.
8. Jintai Ding and Dieter Schmidt, Rainbow, a New Multivariable Polynomial Signature Scheme, Applied Cryptography and Network Security (ACNS), 2005.
9. J. De Feo, D. Jao, and J. Plût, Towards Quantum-Resistant Cryptosystems from Supersingular Elliptic Curve Isogenies, Journal of Mathematical Cryptology, vol. 8, no. 3, 2014.
10. Charles H. Bennett and Gilles Brassard, Quantum Cryptography: Public Key Distribution and Coin Tossing, Proceedings of the IEEE International Conference on Computers, Systems and Signal Processing (CSSP), 1984.

11. Artur K. Ekert, Quantum Cryptography Based on Bell's Theorem, Physical Review Letters, vol. 67, no. 6, pp. 661–663, 1991.
12. Erwin Alkim, Léo Ducas, Thomas Pöppelmann, and Peter Schwabe, Post-Quantum Key Exchange A New Hope, USENIX Security Symposium, 2016.
13. National Security Agency (NSA), Commercial National Security Algorithm Suite 2.0 (CNSA 2.0), 2022.

☯☯☯