QUANTUM COMPUTING AND ITS IMPACT ON CRYPTOGRAPHY

LOU SERGONNE

TUTEUR: SEBASTIAN RUST

2024-25

Contents

1

Introduction

Quantum computing uses strange but powerful phenomena from quantum mechanics, like superposition and entanglement, to perform certain computations much faster than classical computers ever could [BF23; LF23]. While this opens the door to major breakthroughs, it also threatens one of the core pillars of modern digital life: cryptography.

Some quantum algorithms hit particularly hard. Shor's algorithm [Sho97] breaks the math behind today's most widely used public-key cryptography systems RSA, ECC, and DH. all in polynomial time. Grover's algorithm doesn't break symmetric ciphers outright, but it still weakens them by giving a quadratic speedup in brute-force attacks, effectively halving their security.

The threat isn't theoretical anymore. The "Store Now, Decrypt Later" (SNDL) scenario is real: attackers can harvest encrypted data now and wait for quantum machines to catch up [Moo+24; GM24; Reg24]. That's why we need to move towards post-quantum cryptography (PQC) algorithms designed to hold up against both classical and quantum attacks. But switching over isn't as easy as it sounds. It means rethinking systems, protocols, standards, basically, rebuilding mostly from the ground up [Moo+24; GM24; Reg24].

This memoire looks at how quantum computing changes the game for cryptography. It covers the weaknesses exposed by quantum algorithms, dives into emerging quantum-safe solutions, and tackles the messy reality of moving to a PQC-secure world.

1.1 Structure of this Memoire

Here's how the rest of the memoire is laid out:

- **Chapter 2:** Fundamentals of Quantum Computing; Key concepts needed to understand the cryptographic threat.
- **Chapter 3:** Classical Cryptography; A quick and brief recap of current systems and how they work.
- **Chapter 4:** Classical vs Quantum Computing; What makes quantum computers so different.
- **Chapter 5:** Quantum Impact on Cryptography; A closer look at how specific algorithms break classical crypto.
- **Chapter 6:** Quantum-Resistant Solutions; An overview of PQC algorithms and standardization efforts.
- **Chapter 7:** Challenges in Transitioning to PQC; Real-world issues in rolling out quantum-safe cryptography.
- **Chapter 8:** Conclusion

2

Fundamentals of Quantum Computing

2.1 Quantum Mechanics Principles

The foundation of quantum computing rests on several key quantum mechanical principles [vot24; Ism25; Wik25].

2.1.1 Superposition

In quantum mechanics, a system can exist in multiple states simultaneously until measured [**KyrlynnD2024SuperpositionHow**; Tho25]. Mathematically, a quantum state $|\psi\rangle$ of a single qubit can be expressed as:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{2.1}$$

where $|0\rangle$ and $|1\rangle$ are the computational basis states (analogous to classical bits 0 and 1), and where α and β are complex numbers called probability amplitudes. The squares of their absolute values represent the probabilities of measuring the qubit in the corresponding basis state, satisfying $|\alpha|^2 + |\beta|^2 = 1$. This ability to exist in a combination of states allows quantum computers to explore many possibilities concurrently.

2.1.2 Entanglement

Entanglement is a unique quantum correlation where two or more qubits become linked in such a way that they share the same fate, regardless of the distance separating them [DJ07]. Their states are described by a single, combined quantum state, not independent individual states. For example, if two qubits are entangled in the Bell state:

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

measuring the first qubit to be $|0\rangle$ instantly forces the second qubit to be $|0\rangle$, and measuring the first as $|1\rangle$ forces the second to be $|1\rangle$, even if they are light-years apart. This correlation, famously dubbed "spooky action at a distance" by Einstein, cannot be explained by classical physics (e.g., hidden variables) and is experimentally verified. While entanglement does not allow faster-than-light communication (information transfer still requires classical communication), it is a crucial resource enabling quantum algorithms (like Shor's), quantum teleportation, and certain quantum cryptographic protocols.

2.1.3 Quantum Bits

Unlike classical bits which can only be 0 or 1, quantum bits (qubits) can exist in a superposition of both states, as described above. This property is often visualized using the Bloch sphere (Figure 2.1) [Con25b], where any point on the surface of the sphere represents a possible pure state of a single qubit.

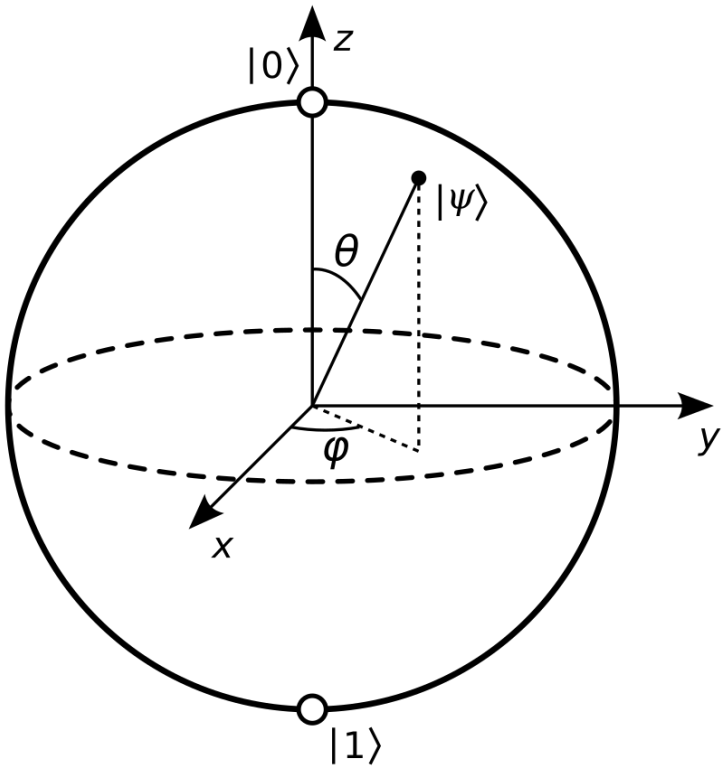

Figure 2.1: The Bloch sphere representation of a single qubit state $|\psi\rangle = \cos(\theta/2)|0\rangle + e^{i\phi}\sin(\theta/2)|1\rangle$. The north pole represents $|0\rangle$, the south pole $|1\rangle$.

Physical Implementations

Realizing qubits physically presents diverse approaches, each with unique advantages and challenges:

- **Superconducting Circuits:** Utilize circuits with Josephson junctions cooled to milli-Kelvin temperatures. These allow for fast gate operations (nanoseconds) and are a leading technology, but require significant cryogenic infrastructure and are sensitive to noise.
- **Trapped Ions:** Charged atoms held by electromagnetic fields. They maintain quantum states for seconds to minutes and perform operations accurately, though more slowly (microseconds) than other platforms.
- **Photonic Qubits:** Employ quantum states of light (e.g., polarization or path encoding). Photons are naturally robust against decoherence and ideal for communication (Quantum Key Distribution), but building universal quantum gates for computation is challenging.

- **Other Platforms:** Include neutral atoms, quantum dots, topological qubits, and NV-centers in diamond, each representing active areas of research.

A major engineering challenge common to all platforms is maintaining quantum coherence, preserving the delicate superposition and entanglement against environmental noise (e.g., thermal fluctuations, stray electromagnetic fields) which causes decoherence (loss of quantum information). This necessitates sophisticated control systems, shielding, and often, operation at extremely low temperatures or high vacuum.

2.2 Quantum Gates and Circuits

Quantum computations are performed by applying sequences of quantum gates to qubits [Con25c]. These gates are equivalent to classical logic gates but operate on quantum states. Mathematically, single-qubit gates are represented by 2×2 unitary transformations ($U^\dagger U = I$), and multi-qubit gates by larger unitary matrices, ensuring that the evolution of quantum states is reversible and preserves probabilities. Below are examples of important quantum gates:

Pauli-X (NOT) Gate

The Pauli-X gate acts as a quantum bit-flip, analogous to the classical NOT gate. It transforms $|0\rangle$ to $|1\rangle$ and $|1\rangle$ to $|0\rangle$. For a general qubit $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$:

$$X|\psi\rangle = X(\alpha|0\rangle + \beta|1\rangle) = \alpha|1\rangle + \beta|0\rangle$$

Matrix form:

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Pauli-Y Gate

The Pauli-Y gate performs a bit-flip combined with phase changes.

$$Y|\psi\rangle = Y(\alpha|0\rangle + \beta|1\rangle) = \alpha(i|1\rangle) + \beta(-i|0\rangle) = -i\beta|0\rangle + i\alpha|1\rangle$$

Matrix form:

$$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$$

Pauli-Z Gate

The Pauli-Z gate acts as a phase-flip, leaving $|0\rangle$ unchanged and mapping $|1\rangle$ to $-|1\rangle$.

$$Z|\psi\rangle = Z(\alpha|0\rangle + \beta|1\rangle) = \alpha|0\rangle - \beta|1\rangle$$

Matrix form:

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Hadamard (H) Gate

The Hadamard gate is crucial for creating superposition. It transforms $|0\rangle$ into an equal superposition of $|0\rangle$ and $|1\rangle$, and $|1\rangle$ into an equal superposition with a phase difference.

$$H|0\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}, \quad H|1\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

Applying H twice returns the original state ($H^2 = I$). Matrix form:

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Phase (S or $\sqrt{Z}$) Gate

The S gate introduces a relative phase shift of $\pi/2$ (or i) between the $|0\rangle$ and $|1\rangle$ components. It is sometimes called the $\sqrt{Z}$ gate as $S^2 = Z$.

$$S|\psi\rangle = S(\alpha|0\rangle + \beta|1\rangle) = \alpha|0\rangle + i\beta|1\rangle$$

Matrix form:

$$S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$$

T (or $\pi/8$) Gate

The T gate introduces a relative phase shift of $\pi/4$. It is important because H, S, and CNOT gates alone are not sufficient for universal quantum computation; adding the T gate completes a common universal set.

$$T|\psi\rangle = T(\alpha|0\rangle + \beta|1\rangle) = \alpha|0\rangle + e^{i\pi/4}\beta|1\rangle$$

Matrix form:

$$T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1+i}{\sqrt{2}} \end{pmatrix}$$

CNOT (Controlled-NOT) Gate

The CNOT gate is a fundamental two-qubit gate. It flips the state of the second qubit (target) if and only if the first qubit (control) is in the state $|1\rangle$.

$$\begin{aligned} \text{CNOT}|00\rangle &= |00\rangle \\ \text{CNOT}|01\rangle &= |01\rangle \\ \text{CNOT}|10\rangle &= |11\rangle \\ \text{CNOT}|11\rangle &= |10\rangle \end{aligned}$$

Matrix form (acting on basis states $|00\rangle, |01\rangle, |10\rangle, |11\rangle$):

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

The CNOT gate is essential for creating entanglement and is a key component in many quantum algorithms and error correction codes.

Toffoli (CCNOT) Gate

The Toffoli gate is a three-qubit gate, acting as a controlled-controlled-NOT. It flips the third qubit if and only if the first two control qubits are both in the state $|1\rangle$.

$$\text{CCNOT}|abc\rangle = |ab(c \oplus (a \cdot b))\rangle$$

where $\oplus$ is addition modulo 2. Matrix form (acting on basis states $|000\rangle$ through $|111\rangle$):

$$\text{CCNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

The Toffoli gate is universal for classical reversible computation. Together with the Hadamard gate (or other suitable single-qubit rotations), it forms a universal set for quantum computation, meaning any quantum computation can be decomposed into a sequence of these gates.

Bell States Creation

Applying a Hadamard gate to the first qubit (initially $|0\rangle$) followed by a CNOT gate controlled by the first qubit acting on the second (initially $|0\rangle$) creates the Bell state $|\Phi^+\rangle$:

1. Start with $|00\rangle$.
2. Apply H to the first qubit: $H|0\rangle \otimes |0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle)$.
3. Apply CNOT (control=1st, target=2nd): $\text{CNOT}\left(\frac{1}{\sqrt{2}}(|00\rangle + |10\rangle)\right) = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) = |\Phi^+\rangle$.

This maximally entangled state demonstrates how quantum gates can generate non-classical correlations essential for quantum algorithms.

2.3 Quantum Algorithms

Quantum algorithms leverage the principles of superposition, entanglement, and quantum interference to perform computations in ways that can dramatically outperform classical algorithms for specific problems. This section introduces three cornerstone quantum algorithms: the Quantum Fourier Transform (QFT), Quantum Phase Estimation (QPE), and Amplitude Amplification (which generalizes Grover's search algorithm), highlighting their mechanisms, interdependencies, and cryptographic relevance.

2.3.1 Quantum Fourier Transform (QFT)

The QFT is the quantum analogue of the classical Discrete Fourier Transform (DFT). It maps a quantum state represented in the computational basis to its representation in the Fourier basis. Its primary strength lies in efficiently finding periodicities in quantum states.

When applied to a computational basis state $|j\rangle$ (where j is an integer represented by n qubits), the QFT generates a superposition state:

$$\text{QFT}|j\rangle = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i jk/N} |k\rangle \tag{2.2}$$

where $N = 2^n$ is the dimension of the state space. While a classical Fast Fourier Transform (FFT) takes $\mathcal{O}(N \log N)$ operations, the QFT circuit can be implemented using only $\mathcal{O}(n^2) = \mathcal{O}((\log N)^2)$ quantum gates (Hadamard and controlled phase rotations R_m), offering an exponential speedup in terms of n.

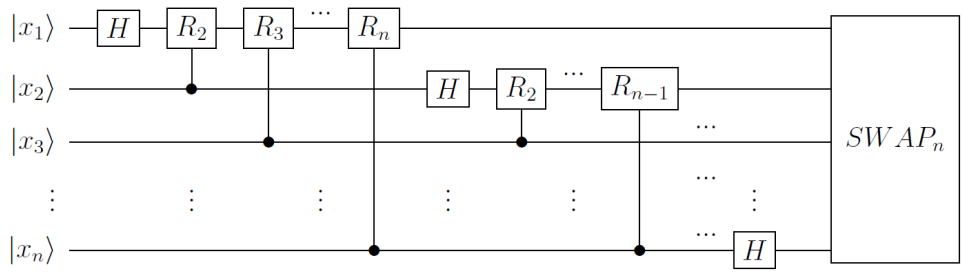

Figure 2.2: Quantum circuit implementing the QFT using layered Hadamard gates (H) and controlled phase rotations (R_m). Qubit ordering typically assumes $|x_1\rangle$ is the most significant qubit and $|x_n\rangle$ the least significant.

The phase rotation gates R_m are defined by:

$$R_m = \begin{pmatrix} 1 & 0 \\ 0 & e^{2\pi i/2^m} \end{pmatrix} \tag{2.3}$$

These gates apply progressively smaller phase shifts.

Cryptographic Relevance: The QFT's ability to efficiently find periods is the core component enabling Shor's algorithm to factor large integers and compute discrete logarithms exponentially faster than the best known classical algorithms. Shor's algorithm uses the QFT to find the period of the modular exponentiation function $f(x) = a^x \bmod N$, which then allows efficient calculation of the factors of N or the discrete logarithm.

2.3.2 Quantum Phase Estimation (QPE)

QPE is a fundamental quantum algorithm used to determine the eigenvalue (specifically, the phase) of an eigenvector of a unitary operator [Con25d]. Given a unitary operator U and one of its eigenstates $|\psi\rangle$ such that $U|\psi\rangle = e^{2\pi i\theta}|\psi\rangle$, QPE efficiently estimates the phase $\theta \in [0, 1)$.

The algorithm uses two registers: the first (measurement register) with m qubits initialized to $|0\rangle^{\otimes m}$, and the second (target register) initialized to the eigenstate $|\psi\rangle$. Key steps:

1. Apply Hadamard gates to the first register: $\frac{1}{\sqrt{2^m}} \sum_{k=0}^{2^m-1} |k\rangle \otimes |\psi\rangle$.
2. Apply controlled-U^{2^j} operations (controlled by the j-th qubit of the first register) to the second register. This encodes the phase θ into the first register's state: $\frac{1}{\sqrt{2^m}} \sum_{k=0}^{2^m-1} e^{2\pi i\theta k}|k\rangle \otimes |\psi\rangle$.
3. Apply the inverse QFT (QFT^{-1}) to the first register.
4. Measure the first register. The measurement outcome provides an m-bit approximation of θ.

The precision of the estimate scales as 2^{-m}.

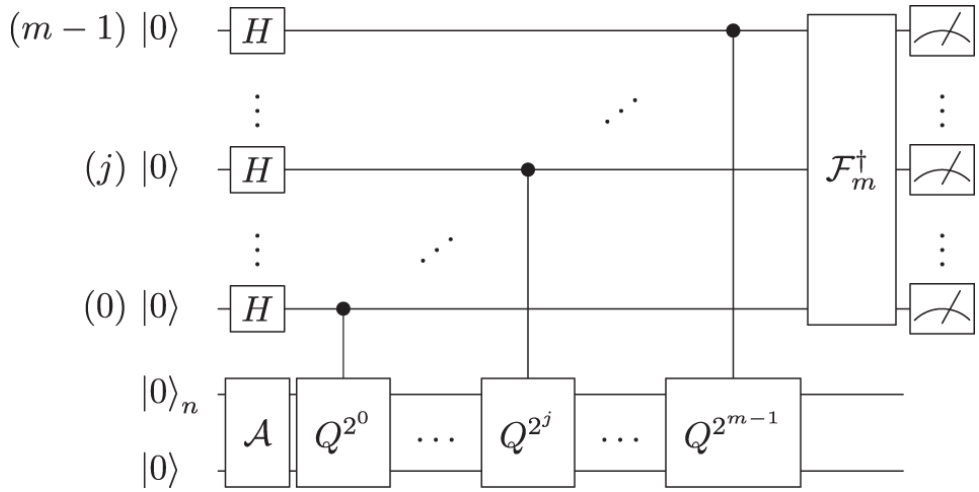

Figure 2.3: Quantum circuit for Phase Estimation. The top m qubits form the measurement register, the bottom n qubits hold the eigenstate $|\psi\rangle$. QFT^{-1} is the inverse Quantum Fourier Transform.

Cryptographic Relevance: QPE is the subroutine within Shor's algorithm that actually extracts the period information. The modular exponentiation operation can be implemented as a unitary operator U, and QPE is used to estimate the phase related to its eigenvalues, which in turn reveals the period needed for factoring or solving the discrete logarithm problem.

2.3.3 Amplitude Amplification and Grover's Algorithm

Amplitude amplification is a quantum technique that generalizes Grover's algorithm [Con25a], enhancing the probability of measuring a desired state in a quantum superposition. As mentioned earlier, it provides a quadratic speedup for unstructured search problems [Sca24].

The algorithm works as follows:

1. Start with an equal superposition of all possible states: $|s\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle$.

2. Apply two operations iteratively, approximately $\mathcal{O}(\sqrt{N})$ times:

   - Quantum oracle U_f: Marks the target state(s) by flipping their sign. For a target state $|w\rangle$, $U_f|w\rangle = -|w\rangle$.
   - Diffusion operator $U_s = 2|s\rangle\langle s| - I$: Performs an inversion about the average amplitude.

3. Measure the system, obtaining the target state with high probability.

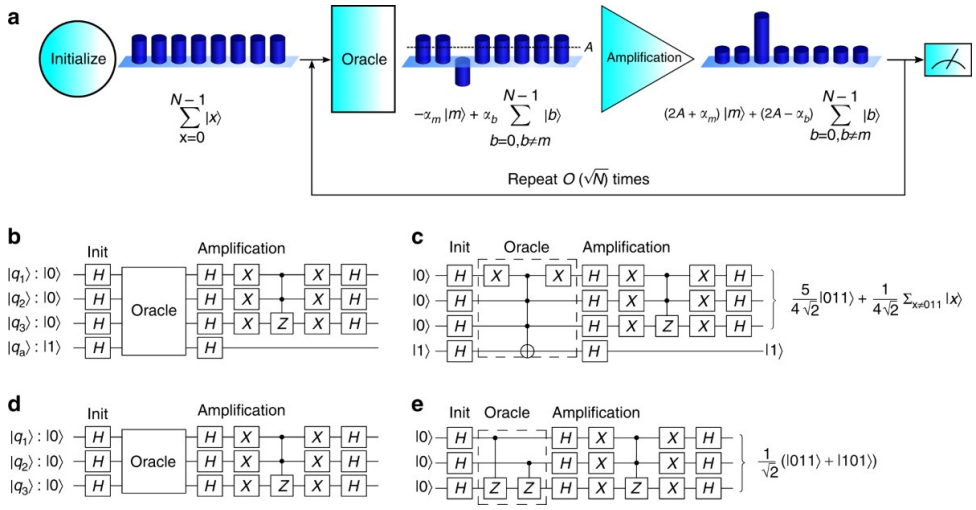

Figure 2.4: Conceptual visualization of Grover's algorithm, showing the iterative enhancement of the target state's amplitude.

Geometric Interpretation: The algorithm can be visualized as a rotation in a two-dimensional space spanned by $|\alpha\rangle$ (superposition of all non-solution states) and $|\beta\rangle$ (superposition of all solution states). Each iteration rotates the state vector closer to $|\beta\rangle$ by a fixed angle, requiring $\mathcal{O}(\sqrt{N/M})$ iterations for M solutions out of N possible states.

Cryptographic Relevance: Grover's algorithm's quadratic speedup impacts symmetric cryptography by reducing the effective key length by half against quantum attacks [Man+24]. For example, AES-128 (with 2^{128} possible keys) would require $\mathcal{O}(2^{64})$ quantum operations to break, equivalent to the classical security of a 64-bit key. This necessitates doubling key sizes (e.g., to AES-256) to maintain the same security level against quantum adversaries.

2.3.4 Algorithm Interconnections and Cryptographic Impact Summary

The fundamental algorithms QFT, QPE, and Amplitude Amplification form an interconnected toolkit with profound cryptographic implications:

- **QFT and QPE** are the core components of Shor's algorithm. They provide an *exponential* speedup for problems like integer factorization and discrete logarithms (both standard and elliptic curve variants). This completely breaks the security foundations of current public-key cryptosystems like RSA, Diffie-Hellman, and ECC.
- **Amplitude Amplification (Grover's algorithm)** provides a *quadratic* speedup for unstructured search problems. This weakens, but does not completely break, symmetric-key ciphers (like AES) and hash functions (like SHA-2, SHA-3) by effectively halving their bit security against brute-force style attacks.

Understanding these algorithms and their impact is crucial for appreciating the need for post-quantum cryptography, as discussed in later chapters.

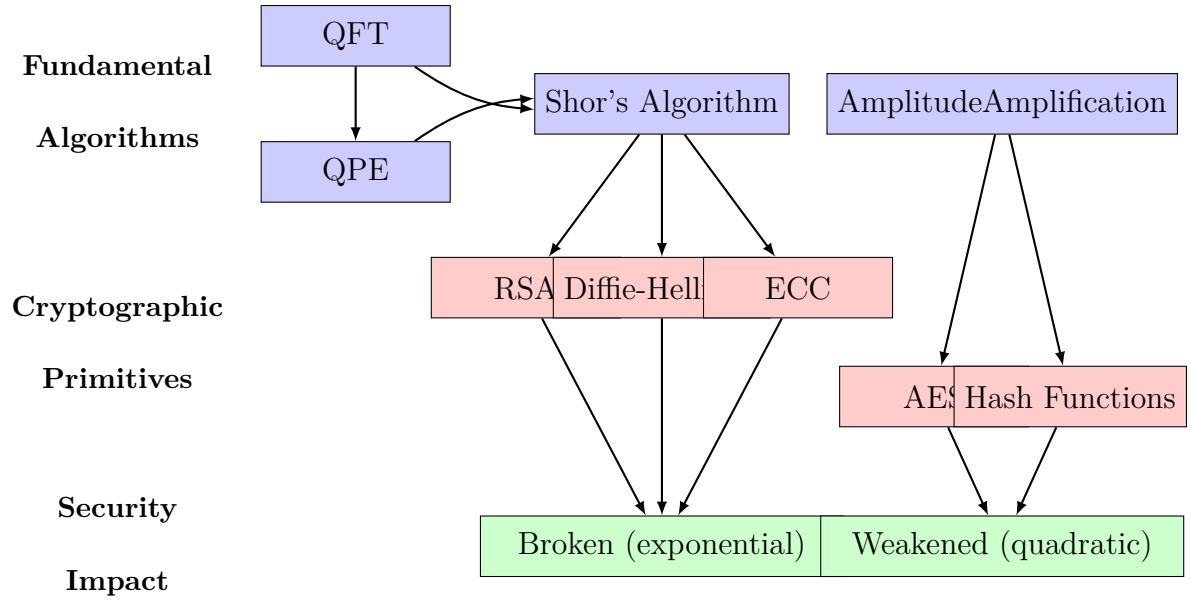

Figure 2.5: Relationships between fundamental quantum algorithms and their cryptographic impact. QFT and QPE are core components of Shor's algorithm, which completely breaks current public-key cryptosystems. Amplitude Amplification provides a quadratic speedup that weakens symmetric-key ciphers and hash functions.

2.4 The NISQ Era and Beyond

Current quantum computers are in the NISQ (Noisy Intermediate-Scale Quantum) era [AS22]. This means they have a limited number of qubits (typically 50-1000) and are susceptible to noise and errors (decoherence). While not yet capable of breaking large-scale RSA encryption, NISQ devices are valuable for exploring quantum algorithms, materials science simulations, and optimization problems. The path towards fault-tolerant quantum computing (FTQC), which requires millions of high-quality qubits and robust quantum error correction, remains a significant scientific and engineering challenge.

QEC (QEC) is theoretically capable of overcoming noise by encoding information redundantly across many physical qubits to create a more stable "logical qubit". However, QEC codes (like the surface code) have extremely high overheads. Current estimates suggest that factoring a 2048-bit RSA number using Shor's algorithm might require millions of high-quality physical qubits to encode the necessary thousands of logical qubits. This resource requirement far exceeds the capabilities of current NISQ hardware.

This discrepancy creates the **"Quantum Advantage Gap"**: we possess algorithms known to break current cryptography, but lack the hardware technology to execute them at scale. Bridging this gap is a central goal of quantum computing research, focusing on:

- Building more stable, higher-fidelity physical qubits with longer coherence times.
- Developing more efficient QEC codes and fault-tolerant architectures.
- Improving quantum compilation techniques to minimize resource requirements.
- Designing potentially useful algorithms that might run effectively on NISQ devices or require fewer resources than initially thought.
- Exploring hybrid quantum-classical approaches that leverage the strengths of both paradigms.

Although the timeline for fault tolerance quantum computing capable of breaking RSA-2048 remains uncertain (with estimates ranging widely), the potential impact necessitates proactive migration to quantum-resistant cryptography.

3

classical cryptography

classical cryptography represents the time tested methods that have protected sensitive information long before quantum computing became a concern. In this chapter, I examine these established techniques, exploring their origins, the mathematical principles they rely on, and the practical algorithms in use today. By doing so, I prepare the ground for Chapter 5, where I assess how quantum advances undermine these very foundations.

3.1 Historical Development of Cryptography

Cryptography has evolved over millennia in response to military, diplomatic, and commercial needs for secrecy [Kah96; Sin99]. Its origins trace back to simple transposition cipher devices like the Spartan scytale (circa 400 BCE) and early symbol substitution ciphers [Mad20].

During the **Classical Period**, monoalphabetic cipher ciphers such as Caesar gave way to polyalphabetic cipher schemes like Vigenère to frustrate frequency analysis; later, digraph cipher ciphers such as Playfair and linear algebraic approaches like the Hill cipher added further complexity [Sta17].

The **Mechanical Era** introduced machines like Enigma machine that automated encryption using rotating wheels (Figure 3.1). Cracking Enigma helped drive early computer development. The **Modern Era** began with Claude Shannon's work in 1949 [Sha49], leading to standards like DES in 1977 [Nat99]. This era established that security should rely on secret keys rather than secret methods, and brought public-key cryptography in 1976 [DH76], transforming how keys are shared securely.

Figure 3.1: Rotor assembly of the Enigma machine, illustrating its polyalphabetic substitution mechanism.

3.2 Fundamentals of Modern Cryptography

Modern cryptography assumes adversaries possess only classical computing resources. Security guarantees derive from well studied mathematical problems.

3.2.1 Core Security Goals

Cryptosystems aim to provide **confidentiality**, **integrity**, **authentication**, and **non-repudiation** through combinations of encryption, hashing, MACs, and digital signatures [KL14; Sta17].

3.2.2 Computational Hardness Assumptions

Key primitives rely on problems believed intractable for classical machines:

- **Integer Factorization (IFP)**: Hard to factor $N = pq$, the basis of RSA.
- **Discrete Logarithm (DLP)**: Hard to find x from $g^x = h$ in a finite group, underpinning Diffie Hellman and DSA.
- **Elliptic Curve DLP (ECDLP)**: Hard to find scalar k from $Q = kP$ on an elliptic curve, enabling ECC.

These support one-way function and trapdoor function functions whose classical difficulty will be reexamined under quantum attacks.

3.3 symmetric-key cryptography

Symmetric algorithms use a shared secret key for encryption and decryption. They excel at high speed bulk encryption but require secure key distribution [KL14].

3.3.1 Design and Security

block ciphers like AES combine substitution (confusion) and permutation (diffusion) across multiple rounds, with a complex key schedule. A key of n bits yields a search space of 2^n, giving n bit classical security (Equation 3.1). Implementations must also resist side-channel attacks such as timing attack and power analysis to realize this security in practice [Koc96; MOP07].

$$\text{Security level (bits)} = n = \log_2(2^n) \tag{3.1}$$

3.3.2 Modes of Operation

To encrypt long messages, block ciphers use mode of operations such as CBC and GCM [Dwo01; Nat07]. CBC hides patterns across blocks. GCM adds integrity and authentication. ECB is insecure since identical plaintext blocks yield identical ciphertext (Figure 3.2).

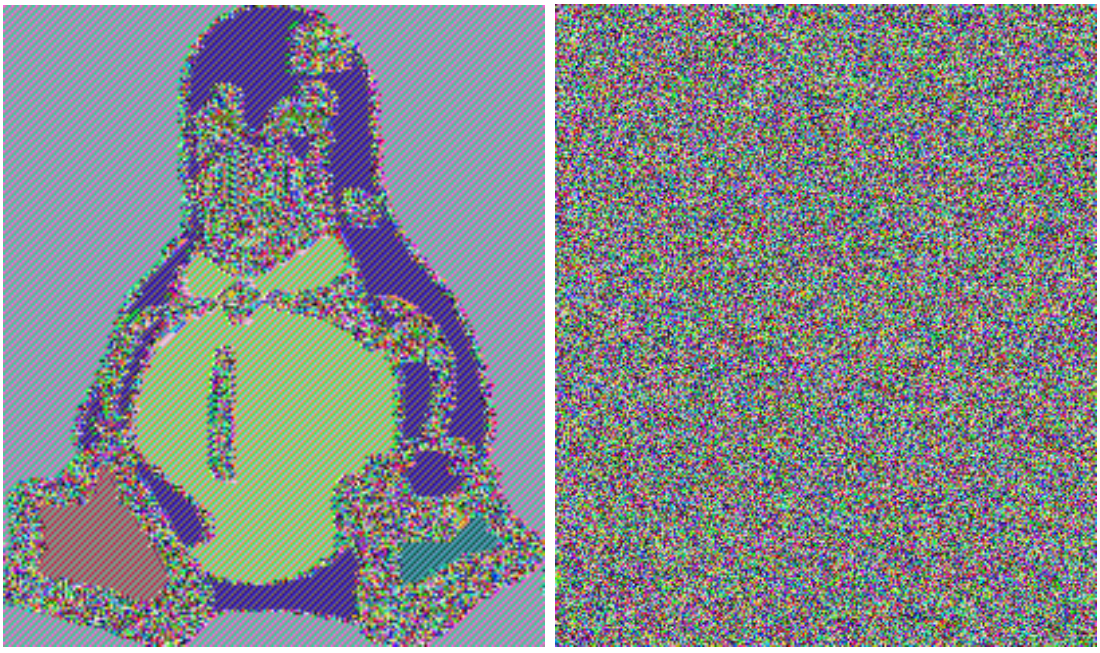

Figure 3.2: ECB mode leaks patterns while CBC mode conceals them.

3.4 public-key cryptography

Asymmetric cryptography solves key distribution by using a public private key pair. Its security depends on IFP/DLP type assumptions that quantum algorithms will break.

3.4.1 RSA

RSA publishes (n, e) and keeps (n, d) secret with $ed \equiv 1 \pmod{\phi(n)}$. Encryption $C = M^e \bmod n$. Decryption $M = C^d \bmod n$. Recommended sizes 2048 to 3072 bits [Nat20a]. Padding schemes such as OAEP prevent adaptive attacks [Ble98]. Shor's algorithm breaks RSA by factoring n efficiently.

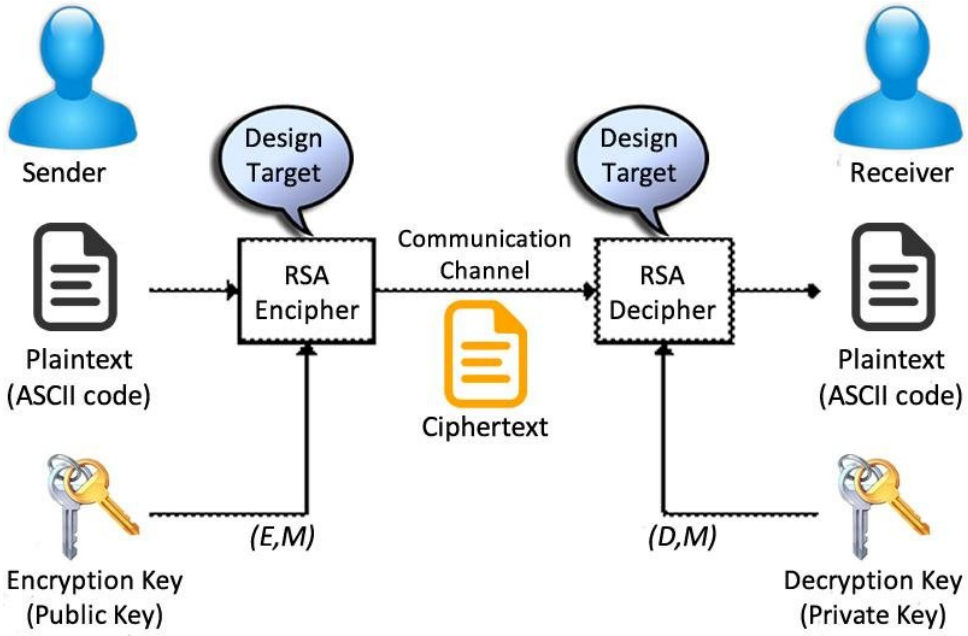

Figure 3.3: RSA encryption and decryption process.

3.4.2 Diffie-Hellman Key Exchange Key Exchange

DH enables two parties to agree on a shared secret $S = g^{ab} \bmod p$ without prior secrets [DH76]. Public values $p, g, A = g^a, B = g^b$. Shared secret $S = B^a = A^b$. Shor's algorithm recovers a, b, breaking DH.

3.4.3 ECC (Elliptic Curve Cryptography)

ECC achieves equivalent security to RSA with much smaller keys (a 256 bit ECC key is roughly equivalent to a 3072 bit RSA key) [Nat20a]. It relies on the hardness of ECDLP. Shor's algorithm also breaks ECC by solving ECDLP efficiently.

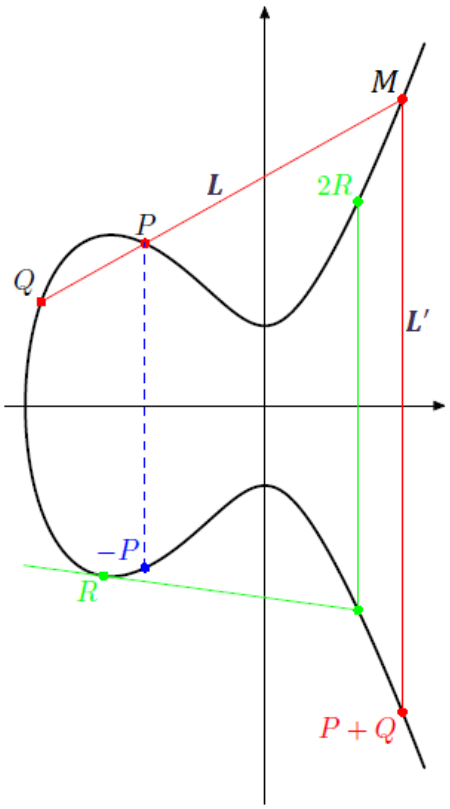

Figure 3.4: Elliptic curve point addition.

3.5 hash functions

Cryptographic hashes map any input to a fixed size digest. They ensure integrity and underpin signatures and MACs [MOV96b].

3.5.1 Security Requirements

A secure cryptographic hash function must satisfy several key properties. **preimage resistance** ensures that given a hash output h, it is computationally infeasible to find any input message m such that $H(m) = h$. **second preimage resistance** means that given an input m_1, it is computationally infeasible to find a distinct input $m_2 \neq m_1$ such that $H(m_1) = H(m_2)$. Finally, **collision resistance** dictates that it must be computationally infeasible to find any pair of distinct inputs m_1, m_2 where $m_1 \neq m_2$ but $H(m_1) = H(m_2)$. While older standards like MD5 and SHA-1 have been demonstrably broken with respect to collision resistance [**wang2005finding**], modern standards such as SHA-2 and SHA-3 are designed to uphold these properties and remain secure against known classical attacks [Nat15; OCo+20].

3.5.2 Merkle Damgard Construction

Many hashes follow the Merkle-Damgård construction design (Figure 3.5) iterating a compression function over message blocks.

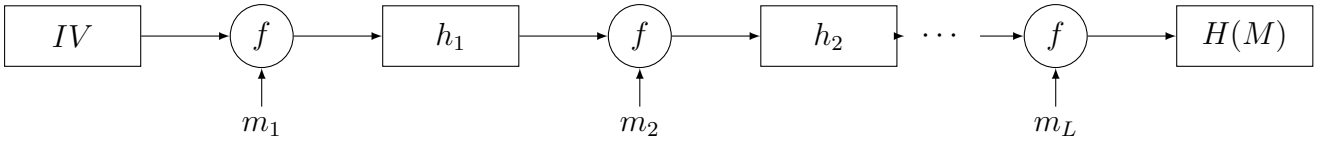

Figure 3.5: Merkle-Damgård construction used in hash functions like SHA-2.

3.6 digital signatures and PKI

digital signatures combine hashes with asymmetric keys to provide authentication integrity and non-repudiation [MOV96a].

3.6.1 Signing and Verification

The Signer computes $H(M)$ then $S = \text{Sign}_{K_{\text{priv}}}(H(M))$, and the Verifier checks $\text{Verify}_{K_{\text{pub}}}(S) \stackrel{?}{=} H(M)$.

3.6.2 PKI Trust Model

Certificate Authorities vouch for public keys enabling trust in TLS and HTTPS (Figure 3.6) [Sta17].

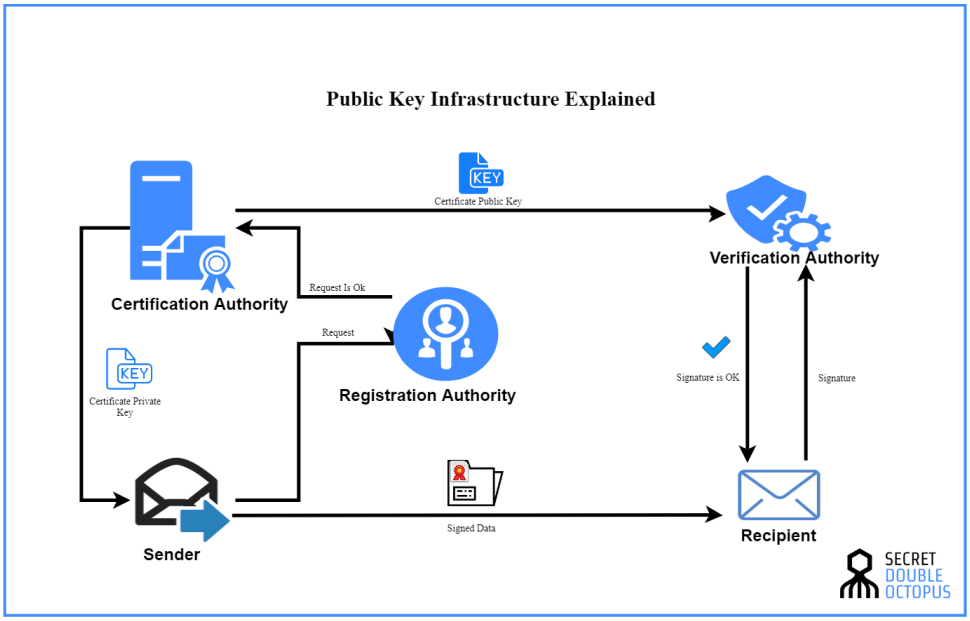

Figure 3.6: PKI trust hierarchy.

3.7 Protocol and Implementation Security

Strong algorithms alone do not guarantee security. Protocol design and implementation rigor are equally vital [BS20].

3.7.1 Protocol Vulnerabilities

Man in the Middle replay and downgrade attacks arise when authentication or freshness checks are missing.

3.7.2 Implementation Vulnerabilities

Side channel leaks such as timing power and electromagnetic analysis padding oracles memory errors and poor randomness can all break otherwise secure schemes.

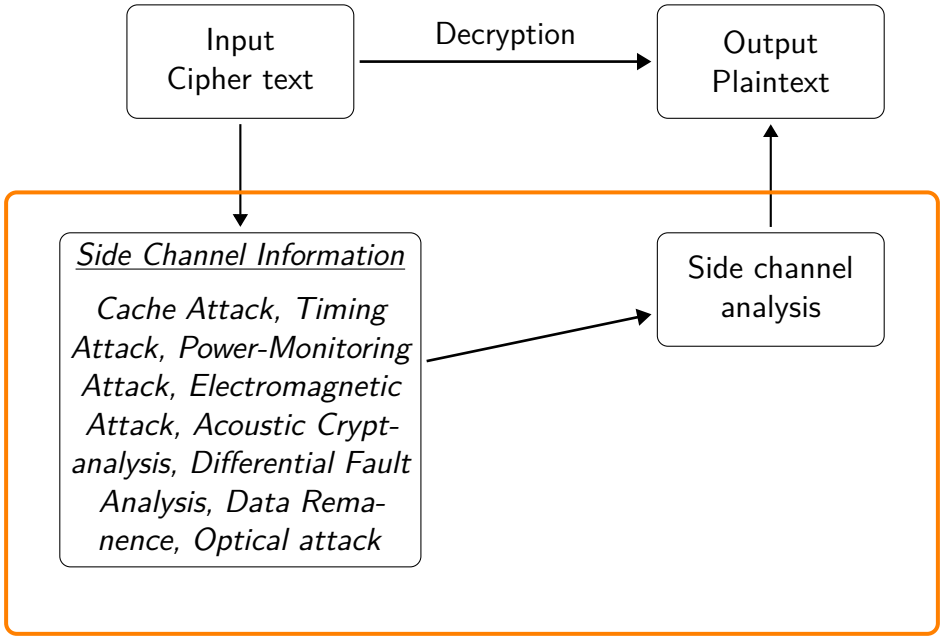

Figure 3.7: Side-Channel Attack Concept: Information leakage during cryptographic operations.

3.8 Conclusion: The Classical Security Model and Its Limits

This chapter surveyed how classical cryptography safeguards information using symmetric ciphers public key mechanisms cryptographic hashes and trust infrastructures all grounded in problems believed to resist classical computation. However these foundations become fragile under quantum scrutiny. Algorithms such as Shor's and Grover's expose their vulnerabilities. Understanding these limitations is critical before turning to quantum resilient alternatives in Chapter 6.

4

Classical versus Quantum Computing

This chapter contrasts the core principles forming the distintion between classical and quantum computing. By examining their fundamental units of information, operational models, and resulting computational power, we establish the context necessary to understand why quantum systems pose a unique and significant challenge to modern cryptography, a topic explored in detail in subsequent chapters.

4.1 Fundamental Differences in Information Representation

The most profound difference lies in the basic unit of information. Classical computers operate using **bits**, which adhere to binary logic, representing either a 0 or a 1 at any given time. Quantum computers, conversely, utilize **qubits** (quantum bits). As introduced in Chapter 2, a qubit is governed by the principles of quantum mechanics and can exist in a state of superposition, meaning it can represent a combination of both 0 and 1 simultaneously [NC10].

Mathematically, the state of a single qubit, $|\psi\rangle$, is described as a linear combination of the basis states $|0\rangle$ and $|1\rangle$:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{4.1}$$

where α and β are complex numbers known as probability amplitudes, satisfying the normalization condition $|\alpha|^2+|\beta|^2 = 1$. The values $|\alpha|^2$ and $|\beta|^2$ represent the probabilities of measuring the qubit in the state $|0\rangle$ or $|1\rangle$, respectively. This ability to exist in multiple states at once allows quantum systems to explore a vastly larger computational state space compared to classical systems of the same size.

The power of superposition becomes exponentially more potent when considering multiple qubits. A classical register of n bits can store exactly one of 2^n possible configurations. In stark contrast, a quantum register of n qubits can, through superposition, simultaneously represent all 2^n configurations.

Furthermore, as discussed in Section 2.1.2, qubits can exhibit entanglement, a uniquely quantum correlation where the state of multiple qubits becomes intrinsically linked, regardless of physical separation [NC10]. Measuring the state of one entangled qubit instantaneously influences the state of the others in a way that cannot be explained by classical physics. This non-local correlation is a crucial resource for many quantum algorithms and communication protocols.

Another fundamental distinction arises from the No-Cloning Theorem in quantum mechanics, which states that it is impossible to create an identical, independent copy of an arbitrary unknown quantum state [NC10]. This contrasts sharply with classical information, which can be copied freely. The no-cloning principle has profound implications for quantum information processing and security.

Table 4.1 provides a concise summary comparing these key characteristics of classical and quantum systems.

Table 4.1: Key Differences Between Classical and Quantum Computing

Feature	**Classical Computer**	**Quantum Computer**
Basic Unit	Bit	Qubit
State	0 or 1	Superposition of $\|0\rangle$ and $\|1\rangle$
Multiple Units	n bits store one of 2^n states	n qubits represent 2^n states simultaneously
Key Principles	Boolean Logic	Superposition, Entanglement, Interference
Operations	Logic Gates (AND, OR, NOT)	Quantum gates (Hadamard, CNOT, Pauli)
Reversibility	Generally not reversible	Reversible (Unitary operations)
Copying Data	Easy	Impossible (No-Cloning Theorem)
Error Handling	Mature Error Correction Codes	Complex Quantum Error Correction
Key Algorithms	Sorting, Searching (Linear/Log)	Shor's algorithm, Grover's algorithm
Complexity Class	P, BPP	BQP
Current Status	Mature, Ubiquitous	Emerging, NISQ Era, Specialized

4.2 Computational Models and Information Processing

Building on these differences in information representation, the computational models themselves are fundamentally distinct.

4.2.1 Classical Computing Model

Classical computation largely follows the Turing machine model or the von Neumann architecture. Information is processed sequentially (or in parallel across multiple classical cores) using deterministic logic gates operating on bits. The state of an n-bit classical system is described by a single binary string of length n. While powerful for a vast range of tasks, this model faces inherent limitations when tackling problems whose complexity grows exponentially with input size, such as factoring large numbers or simulating complex quantum systems.

4.2.2 Quantum Computing Model

Quantum computation operates within the framework of quantum mechanics, typically visualized using the quantum circuit model. Qubits states evolve through the application of quantum gates, which are mathematically represented by unitary transformations acting on the state vector within a complex Hilbert space. The state of an n-qubit system is described by a vector in this 2^n-dimensional space:

$$|\psi\rangle = \sum_{i=0}^{2^n-1} \alpha_i |i\rangle, \quad \text{where} \sum_{i=0}^{2^n-1} |\alpha_i|^2 = 1 \tag{4.2}$$

Each $|i\rangle$ corresponds to a classical bit string (e.g., $|011\rangle$), and α_i is its complex amplitude.

The quantum circuit model leverages superposition and entanglement, allowing quantum gates to simultaneously transform all 2^n components of the state vector. This leads to what is often termed quantum parallelism, where a single quantum operation effectively performs a massive parallel computation across the entire state space. However, this parallelism is not directly accessible. Upon measurement, the quantum state collapses probabilistically into just one of the basis states $|i\rangle$ according to the probability $|\alpha_i|^2$. The true power of quantum algorithms, therefore, lies not just in parallelism but in harnessing quantum interference—carefully choreographing the unitary evolution to constructively amplify the amplitudes of desired outcomes and destructively cancel the amplitudes of undesired ones before measurement [NC10].

4.3 Comparing Computational Power and Complexity

These distinct operational models lead to different computational capabilities, which can be formally compared using computational complexity theory. This theory classifies problems based on the resources (like time or memory) required to solve them as the input size grows.

Key complexity classes relevant to this comparison include:

- **P (Polynomial time):** Problems solvable by a deterministic classical computer in time polynomial in the input size. These are considered "efficiently solvable" classically.
- **NP (Nondeterministic Polynomial time):** Problems for which a proposed solution can be verified efficiently by a classical computer. It is famously unknown whether P = NP.
- **BPP (Bounded-error Probabilistic Polynomial time):** Problems solvable by a probabilistic classical computer in polynomial time with a bounded error probability (e.g., error $< 1/3$). BPP represents problems efficiently solvable by practical randomized classical algorithms and is often considered the class of "efficiently solvable" problems in the classical world.
- **BQP (Bounded-error Quantum Polynomial time):** Problems solvable by a quantum computer in polynomial time with a bounded error probability. This class captures the power of efficient quantum computation.

[AB09]

The known relationships between these classes are $\mathbf{P} \subseteq \mathbf{BPP} \subseteq \mathbf{BQP}$. The crucial insight for cryptography is that BQP contains problems like integer factorization and the discrete logarithm problem, which are essential to the security of RSA, ECC, and Diffie-Hellman. These problems are strongly believed *not* to be in BPP, meaning they are considered intractable for classical computers. However, Shor's algorithm demonstrates they are efficiently solvable by quantum computers, placing them firmly within BQP [Sho97]. (The relationship between BQP and NP remains an open research question.)

While Shor's algorithm provides an exponential speedup for specific structured problems, Grover's algorithm (Section 2.3.3) offers a more general quadratic speedup for unstructured search problems. Classically, finding an item in an unsorted database of size N takes $O(N)$ time on average. Grover's algorithm achieves this in $O(\sqrt{N})$ quantum time. While significant, this quadratic speedup typically does not move problems between major complexity classes (like placing an NP-complete problem into BQP) in the same way Shor's algorithm does for factoring. However, as discussed later, it still impacts the practical security parameters of symmetric ciphers and hash functions (Chapter 5).

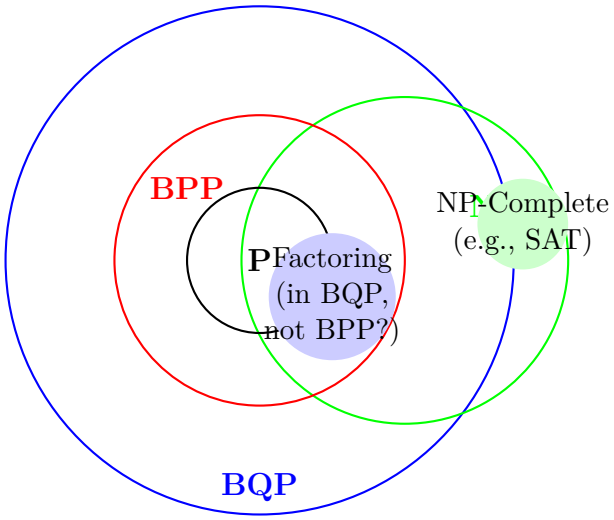

Figure 4.1: Conceptual Venn diagram of complexity classes P, NP, BPP, and BQP, illustrating their suspected relationships and the placement of key problems like Factoring.

In essence, classical computers are restricted to definite 0s or 1s, whereas quantum computers leverage the ability to represent both simultaneously, enabling them to address certain problems intractable for classical machines.

5

Quantum Computing's Impact on Cryptography

Building upon the foundations of quantum computing principles (Chapter 2), the established landscape of classical cryptography (Chapter 3), and the fundamental differences between computational models (Chapter 4), this chapter analyzes the specific and profound impact of quantum algorithms on the security of currently deployed cryptographic systems. We delve into the mechanisms and consequences of Shor's algorithm for public-key systems and Grover's algorithm for symmetric primitives and hashes, quantifying their threat. Furthermore, we examine the strategic implications, such as the "Store Now, Decrypt Later" scenario, underscoring the urgent need for quantum-resistant solutions explored in subsequent chapters.

5.1 Shor's Algorithm: Breaking Public-Key Cryptography

In 1994, Peter Shor introduced an algorithm that fundamentally altered the landscape of cryptographic security [Sho94]. Shor's algorithm represents the single most significant quantum threat to modern cryptography by achieving an exponential speedup over classical algorithms for two foundational problems: the Integer Factorization Problem (IFP) and the Discrete Logarithm Problem (DLP). This revolutionary capability emerges from its elegant application of quantum phenomena, particularly the QFT (Section 2.3.1) and QPE (Section 2.3.2).

5.1.1 The Quantum Advantage in Integer Factorization

The security of much modern digital infrastructure relies on computational problems believed intractable for classical computers. The Integer Factorization Problem (IFP)—finding the prime factors of large composite numbers—stands as the cornerstone of RSA cryptography. While the best classical algorithms require sub-exponential time (like the GNFS), Shor's algorithm achieves factorization in polynomial time, dramatically reducing the resources required, as illustrated in Figure 5.1.

The algorithm's core strength lies in transforming the factorization problem into a period-finding problem. By preparing a quantum superposition, applying modular exponentiation, and using the QFT, Shor's algorithm efficiently discovers the period of

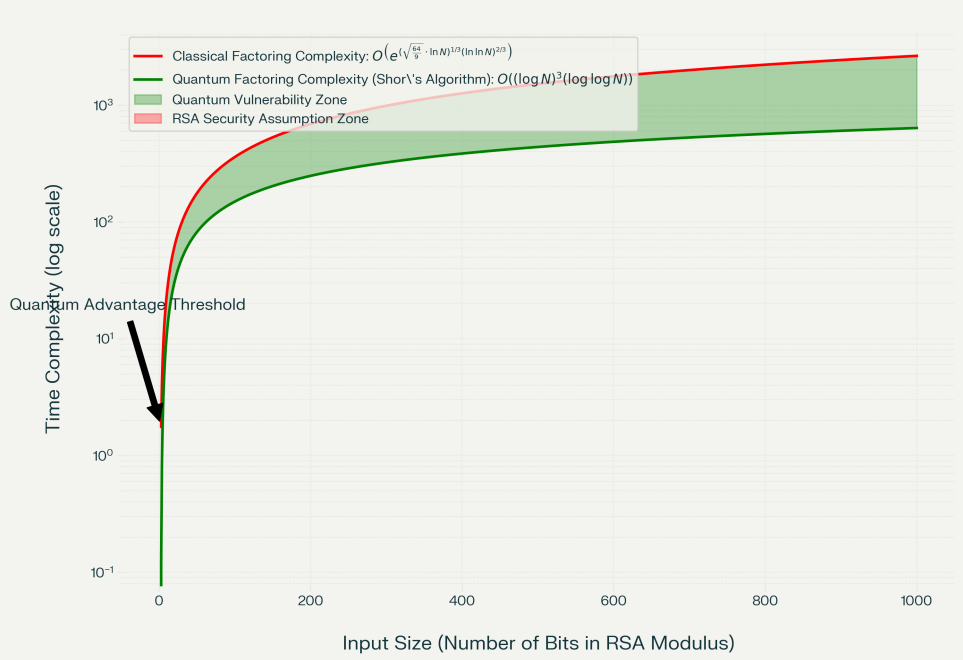

Figure 5.1: **Classical vs Quantum Factoring Complexity.** This graph shows the sub-exponential complexity of classical factoring (e.g., GNFS, red curve) versus the polynomial complexity of Shor's quantum algorithm (green curve, $O((\log N)^3)$). The widening gap (green shading) illustrates the vulnerability of classical systems like RSA to quantum computers, whose advantage grows significantly with key size (N).

this function, which then allows efficient calculation of the factors via classical methods.

5.1.2 Cryptographic Systems Under Threat

The exponential speedup provided by Shor's algorithm directly undermines the security assumptions of virtually all widely deployed public-key cryptosystems [BL17]:

- **RSA Encryption and Signatures:** The security of RSA (Section 3.4.1) relies entirely on the difficulty of factoring the public modulus N. Shor's algorithm makes factoring N feasible, allowing an attacker with a CRQC to derive the private key d from the public key (N, e), breaking the system completely, regardless of key length.
- **DH and DSA:** These protocols (Section 3.4.2) depend on the classical hardness of the Discrete Logarithm Problem (DLP) (DLP) in finite fields. Shor's algorithm provides an efficient quantum method to solve the DLP, rendering DH and the related DSA insecure.
- **Elliptic Curve Cryptography (ECC):** ECC systems like ECDH and ECDSA (Section 3.4.3) rely on the hardness of the ECDLP. Shor's algorithm, adapted for elliptic curve groups, also solves ECDLP efficiently, compromising the security of these compact and widely used schemes.

The security implications extend beyond theoretical concerns. Current resource estimates, while substantial (e.g., millions of physical qubits for RSA-2048 [GE21]),

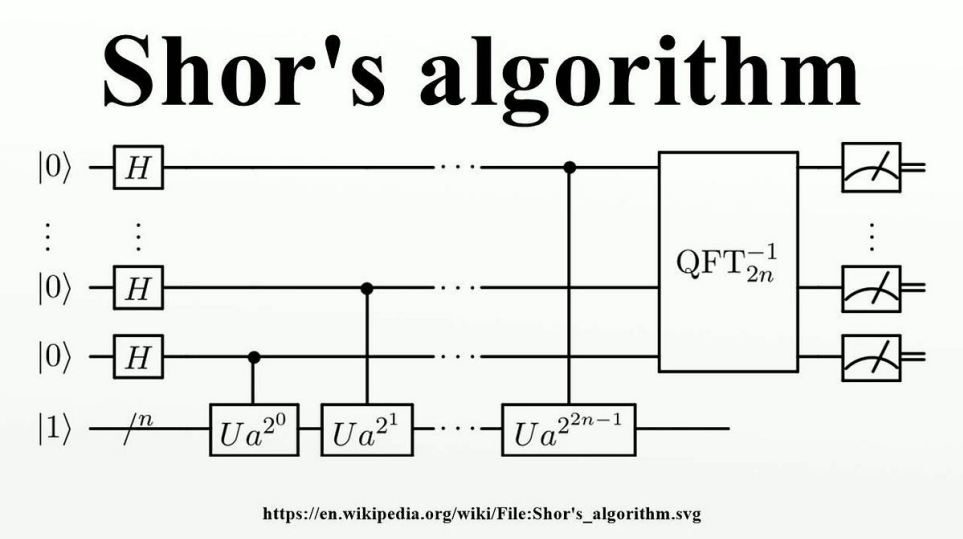

Figure 5.2: Conceptual Diagram of Shor's Algorithm Steps: Utilizes quantum parallelism and QFT for efficient period-finding, enabling factorization.

establish a clear trajectory where these cryptosystems become insecure, driving the need for alternatives.

5.1.3 Broader Implications for Digital Security

The threat from Shor's algorithm cascades through the entire digital ecosystem. Core internet security protocols, including TLS (securing HTTPS), SSH (secure remote access), IPsec (VPNs), and the PKI infrastructure built upon these algorithms, become vulnerable.

Unlike classical vulnerabilities often mitigated by patches or larger keys, the quantum threat is fundamental. No increase in RSA or ECC key size offers long-term protection against Shor's algorithm. This necessitates the development and deployment of fundamentally different, quantum-resistant cryptographic approaches.

While Shor's algorithm poses an existential threat to public-key cryptography, another quantum algorithm, Grover's, presents a different kind of challenge to symmetric primitives.

5.2 Grover's Algorithm: Weakening Symmetric Cryptography and Hashes

In contrast to Shor's exponential speedup for specific problems, Grover's algorithm [Gro96] offers a more general *quadratic* speedup for unstructured search problems. This impacts cryptographic primitives whose security relies partly on the difficulty of exhaustive search: symmetric-key ciphers and . It operates by amplifying the probability amplitude of target states within a quantum superposition (see Section 2.3.3).

5.2.1 Mechanism: Quantum Amplitude Amplification

Grover's algorithm initializes a system into an equal superposition of all N possible states in a search space. It then iteratively applies an "Grover oracle" U_f that marks target state(s) (e.g., by phase inversion) and a "Grover diffusion operator" operator U_s that amplifies the amplitude of marked states while reducing others (performing amplitude amplification). This process, shown conceptually in Figure 5.3, finds a target state with high probability in about $O(\sqrt{N})$ quantum queries, compared to $O(N)$ classical queries for unstructured search [NC10].

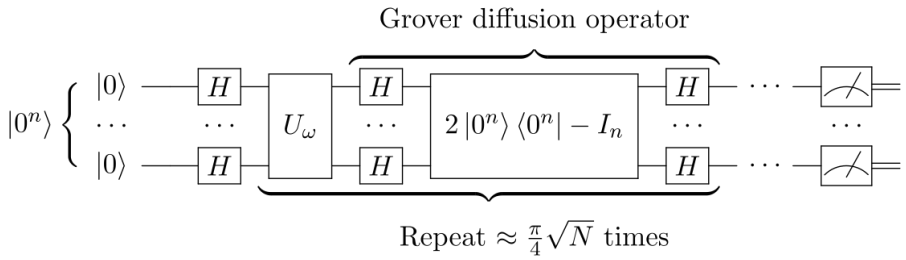

Figure 5.3: Conceptual quantum circuit for Grover's algorithm, showing the iterative application of the Oracle (U_f) and Diffusion (U_s) operators to amplify the target state's amplitude.

5.2.2 Impact on Symmetric Keys and Hash Functions

The quadratic speedup impacts the effective security of symmetric primitives [BL17]:

- **Symmetric Key Search:** Recovering an n-bit symmetric key (e.g., for AES, Section 3.3) classically takes $O(2^n)$ operations via brute force. Grover's reduces the quantum complexity to $O(\sqrt{2^n}) = O(2^{n/2})$ operations. This effectively halves the bit security against quantum brute-force search. For example, AES-128 (128-bit classical security) offers only about $128/2 = 64$ bits of security against Grover's algorithm.
- **Hash Function Pre-images:** Finding an input x for a target hash output y (i.e., $H(x) = y$, Section 3.5) is an unstructured search. For an ideal n-bit hash, Grover's reduces the complexity of a preimage attack from classical $O(2^n)$ to quantum $O(2^{n/2})$.
- **Hash Function Collisions:** Finding distinct inputs $x_1 \neq x_2$ where $H(x_1) = H(x_2)$ (a collision attack) classically uses the birthday attack ($O(2^{n/2})$ complexity). Quantum algorithms improve on this, potentially reaching $O(2^{n/3})$ complexity in some models [BL17], although the practical advantage over the classical birthday bound is still debated for standard hashes. The primary defense remains using a hash function with a large enough output size (n) such that $n/2$ provides sufficient collision resistance.

Crucially, Grover's algorithm weakens these primitives but does not break them exponentially like Shor does for public-key systems. The standard mitigation is to increase key or hash output sizes. For instance, using AES-256 provides approximately 128 bits of quantum security against key search. Using SHA-256 or SHA-384 provides 128 or 192 bits of quantum security against , respectively, and their collision resistance remains dominated by the classical birthday attack bound.

Having examined the distinct impacts of Shor's and Grover's algorithms, we can now synthesize the overall threat landscape and discuss the timeline for action.

5.3 Synthesizing the Threat: Security Levels and Comparison

The differing impacts of Shor's and Grover's algorithms necessitate distinct mitigation strategies and define the tiered nature of the quantum threat.

5.3.1 Required Security Level Adjustments

To maintain a target security level (e.g., 128 bits, corresponding roughly to NIST security levels 1 or 3) against both classical and quantum adversaries, parameters must be chosen carefully:

- **Shor-Vulnerable Systems (RSA, DH, ECC):** Offer essentially zero security against a CRQC. They must be replaced entirely by PQC alternatives (Chapter 6).
- **Grover-Vulnerable Systems (Symmetric Ciphers, Hashes):** To achieve k bits of quantum security against brute-force key search or , the key or hash output size generally needs to be at least $2k$ bits.
  - For 128-bit quantum security (key search) $\implies$ Use AES-256.
  - For 128-bit quantum security (hash pre-image) $\implies$ Use SHA-256 or larger.
  - For collision resistance (dominated by classical $O(2^{n/2})$ birthday attack bound), SHA-256 (providing 128-bit collision resistance) is often deemed sufficient, though larger hashes offer more margin.

This leads to the practical rules: for public-key crypto, Quantum Security ≈ 0; for symmetric/hash crypto, Quantum Security $\approx$ Classical Security / 2 (for search-based attacks).

5.3.2 Comparative Impact Summary

Table 5.1 summarizes the estimated security levels of common cryptographic primitives against classical and quantum attacks, reflecting current understanding and the impacts discussed [Ala+22].

Table 5.1: Estimated Security Levels (in bits) Against Classical and Quantum Attacks

Algorithm/Parameter	**Classical Security (bits)**	**Quantum Security (bits)**
RSA-2048	∼112	≈ 0 (Shor)
RSA-3072	∼128	≈ 0 (Shor)
ECC P-256 / secp256k1	∼128	≈ 0 (Shor)
ECC P-384 / secp384r1	∼192	≈ 0 (Shor)
AES-128	128	≈ 64 (Grover - Key Search)
AES-192	192	≈ 96 (Grover - Key Search)
AES-256	256	≈ 128 (Grover - Key Search)
SHA-256 (Pre-image)	256	≈ 128 (Grover - preimage attack)
SHA-256 (Collision)	128	≈ 128* (Classical birthday attack)
SHA-384 (Pre-image)	384	≈ 192 (Grover - preimage attack)
SHA-384 (Collision)	192	≈ 192* (Classical birthday attack)
SHA-512 (Pre-image)	512	≈ 256 (Grover - preimage attack)
SHA-512 (Collision)	256	≈ 256* (Classical birthday attack)

*Collision resistance against quantum attacks is generally limited by the $O(2^{n/2})$ classical birthday attack complexity. Quantum collision finding algorithms offer limited practical advantage here.

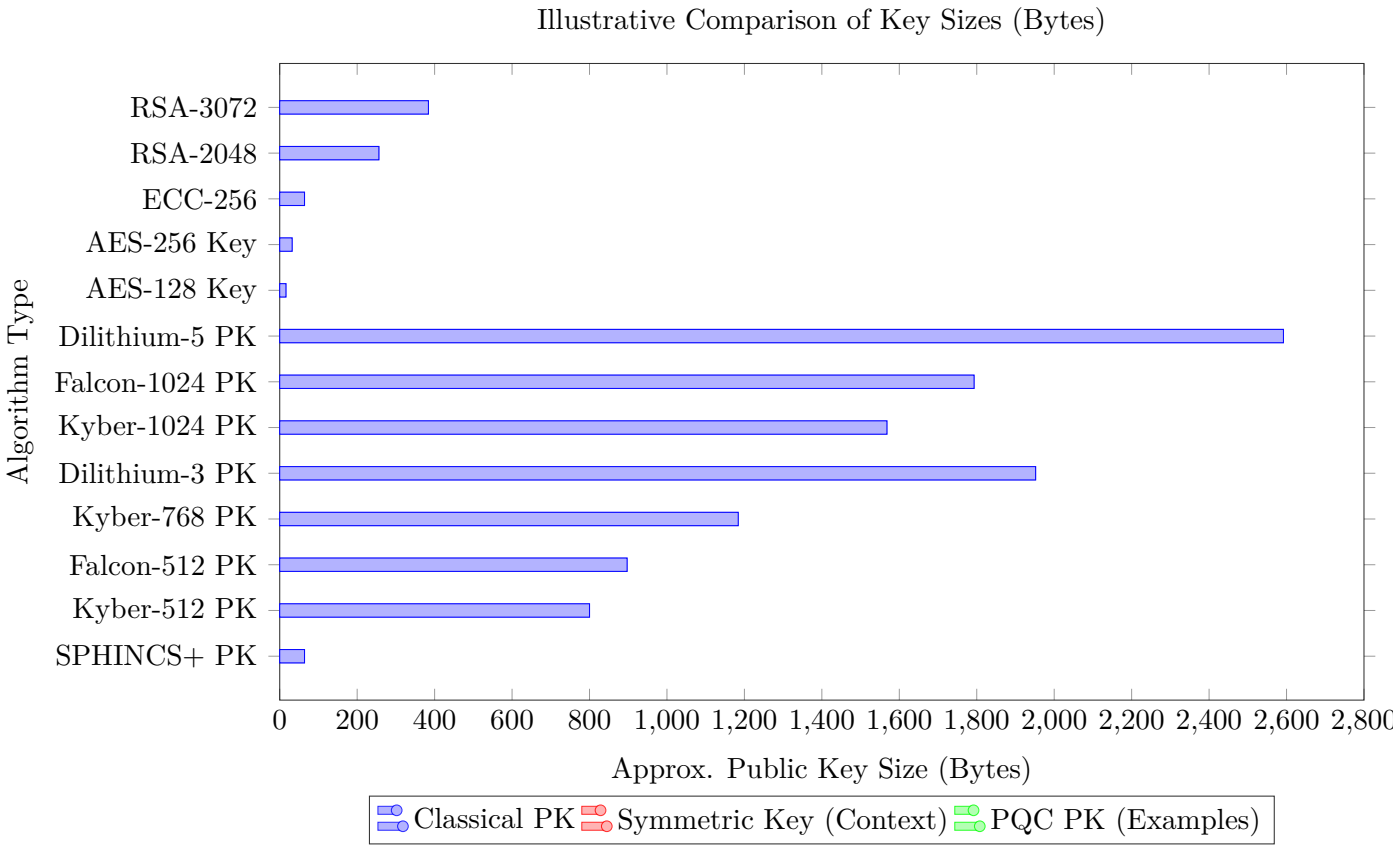

Figure 5.4: Illustrative comparison of public key sizes (Bytes) for classical and example post-quantum algorithms. PQC sizes vary significantly. Symmetric key sizes shown for context (not public keys). *SPHINCS+ public key is small, but signatures are very large. Classic McEliece public keys are much larger still (hundreds of KB) and not shown to scale.

5.4 Timeline and the Urgency of Transition

Predicting the arrival of a CRQC capable of breaking current public-key cryptography is uncertain, yet the risk necessitates proactive migration.

5.4.1 Estimates and Uncertainties

Expert estimates for a CRQC often range from 10 to 20+ years, but significant uncertainty exists [Mos18; Moo+24]. Hardware challenges remain, particularly achieving fault-tolerance via QEC at the scale needed for Shor's algorithm against large keys [Pre18]. Progress continues, but the timeline is difficult to pinpoint, making preparation essential regardless of the exact date.

5.4.2 The "Store Now, Decrypt Later" (SNDL) Threat

The SNDL attack vector creates immediate urgency. Adversaries can capture encrypted data transmitted today (e.g., via TLS) and store it. Once a CRQC is available, this data, if still sensitive, can be decrypted retrospectively. This threat is independent of when the CRQC arrives; it depends on the required confidentiality lifetime of the data.

Mosca's Inequality highlights this urgency:

$$X + Y > Z \tag{5.1}$$

where X = Security Shelf Life of data, Y = Migration Time to PQC, Z = Time until CRQC exists. If $X + Y > Z$, data encrypted today is already at risk. Since Y can be 5-15 years for large organizations and X can be decades, the migration (Y) must begin long before Z arrives [Mos18].

5.5 Conclusion: The Imperative for Quantum Resistance

This chapter has demonstrated that quantum computing poses a concrete and potentially catastrophic threat to current cryptographic standards. Shor's algorithm fundamentally breaks the mathematical foundations of widely used public-key cryptography (RSA, DH, ECC), while Grover's algorithm significantly weakens symmetric-key cryptography algorithms (AES) and (SHA-2, SHA-3) by enabling faster brute-force style attacks (, key search). The looming SNDL threat transforms this from a future possibility into a present-day risk assessment requirement for data with long-term confidentiality needs.

6

Quantum-Resistant Cryptography

The demonstration in Chapter 5 that quantum computers, particularly leveraging Shor's algorithm, can efficiently break currently deployed public-key cryptography systems like RSA and ECC, necessitates a fundamental shift towards cryptographic algorithms secure against both classical and quantum adversaries. This field is known as post-quantum cryptography (PQC) or quantum-resistant cryptography. PQC does not rely on quantum phenomena itself for security; rather, it employs classical cryptographic techniques based on mathematical problems believed to be computationally hard even for large-scale quantum computers [BL17]. This chapter introduces the primary families of PQC algorithms, discusses the ongoing standardization efforts, and touches upon key implementation considerations.

6.1 Introduction to Post-Quantum Cryptography

Recognizing the threat posed by quantum computers, the cryptographic community has been actively researching and developing post-quantum cryptography (PQC)—algorithms designed to be secure against attacks from both classical and sufficiently powerful quantum computers [BL17].

6.2 Major Families of Post-Quantum Cryptography

Post-quantum algorithms are typically categorized into several main families based on the underlying mathematical problems they rely on for security. These problems are believed to be hard for both classical and quantum computers to solve efficiently. The primary families include:

6.2.1 Lattice-Based Cryptography

Among the diverse approaches to post-quantum cryptography, schemes based on mathematical lattices have emerged as particularly prominent and versatile. Their security is rooted in the presumed computational difficulty of solving certain geometric problems defined over these lattices, problems believed to remain hard even for quantum computers [BL17].

Understanding Lattices

Imagine a regular, repeating pattern of points extending infinitely in multiple dimensions. This is the essence of a mathematical lattice. More formally, given a set of linearly independent vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ in n-dimensional space (called the basis vectors), the lattice L generated by this basis is the set of all possible points you can reach by taking integer linear combinations of these basis vectors:

$$L = \left\{ \sum_{i=1}^{n} a_i \mathbf{v}_i \mid a_i \in \mathbb{Z} \right\}$$

where $\mathbb{Z}$ represents the set of all integers (positive, negative, and zero). Think of a 2D lattice like an infinite sheet of graph paper, but where the grid lines might be skewed or stretched depending on the choice of basis vectors (as conceptually shown in Figure 6.1). The same basis vectors can generate the same infinite set of points, making finding the "best" basis a non-trivial task.

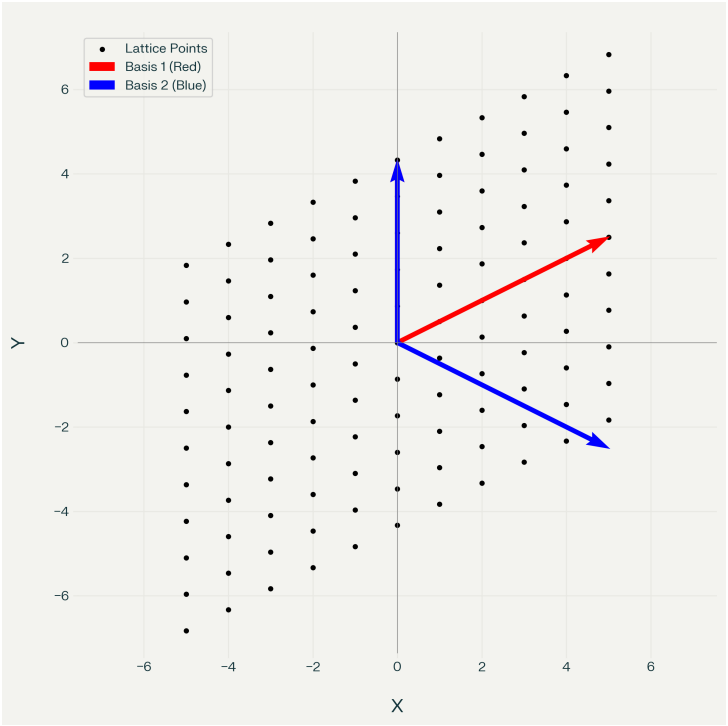

Figure 6.1: A 2D lattice showing two different bases (red and blue vectors) that generate the same set of points.

Hard Problems on Lattices

While lattices have a regular structure, certain geometric problems defined on them become computationally very hard, especially as the number of dimensions increases. The security of lattice-based cryptosystems typically relies on the difficulty of problems like:

- **Shortest Vector Problem (SVP):** Finding the non-zero lattice point closest to the origin (the zero vector). While easy to visualize in 2D (see Figure 6.2), finding this shortest vector becomes exponentially difficult in high dimensions for known classical and quantum algorithms.
- **Closest Vector Problem (CVP):** Given a target point $\mathbf{t}$ in the space (which may not be a lattice point itself), find the lattice point $\mathbf{p}$ that is closest to $\mathbf{t}$. Again, this is computationally hard in high dimensions (see Figure 6.2).

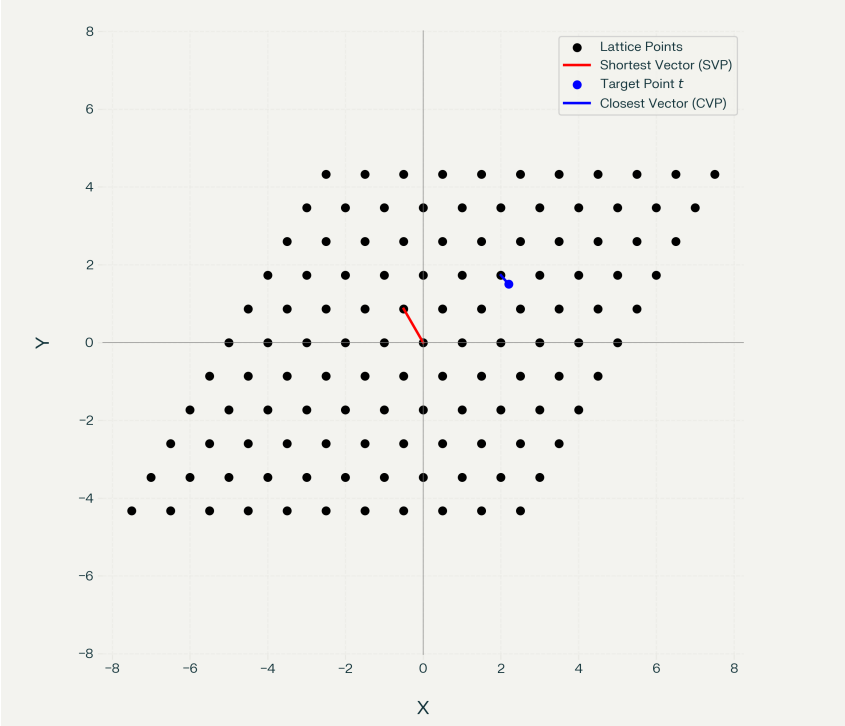

Figure 6.2: Illustration of Lattice Problems: Shortest Vector Problem (SVP - find shortest vector, red) and Closest Vector Problem (CVP - find lattice point nearest target t, blue).

While SVP and CVP are fundamental, modern lattice cryptography often relies more directly on the **LWE (Learning With Errors)** problem, introduced by Oded Regev [Pei16]. LWE provides a powerful framework for building cryptographic schemes.

The Learning With Errors (LWE) Problem

Imagine trying to determine a secret set of numbers (a secret vector $\mathbf{s}$) when you are only given clues in the form of approximate linear equations. This is the core idea behind LWE. An adversary is given access to multiple samples $(\mathbf{a}_i, b_i)$. Each $\mathbf{a}_i$ is a known, randomly chosen vector, and b_i is approximately equal to the inner product (dot product) of $\mathbf{a}_i$ and the secret $\mathbf{s}$, but with a small amount of random "noise" or "error" e_i added:

$$b_i \approx \langle \mathbf{a}_i, \mathbf{s} \rangle + e_i \pmod{q}$$

Here, q is a modulus (typically a prime number), and the error e_i is drawn from a specific probability distribution, usually centered around zero with a small standard deviation (like a discrete Gaussian distribution). The challenge for the adversary is to recover the secret vector $\mathbf{s}$ using only the publicly known $\mathbf{a}_i$ vectors and the noisy results b_i.

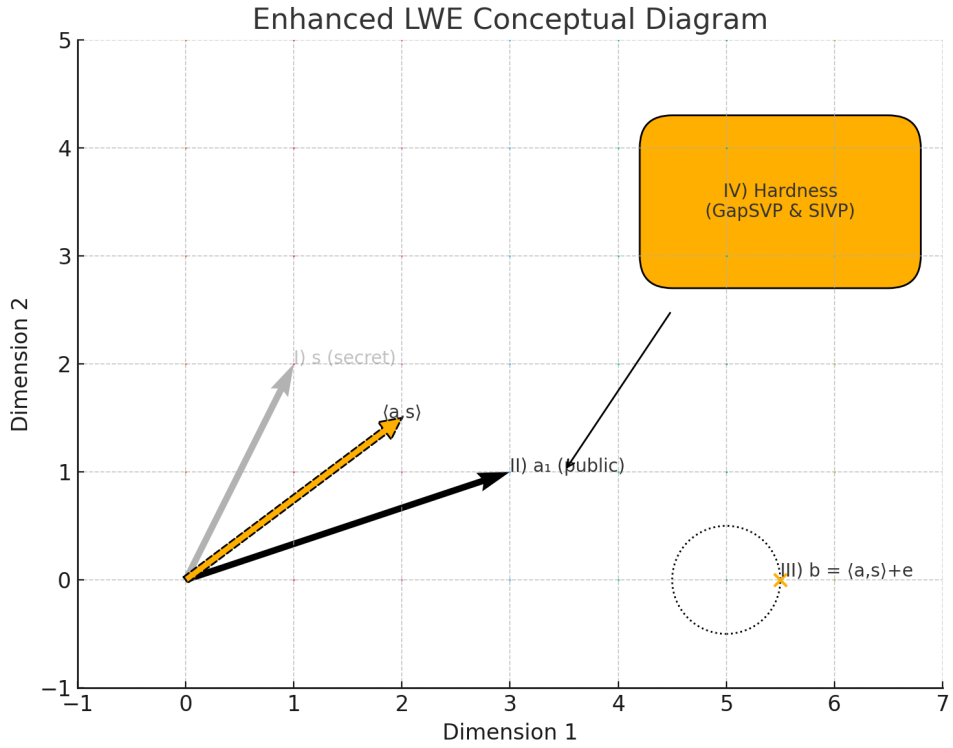

Figure 6.3: Conceptual illustration of the LWE problem: Recovering the secret $\mathbf{s}$ from public samples $(\mathbf{a}_i, b_i)$ where b_i is the inner product $\langle \mathbf{a}_i, \mathbf{s} \rangle$ plus small noise e_i.

Without the noise ($e_i = 0$), finding $\mathbf{s}$ would be a standard linear algebra problem, easily solvable classically. However, the addition of these small, random errors fundamentally changes the problem's nature, making it computationally difficult. It essentially hides the secret $\mathbf{s}$ within the uncertainty introduced by the noise. The security relies on the fact that distinguishing the distribution of $(\mathbf{a}_i, b_i)$ pairs from a distribution where b_i is simply chosen uniformly at random is computationally hard.

Quantum Resistance of LWE

The reason LWE is believed to be resistant to quantum computers stems from this reliance on noise and the lack of apparent exploitable structure. Unlike integer factorization or the discrete logarithm problem, which possess a periodic structure that Shor's algorithm can efficiently find using the QFT, the LWE problem does not seem to exhibit such periodicity [BL17]. The randomness introduced by the errors effectively masks any underlying algebraic structure that current quantum algorithms might target. While quantum computers might offer some speedup for lattice problems (e.g., using Grover's algorithm for search aspects), they do not provide the exponential advantage seen with Shor's algorithm against classical public-key systems. Therefore, LWE and related lattice problems form a strong foundation for building PQC schemes.

Efficiency Variants: Ring-LWE and Module-LWE

While standard LWE provides strong security guarantees, its practical implementations can lead to relatively large keys and slower computations. To improve efficiency, structured variants were developed:

- **Ring-LWE:** Instead of working with generic vectors and matrices, Ring-LWE operates within the richer algebraic structure of polynomial rings. This allows mathematical operations to be performed much more efficiently using tools like the Number Theoretic Transform (NTT), significantly reducing key sizes and speeding up computations.
- **Module-LWE:** This variant acts as an intermediate step between standard LWE and Ring-LWE. It works over mathematical structures called modules, which generalize the ring structure. Module-LWE aims to retain most of the efficiency benefits of Ring-LWE while potentially offering security guarantees closer to the more general (and arguably more conservative) standard LWE problem.

Many leading PQC candidates, including those selected by NIST, leverage the efficiency gains offered by Module-LWE [LPR24]. However, the introduction of this extra algebraic structure necessitates ongoing security analysis to ensure no new vulnerabilities are inadvertently created.

Strengths and Weaknesses

Lattice-based cryptography offers a compelling package for the post-quantum era. Its **strengths** include strong theoretical security foundations often linked to worst-case problem hardness, versatility in building different cryptographic tools like KEMs and signatures, and generally efficient performance compared to some other PQC families. The underlying LWE concept is also relatively accessible compared to the intricacies of elliptic curve mathematics.

However, it is not without its **weaknesses**. A primary concern is the size overhead: public keys, ciphertexts, and signatures are typically larger than their counterparts in traditional ECC systems, which can pose challenges for protocols sensitive to bandwidth or storage (like TLS or constrained IoT devices). Selecting appropriate parameters (dimensions, modulus, noise level) to ensure both security and efficiency is a complex task requiring specialized knowledge. Furthermore, like all cryptographic implementations, lattice-based schemes require careful engineering to prevent vulnerabilities arising from side-channel attacks. Finally, while the underlying mathematical problems are well-studied, their widespread cryptographic application is newer than RSA or ECC, meaning they have faced less long-term, real-world cryptanalytic scrutiny, although the intensive NIST evaluation process has significantly mitigated this concern.

NIST Standardized Examples

The practical potential of lattice-based cryptography is underscored by its strong representation in the NIST PQC standardization outcome:

- **CRYSTALS-Kyber:** Chosen as the primary standard for KEMs, Kyber leverages Module-LWE to provide a well-rounded balance of security, operational speed, and key/ciphertext sizes suitable for general-purpose key exchange. Kyber's design specifically targets efficiency and ease of implementation while maintaining robust security against known classical and quantum attacks. Its operations involve polynomial arithmetic in a ring modulo q, combined with techniques to generate shared secrets that are computationally indistinguishable from random values for an adversary without the private key. The security levels defined by NIST (Levels 1, 3, 5) correspond to varying parameter sets, offering different trade-offs between security margin and performance overhead [Nat23a]. The selection of Kyber reflects confidence in the underlying Module-LWE problem and its suitability for widespread deployment in protocols like TLS.

- **CRYSTALS-Dilithium:** Selected as a primary standard for digital signatures, Dilithium is also based on Module-LWE and offers efficient signing and verification, making it a strong general-purpose signature scheme.

- **Falcon:** Another primary signature standard, Falcon is built upon the NTRU lattice problem (closely related to Ring-LWE). Its main advantage is producing significantly smaller signatures than Dilithium, which is valuable in size-constrained scenarios. However, its implementation is more complex, notably requiring floating-point arithmetic during the signing process.

These standardized algorithms provide developers with concrete, rigorously analyzed options for implementing quantum-resistant key establishment and digital authentication.

6.2.2 Hash-Based Cryptography

Hash-based cryptography, particularly for digital signatures, relies solely on the security properties of cryptographic hash functions (like SHA-2 or SHA-3, see Section 3.5). Since hash functions are generally believed to be more resistant to quantum attacks (only quadratically weakened by Grover's algorithm), hash-based signatures offer strong security guarantees with minimal mathematical assumptions beyond the hash function's security [HBP24].

Early hash-based signatures (e.g., Lamport signatures) were one-time signatures (OTS), meaning a key pair could only sign a single message securely. Modern schemes build upon these using Merkle trees to combine many OTS keys into a single public key that can sign multiple messages. There are two main types:

- **Stateful Signatures (e.g., XMSS, LMS):** Offer smaller signatures and faster signing but require the signer to securely maintain state (e.g., tracking which OTS key was used last). Using the same OTS key twice breaks security completely.
- **Stateless Signatures (e.g., SPHINCS+):** Eliminate the need for state management, making them safer to deploy in practice. However, this comes at the cost of significantly larger signature sizes (often tens of kilobytes) and slower signing performance compared to stateful schemes or lattice-based signatures.

NIST PQC Standardization selected **SPHINCS+** as the standard for stateless hash-based signatures due to its strong security foundations and avoidance of state management risks.

6.2.3 Code-Based Cryptography

Code-based cryptography, pioneered by McEliece [McE78], relies on the difficulty of decoding a general linear code, which is an NP-hard problem. The foundational scheme is the McEliece cryptosystem, proposed in 1978. In McEliece, the public key is a generator matrix G' for a specific code family (e.g., binary Goppa codes) that possesses an efficient decoding algorithm. This matrix G' is constructed by taking the original generator matrix G of the chosen code and obfuscating it using a random non-singular matrix S and a random permutation matrix P, such that $G' = SGP$. This transformation makes G' appear as a generator matrix for a general random linear code, for which decoding is known to be an NP-hard problem [Ber09].

Encryption involves representing the message m as a vector, computing the codeword $c = mG'$, and adding a random error vector e of a specific weight t (number of non-zero entries) to produce the ciphertext $y = c + e = mG' + e$. Decryption requires knowledge of the secret components S, G, and P. The recipient computes $y' = yP^{-1} = mSG + eP^{-1}$. Since P is a permutation matrix, $e' = eP^{-1}$ has the same weight t. The recipient then uses the efficient decoding algorithm associated with the original code G to decode y' and recover mS. Finally, multiplying by S^{-1} yields the original message m. The security relies on the intractability of decoding the publicly known code generated by G' without knowledge of its hidden structure (S, G, P).

Code-based cryptography offers fast encryption and decryption but is primarily suited for encryption/KEMs. A major drawback has historically been very large public key sizes (hundreds of kilobytes to megabytes), although the ciphertext overhead is often small. Recent research and variants (like Niederreiter) aim to reduce key sizes. NIST PQC Standardization is standardizing **Classic McEliece** (using binary Goppa codes)

as an additional KEM, valued for its long history (having withstood cryptanalysis since 1978), distinct security assumptions compared to lattices, and conservative parameter choices [STC24; Ala+22]. Its inclusion in the NIST portfolio provides an alternative for applications where the large key size is acceptable and a diversity of cryptographic assumptions is desired.

6.2.4 Multivariate Cryptography

Multivariate cryptography bases its security on the difficulty of solving systems of multivariate polynomial equations over a finite field (the "MQ problem"). These schemes are generally better suited for digital signatures than encryption. They often feature very fast signature generation and verification and relatively small signatures. However, designing secure multivariate schemes has proven challenging; several proposals have been broken over the years. Furthermore, public keys can sometimes be large. NIST selected **Rainbow** as a finalist, but it was subsequently broken and is no longer being standardized [DP21]. Research continues in this area, but it currently holds fewer standardized candidates compared to lattice or hash-based approaches.

6.3 NIST Standardization Process

Recognizing the existential threat posed by quantum computers, the U.S. National Institute of Standards and Technology (NIST) initiated a public process in 2016 to solicit, evaluate, and standardize quantum-resistant cryptographic algorithms [Nat20b]. This multi-year effort involved several rounds of submission, public analysis, and feedback from the global cryptographic community.

The goals were to select a portfolio of algorithms offering strong security against both classical and quantum attacks, good performance characteristics, and representing diverse mathematical approaches (to hedge against future cryptanalytic breakthroughs targeting a single family).

In July 2022, NIST announced its first set of selections for standardization [Ala+22]:

- **For Public-Key Encryption / KEMs:** CRYSTALS-Kyber (primary standard).
- **For Digital Signatures:** CRYSTALS-Dilithium, Falcon, and SPHINCS+ (primary standards).

Additionally, NIST selected several algorithms for further evaluation in a fourth round, focusing on KEMs with different characteristics, including Classic McEliece, BIKE, and HQC. Final standards for the initial selections are expected soon, providing official specifications for implementation and deployment. This standardization process is crucial for enabling widespread adoption and interoperability of PQC solutions.

6.4 Implementation Considerations

Deploying PQC algorithms involves practical considerations beyond theoretical security [SGK24]:

- **Performance Trade-offs:** As discussed, different PQC families exhibit varying performance in terms of key generation speed, encryption/encapsulation speed, decryption/decapsulation speed, and the size of public keys, private keys, ciphertexts, and signatures. These trade-offs must be evaluated in the context of specific applications and protocols (e.g., TLS handshakes, code signing, disk encryption).
- **Implementation Security:** Like classical algorithms, PQC implementations are vulnerable to side-channel attacks (timing, power analysis, etc.) if not carefully implemented. Developing constant-time and otherwise hardened implementations is an active area of research and engineering [RPS24].
- **Integration Complexity:** Integrating PQC into existing protocols and systems (like TLS, SSH, PKI) requires careful engineering, potentially involving protocol modifications to handle larger key/signature sizes and ensuring backward compatibility during the transition phase (see Chapter 7).

6.5 Hybrid Approaches

Given the uncertainties in quantum computing timelines and the maturity of newly standardized PQC algorithms, a common transition strategy is the use of **hybrid modes** [OPD24]. In a hybrid approach, systems use both a classical algorithm (e.g., ECDH) and a PQC algorithm (e.g., Kyber) simultaneously to establish a shared secret or verify a signature.

For key exchange, the final shared secret might be derived by combining the outputs of both the classical and PQC key exchanges (e.g., $K = \text{KDF}(\text{ClassicalSecret}||\text{PQCSecret})$). For signatures, both a classical and a PQC signature might be sent and verified.

The rationale is to maintain security against classical attacks (relying on the proven classical algorithm) while also gaining protection against future quantum attacks (relying on the PQC algorithm). If the classical algorithm is broken by quantum computers, the PQC component still provides security. If the PQC algorithm is unexpectedly broken classically, the classical component maintains security against current threats. While hybrid modes increase complexity and overhead (latency, bandwidth), they offer a conservative approach during the migration period [CPS24; KLS24].

6.6 Conclusion

Quantum-resistant cryptography is rapidly transitioning from theory to practice through NIST standardization. Lattice-based, hash-based, and code-based algorithms offer solutions based on problems believed hard for quantum computers. While providing quantum resilience, deployment requires managing performance trade-offs, implementation security, and system integration complexities. The next chapter explores these practical challenges in transitioning global infrastructure to this new cryptography.

7

Challenges and Considerations for Transitioning to Post-Quantum Cryptography

The shift from classical cryptography to post-quantum cryptography (PQC) is more than just swapping algorithms; it's a complex, global undertaking fraught with challenges. While Chapter 6 detailed the PQC algorithms themselves, this chapter explores the practical hurdles involved in their real-world deployment. Successfully navigating this transition requires addressing technical performance issues, complex system integrations, careful migration strategies, resource limitations, new security considerations, ongoing standardization efforts, and significant costs.

7.1 Technical Challenges: Performance and Size

One of the primary technical hurdles in adopting PQC algorithms is their often significantly different performance characteristics and larger key/signature sizes compared to their classical counterparts (e.g., RSA, ECC) [DBO25].

7.1.1 Performance Overhead

A primary technical hurdle is that many leading PQC algorithms, including the NIST standards like CRYSTALS-Kyber and CRYSTALS-Dilithium, exhibit different performance profiles [BL17]. These differences can impact user experience, system throughput, and protocol design:

- **Computational Speed:** While some PQC operations are surprisingly fast (e.g., Kyber key encapsulation can be faster than RSA decryption), others might be slower than their highly optimized classical counterparts (e.g., SPHINCS+ signing). This affects applications sensitive to latency, such as real-time communication or high-frequency operations.

- **Key and Signature Sizes:** PQC algorithms often require larger keys or produce larger signatures compared to ECC, which is known for its compactness.
  - *Public Keys:* Range from slightly larger than ECC (SPHINCS+) to significantly larger (Kyber, Dilithium) or even vastly larger (Classic McEliece).
  - *Ciphertexts/Encapsulations:* Often larger than classical equivalents, impacting bandwidth.
  - *Signatures:* Lattice-based signatures are much larger than ECDSA; hash-based signatures (SPHINCS+) are larger still (tens of kilobytes).

  This size increase directly impacts network bandwidth consumption (especially during protocol handshakes like TLS), storage requirements (for keys, certificates), and processing on resource-constrained devices.

7.2 Implementation and Integration Challenges

Integrating new PQC algorithms into the vast and varied landscape of existing digital systems presents substantial practical hurdles [MPR24].

7.2.1 System Integration Complexity

Replacing deeply embedded classical cryptography requires more than just updating libraries:

- **Legacy Systems:** Many critical systems run on older hardware or software that may be difficult or impossible to upgrade. This includes industrial control systems, embedded devices, mainframes, and older operating systems. Finding compatible PQC solutions or secure workarounds is a major challenge.
- **Protocol Modifications:** Standard network protocols like TLS, SSH, IPsec, DNSSEC were often designed assuming smaller classical key/signature sizes. Accommodating larger PQC payloads might require protocol revisions to avoid fragmentation, performance degradation (e.g., increased handshake latency), or breaking compatibility with middleboxes [**cloud_pqc_2024**].
- **Hardware Constraints:** Resource-constrained environments (IoT devices, smart cards) may lack the processing power, memory (RAM), or code space (ROM) to efficiently run PQC algorithms or store larger keys/signatures. Research into lightweight PQC implementations is ongoing but often involves trade-offs [CDX24].
- **Software Ecosystem:** Updating cryptographic libraries (e.g., OpenSSL, BoringSSL, Libgcrypt), operating systems, browsers, servers, and end-user applications across the entire software stack is a massive undertaking requiring significant development, testing, and coordination [CKS24]. Dependencies between components can create complex upgrade paths.

- **Infrastructure Updates:** Core security infrastructure needs adaptation:
  - *Key Management Systems (KMS):* Must handle new key types, formats, generation processes, and potentially larger key volumes.
  - *Hardware Security Modules (HSMs):* May require firmware updates or hardware replacement to support PQC operations securely.
  - *Public Key Infrastructure (PKI):* CAs must issue certificates with PQC public keys. This involves defining new OIDs, handling larger certificate sizes (affecting storage, transmission, and validation time), and potentially updating revocation mechanisms (CRLs, OCSP) [SMK24].

7.2.2 Implementation Security

Beyond functional integration, ensuring the *secure* implementation of PQC is critical but challenging:

- **Side-Channel Vulnerabilities:** PQC algorithms, like their classical predecessors, can be vulnerable to side-channel attacks if not implemented carefully. Timing variations, power consumption patterns, electromagnetic emissions, or cache access patterns can leak secret key information [RPS24]. Developing constant-time or otherwise masked/hardened implementations requires specialized expertise and thorough testing [RPC24].
- **Algorithmic Complexity:** Some PQC algorithms (e.g., Falcon, certain code-based schemes) involve complex mathematical operations that increase the risk of subtle implementation bugs compared to more straightforward classical algorithms.
- **Lack of Mature Tooling:** While improving, the ecosystem of development tools, testing frameworks, and formal verification methods specifically tailored for PQC security is less mature than for classical cryptography.

7.3 Migration Strategy Challenges

The transition itself, likely spanning several years or even decades, introduces specific logistical and security complexities [CMK24].

7.3.1 Managing the Transition Period

The extended period during which both classical and PQC algorithms coexist requires careful planning and execution:

- **Cryptographic Agility:** Organizations need to build systems and processes that allow them to switch cryptographic algorithms relatively easily in the future, not just for the current PQC transition, but potentially for future cryptographic breaks. This involves inventorying crypto usage, designing flexible interfaces, and avoiding hardcoding algorithms [Nat23b].
- **Hybrid Modes:** Deploying systems that use both classical (e.g., ECDH, ECDSA) and PQC (e.g., Kyber, Dilithium) algorithms simultaneously is a common interim strategy [OPD24].
  - *Rationale:* Provides defense-in-depth. Security relies on the hardness of *at least one* of the algorithms (classical against classical attacks, PQC against quantum attacks).
  - *Overhead:* Increases complexity, code size, potential attack surface, and performance cost (latency, bandwidth).
  - *Design Choices:* Securely combining outputs (e.g., key derivation functions for KEMs) requires careful design and standardization [CPS24].
- **Backward Compatibility and Interoperability:** Ensuring new PQC-enabled systems can still communicate securely with older, non-upgraded systems is critical but challenging.
  - *Negotiation:* Protocols need robust mechanisms to negotiate mutually supported algorithms.
  - *Downgrade Attacks:* Preventing attackers from forcing systems to fall back to insecure classical modes requires careful protocol design.
  - *Testing:* Extensive interoperability testing between different vendor implementations is essential before widespread deployment.
- **Inventory and Prioritization:** The first step is often the hardest: identifying *all* cryptographic dependencies within an organization's hardware, software, protocols, and data stores. This "crypto-inventory" is crucial for prioritizing upgrades based on data sensitivity (SNDL), system lifespan, ease of upgrade, and regulatory requirements [**migration_risk_2024**; Nat24].

Figure 7.1 illustrates a possible phased approach to migration.

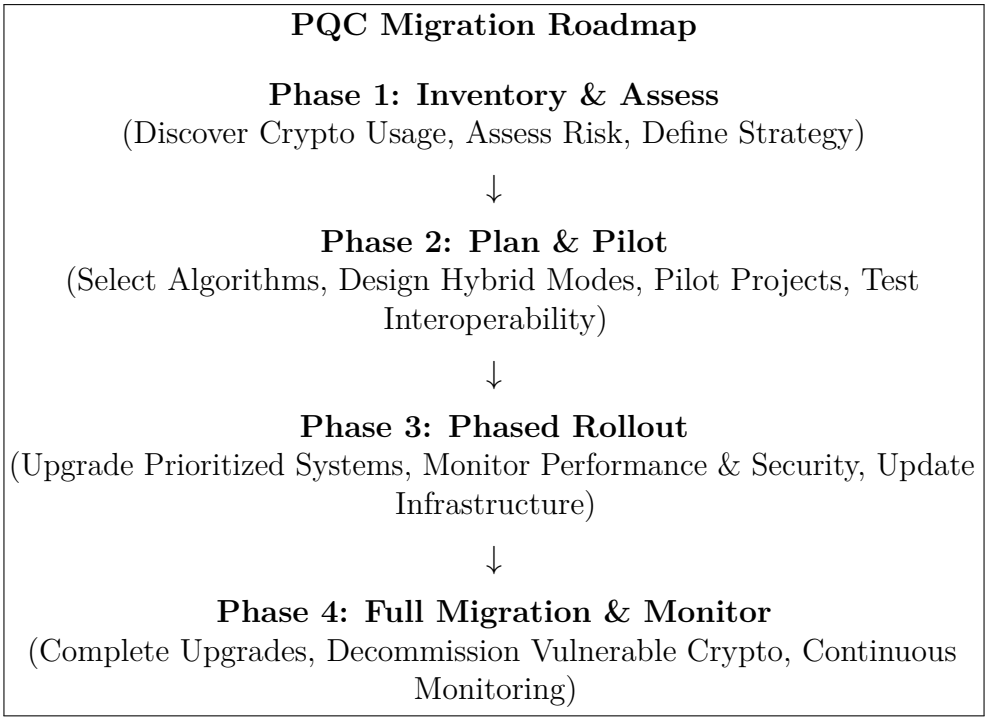

Figure 7.1: Conceptual Phased Roadmap for PQC Migration.

7.4 Resource Constraints and Availability

The transition demands significant resources, both technical and human.

7.4.1 Hardware and Software Resource Demands

As noted earlier, PQC algorithms can place greater demands on system resources compared to highly optimized classical counterparts:

- **Memory Usage:** Larger keys need more storage. Some PQC operations might require substantial RAM during execution, potentially exceeding the limits of constrained devices. Increased memory footprints can also negatively impact CPU cache performance.
- **Processing Power:** While often asymptotically efficient, the concrete computational cost of PQC operations can be significant. Supporting hybrid modes doubles the cryptographic workload during the transition. Real-time systems might struggle with increased latency unless hardware acceleration is employed [OGF24].
- **Bandwidth:** Larger keys, ciphertexts, and especially signatures consume more network bandwidth, impacting protocol efficiency, particularly during initial handshakes.

7.4.2 Expertise and Personnel

A significant bottleneck is the availability of personnel with the necessary expertise:

- **Specialized Knowledge:** Deep understanding of PQC algorithms, their security assumptions, performance characteristics, and secure implementation techniques is currently scarce.
- **Training Needs:** Organizations need to invest heavily in training developers, security architects, IT staff, and auditors on PQC concepts, risks, and migration best practices [MSP24].
- **Vendor Support:** Reliance on vendors for PQC-enabled hardware, software, and libraries requires vetting vendor expertise and ensuring long-term support.

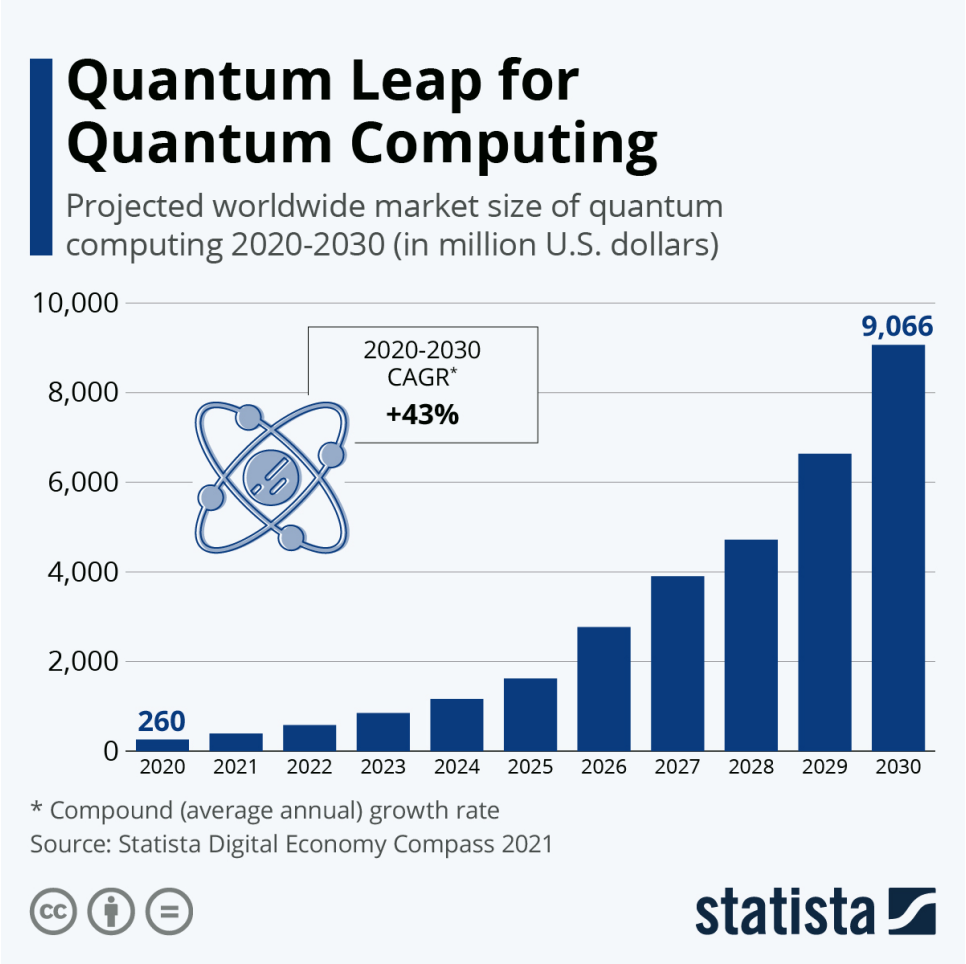

Figure 7.2: Projected worldwide market size of quantum computing (2020-2030), illustrating the rapid growth driving the demand for specialized expertise (Source: Statista Digital Economy Compass 2021).

7.5 Security Confidence and Risk Management

Transitioning involves managing both the risks of inaction (quantum threat) and the risks inherent in adopting new cryptography.

7.5.1 Trust in New Algorithms

While PQC algorithms are designed based on problems believed hard for quantum computers, they lack the decades of intense public scrutiny that classical algorithms like RSA and ECC have undergone [FKP24]:

- **Mathematical Assumptions:** Confidence relies on the assumed hardness of underlying problems (e.g., lattice problems, MQ problem, decoding). While well-studied, they are less "battle-tested" cryptographically than factoring or DLP. Future breakthroughs in classical or quantum cryptanalysis, though considered unlikely for the standardized schemes, cannot be entirely ruled out [ALP24].
- **Parameter Selection:** Choosing secure parameters involves complex trade-offs. Parameters selected today are based on current knowledge of attacks; future analysis might reveal weaknesses or require adjustments.
- **Implementation Vulnerabilities:**Semantic ScholarÂ͡s discussed, subtle bugs or side-channels specific to PQC implementations could emerge as attackers gain experience targeting these new systems [RPC24].

The NIST process aims to build confidence through public scrutiny, but long-term assurance will only come with time and continued analysis.

7.6 Standardization and Interoperability Hurdles

Achieving global consensus and ensuring systems can talk to each other securely using PQC is essential but complex.

7.6.1 Global Coordination

While NIST leads a major effort, PQC standardization is a global concern:

- **NIST Process Timeline:** Although initial standards are emerging, the process is ongoing (e.g., Round 4 candidates). Delays or changes can impact industry planning [**nist_pqc_status_2024**].
- **International Harmonization:** Other standards bodies (e.g., ISO/IEC, ETSI, IETF) are also working on PQC. Ensuring alignment and avoiding conflicting standards is crucial for global interoperability [RMS24].
- **Testing and Validation:** Establishing robust conformance testing and validation programs (like FIPS validation for classical crypto) for PQC implementations is necessary to ensure correctness and security.
- **Export Controls and National Policies:** Governmental regulations regarding the use and export of strong cryptography might evolve to address PQC, potentially creating regional variations [Bur24].

7.7 Cost and Economic Impact

The transition to PQC represents a significant financial and resource investment for organizations worldwide.

7.7.1 Financial and Operational Costs

The economic implications are substantial:

- **Direct Costs:** Significant investment in potentially new hardware (servers, HSMs, network gear, embedded devices), software licenses or development effort, integration services, and extensive testing infrastructure [MSS24].
- **Indirect Costs:** Includes the cost of training personnel, developing new operational procedures, updating documentation, managing the complexities of the extended migration period (including hybrid modes), potential productivity impacts due to performance changes, and auditing/compliance efforts [KCG24].
- **Opportunity Costs:** Resources dedicated to PQC migration might be diverted from other business or innovation initiatives.

While the costs are high, they must be weighed against the potentially catastrophic cost of widespread security failures if systems remain vulnerable when cryptographically relevant quantum computers arrive.

7.8 Conclusion

The journey to a post-quantum secure world is complex and multifaceted. Overcoming the challenges outlined in this chapter, spanning technical performance, intricate implementations, strategic migration, resource allocation, security assurance, global standardization, and economic costs, requires a concerted, collaborative effort from researchers, engineers, standards bodies, businesses, and governments. The imperative, driven by the potential of future quantum computers, is clear, but the path requires careful navigation, significant investment, and sustained commitment over the coming years.

8

Conclusion

To conclude, this memoire set out to explore how quantum computing challenges the foundations of modern cryptography, and why that matters. Each chapter tackled a different angle of this shift, helping me piece together the big picture: that adapting to disruptive technologies isn't optional, it's a must.

I started by diving into the fundamentals of quantum computing. Concepts like superposition and entanglement weren't just theoretical curiosities, they explain how quantum computers can break things classical ones can't. From there, I revisited classical cryptography, the kind that secures almost everything online today. Understanding how fragile that security becomes in a quantum context made the risk feel very real.

Looking into Shor's and Grover's algorithms showed just how deep the threat runs. Systems we trust daily, from banking logins to encrypted emails, could be rendered obsolete.

That's why I found post-quantum cryptography (PQC) so important. These new cryptographic approaches are designed to survive quantum attacks, using hard mathematical problems that remain secure even in a quantum world. But exploring Chapter 7 made one thing very clear: switching to PQC is not easy. It's technically complex, expensive, and forces organizations to rethink their infrastructure.

One of the key takeaways from this memoire is that resisting change just because it's inconvenient is not an option when it comes to security. Yes, the transition to PQC comes with overhead, in time, money, and expertise. But the cost of doing nothing is far greater.

Personally, working on this topic has shown me just how interconnected technology and society are. The tools we build, or fail to update, have real-world consequences. This research has strengthened my belief that staying ahead in tech isn't just about performance or innovation. Sometimes, it's about resilience, preparation, and having the humility to change course when new realities emerge.

In short, this memoire succeeded in showing how critical it is to adapt, even when it's hard. Quantum computing is coming, as I write advancements are being made, and the systems we rely on must evolve to meet it, not when it's convenient, but before it's too late.

Glossary of Terms and Concepts

AES Advanced Encryption Standard; the standard symmetric block cipher widely used today. See Section 3.3. 17, 31–33, 35, 54, 64

amplitude amplification A general quantum algorithm technique that increases the probability amplitude of desired states, generalizing Grover's algorithm. See Section 2.3.3. 13, 30, 54

authentication Security goal verifying the identity of a user or system. Mentioned in Section 3.2.. 17, 21, 54

Bell state One of a set of four specific maximally entangled quantum states of two qubits. Example: $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. See Section 2.1.2. 6, 54

birthday attack A type of collision attack based on the mathematics of the birthday problem, allowing collisions in hash functions to be found significantly faster ($O(2^{n/2})$) than brute force ($O(2^n)$). See Section 5.2.2.. 31–33, 54

bit The fundamental unit of information in classical computing, representing either a 0 or a 1. Contrasted with qubit. See Section 4.1.. 23, 24, 54

Bloch sphere A geometrical representation of the pure state space of a single qubit. See Section 2.1.3. 7, 54

block cipher A symmetric cipher operating on fixed-length blocks of data. See Section 3.3.. 17, 18, 54

BPP Bounded-error Probabilistic Polynomial time; the class of problems efficiently solvable by a classical probabilistic computer. See Section 4.3. 25, 26, 54, 56

BQP Bounded-error Quantum Polynomial time; the class of problems efficiently solvable by a quantum computer. See Section 4.3. 24–26, 54, 56, 61

brute-force attack A cryptanalytic attack that tries all possible keys or passwords until the correct one is found. Grover's algorithm provides a quadratic speedup for these attacks against symmetric ciphers. Mentioned in Chapter 1. 4, 54

CA Certificate Authority; a trusted entity that issues and signs digital certificates in a PKI. See Section 3.6. 47, 54

CBC Cipher Block Chaining. A mode of operation for a block cipher where each block of plaintext is XORed with the previous ciphertext block before being encrypted.. 18, 54

Classic McEliece A specific parameterization of the McEliece cryptosystem using binary Goppa codes, chosen by NIST as a post-quantum KEM candidate for standardization, valued for its long history and conservative design. See Section 6.2.3.. 54

classical computer A computer that operates based on classical physics, using bits (0s and 1s) as the basic unit of information. Contrasted with quantum computers. Mentioned in Chapter 1. 4, 54

classical core A standard processing unit within a classical computer's central processing unit (CPU) that executes instructions sequentially or in parallel with other cores. Mentioned in Section 4.2.1 in the context of parallel processing.. 24, 54

classical cryptography Cryptographic techniques developed before the advent of quantum computing, primarily relying on computational hardness assumptions believed to hold for classical computers. See Chapter 3.. 1, 16, 54

code-based cryptography PQC approach based on the difficulty of decoding general linear error-correcting codes. See Section 6.2.3. 54, 60

collision attack A cryptanalytic attack on hash functions that tries to find two distinct inputs that produce the same hash output. See Section 5.2.2.. 31, 54

collision resistance Hash property: hard to find two distinct inputs m1, m2 such that H(m1)=H(m2). Mentioned in Section 3.5.. 20, 32, 54

compression function Core component in Merkle-Damgård hashes, processes message blocks. Mentioned in Section 3.5.. 21, 54

computational complexity theory The branch of theoretical computer science that studies the resources (e.g., time, memory) required to solve computational problems, classifying them into complexity classes like P, NP, BPP, and BQP. See Section 4.3.. 25, 54

computational hardness assumption The belief that certain mathematical problems are too difficult to solve efficiently with current computing technology (classical or quantum). See Section 3.2. 54

confidentiality Security goal ensuring information is not disclosed to unauthorized parties. Mentioned in Section 3.2.. 17, 54

confusion Cryptographic principle obscuring the relationship between the key and ciphertext (e.g., via substitution). Mentioned in Section 3.3.. 17, 54

CRQC Cryptographically Relevant Quantum Computer; a quantum computer with sufficient qubits and error correction to pose a real threat to traditional cryptographic systems by running algorithms like Shor's. Mentioned in Chapter 5. 28, 32, 34, 35, 54

cryptographic agility The ability of a system to easily switch between different cryptographic algorithms. Mentioned in Section 7.1. 54

cryptography The practice and study of techniques for secure communication in the presence of third parties called adversaries. Mentioned in Chapter 1. 4, 54

CRYSTALS-Dilithium A lattice-based digital signature scheme based on Module-LWE, selected by NIST as a primary standard for post-quantum digital signatures. See Section 6.2.1.. 54

CRYSTALS-Kyber A lattice-based Key Encapsulation Mechanism (KEM) based on Module-LWE, selected by NIST as a primary standard for post-quantum public-key encryption/key establishment. See Section 6.2.1.. 54

CVP Closest Vector Problem. A computationally hard problem on lattices, which involves finding the lattice point closest to a given target point in the space. See Section 6.2.1.. 54

decoherence Loss of quantum properties due to environmental interactions. See Section 2.4. 7, 8, 54

DES Data Encryption Standard; a symmetric-key block cipher (1977), now insecure due to small key size. Mentioned in Section 3.2.. 16, 54

DH Diffie-Hellman key exchange; a method for securely exchanging cryptographic keys over a public channel. Vulnerable to Shor's algorithm. Mentioned in Chapter 1. 4, 28, 35, 54

Diffie-Hellman Key Exchange Method to establish a shared secret over an insecure channel (DH). See Section 3.4.2.. 1, 19, 54

diffusion Cryptographic principle spreading plaintext influence across ciphertext (e.g., via permutation). Mentioned in Section 3.3.. 17, 54

diffusion operator An operation used in Grover's algorithm and amplitude amplification that performs an inversion about the average amplitude, amplifying the marked states. See Section 2.3.3. 13, 54

digital signature Scheme for verifying authenticity, integrity, and non-repudiation using asymmetric crypto. See Section 3.6.. 1, 17, 21, 54

digraph cipher A cipher encrypting pairs of letters (digraphs) rather than single letters. Mentioned in Section 3.2.. 16, 54

Discrete Logarithm Problem (DLP) The computational problem of finding the exponent x such that $g^x \equiv h \pmod{p}$ for given group elements g, h and modulus p. The presumed difficulty underlies the security of DH, DSA, and ECC. See Section 5.1.2.. 27, 28, 54

downgrade attack Attack forcing a system to use a less secure mode of operation. Mentioned in Section 3.7.. 54

DSA Digital Signature Algorithm; a U.S. federal standard for digital signatures based on the discrete logarithm problem. Vulnerable to Shor's algorithm. See Section 5.1.2.. 28, 54

ECB Electronic Codebook; the simplest block cipher mode, encrypting blocks independently (insecure). See Section 3.3.. 18, 54

ECC Elliptic Curve Cryptography; public-key cryptography using elliptic curves, offering smaller keys than RSA. See Section 3.4.3. 1, 4, 17, 20, 26, 28, 29, 32, 33, 35, 36, 45, 54

ECDH Elliptic Curve Diffie-Hellman; a key agreement protocol that allows two parties, each having an elliptic-curve public-private key pair, to establish a shared secret over an insecure channel. Vulnerable to Shor's algorithm. See Section 5.1.2.. 28, 54

ECDLP Elliptic Curve Discrete Logarithm Problem; the hard mathematical problem underlying ECC security. See Section 3.4.3. 20, 28, 54

ECDSA Elliptic Curve Digital Signature Algorithm; a variant of the Digital Signature Algorithm (DSA) which uses elliptic curve cryptography. Vulnerable to Shor's algorithm. See Section 5.1.2.. 28, 54

Enigma machine Electro-mechanical rotor cipher machine used for encrypting secret messages. Mentioned in Section 3.2.. 16, 54

entanglement A quantum phenomenon where multiple qubits become correlated. See Section 2.1.2. 4, 6, 11, 23, 24, 53, 54

Euler's totient function Function $\phi(n)$ counting positive integers less than or equal to n that are relatively prime to n. Used in RSA key generation. See Section 3.4.1. 54

Falcon A lattice-based digital signature scheme based on the NTRU problem (related to Ring-LWE), selected by NIST as a primary standard. Known for its particularly small signature sizes. See Section 6.2.1.. 54

fault tolerance The ability of a quantum computer to perform reliable computations even when its underlying components (qubits, gates) are imperfect or noisy, typically achieved through quantum error correction. See Section 2.4. 15, 54

frequency analysis The study of letter/group frequencies in ciphertext to break classical ciphers. Mentioned in Section 3.2.. 16, 54

GC Galois/Counter Mode. An authenticated encryption mode that provides both confidentiality and data authenticity. Often referred to as GCM.. 18, 54

GNFS General Number Field Sieve; the most efficient known classical algorithm for factoring large integers. Mentioned in Section 5.1.2. 27, 54

Grover diffusion operator A component in Grover's algorithm (denoted U_s) that amplifies the amplitude of the marked state(s) and decreases the amplitude of others, effectively performing inversion about the mean. See Section 5.2.1.. 30, 54

Grover oracle A component in Grover's algorithm (denoted U_f) that identifies and "marks" the target state(s) within the quantum superposition, typically by applying a phase shift. See Section 5.2.1.. 30, 54

Grover's algorithm Quantum search algorithm providing quadratic speedup for unstructured search. See Section 5.2. 4, 13, 24, 26, 27, 30, 35, 42, 54

hash function A cryptographic function that takes an arbitrary input size and produces a fixed-size output (hash value). Key properties include determinism, preimage resistance (hard to find input for a given output), second preimage resistance (hard to find a different input with the same output), and collision resistance (hard to find two different inputs with the same output).. 1, 20, 54

hash-based cryptography PQC approach (primarily for signatures) relying only on the security of cryptographic hash functions. See Section 6.2.2. 42, 54, 61

Hilbert space Mathematical space describing quantum states. See Section 4.2. 25, 54

Hill cipher A polygraphic substitution cipher based on linear algebra. Mentioned in Section 3.2.. 16, 54

hybrid cryptography Combined use of classical and post-quantum algorithms during the transition period. See Section 7.3.1. 54, 59

hybrid mode Synonym for hybrid cryptography. The simultaneous use of both a classical and a post-quantum cryptographic algorithm to provide security against both classical and quantum adversaries during the transition period. See Section 6.5.. 54

IFP/DLP Integer Factorization Problem / Discrete Logarithm Problem; the hard mathematical problems underlying RSA and DH security respectively. See Section 3.2. 19, 54

Integer Factorization Problem (IFP) The computational problem of finding the prime factors of a given composite integer. The presumed difficulty of this problem underlies the security of RSA. See Section 5.1.. 27, 54

integrity Security goal ensuring data has not been altered unauthorizedly. Mentioned in Section 3.2.. 17, 54

interference Quantum phenomenon where probability amplitudes can cancel or reinforce each other, key to algorithm speedups. See Section 4.2. 11, 25, 54

IPsec Internet Protocol Security; a secure network protocol suite that authenticates and encrypts packets of data sent over an Internet Protocol network. Often relies on algorithms vulnerable to Shor's algorithm. Mentioned in Section 5.1.. 29, 54

KEM Key Encapsulation Mechanism; cryptographic technique used to securely establish shared keys, often used in PQC standards.. 41–43, 54, 56

key schedule Algorithm generating round keys from the main key in block ciphers. Mentioned in Section 3.3.. 17, 54

lattice A lattice. 1, 54

lattice-based cryptography PQC approach based on computationally hard problems on mathematical lattices, such as LWE. See Section 6.2.1. 54, 60

logical qubit A qubit encoded using multiple physical qubits and quantum error correction to protect it from noise. See Section 2.4. 15, 54

LWE Learning With Errors; a mathematical problem believed to be hard for both classical and quantum computers, basis for lattice-based PQC. See Section 6.2.1. 38, 54, 63

MAC Message Authentication Code; used to verify data integrity and authenticity using a shared secret key. Mentioned in Section **??**. 17, 54

Man-in-the-Middle attack Attack relaying/altering communication between two parties (MitM). Mentioned in Section 3.7.. 54

mathematical lattice A periodic arrangement of points in n-dimensional space, formed by integer linear combinations of a set of basis vectors. Used as the foundation for lattice-based cryptography. See Section 6.2.1.. 54

McEliece cryptosystem A foundational code-based cryptography scheme proposed in 1978, whose security relies on the difficulty of decoding general linear codes (specifically, often using Goppa codes). See Section 6.2.3.. 54, 56

MD5 Message Digest 5; 128-bit hash function, now broken (collisions found). Mentioned in Section 3.5.. 20, 54

Merkle-Damgård construction A method for building collision-resistant hash functions from compression functions. See Section 3.5. 21, 54

mode of operation Method for using block ciphers on messages longer than one block. See Section 3.3.. 18, 54

Module-LWE A variant of the LWE problem used in schemes like CRYSTALS-Kyber. See Section 6.2.1. 40, 54, 57

monoalphabetic cipher A substitution cipher using a single, fixed substitution alphabet for the entire message. Mentioned in Section 3.2.. 16, 54

Mosca's Inequality An inequality ($X + Y > Z$) highlighting the urgency of PQC migration, where X is data confidentiality lifetime, Y is migration time, and Z is time until a CRQC exists. See Section 5.4.2.. 35, 54

MQ problem The problem of solving systems of multivariate quadratic polynomial equations over a finite field. The presumed difficulty of this problem underlies multivariate cryptography. See Section 6.2.4.. 54

multivariate cryptography PQC approach based on the difficulty of solving systems of multivariate polynomial equations. See Section 6.2.4. 43, 54, 60

NISQ Noisy Intermediate-Scale Quantum; the current era of quantum hardware with limited qubits and no fault tolerance. See Section 2.4. 15, 24, 54

NIST PQC Standardization The ongoing process led by the U.S. National Institute of Standards and Technology (NIST) to select and standardize post-quantum cryptographic algorithms. See Chapter 6.1. 42, 54

NIST security levels Standardized categories (1-5) defined by NIST to compare the strength of PQC algorithms against known attacks. See Section 5.3. 32, 54

No-Cloning Theorem Fundamental principle stating that an arbitrary unknown quantum state cannot be perfectly copied. See Section 4.1. 24, 54

non-repudiation Security goal preventing denial of authenticity of a signature or message origin. Mentioned in Section 3.2.. 17, 21, 54

NP Nondeterministic Polynomial time; the class of problems where proposed solutions can be efficiently verified classically. See Section 4.3. 25, 26, 54, 56

NP-hard A class of computational problems that are at least as hard as the hardest problems in NP (Nondeterministic Polynomial time).. 42, 54

NTT Number Theoretic Transform. An algorithm analogous to the Fast Fourier Transform (FFT), but operating over finite fields or rings. Used to efficiently perform polynomial multiplication in schemes like Ring-LWE. See Section 6.2.1.. 54

OAEP Optimal Asymmetric Encryption Padding; padding scheme for RSA to prevent attacks. Mentioned in Section 3.4.1.. 19, 54

one-way function A function easy to compute but hard to invert. Mentioned in Section 3.2.2.. 17, 54

OTS One-Time Signature. A type of digital signature scheme where a key pair can only be used to sign a single message securely. Used as building blocks in hash-based cryptography. See Section 6.2.2.. 54, 63

P Polynomial time; the class of problems efficiently solvable by a classical deterministic computer. See Section 4.3. 25, 54, 56

padding oracle attack Attack using padding validation errors to decrypt ciphertext. Mentioned in Section 3.7.. 54

physical qubit An actual physical system (e.g., trapped ion, superconducting circuit) used to represent a qubit in a quantum computer, susceptible to noise. Contrasted with logical qubit. See Section 2.4. 15, 54

PKI Public Key Infrastructure; framework for managing public keys and digital certificates. See Section 3.6. 1, 21, 29, 54

polyalphabetic cipher A substitution cipher using multiple substitution alphabets, making frequency analysis harder. Mentioned in Section 3.2.. 16, 54

polynomial time An algorithm runtime complexity where the number of steps scales polynomially with the input size. Considered efficient for classical computation. See also BQP. Mentioned in Chapter 1. 4, 54

post-quantum cryptography Cryptographic algorithms designed to be secure against attacks by both classical and quantum computers. Often abbreviated as PQC. See Chapter 6.1. 4, 36, 45, 54

power analysis Side-channel attack studying power consumption of cryptographic hardware. Mentioned in Section 3.3 and 3.7.. 17, 54

PQC Post-Quantum Cryptography. 4, 32, 54

preimage attack A cryptographic attack where, given a hash value, the attacker tries to find any input message that hashes to that specific value. For a secure hash function, this should be computationally infeasible.. 31, 33, 54

preimage resistance Hash property: hard to find input m for a given hash h such that H(m)=h. Mentioned in Section 3.5.. 20, 54

probability amplitude A complex number whose squared magnitude represents the probability of measuring a quantum system (like a qubit) in a specific basis state. Used in the description of superposition. See Section 4.1.. 23, 54

public-key cryptography Asymmetric encryption using a public key for encryption/verification and a private key for decryption/signing. See Section 3.4. 1, 4, 16, 19, 35, 36, 54

QEC Quantum Error Correction. 34, 54

QEC Quantum Error Correction; techniques to protect quantum information from noise. See Section 2.4. 15, 54

QFT Quantum Fourier Transform; a core component of Shor's algorithm. See Section 2.3.1. 11, 12, 27, 40, 54

QKD Quantum Key Distribution. 54

QPE Quantum Phase Estimation; a quantum algorithm used within Shor's algorithm. See Section 2.3.2. 12, 27, 54

quantum circuit Model representing quantum computations as sequences of quantum gates acting on qubits. See Section 4.2. 25, 54

quantum gate Basic operation in quantum circuits that transforms qubit states. See Section 2.2. 8, 24, 25, 54

quantum mechanics The fundamental theory in physics describing the properties of nature at the scale of atoms and subatomic particles. Relevant concepts include superposition and entanglement. Mentioned in Chapter 1. 4, 54

quantum oracle A 'black box' operation in a quantum algorithm that implements a specific function, often used to mark target states (e.g., in Grover's algorithm). See Section 2.3.3. 13, 54

quantum parallelism Ability of quantum computers to perform computations on multiple states simultaneously via superposition. See Section 4.2. 25, 54

qubit The fundamental unit of quantum information. See Chapter 2. 6–8, 23–25, 54, 55, 62

replay attack Network attack repeating/delaying valid data transmission. Mentioned in Section 3.7.. 54

Ring-LWE Ring Learning With Errors. An efficiency variant of the LWE problem that operates within polynomial rings, enabling smaller keys and faster computations. See Section 6.2.1.. 54, 58, 61

RSA Public-key cryptosystem based on the difficulty of factoring large integers. See Section 3.4.1. 1, 4, 19, 26–29, 32, 33, 35, 36, 45, 54

second preimage resistance Hash property: hard to find different input m2 for a given m1 such that H(m1)=H(m2). Mentioned in Section 3.5.. 20, 54

SHA-1 Secure Hash Algorithm 1; 160-bit hash function, deprecated (collisions found). Mentioned in Section 3.5.. 20, 54

SHA-2 Secure Hash Algorithm 2; family of hash functions (e.g., SHA-256), currently secure. Mentioned in Section 3.5.. 20, 54

SHA-3 Secure Hash Algorithm 3; based on sponge construction, currently secure. Mentioned in Section 3.5.. 20, 54

Shor's algorithm Quantum algorithm that efficiently solves integer factorization and discrete logarithms. See Section 5.1. 4, 11, 12, 14, 19, 20, 24, 26, 27, 35, 36, 40, 54

side-channel attack Attack exploiting information leaked from a cryptosystem's physical implementation (e.g., timing, power consumption). See Section 3.7. 17, 41, 44, 47, 54

SNDL Store Now, Decrypt Later; attack strategy involving storing encrypted data today for future decryption with quantum computers. See Section 5.4.2. 4, 35, 48, 54

SPHINCS+ A stateless hash-based digital signature scheme selected by NIST as a standard for post-quantum signatures. Known for its strong security based solely on hash function properties, at the cost of large signature sizes. See Section 6.2.2.. 54, 63

SSH Secure Shell. 29, 46, 54

stateful signature A type of hash-based signature scheme (e.g., XMSS, LMS) that requires the signer to securely maintain state (like the index of the last used OTS key) to avoid key reuse. Generally faster and produces smaller signatures than stateless schemes. See Section 6.2.2.. 54

stateless signature A type of hash-based signature scheme (e.g., SPHINCS+) that does not require the signer to maintain state, making it more robust against key reuse vulnerabilities but typically resulting in larger signatures and slower performance. See Section 6.2.2.. 54

substitution cipher A method of encryption where units of plaintext are replaced with ciphertext according to a fixed system. Mentioned in Section 3.2.. 16, 54

superposition A quantum state where a system exists in multiple states simultaneously. See Section 2.1.1. 4, 7, 11, 23, 24, 53, 54, 62

SVP Shortest Vector Problem. A computationally hard problem on lattices, which involves finding the shortest non-zero vector in a given lattice. See Section 6.2.1.. 54

symmetric cipher A type of encryption algorithm that uses the same key for both encryption and decryption (e.g., AES). Weakened by Grover's algorithm. Mentioned in Chapter 1. 4, 54

symmetric-key cryptography Encryption methods using the same key for encryption and decryption. See Section 3.3.. 1, 17, 35, 54

timing attack Side-channel attack analyzing time taken for cryptographic operations. Mentioned in Section 3.3 and 3.7.. 17, 54

TLS Transport Layer Security. 29, 35, 46, 54

transposition cipher A method of encryption where the positions held by units of plaintext are shifted according to a regular system. Mentioned in Section 3.2.. 16, 54

trapdoor function A one-way function hard to invert without secret information (the trapdoor). Mentioned in Section 3.2.2.. 17, 54

Turing machine A mathematical model of computation that defines an abstract machine manipulating symbols on a strip of tape according to rules. A fundamental model for classical computation. Mentioned in Section 4.2.1.. 24, 54

unitary transformation A mathematical operation preserving length and angles, representing the evolution of quantum states via quantum gates. See Section 2.2. 8, 54

unstructured search A search problem where there is no known structure in the search space that can be exploited to find the target element faster than checking items one by one (classically) or using Grover's algorithm (quantumly). See Section 5.2.. 30, 31, 54

von Neumann architecture A computer architecture based on the concept of stored-program computers where instruction data and program data are stored in the same memory. The basis for most modern classical computers. Mentioned in Section 4.2.1.. 24, 54

Sources

Alagic et al.: Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process **nist_pqc_status_report_ir8413**

Gorjan Alagic et al. *Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process*. NISTIR 8413. Comprehensive report on NIST's PQC third round results. National Institute of Standards and Technology, 2022. URL: https://doi.org/10.6028/NIST.IR.8413.

Albrecht et al.: New Cryptanalytic Techniques for Post-Quantum Cryptographic Schemes **cryptanalysis_2024**

Martin Albrecht, Tancrède Lepoint, and Kenneth Paterson. "New Cryptanalytic Techniques for Post-Quantum Cryptographic Schemes". In: *Journal of Cryptology* 37.2 (2024), pp. 121–156. DOI: 10.1007/s00145-023-09465-3.

Ali et al.: Quantum Computing - Introduction to the special theme **Ali2022QuantumC**

Shaukat Ali and S. Selstø. "Quantum Computing - Introduction to the special theme". In: *ERCIM News* 2022 (2022).

Arora et al.: Computational Complexity: A Modern Approach **arora_barak_complexity**

Sanjeev Arora and Boaz Barak. *Computational Complexity: A Modern Approach*. Cambridge, UK: Cambridge University Press, 2009.

Berberich et al.: Quantum Computing Through the Lens of Control: A Tutorial Introduction **Berberich2023QuantumCT**

Julian Berberich and Daniel Fink. "Quantum Computing Through the Lens of Control: A Tutorial Introduction". In: *IEEE Control Systems* 44 (2023), pp. 24–49.

Bernstein: Introduction to post-quantum cryptography **bernstein2008introduction**

Daniel J. Bernstein. "Introduction to post-quantum cryptography". In: *Post-Quantum Cryptography*. Ed. by Daniel J. Bernstein, Johannes Buchmann, and Erik Dahmen. Springer, 2009, pp. 1–14. DOI: 10.1007/978-3-540-88702-7_1.

Bernstein et al.: Post-Quantum Cryptography **bernstein2017post**

Daniel J. Bernstein and Tanja Lange. "Post-Quantum Cryptography". In: *Nature* 549.7671 (2017). Overview of post-quantum cryptography approaches., pp. 188–194.

Bleichenbacher: Chosen ciphertext attacks against protocols based on the RSA encryption standard PKCS #1 **bleichenbacher1998chosen**

Daniel Bleichenbacher. "Chosen ciphertext attacks against protocols based on the RSA encryption standard PKCS #1". In: *Advances in Cryptology — CRYPTO '98*. Springer. 1998, pp. 1–12.

Boneh: Cryptography Full Course Part 1 **crypto1**

Dan Boneh. *Cryptography Full Course Part 1*. YouTube. Stanford University course covering stream ciphers, block ciphers, and message integrity. 2021. URL: https://www.youtube.com/watch?v=j_8PLI_wCVU.

Boneh et al.: A Graduate Course in Applied Cryptography **boneh2020graduate**

Dan Boneh and Victor Shoup. *A Graduate Course in Applied Cryptography*. https://toc.cryptobook.us/. Stanford University, 2020.

Bureau of Industry and Security: Cryptography Export Controls and Post-Quantum Cryptography: Policy Implications **crypto_export_controls**

Bureau of Industry and Security. *Cryptography Export Controls and Post-Quantum Cryptography: Policy Implications*. Technical Report BIS-TRD-24-01. U.S. Department of Commerce, Feb. 2024. URL: https://www.bis.doc.gov/index.php/documents/policy-guidance/3270-quantum-cryptography-export-controls.

Byrnes: Introduction to Quantum Computing Course **byrnes2022**

Tim Byrnes. *Introduction to Quantum Computing Course*. 2022. URL: https://www.youtube.com/watch?v=L-vjihvQnd0.

Cheng et al.: Lightweight Post-Quantum Cryptography Implementations for Constrained IoT Devices **iot_pqc_2024**

Hao Cheng, Léo Ducas, and Xiaoyang Xie. "Lightweight Post-Quantum Cryptography Implementations for Constrained IoT Devices". In: *IEEE Internet of Things Journal* 11.5 (2024), pp. 8261–8276. DOI: 10.1109/JIOT.2023.3341825.

Chou et al.: New Techniques for Efficient Software Implementations of Post-Quantum Cryptography **software_optimization_2024**

Tung Chou, Matthias J. Kannwischer, and Peter Schwabe. "New Techniques for Efficient Software Implementations of Post-Quantum Cryptography". In: *IEEE Transactions on Computers* 73.5 (2024), pp. 1305–1318. DOI: 10.1109/TC.2023.3337842.

Contributors to Wikimedia projects: Amplitude amplification — Wikipedia, The Free Encyclopedia **wikipediaAmplitudeAmp**

Contributors to Wikimedia projects. *Amplitude amplification — Wikipedia, The Free Encyclopedia*. [Online; accessed 2-May-2025]. 2025. URL: https://en.wikipedia.org/wiki/Amplitude_amplification (visited on 05/02/2025).

Contributors to Wikimedia projects: Bloch sphere — Wikipedia, The Free Encyclopedia **wikipediaBlochSphere**

Contributors to Wikimedia projects. *Bloch sphere — Wikipedia, The Free Encyclopedia*. [Online; accessed 2-May-2025]. 2025. URL: https://en.wikipedia.org/wiki/Bloch_sphere (visited on 05/02/2025).

Contributors to Wikimedia projects: Quantum circuit — Wikipedia, The Free Encyclopedia **wikipediaQuantumCircuit**

Contributors to Wikimedia projects. *Quantum circuit — Wikipedia, The Free Encyclopedia*. [Online; accessed 2-May-2025]. 2025. URL: https://en.wikipedia.org/wiki/Quantum_circuit (visited on 05/02/2025).

Contributors to Wikimedia projects: Quantum phase estimation algorithm — Wikipedia, The Free Encyclopedia **wikipediaQPE**

Contributors to Wikimedia projects. *Quantum phase estimation algorithm — Wikipedia, The Free Encyclopedia*. [Online; accessed 2-May-2025]. 2025. URL: https://en.wikipedia.org/wiki/Quantum_phase_estimation_algorithm (visited on 05/02/2025).

Crockett et al.: Post-Quantum Cryptography Migration Strategies: Current Research and Best Practices **pqc_migration_2024**

Emily Crockett, Michele Mosca, and Panos Kampanakis. "Post-Quantum Cryptography Migration Strategies: Current Research and Best Practices". In: *Cryptography* 8.1 (2024), p. 2. DOI: 10.3390/cryptography8010002.

Crockett et al.: New Approaches to Combining Classical and Post-Quantum Algorithms in Protocol Design **hybrid_protocols_2024**

Emily Crockett, Christian Paquin, and Daniel Smith-Tone. "New Approaches to Combining Classical and Post-Quantum Algorithms in Protocol Design". In: *Journal of Mathematical Cryptology* 18.1 (2024), pp. 20–42. DOI: 10.1515/jmc-2023-0055.

Demir et al.: Performance Analysis and Industry Deployment of Post-Quantum Cryptography Algorithms **pqc_performance_benchmarking_2023**

Elif Dicle Demir, Buse Bilgin, and Mehmet Cengiz Onbasli. "Performance Analysis and Industry Deployment of Post-Quantum Cryptography Algorithms". In: *arXiv preprint* (2025). DOI: 10.48550/arXiv.2503.12952. URL: https://arxiv.org/abs/2503.12952.

Diffie et al.: New directions in cryptography **diffie1976new**

Whitfield Diffie and Martin E. Hellman. "New directions in cryptography". In: *IEEE Transactions on Information Theory* 22.6 (1976). Introduced public-key cryptography and key exchange, pp. 644–654.

Ding et al.: Review on the study of entanglement in quantum computation speedup **Ding2007EntanglementReview**

ShengChao Ding and Zhi Jin. "Review on the study of entanglement in quantum computation speedup". In: *Chinese Science Bulletin* 52.16 (2007), pp. 2161–2166. DOI: 10.1007/s11434-007-0324-8. URL: https://doi.org/10.1007/s11434-007-0324-8.

Djelic et al.: NIST Post-Quantum Cryptography Standardization - Third Round Finalist Digital Signature Schemes **djelic2021nist**

Marko Djelic and Mario Poljak. "NIST Post-Quantum Cryptography Standardization - Third Round Finalist Digital Signature Schemes". In: *Automatic Control and Computer Sciences* 55.1 (2021), pp. 86–95.

Dworkin: Recommendation for Block Cipher Modes of Operation: Methods and Techniques **dworkin**

Morris J. Dworkin. *Recommendation for Block Cipher Modes of Operation: Methods and Techniques*. Tech. rep. Special Publication 800-38A. National Institute of Standards and Technology, 2001. DOI: 10.6028/NIST.SP.800-38A. URL: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a.pdf.

Edureka: Cryptography for Beginners **crypto5**

Edureka. *Cryptography for Beginners*. YouTube. Foundational concepts with demonstrations of encryption algorithms. 2018. URL: https://www.youtube.com/watch?v=5jpgMXt1Z9Y.

Ekert: Oxford Quantum Computing Lectures **ekert2021**

Artur Ekert. *Oxford Quantum Computing Lectures*. 2021. URL: https://www.youtube.com/playlist?list=PLkespgaZN4gmu0nWNmfMflVRqw0VPkCGH.

Eye on Tech: What is Quantum Cryptography? An Introduction **EyeOnTech2022**

Eye on Tech. *What is Quantum Cryptography? An Introduction*. Accessed: 2025-05-02. YouTube video. 2022. URL: https://www.youtube.com/watch?v=_5NQf8k3Jo0.

Festival: Quantum Computing: Hype vs. Reality **wsf2024**

World Science Festival. *Quantum Computing: Hype vs. Reality*. 2024. URL: https://www.youtube.com/watch?v=-1PsQIciMEc.

Fischlin et al.: Enhanced Security Proofs for Post-Quantum Cryptographic Protocols **security_proofs_2024**

Marc Fischlin, Eike Kiltz, and Krzysztof Pietrzak. "Enhanced Security Proofs for Post-Quantum Cryptographic Protocols". In: *Theory of Computing* 20.1 (2024), pp. 1–37. DOI: 10.4086/toc.2024.v20a1.

Geremew et al.: Preparing Critical Infrastructure for Post-Quantum Cryptography: Strategies for Transitioning Ahead of Cryptanalytically Relevant Quantum Computing **Geremew2024PreparingCI**

Amare Geremew and Atif Mohammad. "Preparing Critical Infrastructure for Post-Quantum Cryptography: Strategies for Transitioning Ahead of Cryptanalytically Relevant Quantum Computing". In: *International Journal on Engineering, Science and Technology* (2024).

Gharibian: Paderborn University Quantum Computing Courses **gharibian2021**

Sevag Gharibian. *Paderborn University Quantum Computing Courses*. 2021. URL: https://youtube.com/playlist?list=[specified_in_source2].

Gidney et al.: How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits **gidney2021factor**

Craig Gidney and Martin Ekerå. "How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits". In: *Quantum* 5 (2021). Detailed resource estimates for quantum factoring, p. 433.

Grover: A fast quantum mechanical algorithm for database search **grover1996fast**

Lov K. Grover. "A fast quantum mechanical algorithm for database search". In: *Proceedings of the Twenty-eighth Annual ACM Symposium on Theory of Computing*. Presents Grover's quantum search algorithm. ACM. 1996, pp. 212–219.

Hülsing et al.: Recent Developments in Hash-Based Signatures: Addressing Key Challenges **hash_based_advances_2024**

Andreas Hülsing, Daniel J. Bernstein, and Lorenz Panny. "Recent Developments in Hash-Based Signatures: Addressing Key Challenges". In: *Designs, Codes and Cryptography* 92.4 (2024), pp. 725–753. DOI: 10.1007/s10623-023-01273-x.

IIT KANPUR-NPTEL: Lecture 26: Quantum Cryptography **IITKanpur2024**

IIT KANPUR-NPTEL. *Lecture 26: Quantum Cryptography*. Accessed: 2025-05-02. YouTube video. 2024. URL: https://www.youtube.com/watch?v=71Hvk7r2n44.

Ismael: Quantum Mechanics **sepQM2025**

Jenann Ismael. *Quantum Mechanics*. Ed. by Edward N. Zalta and Uri Nodelman. 2025. URL: https://plato.stanford.edu/archives/spr2025/entries/qm/ (visited on 05/02/2025).

Kahn: The Codebreakers: The Story of Secret Writing **kahn1996codebreakers**

David Kahn. *The Codebreakers: The Story of Secret Writing*. Scribner, 1996.

Katz et al.: Economic Impact of Post-Quantum Cryptography on the Healthcare Sector **pqc_economic_impact_healthcare**

Jonathan Katz, Adam Cohen, and Craig Gentry. "Economic Impact of Post-Quantum Cryptography on the Healthcare Sector". In: *Journal of Health Economics* 93 (2024), p. 102842. DOI: 10.1016/j.jhealeco.2023.102842.

Katz et al.: Introduction to Modern Cryptography **katz2014introduction**

Jonathan Katz and Yehuda Lindell. *Introduction to Modern Cryptography*. 2nd. Widely used modern textbook. CRC Press, 2014.

Kocher: Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems **kocher1996timing**

Paul C. Kocher. "Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems". In: *Advances in Cryptology — CRYPTO '96* (1996), pp. 104–113.

Kwiatkowski et al.: Performance Analysis of Hybrid Cryptographic Systems: Optimized Implementations Reducing Overhead **hybrid_performance_2024**

Krzysztof Kwiatkowski, Adam Langley, and Nick Sullivan. "Performance Analysis of Hybrid Cryptographic Systems: Optimized Implementations Reducing Overhead". In: *IEEE Transactions on Information Theory* 70.6 (2024), pp. 3746–3761. DOI: 10.1109/TIT.2024.3352778.

Learning: Quantum Explained: Fundamentals and Computing Realities **mit2024**

MIT Open Learning. *Quantum Explained: Fundamentals and Computing Realities*. 2024. URL: https://www.youtube.com/watch?v=jk0jWzlvA5w.

Liu et al.: Introduction to Quantum Computing for Everyone: Experience Report **Liu2023IntroductionTQ**

Jonathan Liu and Diana Franklin. "Introduction to Quantum Computing for Everyone: Experience Report". In: *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1* (2023).

Lucerne University of Applied Sciences and Arts: A Quick Guide to Quantum Cryptography **Lucerne2024**

Lucerne University of Applied Sciences and Arts. *A Quick Guide to Quantum Cryptography*. Accessed: 2025-05-02. YouTube video. 2024. URL: https://www.youtube.com/watch?v=kpMwrnrEa-o.

Lyubashevsky et al.: Recent Theoretical Advances in Lattice-Based Cryptography: Security Proofs and Practical Efficiency **lattice_theory_2024**

Vadim Lyubashevsky, Chris Peikert, and Oded Regev. "Recent Theoretical Advances in Lattice-Based Cryptography: Security Proofs and Practical Efficiency". In: *Journal of Cryptology* 37.1 (2024), pp. 1–48. DOI: 10.1007/s00145-023-09456-4.

Madness: Classical Cryptography **madness2020classical**

Crypto Madness. *Classical Cryptography*. Specify Source: e.g., Website, Blog Post, Course Notes. 2020.

Magazine: Quantum Computers, Explained With Quantum Physics **quanta2021**

Quanta Magazine. *Quantum Computers, Explained With Quantum Physics*. 2021. URL: https://www.youtube.com/watch?v=jHoEjvuPoB8.

Makarov: Introduction to Quantum Cryptography **Makarov2014**

Vadim Makarov. *Introduction to Quantum Cryptography*. Accessed: 2025-05-02. YouTube video. 2014. URL: https://www.youtube.com/watch?v=ToOLbdrWst4.

Mandal et al.: Implementing Grover's on AES-based AEAD schemes **Mandal2024ImplementingGO**

Surajit Mandal et al. "Implementing Grover's on AES-based AEAD schemes". In: *Scientific Reports* 14 (2024).

Mangard et al.: Power Analysis Attacks: Revealing the Secrets of Smart Cards **mangard2007power**

Stefan Mangard, Elisabeth Oswald, and Thomas Popp. *Power Analysis Attacks: Revealing the Secrets of Smart Cards*. Springer, 2007.

McEliece: A Public-Key Cryptosystem Based On Algebraic Coding Theory **mceliece1978public**

Robert J. McEliece. *A Public-Key Cryptosystem Based On Algebraic Coding Theory*. DSN Progress Report 42-44. Jet Propulsion Laboratory, 1978, pp. 114–116. URL: https://ipnpr.jpl.nasa.gov/progress_report2/42-44/44N.PDF.

Menezes et al.: Digital Signatures **menezes_signatures**

Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone. "Digital Signatures". In: *Handbook of Applied Cryptography*. CRC Press, 1996. Chap. 11, pp. 425–486. ISBN: 978-0849385230. URL: https://cacr.uwaterloo.ca/hac/about/chap11.pdf.

Menezes et al.: Hash Functions and Data Integrity **menezes_hash**

Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone. "Hash Functions and Data Integrity". In: *Handbook of Applied Cryptography*. CRC Press, 1996. Chap. 9, pp. 321–383. ISBN: 978-0849385230. URL: https://cacr.uwaterloo.ca/hac/about/chap9.pdf.

Moody et al.: Post-Quantum Cryptography Deployment: Case Studies and Lessons Learned **pqc_deployment_cases_2024**

Dustin Moody, Ray A. Perlner, and Andrew Regenscheid. "Post-Quantum Cryptography Deployment: Case Studies and Lessons Learned". In: *Journal of Information Security and Applications* 78 (2024), p. 103601. DOI: 10.1016/j.jisa.2024.103601.

Moody et al.: Post-Quantum Cryptography Standardization: Timeline and Roadmap **nist_pqc_timeline_2024**

Dustin Moody et al. "Post-Quantum Cryptography Standardization: Timeline and Roadmap". In: *NIST Special Publications* 800-208 (Jan. 2024). URL: https://doi.org/10.6028/NIST.SP.800-208.

Mosca: Cybersecurity in an Era with Quantum Computers: Will We Be Ready? **mosca2018cybersecurity**

Michele Mosca. "Cybersecurity in an Era with Quantum Computers: Will We Be Ready?" In: *IEEE Security & Privacy* 16.5 (2018). Discusses quantum threats to current cryptographic systems and preparation strategies., pp. 38–41.

Mosca et al.: Economic Impact of Post-Quantum Cryptography on the Finance Sector **pqc_economic_impact_finance**

Michele Mosca, Prashant Sharma, and Douglas Stebila. "Economic Impact of Post-Quantum Cryptography on the Finance Sector". In: *Journal of Banking & Finance* 160 (2024), p. 106958. DOI: 10.1016/j.jbankfin.2023.106958.

Mosca et al.: Report on Post-Quantum Cryptography Workforce Requirements: Training, Best Practices, and Migration Strategies **pqc_workforce_report**

Michele Mosca, Douglas Stebila, and Christian Paquin. "Report on Post-Quantum Cryptography Workforce Requirements: Training, Best Practices, and Migration Strategies". In: *ACM Transactions on Management Information Systems* 15.2 (2024), 16:1–16:30. DOI: 10.1145/3593360.

National Cybersecurity Center of Excellence (NCCoE): Framework for Prioritizing Systems for Quantum-Resistant Cryptographic Upgrades **quantum_mitigation_framework_2024**

National Cybersecurity Center of Excellence (NCCoE). *Framework for Prioritizing Systems for Quantum-Resistant Cryptographic Upgrades*. Special Publication 1800-47. National Institute of Standards and Technology, Jan. 2024. URL: https://www.nccoe.nist.gov/projects/building-blocks/post-quantum-cryptography.

National Institute of Standards and Technology: Cryptographic Algorithm Validation Program (CAVP) Requirements and Procedures **nist_pqc_fips_205**

National Institute of Standards and Technology. *Cryptographic Algorithm Validation Program (CAVP) Requirements and Procedures*. Federal Information Processing Standards Publication FIPS 205. U.S. Department of Commerce, Aug. 2023. URL: https://doi.org/10.6028/NIST.FIPS.205.

National Institute of Standards and Technology (NIST): Data Encryption Standard (DES) **nist1999des**

National Institute of Standards and Technology (NIST). *Data Encryption Standard (DES)*. FIPS PUB 46-3. Withdrawn 2005. U.S. Department of Commerce, 1999.

National Institute of Standards and Technology (NIST): Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC **nist_sp800_38d**

National Institute of Standards and Technology (NIST). *Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC*. Special Publication (SP) 800-38D. U.S. Department of Commerce, 2007.

National Institute of Standards and Technology (NIST): SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions **nist_fips202**

National Institute of Standards and Technology (NIST). *SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions*. FIPS PUB 202. U.S. Department of Commerce, 2015.

National Institute of Standards and Technology (NIST): Recommendation for Key Management, Part 1: General **nist_sp800_57p1r5**

National Institute of Standards and Technology (NIST). *Recommendation for Key Management, Part 1: General*. Special Publication (SP) 800-57 Part 1 Rev. 5. U.S. Department of Commerce, 2020.

National Institute of Standards and Technology (NIST): Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardization Process **nist_pqc**

National Institute of Standards and Technology (NIST). *Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardization Process*. NIST Internal Report (NISTIR) 8309. Represents the general NIST PQC effort; cite specific documents when possible. U.S. Department of Commerce, 2020. URL: https://csrc.nist.gov/projects/post-quantum-cryptography.

National Security Agency (NSA) Cybersecurity Directorate: Achieving Cryptographic Agility **cryptographic-agility**

National Security Agency (NSA) Cybersecurity Directorate. *Achieving Cryptographic Agility*. Cybersecurity Information Sheet. Accessed: 2024-05-15. Sept. 2023. URL: https://media.defense.gov/2023/Sep/19/2003304377/-1/-1/0/CSI-Achieving-Cryptographic-Agility.PDF.

Nielsen et al.: Quantum Computation and Quantum Information **nielsen_chuang_book**

Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. 10th Anniversary Edition. Cambridge, UK: Cambridge University Press, 2010.

O'Connor et al.: BLAKE3 Cryptographic Hash Function **blake3**

Jack O'Connor et al. *BLAKE3 Cryptographic Hash Function*. https://blake3.io/. Specification available at https://github.com/BLAKE3-team/BLAKE3-specs/blob/master/blake3.pdf. 2020.

Oder et al.: Hardware Acceleration for Post-Quantum Cryptography: Designs and Implementations **hardware_acceleration_2024**

Tobias Oder, Tim Güneysu, and Eiichiro Fujisaki. "Hardware Acceleration for Post-Quantum Cryptography: Designs and Implementations". In: *IEEE Transactions on Computers* 73.4 (2024), pp. 1127–1142. DOI: 10.1109/TC.2023.3329875.

Ounsworth et al.: Hybrid Cryptography: Combining Classical and Post-Quantum Algorithms for Enhanced Security **hybrid_crypto_2024**

Mike Ounsworth, Christian Paquin, and Tim Dierks. "Hybrid Cryptography: Combining Classical and Post-Quantum Algorithms for Enhanced Security". In: *Journal of Cryptographic Engineering* 14.1 (2024), pp. 61–76. DOI: 10.1007/s13389-023-00327-z.

Peikert: A decade of lattice cryptography **peikert2016decade**

Chris Peikert. "A decade of lattice cryptography". In: *Foundations and Trends in Theoretical Computer Science* 10.4 (2016). Excellent survey of lattice-based crypto, pp. 283–424.

Preskill: Quantum Computing in the NISQ Era and Beyond **preskill2018quantum**

John Preskill. "Quantum Computing in the NISQ Era and Beyond". In: *Quantum* 2 (2018). Discusses near-term quantum computing and its limitations., p. 79.

Preskill: Ph/CS 219A Quantum Computation Lectures **preskill2021**

John Preskill. *Ph/CS 219A Quantum Computation Lectures*. 2021. URL: https://www.youtube.com/playlist?list=PL0ojjrEqIyPy-1RRD8cTD_lF1hflo89Iu.

QNu Labs: Quantum Cryptography Explained Step by Step **QNuLabs2024**

QNu Labs. *Quantum Cryptography Explained Step by Step*. Accessed: 2025-05-02. YouTube video. 2024. URL: https://www.youtube.com/watch?v=SRLfQ5rC-Jw.

Quantum Computing Explained: How Does Quantum Cryptography Work? **QuantumBasics2022**

Quantum Computing Explained. *How Does Quantum Cryptography Work?* Accessed: 2025-05-02. YouTube video. 2022. URL: https://www.youtube.com/watch?v=SPNehFLwbb8.

Ravi et al.: New Metrics for Implementation Security of Post-Quantum Cryptographic Algorithms **implementation_security_2024**

Prasanna Ravi, Romain Poussier, and Anupam Chattopadhyay. "New Metrics for Implementation Security of Post-Quantum Cryptographic Algorithms". In: *IACR Transactions on Cryptographic Hardware and Embedded Systems* 2024.1 (2024), pp. 226–257. DOI: 10.46586/tches.v2024.i1.226-257.

Ravi et al.: Side-Channel Resistant Implementations of Post-Quantum Cryptographic Algorithms **side_channel_2024**

Prasanna Ravi, Romain Poussier, and François-Xavier Standaert. "Side-Channel Resistant Implementations of Post-Quantum Cryptographic Algorithms". In: *IACR Transactions on Cryptographic Hardware and Embedded Systems* 2024.1 (2024), pp. 193–225. DOI: 10.46586/tches.v2024.i1.193-225.

Regenscheid: Transition to Post-Quantum Cryptography Standards **Regenscheid2024TransitionTP**

Andrew Regenscheid. "Transition to Post-Quantum Cryptography Standards". In: 2024.

Regenscheid et al.: International Cooperation on Post-Quantum Cryptography Standards **pqc_standards_cooperation**

Andrew Regenscheid, Dustin Moody, and Rene Struik. "International Cooperation on Post-Quantum Cryptography Standards". In: *IEEE Communications Standards Magazine* 8.1 (2024), pp. 32–43. DOI: 10.1109/MCOMSTD.2023.3337521.

Scaibu: Grover's Search Algorithm: A Complete Guide with Code Examples **ScaibuGroversGuide**

Scaibu. *Grover's Search Algorithm: A Complete Guide with Code Examples*. 2024. URL: https://scaibu.substack.com/p/grovers-search-algorithm-a-complete (visited on 05/02/2025).

Schwabe et al.: Optimizing Post-Quantum Cryptography Implementations: Recent Research Developments **pqc_optimization_2024**

Peter Schwabe, Tim Güneysu, and Matthias J. Kannwischer. "Optimizing Post-Quantum Cryptography Implementations: Recent Research Developments". In: *ACM Computing Surveys* 56.6 (2024). This is a survey. If a specific Kyber optimization paper was cited, replace this reference., 123:1–123:36. DOI: 10.1145/3609961.

Sendrier et al.: New Constructions in Code-Based Cryptography: Addressing Size and Performance Challenges **code_based_2024**

Nicolas Sendrier, Jean-Pierre Tillich, and Tung Chou. "New Constructions in Code-Based Cryptography: Addressing Size and Performance Challenges". In: *IEEE Transactions on Information Theory* 70.5 (2024), pp. 3142–3158. DOI: 10.1109/TIT.2024.3347142.

Shannon: Communication Theory of Secrecy Systems **shannon1949communication**

Claude E. Shannon. "Communication Theory of Secrecy Systems". In: *Bell System Technical Journal* 28.4 (1949), pp. 656–715.

Shor: Algorithms for quantum computation: discrete logarithms and factoring **shor1994algorithms**

Peter W. Shor. "Algorithms for quantum computation: discrete logarithms and factoring". In: *Proceedings 35th Annual Symposium on Foundations of Computer Science*. The seminal paper on Shor's algorithm. IEEE. 1994, pp. 124–134.

Shor: Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer **shor1997polynomial**

Peter W. Shor. "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer". In: *SIAM Journal on Computing* 26.5 (1997). The foundational paper describing Shor's algorithm for integer factorization., pp. 1484–1509.

Simplilearn: Cryptography And Network Security Full Course **crypto6**

Simplilearn. *Cryptography And Network Security Full Course*. YouTube. 3-hour comprehensive course covering DES, AES, RSA, and SSL/TLS. 2021. URL: https://www.youtube.com/watch?v=C7vmouDOJYM.

Simplilearn: Cryptography Tutorial **crypto2**

Simplilearn. *Cryptography Tutorial*. YouTube. Introduction to cryptography fundamentals and real-world applications. 2021. URL: https://www.youtube.com/watch?v=rjWx39mB4Sc.

Singh: The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography **singh1999code**

Simon Singh. *The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography*. Anchor Books, 1999.

SOAR: Quantum Computing Course – Math and Theory for Beginners **soar2024**

Quantum SOAR. *Quantum Computing Course – Math and Theory for Beginners*. 2024. URL: https://www.youtube.com/watch?v=tsbCSkvHhMo.

Stallings: Cryptography and Network Security: Principles and Practice **stallings2017cryptography**

William Stallings. *Cryptography and Network Security: Principles and Practice*. 7th. Comprehensive, practical focus. Pearson, 2017.

Stebila et al.: Applications of Post-Quantum Cryptography: Current State and Future Directions **pqc_applications_2024**

Douglas Stebila, Michele Mosca, and Panos Kampanakis. "Applications of Post-Quantum Cryptography: Current State and Future Directions". In: *ACM Computing Surveys* 56.5 (2024), 102:1–102:36. DOI: 10.1145/3603606.

Thomson: What is quantum superposition? **Thomson2025LiveScienceSuperposition**

Jess Thomson. *What is quantum superposition?* Live Science. Apr. 2025. URL: https://www.livescience.com/technology/computing/what-is-quantum-superposition-and-what-does-it-mean-for-quantum-computing (visited on 05/02/2025).

Up and Atom: Quantum Cryptography in 6 Minutes **UpAndAtom2017**

Up and Atom. *Quantum Cryptography in 6 Minutes*. Accessed: 2025-05-02. YouTube video. 2017. URL: https://www.youtube.com/watch?v=uiiaAJ3c6dM.

Veritasium: Quantum Computers, Explained with MKBHD **veritasium2023**

Veritasium. *Quantum Computers, Explained with MKBHD*. 2023. URL: https://www.youtube.com/watch?v=e3fz3dqhN44.

votatera: Quantum Mechanics | Principles, Applications & Theory **votatera2024quantum**

votatera. *Quantum Mechanics | Principles, Applications & Theory*. urlhttps://modern-physics.org/quantum-mechanics/. Accessed: 2025-05-02. May 2024.

What is Cryptography Full Tutorial **crypto3**

What is Cryptography Full Tutorial. YouTube. Comprehensive tutorial covering encryption algorithms and cryptographic protocols. 2021. URL: https://www.youtube.com/watch?v=CajoamYNxz0.

WhiteboardDoodles: Cryptography Basics: Intro to Cybersecurity **crypto4**

WhiteboardDoodles. *Cryptography Basics: Intro to Cybersecurity*. YouTube. Visual explanation of encryption, hashing, and PKI fundamentals. 2024. URL: https://www.youtube.com/watch?v=2oXKjPwBSUk.

Wiebe: Advanced Quantum Algorithms Lectures **wiebe2021**

Nathan Wiebe. *Advanced Quantum Algorithms Lectures*. 2021. URL: https://youtube.com/playlist?list=[specified_in_source2].

Wikipedia contributors: Quantum mechanics — Wikipedia, The Free Encyclopedia **wikipedia2025quantum**

Wikipedia contributors. *Quantum mechanics — Wikipedia, The Free Encyclopedia*. [Online; accessed 2-May-2025]. 2025. URL: https://en.wikipedia.org/w/index.php?title=Quantum_mechanics&oldid=1286231207 (visited on 05/02/2025).