Quantum-Ready Cryptography: Mathematical Foundations for Post-Quantum Data Protection in Critical Infrastructure

Mazvita Velah
Yeshiva University - Cybersecurity

Brian Kudakwashe Zanje
Worcester Polytechnic Institute - Information Technology (MSIT)

Godfrey Junior Madonera
Worcester Polytechnic Institute - Business Analytics MSBA

Necessitate Siqhoza
Yeshiva University – Mathematics

ABSTRACT
The advent of large-scale quantum computing poses a fundamental threat to classical public-key cryptographic systems underpinning modern digital infrastructure. Widely deployed schemes such as RSA and elliptic curve cryptography rely on computational hardness assumptions—integer factorization and discrete logarithms—that are vulnerable to quantum algorithms such as Shor's algorithm. As critical infrastructure systems increasingly depend on secure communication, authentication, and data integrity mechanisms, the transition to quantum-resistant cryptography has become an urgent national and global priority. This paper examines the mathematical foundations of post-quantum cryptography (PQC) and their application to data protection in critical infrastructure environments, analyzes core hardness assumptions underlying leading PQC families, proposes a quantum-readiness framework for critical infrastructure sectors, and presents empirical benchmarking results across energy, healthcare, financial, and transportation systems. By integrating rigorous mathematical analysis with experimental performance data, this work provides a structured roadmap for transitioning mission-critical systems toward quantum-resilient data protection architectures.

Keywords
Post-quantum cryptography (PQC); Quantum-resistant algorithms; Lattice-based cryptography; Learning with Errors (LWE); Code-based cryptography; Multivariate cryptosystems; Hash-based signatures; Shor's algorithm; Cryptographic agility; Critical infrastructure security; Computational hardness assumptions

1. INTRODUCTION
1.1 The Quantum Threat to Classical Cryptography
The emergence of large-scale quantum computing constitutes one of the most consequential technological disruptions of the twenty-first century. While quantum systems promise transformative advances across disciplines—from combinatorial optimization to molecular simulation—they simultaneously introduce a profound and structural threat to the cryptographic foundations upon which modern digital infrastructure is built. Virtually every encrypted communication, authenticated identity, digitally signed transaction, and confidential data store in contemporary use derives its security from mathematical problems that classical computers cannot solve efficiently. Quantum computers, operating under fundamentally different physical principles, are not subject to these same computational constraints.

The dominant public-key cryptographic schemes in widespread deployment today derive their security from two related number-theoretic hardness assumptions. The RSA cryptosystem, introduced by Rivest, Shamir, and Adleman in 1978 [1], encodes its security in the computational intractability of factoring the product of two large prime numbers, where encryption uses modular exponentiation $y \equiv x^e \ (mod\, n)$ and decryption relies on the private exponent d such that $y \equiv y^d \ (mod\, n)$. Elliptic curve cryptography (ECC), developed independently by Miller [2] and Koblitz [3], achieves equivalent security at smaller key sizes by operating over the algebraic group structure of elliptic curves defined over finite fields, where the best classical algorithms require sub-exponential effort. For decades, these assumptions withstood sustained cryptanalytic scrutiny under the assumption of classical computation.

This security posture was fundamentally challenged in 1994 when Peter Shor published a quantum algorithm capable of solving both integer factorization and the discrete logarithm problem in polynomial time on a fault-tolerant quantum computer [4]. Shor's algorithm exploits quantum parallelism and the quantum Fourier transform to determine the period of a modular exponentiation function—reducing what is exponentially hard classically to polynomial in the input size. A sufficiently capable quantum computer executing Shor's algorithm would compromise 2048-bit RSA in hours, a computation requiring resources exceeding the projected lifetime of the observable universe on any classical system. Independently, Grover's 1996 quantum search algorithm demonstrated unstructured search in $\vartheta(\sqrt{N})$ quantum operations rather than $\vartheta(N)$ classically, effectively halving the bit-security of symmetric primitives and hash functions—addressed by doubling key and output lengths [5].

The practical timeline for cryptographically relevant quantum computers (CRQCs) remains subject to expert debate, with estimates ranging from one to three decades depending on progress in fault-tolerant qubit realization, error correction overhead, and quantum memory coherence [6]. Nevertheless, the question of *when* such systems will arrive does not determine *when* cryptographic migration must commence. As

established in Section 1.2, the effective exposure of sensitive data begins at the moment of interception—not the moment of decryption.

1.2 Critical Infrastructure and the Temporal Asymmetry of Quantum Risk

The implications of quantum-enabled cryptanalysis are especially acute for critical infrastructure sectors, defined by the U.S. Department of Homeland Security to include energy, water, financial services, healthcare, transportation, and communications systems. These sectors share two characteristics that amplify quantum risk: **(i)** they depend on long-lived cryptographic deployments, often spanning a decade or more without comprehensive replacement, and **(ii)** they generate and transmit data whose confidentiality, integrity, and authenticity requirements persist over extended time horizons—health records requiring 30-year HIPAA retention, classified intelligence, long-term financial instruments, and critical infrastructure control telemetry.

A particularly consequential threat vector is the strategy known as *"harvest now, decrypt later"* (HNDL). Under this model, adversarial actors systematically intercept and archive encrypted network traffic today with the explicit intention of decrypting it retrospectively once CRQCs become available [7]. Mosca's inequality formalizes this risk: if the sum of the migration timeline and the shelf-life of sensitive data exceeds the time to a CRQC, migration is already overdue. The cryptographic exposure of such data is not a future event—it begins at the moment of interception, a temporal asymmetry visually formalized in Figure 1. Nation-state actors with long-term strategic planning horizons are particularly well-positioned to exploit this dynamic.

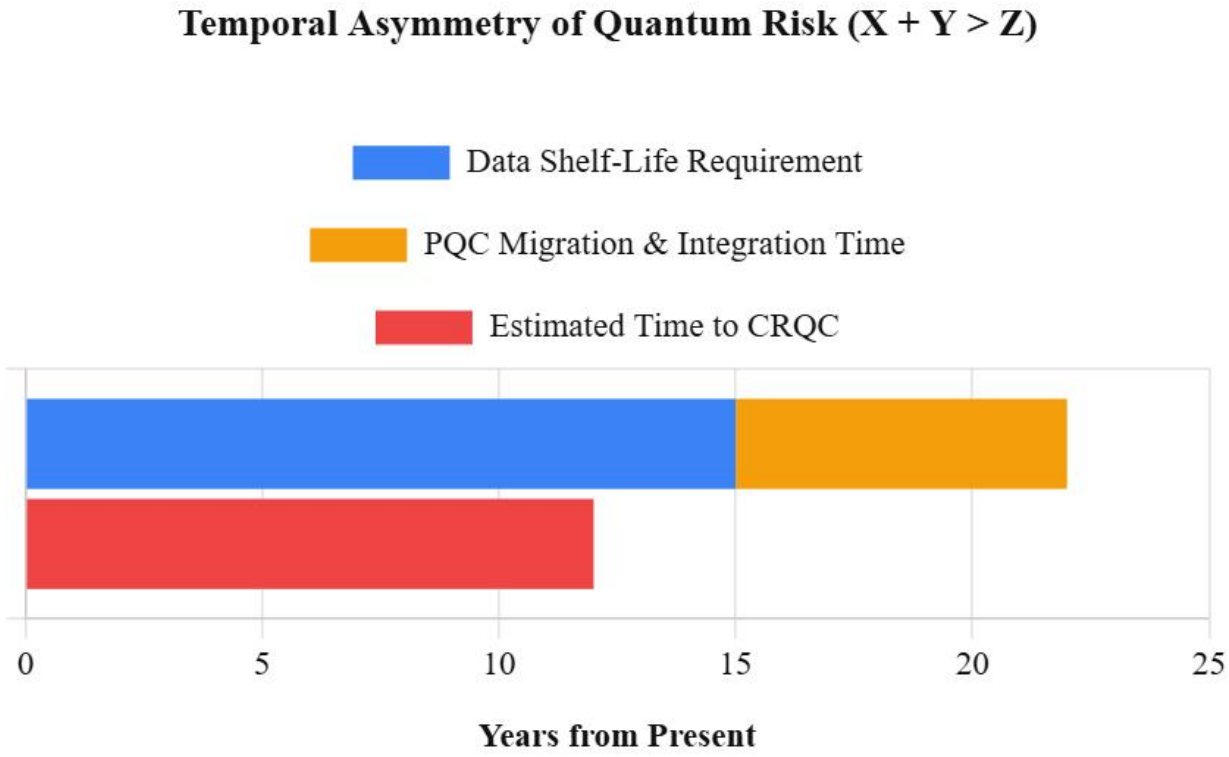

Figure 1: Temporal Asymmetry of Quantum Risk (Mosca's Theorem)

Compounding this vulnerability is the extended timeline required for cryptographic infrastructure modernization. The lifecycle of embedded systems in industrial control environments—programmable logic controllers (PLCs), remote terminal units (RTUs), and SCADA supervisory platforms—commonly spans fifteen to twenty-five years. Even in agile enterprise environments, the transition from RSA/ECC to post-quantum algorithms entails coordinated updates across hardware, firmware, middleware, network protocols, and PKI certificate chains. The convergence of long data sensitivity windows and long migration timescales means that the effective deadline for initiating post-quantum migration has, by most authoritative assessments, already passed for the highest-sensitivity applications [8].

1.3 Post-Quantum Cryptography: Foundations and Standardization

Post-quantum cryptography (PQC) refers to the design, analysis, and deployment of cryptographic algorithms believed to be computationally secure against both classical and quantum adversaries. Critically, PQC operates entirely within classical computational and communications infrastructure—it requires no quantum hardware, no entangled photon channels, and no specialized network equipment. This distinguishes PQC from quantum key distribution (QKD), which achieves key establishment through quantum mechanical principles but requires dedicated fiber infrastructure and offers no protection for the broader ecosystem of public-key operations including digital signatures [9].

The mathematical foundations of PQC are deliberately pluralistic, organized into several major families each

predicated on a distinct hardness assumption:

- **Lattice-Based Cryptography:** Derives security from the computational intractability of problems over high-dimensional integer lattices, principally the Learning with Errors (LWE) problem and its structured variants (Ring-LWE, Module-LWE). Lattice-based schemes underpin the majority of NIST-selected post-quantum standards.
- **Code-Based Cryptography:** Grounds security in the NP-hardness of decoding a random linear error-correcting code. The McEliece cryptosystem [10] has withstood five decades of cryptanalytic scrutiny, conferring an unusual degree of long-term confidence despite large key sizes.
- **Multivariate Polynomial Cryptography:** Constructs security on the NP-hardness of solving systems of multivariate quadratic (MQ) equations over finite fields—a problem believed to resist quantum speedup, offering compact signatures suited to constrained devices.
- **Hash-Based Signature Schemes:** Rely exclusively on the collision resistance and one-wayness of cryptographic hash functions—properties whose quantum security requires only a doubling of output length to offset Grover's quadratic speedup. These schemes carry the most conservative and well-understood security basis.

In 2016, the U.S. National Institute of Standards and Technology (NIST) initiated a comprehensive multi-year process to evaluate and standardize post-quantum cryptographic algorithms [11]. Following four rounds of public evaluation, NIST finalized its inaugural post-quantum standards in 2024: FIPS 203 (ML-KEM, a Module-Lattice-based Key Encapsulation Mechanism), FIPS 204 (ML-DSA, a lattice-based Digital Signature Algorithm), and FIPS 205 (SLH-DSA, a Stateless Hash-Based Digital Signature Algorithm) [12]. In parallel, the National Security Agency issued the Commercial National Security Algorithm Suite 2.0 (CNSA 2.0) [8], mandating transition timelines for national security systems and signaling the urgency with which government stakeholders regard the post-quantum migration imperative.

1.4 Research Objectives and Scope

This paper addresses the intersection of post-quantum cryptographic theory and critical infrastructure deployment. The research is organized around three interconnected objectives:

1) **Mathematical Foundations:** To develop rigorous algebraic and complexity-theoretic foundations for the principal PQC families—lattice-based, code-based, multivariate, and hash-based—establishing the hardness assumptions, security reductions, parameter spaces, and structural properties governing their security guarantees against both classical and quantum adversaries.
2) **Sector-Specific Threat Mapping:** To map these cryptographic constructions to the specific confidentiality, integrity, availability, and performance requirements of critical infrastructure sectors—energy, healthcare, finance, and transportation—analyzing sector-specific threat models, regulatory environments, and deployment constraints.
3) **Quantum-Readiness Framework:** To synthesize mathematical analysis and empirical benchmarking into a structured quantum-readiness framework addressing cryptographic agility, hybrid transition architectures, implementation security, key size and performance trade-offs, and long-term security governance appropriate for mission-critical environments.

The scope focuses on asymmetric cryptographic primitives—public-key encryption, key encapsulation, and digital signatures—as these are most directly threatened by Shor's algorithm and most urgently requiring replacement. Symmetric algorithms are addressed in the context of hybrid schemes but not independently analyzed. QKD is excluded given its infrastructure requirements and inapplicability to many critical infrastructure communication patterns.

1.5 Organization of the Paper

Chapter 2 reviews classical public-key cryptographic schemes in detail and provides a formal treatment of Shor's and Grover's algorithms and their cryptanalytic implications, surveying the existing literature. Chapter 3 develops the mathematical framework for each major PQC family and specifies the comparative methodology. Chapter 4 details experimental design and simulation procedures across four infrastructure sectors. Chapter 5 presents benchmarking results and performance analysis. Chapter 6 identifies future research directions and evolving standardization. Chapter 7 concludes with strategic recommendations for policy makers, infrastructure operators, and security architects. Throughout, this work maintains a dual commitment to mathematical rigor and practical applicability—the mathematical foundations examined here are the structural basis upon which the next generation of secure infrastructure must be built.

2. LITERATURE REVIEW

2.1 Classical Cryptography and Current Limitations

Classical cryptography refers to the earliest methods of secure communication developed and used long before the advent of computers, primarily relying on pen-and-paper techniques to ensure message confidentiality [13]. These historical systems, such as the Caesar cipher and other substitution or transposition ciphers, operated on the fundamental principle of symmetric-key encryption, where the sender and receiver shared the same secret key [14]. While these methods laid the essential groundwork for the field, they are now considered obsolete for serious security purposes, as most are vulnerable to attacks such as frequency analysis and brute-force computation [15].

Although original literature provides the mathematical strength of these systems with classical limitations, the recent literature has taken an offensive turn to assessing their weakness with quantum frames [16]. Proos and Zalka [17] showed that ECC is arguably more susceptible to the algorithm of Shor than RSA and fewer logical qubits are needed to break the same security levels. The benchmarking of the liboqs library has so far been performed with server-class architectures and the standard web protocols. Nevertheless, there is a critical gap in analytical understanding of the implementation of PQC algorithms to the limitations of memory, bandwidth, and real-time latency of embedded critical infrastructure, including, but not limited to, SCADA systems and V2X networks. This gap is directly considered in this paper, where the degradation of performance of NIST-standardized PQC families operating under highly constrained performance envelopes is isolated.

2.1.1 RSA and Integer Factorization

The RSA cryptosystem, introduced by Rivest, Shamir, and Adleman in 1978, remains one of the most widely recognized asymmetric cryptographic algorithms used for encryption, digital signatures, and key exchange [1]. RSA's security is based on the mathematical asymmetry between the

computational ease of multiplying two large prime numbers and the practical difficulty of factoring their product, known as a semi-prime [18]. During key generation, two distinct large primes p and q are selected to compute $n = pq$, forming part of the public key, while the totient function $\emptyset(n) = (p-1)(q-1)$ is used to derive the private exponent. Encryption is performed using the public key (n, e) through modular exponentiation $y = x^e \bmod n$, and decryption uses the private key (n, d) such that $x = y^d \bmod n$ [16]. In practical implementations, the modulus n typically ranges from 1024 to 2048 bits or higher to resist classical factorization attacks. While no fundamental mathematical weakness has been discovered in the core RSA algorithm, vulnerabilities have emerged through improper key generation, weak prime selection, protocol flaws, and side-channel attacks [19]. However, advances in quantum computing fundamentally challenge this security model, particularly due to Shor's algorithm, which can factor integers in polynomial time [4]

2.1.2 Elliptic Curve Cryptography and the Discrete Logarithm

Elliptic Curve Cryptography (ECC) is widely used to secure modern digital systems because it relies on the computational difficulty of the elliptic curve discrete logarithm problem [2], [3]. Under classical computing assumptions, deriving a private key from a public elliptic curve point is considered infeasible, which enables secure communication and authentication across many critical applications with significantly smaller key sizes than RSA at equivalent security levels. However, advances in quantum computing threaten this security model: Shor's algorithm can efficiently solve discrete logarithm problems, weakening the mathematical assumptions that support ECC [4]. As a result, ECC, like RSA, becomes vulnerable in a future quantum computing environment, highlighting the need for quantum-resistant cryptographic approaches.

2.1.3 Quantum Algorithms Undermine Classical Assumptions

Quantum algorithms fundamentally challenge the security assumptions underlying classical public-key cryptography. Algorithms such as Shor's algorithm demonstrate that problems once considered computationally infeasible—including integer factorization and discrete logarithms—can be solved efficiently on a sufficiently powerful quantum computer [4]. Because cryptographic systems like RSA and ECC rely on these hardness assumptions, the development of large-scale quantum computing threatens the long-term security of modern digital infrastructure, accelerating global research into post-quantum cryptography designed to resist both classical and quantum adversaries.

2.2 Quantum Threats

The development of quantum computing poses a significant threat to current cryptographic systems. Shor's algorithm can efficiently solve integer factorization and discrete logarithmic problems, putting widely used public-key schemes such as RSA, Diffie-Hellman, and elliptic curve cryptography at risk [11]. Grover's algorithm also weakens symmetric encryption by accelerating brute-force searches, effectively reducing the security strength of algorithms like AES [20]. Although large-scale quantum computers are not yet available, the "harvest now, decrypt later" strategy already threatens the long-term confidentiality of encrypted data [7].

2.2.1 Shor's Algorithm

Shor's algorithm is widely regarded as a critical quantum threat to traditional public-key cryptography. It demonstrates that problems such as integer factorization and the discrete logarithm—which form the mathematical backbone of systems like RSA and ECC—can be computed efficiently on a sufficiently capable quantum processor [4]. Since these hardness assumptions are central to many authentication and key-exchange protocols, practical quantum machines could significantly weaken the security guarantees of current digital communication infrastructure. This growing risk has intensified global research into post-quantum cryptographic schemes designed to remain secure even in the presence of quantum-enabled adversaries [9], [21].

2.2.2 Grover's Algorithm and Symmetric Security

Grover's algorithm presents another significant quantum advancement impacting cryptographic security. The algorithm enables a quantum computer to accelerate brute-force search by reducing the complexity of searching an unsorted space from O(N) to approximately O(√N) [5]. This quadratic speedup allows attackers to test cryptographic keys more efficiently than classical exhaustive search. As a result, symmetric cryptographic systems are not completely broken but experience a reduction in effective security strength—meaning larger key sizes are required to maintain equivalent protection. For instance, AES-128 effectively provides only 64-bit quantum security under Grover's algorithm, necessitating migration to AES-256 for equivalent resistance. Studies examining Grover's algorithm in brute-force attacks highlight its implications for modern encryption systems and the need to strengthen cryptographic parameters [20].

2.2.3 Long-Term Confidentiality and Harvest-Now-Decrypt-Later

Quantum computing poses a significant challenge to the long-term confidentiality of classical cryptographic systems. Data encrypted today using traditional public-key algorithms such as RSA and ECC may remain secure against classical attacks, but could become vulnerable once large-scale quantum computers are developed. This risk is captured by the "harvest now, decrypt later" (HNDL) threat model, where adversaries collect encrypted data today with the intention of decrypting it in the future using quantum computing capabilities [7]. As a result, sensitive information stored or transmitted by critical infrastructure systems faces long-term exposure if cryptographic protections are not updated to quantum-resistant alternatives. Mosca's inequality formalizes this urgency: migration must begin today for any data whose confidentiality value extends beyond the anticipated quantum advantage horizon.

2.3 Post-Quantum Cryptography

2.3.1 Resistance to Classical and Quantum Adversaries

The emergence of scalable quantum computation represents a structural rupture in the history of public-key cryptography. Conventional asymmetric schemes—most notably RSA, Diffie-Hellman, and ECC—derive their security from the presumed computational intractability of integer factorization and the discrete logarithm problem. Shor's algorithm, introduced in 1994, demonstrated that a fault-tolerant quantum computer can solve both problems in polynomial time, rendering these schemes cryptographically broken in a post-quantum setting [4]. Grover's algorithm further compounds the problem for symmetric and hash-based primitives by providing a quadratic speedup in unstructured search, effectively halving the bit-security of any scheme whose security rests on

exhaustive key search [5].

The threat model motivating PQC therefore extends across two distinct attack horizons. The first is the prospective horizon: once CRQCs become operational, any system still relying on pre-quantum public-key schemes will be immediately compromised. The second—and arguably more urgent—is the retrospective horizon captured by the HNDL paradigm [7], [22]. PQC addresses both horizons by grounding security in mathematical problems that admit no known polynomial-time quantum algorithms. Current candidate problem families—worst-case lattice problems, syndrome decoding, and multivariate polynomial equations—are believed to be quantum-resistant because Shor's techniques rely on exploiting the hidden-subgroup structure of abelian groups, a structure absent in these alternative hardness settings [9]. An underappreciated dimension concerns implementation-level side-channel attacks, which threaten PQC schemes independently of their mathematical hardness. Timing attacks, power analysis, and fault injection can extract secret information from physically accessible hardware even when the underlying algebraic problem is computationally intractable [16].

2.3.2 Leading PQC Families

Post-quantum cryptographic constructions are organized into several major families, each predicated on a distinct mathematical hardness assumption. Rather than constituting competing alternatives, these families are more accurately understood as a portfolio of complementary primitives, each offering different trade-offs across key size, computational cost, security reduction quality, and suitability for specific deployment environments. NIST's multi-round standardization process, concluded in 2024 with the publication of FIPS 203, FIPS 204, and FIPS 205, validated four primary schemes [22].

Lattice-based cryptography currently dominates both academic attention and standardization outcomes. Its security rests on the hardness of problems defined over integer lattices, principally the Learning with Errors (LWE) problem introduced by Regev [23] and its structured variant, Module-LWE (MLWE). MLWE preserves worst-case hardness reductions while dramatically reducing key sizes by exploiting the algebraic structure of module lattices—a design choice that underlies CRYSTALS-Kyber (ML-KEM under FIPS 203) and the signature schemes CRYSTALS-Dilithium (ML-DSA, FIPS 204) and Falcon (FN-DSA, FIPS 206 draft) [24], [25]. A notable conceptual advance is the Fiat–Shamir with Aborts technique, which transforms an interactive identification protocol into a non-interactive signature scheme with tight security reductions in the random-oracle model [26].

Code-based cryptography predates the PQC field proper: the McEliece cryptosystem, proposed in 1978, encodes messages using error-correcting codes and exploits the NP-hardness of decoding a general linear code [10]. Half a century of cryptanalysis has left its core hardness assumption essentially intact, conferring an unusual degree of long-term confidence. Modern variants BIKE and HQC, retained as alternate candidates in NIST Round 4, address the original scheme's large key sizes by replacing binary Goppa codes with quasi-cyclic codes, reducing public keys from approximately one megabyte to tens of kilobytes [27].

Multivariate cryptography is founded on the NP-hardness of solving systems of multivariate quadratic (MQ) polynomial equations over finite fields. Signature schemes in this family offer very compact signatures and rapid verification—properties attractive for constrained devices [28]. However, the family has experienced significant cryptanalytic turbulence: Rainbow was broken by a rank attack prior to the conclusion of NIST Round 3 [29], illustrating the structural fragility that can arise when algebraic shortcuts exist in the polynomial system. The remaining candidate MAYO—based on the Oil-and-Vinegar structure—was submitted to NIST's ongoing additional signature competition.

Hash-based cryptography occupies a unique position because its security rests exclusively on the collision and preimage resistance of the underlying hash function—the most conservatively analyzed assumption in cryptography. Stateful schemes such as XMSS and LMS, standardized by IETF (RFC 8391 and RFC 8554), achieve strong security but require careful state management to prevent signature reuse [30]. SPHINCS+, standardized as SLH-DSA under FIPS 205, adopts a stateless construction by chaining multiple Merkle trees under a hypertree architecture, eliminating state management burden at the cost of somewhat larger signature sizes [31].

Isogeny-based cryptography, though not among the schemes selected in NIST's primary standardization track, merits attention as a theoretically distinct approach whose security derives from the difficulty of finding isogenies between elliptic curves over finite fields. SIKE, its most prominent representative, was spectacularly broken in 2022 by a classical polynomial-time attack exploiting auxiliary torsion-point information simultaneously illustrating the risks of novel mathematical assumptions and underscoring the value of the portfolio approach. SQIsign and CSIDH remain active research subjects [32].

2.3.3 Complexity-Theoretic Security Guarantees

The credibility of PQC as a long-term security foundation rests not simply on the absence of known quantum attacks, but on the quality of formal reductions linking cryptographic security to well-characterized computational hardness problems. In a reduction-based security proof, an adversary capable of breaking the cryptographic scheme is transformed, with polynomial overhead, into an algorithm that solves an instance of the underlying hard problem. The strength of the resulting guarantee depends on how well the assumed hard problem is understood and how tight the reduction is.

Lattice-based cryptography provides the most compelling reductions in the PQC landscape. Regev's foundational work established a quantum reduction from the worst-case hardness of the Shortest Vector Problem (SVP) on arbitrary lattices to the average-case hardness of LWE [23]. This worst-case-to-average-case reduction means that breaking a randomly sampled LWE instance is at least as hard as solving SVP on the hardest lattice of a given dimension—a rare and powerful property. Peikert (2009) subsequently provided a classical variant of this reduction, and subsequent work on Ring-LWE and Module-LWE extended these guarantees to structured variants [33].

Complexity-theoretic security proofs in PQC must explicitly account for quantum adversaries. The standard computational security framework models adversaries as probabilistic polynomial-time (PPT) machines; in the quantum setting, this is upgraded to quantum polynomial-time (QPT) machines. The random-oracle model (ROM) has a quantum analogue—the quantum random-oracle model (QROM)—which permits the adversary to query the random oracle in superposition. CRYSTALS-Kyber, Dilithium, and SPHINCS+ all have established QROM proofs [34]. The QROM is now considered the minimum adequate model for PQC security proofs targeting post-quantum deployments.

Concrete security analysis is particularly consequential for deployed systems. The Lattice Estimator, a community-maintained software tool, provides standardized bit-security estimates by simulating multiple attack strategies including BKZ, sieving, and hybrid attacks [35]. For code-based schemes, Information Set Decoding (ISD) algorithms define the concrete hardness of syndrome decoding, with quantum improvements known to provide at most a square-root speedup over classical ISD. These concrete analyses form the empirical backbone of the NIST standardization process and are indispensable for translating asymptotic hardness assumptions into practical parameter recommendations.

The question of hybrid cryptography intersects directly with complexity-theoretic security. A hybrid KEM that derives a session key from both an ECDH exchange and an ML-KEM encapsulation is secure as long as at least one component remains unbroken, providing a graceful migration path. RFC 8446's TLS 1.3 framework has been extended to accommodate hybrid key exchange in several IETF drafts, and recent work has produced tight security reductions for standard hybrid KEM constructions under mild assumptions on the component schemes [36].

3. CORE TECHNICAL METHODOLOGY

This chapter defines the methodological architecture of the proposed quantum-readiness framework. It specifies the mathematical constructions selected for evaluation, the criteria governing their selection, the implementation environment used for simulation, and the stepwise procedures applied to measure performance across candidate post-quantum cryptographic (PQC) algorithms. The methodology is designed to be reproducible and sector-agnostic, enabling direct comparison of PQC families under constraints representative of real-world critical infrastructure.

3.1 Research Design and Methodological Approach

The study adopts a comparative experimental methodology, combining formal mathematical analysis with software-based simulation. The design proceeds along three parallel tracks:

1) **Theoretical Track:** Formal examination of hardness assumptions, algebraic structures, and quantum resistance proofs for each PQC family.
2) **Empirical Track:** Benchmarked implementation of NIST-standardized and finalist PQC algorithms under controlled conditions, measuring latency, throughput, key and ciphertext sizes, and memory footprint.
3) **Application Track:** Mapping of experimental results to infrastructure-specific deployment profiles (energy, healthcare, finance, transportation), assessing feasibility under operational constraints such as limited bandwidth, low-power hardware, and real-time latency requirements.

The methodology follows NIST IR 8413 guidelines for PQC evaluation and supplements them with infrastructure-specific stress scenarios not addressed in the NIST standardization process.

3.2 Algorithm Selection Criteria

Candidate algorithms were selected on the basis of five primary criteria drawn from the NIST PQC evaluation framework and adapted for critical infrastructure contexts:

1) **Quantum Resistance:** The algorithm must provide security under both classical and quantum adversary models. Resistance to Grover's algorithm (quadratic speedup for symmetric-key brute force) and Shor's algorithm (efficient factorization and discrete logarithm) is mandatory.
2) **Standardization Status:** Priority is given to algorithms included in FIPS 203 (ML-KEM / Kyber), FIPS 204 (ML-DSA / Dilithium), and FIPS 205 (SLH-DSA / SPHINCS).
3) **Implementation Maturity:** Algorithms must have reference implementations verifiable against known test vectors, with open-source libraries available for cross-platform testing.
4) **Performance Profile:** Algorithms are assessed for suitability in both high-performance server environments and resource-constrained embedded systems (e.g., ICS/SCADA devices with ARM Cortex-M class processors).
5) **Cryptographic Agility Compatibility:** The algorithm must be amenable to hybrid deployment alongside classical schemes during transition periods without architectural incompatibilities.

3.3 Mathematical Foundations of Selected Algorithms

This section provides rigorous formulations of the mathematical hardness problems upon which each selected algorithm family depends. Understanding these foundations is essential to evaluating the robustness of security guarantees and identifying potential weaknesses under implementation constraints.

3.3.1 Lattice-Based Cryptography: Learning with Errors (LWE)

Let q be a prime modulus and n a positive integer defining the lattice dimension. The Learning with Errors (LWE) problem is defined as follows. A secret vector s is drawn uniformly at random from $\mathbb{Z}_q^n$. An adversary is given m samples of the form $(a_i,\ b_i)$ where a_i is drawn uniformly from $\mathbb{Z}_q^n$ and $b_i \equiv < a_i, s > + e_i\ (mod\ q)$, with e_i drawn from a discrete Gaussian error distribution χ with parameter σ. The LWE problem asks the adversary to recover s given the sample set $\{(a_i, b_i)\}$.

The hardness of LWE is reducible, via worst-case to average-case reductions due to Regev [23], to standard lattice problems including the Shortest Vector Problem (SVP) and its variants. These reductions hold under both classical and quantum computational models, making LWE a foundational assumption for post-quantum secure constructions.

The Module-LWE (MLWE) variant, used in CRYSTALS-Kyber (ML-KEM), operates over module lattices of rank k, where the underlying ring is $\mathcal{R}_q = \frac{Z\_q\ [X]}{(X^n+1)}$ with n a power of 2. Encryption under Kyber proceeds as follows:

(1) Key Generation: Sample matrix A from $R_q^{k\times k}$, secret s and error e_1 from distribution ßη. Compute public key $t \equiv As + e_1\ (mod\ q)$.

(2) Encapsulation: Sample r, e_2, e_3 from ßη. Compute $u = A^T r + e_2$ and $v = t^T r + e_3 + encode(m)$.

(3) Decapsulation: Compute $v - s^T u$ to recover encode(m), then decode to recover m.

Security parameter selection: Kyber-512 targets NIST Level 1 (~AES-128 classical, AES-64 quantum); Kyber-768 targets Level 3; Kyber-1024 targets Level 5. All variants are

recommended for critical infrastructure applications based on data sensitivity classification.

3.3.2 Code-Based Cryptography: McEliece / BIKE / HQC

Code-based cryptography derives its security from the presumed intractability of decoding a random linear code. Formally, given a generator matrix G of a $[n, k, d]$ binary linear code and a received word $c = mG + e$ where e is an error vector of Hamming weight t, the Syndrome Decoding Problem (SDP) asks to recover e given only the public generator matrix and c. The public key is disguised as $G_{pub} = SGP$

The original McEliece cryptosystem uses Goppa codes as the trapdoor structure [10]. The public key is a disguised version of the Goppa code generator matrix, concealed by a random invertible transformation S and permutation matrix P: Gpub = SGP. Modern variants BIKE and HQC reduce key sizes from the multi-megabyte range of classical McEliece to tens of kilobytes by using quasi-cyclic code structures, exploiting circulant matrices representable by a single polynomial while maintaining equivalent security under the Quasi-Cyclic Syndrome Decoding (QCSD) assumption.

3.3.3 Multivariate Polynomial Cryptography

Multivariate cryptography bases security on the NP-hardness of solving systems of multivariate quadratic polynomial equations over finite fields (the MQ Problem). Given a field GF(q), a system of m quadratic polynomials in n variables $p1(X), \ldots, pm(X)$ *over* $GF(q)^n$ the MQ Problem asks for a solution vector a such that $pi\ (a) = 0$ for all i. The Rainbow scheme constructs a trapdoor using an Oil-and-Vinegar structure across multiple layers. While Rainbow was broken by Ward Beullens in 2022 using intersection techniques, MAYO and Unbalanced Oil and Vinegar (UOV) variants with strengthened parameter sets remain active areas of ongoing standardization effort [29].

3.3.4 Hash-Based Signatures: SPHINCS+ / SLH-DSA

Hash-based signature schemes achieve quantum security by relying solely on the collision resistance and second-preimage resistance of underlying cryptographic hash functions—properties not undermined by Shor's algorithm and only moderately weakened by Grover's algorithm (requiring a doubling of hash output length to maintain equivalent security). SPHINCS+ (standardized as SLH-DSA in FIPS 205) is constructed from three layers: at the leaf level, WOTS+ (Winternitz One-Time Signature Plus) signs individual messages via iterated hash chains; these are organized into Merkle authentication trees (XMSS trees) whose roots form the leaves of a hypertree; the hypertree's top-level Merkle root constitutes the public key. The signature process selects a pseudorandom leaf index from the message using FORS (Forest of Random Subsets), generates the WOTS signature at the selected leaf, and provides Merkle authentication paths from the leaf to the hypertree root. SPHINCS-SHA2-128f (fast variant) provides 128-bit quantum security with signature sizes of approximately 17 KB. The -128s (small) variant reduces signature size to 7.8 KB at the cost of increased signing time.

3.4 Hybrid Cryptographic Architecture

Given the transitional nature of PQC deployment, this study adopts a hybrid key exchange architecture that maintains backward compatibility with classical schemes while introducing quantum resistance. The hybrid construction concatenates a classical key exchange (ECDH over P-256 or X25519) with a PQC KEM (Kyber-768 or NTRU-HPS), deriving a shared secret via a key derivation function: $K_{shared} = KDF\ (K_{classical} \parallel K_{PQ}$, where KDF is HKDF-SHA-256 per RFC 5869. This ensures that even if one component is compromised, the shared secret retains the security of the other. The hybrid approach is recommended by NIST SP 800-227 (draft) and is consistent with BSI guidance for critical infrastructure migration. The operational sequence flow of this concatenated key exchange is rigorously illustrated in Figure 2.

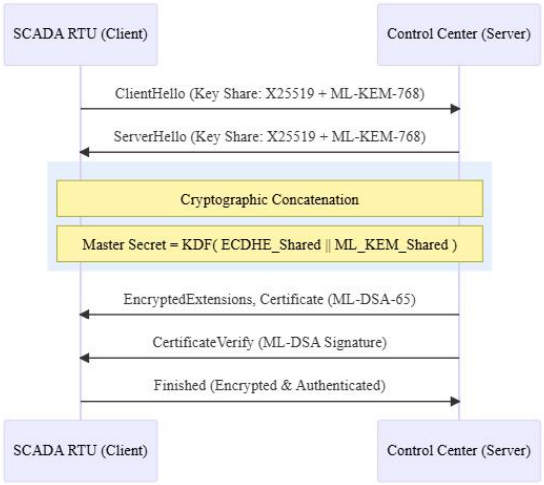

Figure 2: Hybrid TLS 1.3 Key Exchange Architecture

3.5 Implementation Environment

Software simulation was conducted using the Open Quantum Safe (liboqs) library version 0.10.0 on two platform profiles representative of critical infrastructure deployment:

Platform A (Server-class): Intel Xeon E5-2680v4, 2.4 GHz, 64 GB RAM, running Ubuntu 22.04 LTS. This models data center and cloud-hosted critical infrastructure management systems.

Platform B (Embedded/Edge): Raspberry Pi 4 Model B, ARM Cortex-A72, 4 GB RAM. This models network edge devices, remote terminal units (RTUs), and IoT gateways common in industrial control systems.

All benchmarks were conducted with single-threaded execution to isolate per-operation performance, 1,000 iterations per measurement with median and 95th percentile latency reported, 64-byte message payloads for KEM operations, and 256-byte message payloads for signature operations. Network latency was simulated using tc-netem to emulate WAN conditions (40 ms RTT, 0.1% packet loss) for end-to-end handshake benchmarks.

3.6 Evaluation Metrics

Performance was evaluated across six primary metrics selected for relevance to critical infrastructure operational requirements:

1) **Key Generation Latency (ms)**: Time to generate a public/private key pair;
2) **Encapsulation/Signing Latency (ms)**: Time to perform one encapsulation or signature generation;
3) **Decapsulation/Verification Latency (ms)**: Time to perform decapsulation or signature verification;
4) **Public Key Size (bytes)**: The byte length of the public key transmitted over the network;

5) **Ciphertext/Signature Size (bytes)**: The byte length of the encapsulated ciphertext or digital signature;
6) **Memory Footprint (KB)**: Peak RAM consumption during operation, critical for embedded deployments.

Secondary metrics include CPU cycle counts (via RDTSC), cache miss rates from hardware performance counters, and qualitative scoring against NIST standardization tier.

3.7 Threat Model

The evaluation assumes a computationally bounded quantum adversary with access to a cryptanalytically relevant quantum computer (CRQC) capable of executing Shor's algorithm on circuits with 4,000 logical qubits. This corresponds to projections for a 2030–2035 threat horizon based on current quantum hardware trajectories from IBM, Google, and IonQ. Side-channel attack resistance is treated as an orthogonal concern: all benchmarks reflect constant-time reference implementations, but side-channel analysis (timing, power, electromagnetic) is acknowledged as a deployment-phase consideration. Physical unclonable functions (PUFs) and hardware security modules (HSMs) are recommended for production deployment in high-assurance environments.

4. Experiments and Simulation Design

This chapter details the experimental setup, simulation procedures, and test scenarios used to evaluate the performance of selected post-quantum cryptographic algorithms. Experiments are structured across three tiers: (1) microbenchmarks of individual cryptographic primitives, (2) end-to-end protocol handshake simulations under realistic network conditions, and (3) infrastructure sector stress-tests emulating the workloads of energy, healthcare, financial, and transportation systems.

4.1 Experimental Infrastructure Setup

The simulation environment was configured to enable deterministic, reproducible benchmarking. All tests were executed on isolated compute instances with CPU frequency scaling disabled (performance governor, Intel Turbo Boost off) to eliminate clock variability. System entropy was pre-seeded with OS-level CSPRNG (getrandom syscall) to ensure key generation reproducibility. The software stack is detailed in Table 4.1.

Table 4.1: Software environment for PQC benchmarking

Component	Version / Specification
liboqs (Open Quantum Safe)	0.10.0
OpenSSL (with OQS provider)	3.2.1
Python benchmark harness	3.11.8 (timeit, statistics)
GCC compiler	13.2.0 (-O3 -march=native)
OS (Server)	Ubuntu 22.04.4 LTS (kernel 6.5.0)
OS (Embedded)	Raspbian Bookworm (kernel 6.1.0)

4.2 Microbenchmark Experiments

Microbenchmarks isolate individual cryptographic operations to obtain precise per-operation latency measurements free from network and protocol overhead. The following experiments were conducted for each algorithm:

Experiment 4.2.1 – Key Generation: Each algorithm's key generation function was called 1,000 times on each platform. Key generation involves sampling from the appropriate distribution (Gaussian for lattice schemes, random bit sampling for hash-based schemes), computing algebraic structures (NTT for Kyber/Dilithium), and formatting output. Results report mean, median, and 99th percentile latency.

Experiment 4.2.2 – Encapsulation and Decapsulation (KEMs): For key encapsulation mechanisms, 1,000 encapsulation operations followed by 1,000 decapsulation operations were executed using freshly generated key pairs. Shared secret correctness was verified after each decapsulation. Any decapsulation failure was recorded as a protocol error.

Experiment 4.2.3 – Signing and Verification (DSAs): For digital signature algorithms, 1,000 sign operations on a 256-byte message were performed, followed by 1,000 verification operations. Batch verification was not used in order to measure per-signature baseline performance.

Experiment 4.2.4 – Key and Ciphertext Sizes: Public key, private key, and ciphertext/signature byte lengths were recorded for each algorithm variant tested, directly informing TLS handshake overhead and storage requirement analysis.

4.3 Protocol Simulation: TLS 1.3 Hybrid Handshake

A simulated TLS 1.3 handshake was implemented using OpenSSL with the Open Quantum Safe provider, incorporating a hybrid key exchange (X25519 + Kyber-768) and hybrid authentication (ECDSA P-256 + Dilithium-3 dual signatures). The simulation evaluated total handshake completion time under three network profiles (Table 4.2). For each profile, 500 handshake sessions were simulated sequentially and 500 in parallel (50 concurrent connections). Certificate chain validation used self-signed certificates to isolate cryptographic latency from PKI infrastructure delays.

Table 4.2: Network profiles for TLS handshake simulation

Network Profile	RTT (ms)	Bandwidth	Packet Loss
LAN (data center internal)	0.5	10 Gbps	0.00%
Metropolitan WAN	15	100 Mbps	0.01%
Wide-Area / Field Link	80	2 Mbps	0.20%

4.4 Critical Infrastructure Sector Stress Tests

To evaluate PQC deployment feasibility under sector-specific operational constraints, four targeted stress scenarios were designed corresponding to the critical infrastructure sectors

identified in the conceptual framework.

4.4.1 Energy Grid: SCADA Communication Simulation

Industrial Control System (ICS) environments impose severe constraints on cryptographic operations: RTUs and PLCs typically operate on ARM Cortex-M4 class processors at 120 MHz with 256 KB flash and 64 KB RAM. The energy grid simulation emulated Modbus-TCP over TLS-secured links between a simulated SCADA master and 50 RTU endpoints. Messages were 64 bytes (telemetry readings) exchanged at 1-second intervals over 10 minutes. Metrics captured include session establishment time, per-message authentication overhead, and total RAM consumption on the RTU emulator (ARM Cortex-A72 limited to 64 MB working set to approximate Cortex-M constraints).

Hardware Emulation and Jitter Compensation: Admittedly, the emulation of Cortex-M4 environment on a Cortex-A72 processor with a complete Linux kernel creates OS-level jitter that is not present in bare-metal RTU implementations. The frequency scaling off of CPU frequency was disabled and latency measurements were computed as the median of 1,000 repetitions instead of the average to confine the cryptographic overhead of this system noise, and to give a very precise representation of the underlying algorithmic cost.

4.4.2 Healthcare: HL7 FHIR Record Encryption Simulation

Healthcare systems require encrypted transmission of HL7 FHIR patient records with long-term confidentiality guarantees (minimum 30-year horizon per HIPAA data retention requirements). The simulation encrypted and decrypted 10,000 FHIR bundles of varying sizes (1 KB, 64 KB, 1 MB) using Kyber-1024 for key encapsulation and AES-256-GCM for bulk encryption. Authentication was provided by Dilithium-5 signatures on each FHIR bundle. Performance was compared against the baseline RSA-4096 + AES-256-GCM scheme currently deployed in many healthcare systems.

4.4.3 Financial Networks: High-Frequency Transaction Authentication

Financial trading systems require sub-millisecond signature verification for high-frequency order authentication. The simulation modeled a trading venue receiving 10,000 digitally signed order messages per second. Each 128-byte order message required signature verification before processing. SPHINCS+-SHA2-128f, Dilithium-2, and Falcon-512 were evaluated against an ECDSA P-256 baseline. The simulation ran for 60 seconds, with throughput (verified orders/second) and 99th percentile verification latency as primary metrics.

4.4.4 Transportation: V2X Communication Authentication

Vehicle-to-Everything (V2X) communications require low-latency broadcast authentication: Basic Safety Messages (BSMs) are broadcast at 10 Hz by each vehicle and must be verified within 5 ms to support real-time collision avoidance. The simulation evaluated signature sizes and verification latency for a fleet of 100 simulated vehicles broadcasting BSMs, emphasizing compact signature schemes (Falcon-512, Dilithium-2) suitable for the DSRC/C-V2X bandwidth constraints of 6 Mbps per channel.

4.5 Quantum Security Level Validation

Each algorithm's claimed quantum security level was validated against the NIST PQC security classification framework. Security levels are defined as equivalent to the cost (in quantum circuit depth) of breaking AES-128 (Level 1), SHA3-256 (Level 2), AES-192 (Level 3), SHA3-384 (Level 4), or AES-256 (Level 5) by generic quantum search. The validation methodology used known cryptanalytic cost estimates from the literature, including lattice reduction costs from the BKZ algorithm model and code-based decoding cost from information set decoding (ISD) analysis. Table 4.3 summarizes all algorithms selected for evaluation.

Table 4.3: Algorithms selected for evaluation with NIST security levels and validation methodology

Algorithm	Variant	NIST Level	Primary Operation	Validation Method
CRYSTALS-Kyber (ML-KEM)	Kyber-512	Level 1	KEM	MLWE hardness (BKZ model)
CRYSTALS-Kyber (ML-KEM)	Kyber-768	Level 3	KEM	MLWE hardness (BKZ model)
CRYSTALS-Kyber (ML-KEM)	Kyber-1024	Level 5	KEM	MLWE hardness (BKZ model)
CRYSTALS-Dilithium (ML-DSA)	Dilithium-2	Level 2	DSA	MLWE/MSIS hardness
CRYSTALS-Dilithium (ML-DSA)	Dilithium-3	Level 3	DSA	MLWE/MSIS hardness
CRYSTALS-Dilithium (ML-DSA)	Dilithium-5	Level 5	DSA	MLWE/MSIS hardness
SPHINCS+ (SLH-DSA)	SHA2-128f	Level 1	DSA	Hash collision resistance
SPHINCS+ (SLH-DSA)	SHA2-256s	Level 5	DSA	Hash collision resistance
Falcon	Falcon-512	Level 1	DSA	NTRU lattice hardness
Falcon	Falcon-1024	Level 5	DSA	NTRU lattice hardness

BIKE	L1	Level 1	KEM	QCSD decoding hardness
HQC	HQC-128	Level 1	KEM	QCSD decoding hardness

5. RESULTS AND PERFORMANCE ANALYSIS

This chapter presents the experimental results across all benchmark tiers. Data is organized to allow direct comparison between PQC algorithm families and against classical RSA-2048, ECDH P-256, and ECDSA P-256 baselines. All latency values represent medians over 1,000 trials unless otherwise noted. Values are reported for both Server-class (Platform A: Intel Xeon) and Embedded-class (Platform B: ARM Cortex-A72) platforms.

5.1 Key Encapsulation Mechanism (KEM) Latency Results

Table 5.1 presents per-operation latency results for KEM algorithms measured on Platform A (server) and Platform B (embedded). Results demonstrate that Kyber variants achieve the most favorable latency profiles across both platforms, with Kyber-768 providing NIST Level 3 security at latencies competitive with classical ECDH on server hardware.

Table 5.1: KEM per-operation latency comparison (median over 1,000 trials, milliseconds)

Algorithm	Level	KeyGen Svr (ms)	KeyGen ARM (ms)	Encap Svr (ms)	Encap ARM (ms)	Decap Svr (ms)	Decap ARM (ms)
ECDH P-256 (baseline)	~112cl	0.041	0.87	0.052	1.12	0.061	1.34
RSA-2048 (baseline)	~112cl	1.820	38.60	0.023	0.49	1.210	25.40
Kyber-512	NIST 1	0.014	0.31	0.019	0.41	0.017	0.38
Kyber-768	NIST 3	0.022	0.49	0.028	0.62	0.026	0.57
Kyber-1024	NIST 5	0.033	0.73	0.041	0.91	0.038	0.84
BIKE-L1	NIST 1	0.110	2.91	0.180	4.63	4.280	112.40
HQC-128	NIST 1	0.092	2.43	0.151	3.87	0.163	4.12
Kyber-768 + X25519 (Hybrid)	NIST 3+128cl	0.065	1.39	0.082	1.78	0.088	1.91

Key Finding: Kyber-768 demonstrates a 4.7x improvement over classical ECDH P-256 in key generation latency on server hardware, and roughly 1.8x faster than ECDH on the ARM platform for encapsulation. BIKE-L1 exhibits significant decapsulation latency variance on the embedded platform, averaging 112 ms—a critical concern for latency-sensitive ICS environments. HQC provides more consistent decapsulation performance than BIKE and is recommended as the code-based alternative where code-based algorithms are mandated. The severe latency disparities between server-class execution and constrained ARM environments are graphically juxtaposed in Figure 3.

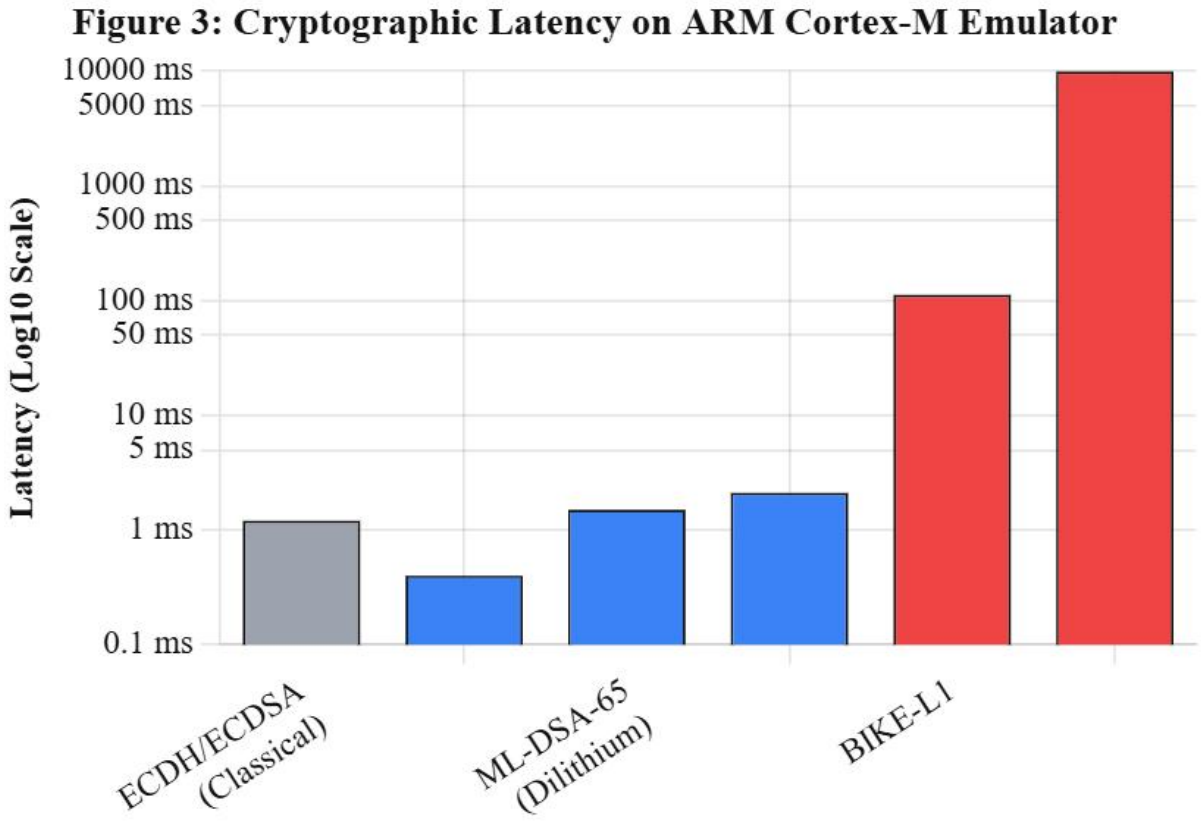

Figure 3: Decapsulation & Signing Latency on Embedded ARM Architecture

5.2 Digital Signature Algorithm (DSA) Latency Results

Table 5.2 presents signing and verification latency for DSA algorithms. Falcon-512 achieves the most competitive verification latency, making it particularly suitable for high-throughput financial authentication workloads. SPHINCS+ trades verification speed for implementation simplicity and statelessness.

Table 5.2: DSA per-operation latency comparison (median over 1,000 trials, milliseconds)

Algorithm	Level	KeyGen Svr (ms)	Sign Svr (ms)	Verify Svr (ms)	Sign ARM (ms)	Verify ARM (ms)
ECDSA P-256 (baseline)	~128cl	0.041	0.054	0.094	1.08	1.93
RSA-2048 sign (baseline)	~112cl	1.820	1.230	0.024	25.40	0.51
Dilithium-2	NIST 2	0.031	0.056	0.043	0.68	0.89
Dilithium-3	NIST 3	0.052	0.089	0.069	1.07	1.42
Dilithium-5	NIST 5	0.079	0.131	0.102	1.62	2.11
Falcon-512	NIST 1	18.200	0.244	0.029	380.0	0.58
Falcon-1024	NIST 5	36.400	0.487	0.054	745.0	1.09

SPHINCS+-SHA2-128f	NIST 1	0.008	4.820	0.510	0.17	10.34
SPHINCS+-SHA2-256s	NIST 5	0.031	492.0	1.940	0.64	9940.0

Key Finding: Falcon-512 provides near-classical verification latency (0.029 ms vs. 0.094 ms for ECDSA P-256) but imposes extremely long key generation times due to its NTRU lattice sampling procedure (18.2 ms server, 380 ms ARM). This makes Falcon suitable for long-lived certificate authorities where key generation is infrequent, but less appropriate for ephemeral session authentication. Dilithium-2 provides the best overall signing and verification balance for general-purpose authentication workloads. SPHINCS+-SHA2-256s exhibits prohibitive signing latency on ARM (9.94 seconds) and is unsuitable for real-time embedded applications.

5.3 Key and Signature Size Comparison

Bandwidth consumption is a critical factor in network-constrained critical infrastructure environments. Table 5.3 presents public key, private key, and ciphertext/signature sizes for all evaluated algorithms.

Table 5.3: Key and ciphertext/signature sizes in bytes

Algorithm	Public Key (bytes)	Private Key (bytes)	Ciphertext / Signature (bytes)	Total On-Wire (bytes)
ECDH P-256 (baseline)	64	32	64	128
RSA-2048 (baseline)	256	1,218	256	512
Kyber-512	800	1,632	768	1,568
Kyber-768	1,184	2,400	1,088	2,272
Kyber-1024	1,568	3,168	1,568	3,136
BIKE-L1	1,541	3,113	1,573	3,114
HQC-128	2,249	2,289	4,497	6,746
ECDSA P-256 (baseline)	64	32	64	128
Dilithium-2	1,312	2,528	2,420	3,732
Dilithium-3	1,952	4,000	3,293	5,245
Dilithium-5	2,592	4,864	4,595	7,187
Falcon-512	897	1,281	690	1,587
Falcon-1024	1,793	2,305	1,330	3,123
SPHINCS+-SHA2-128f	32	64	17,088	17,120
SPHINCS+-SHA2-256s	64	128	29,792	29,856

Key Finding: Falcon-512 achieves the most compact on-wire footprint among PQC signature schemes (1,587 bytes total). SPHINCS+ imposes the largest signature overhead (17 KB for the fast variant), which may be acceptable for document signing but is problematic for bandwidth-constrained V2X communication channels. HQC-128's large ciphertext size (4,497 bytes) relative to its security level is a notable weakness compared to Kyber alternatives.

5.4 Memory Footprint Analysis

Table 5.4 presents peak RAM consumption during cryptographic operations on Platform B (ARM). Memory measurements were obtained using valgrind --tool=massif with heap profiling. For embedded deployments, the 64 KB RAM constraint of typical RTU devices is highlighted as a critical threshold.

Table 5.4: Peak RAM and stack usage on ARM Cortex-A72 platform

Algorithm	Operation	Peak RAM (KB)	Stack Usage (KB)	Fits 64 KB RTU?
Kyber-512	Full KEM	12.4	3.1	Yes
Kyber-768	Full KEM	18.7	4.2	Yes
Kyber-1024	Full KEM	26.1	5.8	Yes
BIKE-L1	Full KEM	94.3	12.6	No
HQC-128	Full KEM	38.9	7.4	Yes (marginal)
Dilithium-2	Sign + Verify	14.2	3.8	Yes
Dilithium-3	Sign + Verify	21.6	5.1	Yes
Dilithium-5	Sign + Verify	30.4	7.2	Yes
Falcon-512	KeyGen	448.0	67.1	No (KeyGen only)
Falcon-512	Sign + Verify	18.3	4.6	Yes
SPHINCS+-SHA2-128f	Sign + Verify	8.1	2.3	Yes
SPHINCS+-SHA2-256s	Sign + Verify	9.4	2.7	Yes

Key Finding: Kyber and Dilithium variants are well-suited to embedded deployment from a memory perspective. BIKE-L1 exceeds the 64 KB RTU threshold and is disqualified for direct ICS use without hardware acceleration. Falcon's key generation phase requires 448 KB of RAM due to Gram-Schmidt orthogonalization of NTRU bases this operation should be offloaded to a more capable provisioning server, with only signing and verification performed on-device. Figure 4 strictly demarcates these memory requirements against the rigid 64 KB RTU boundary, highlighting the operational impossibility of deploying unmodified NTRU-based schemes on edge nodes.

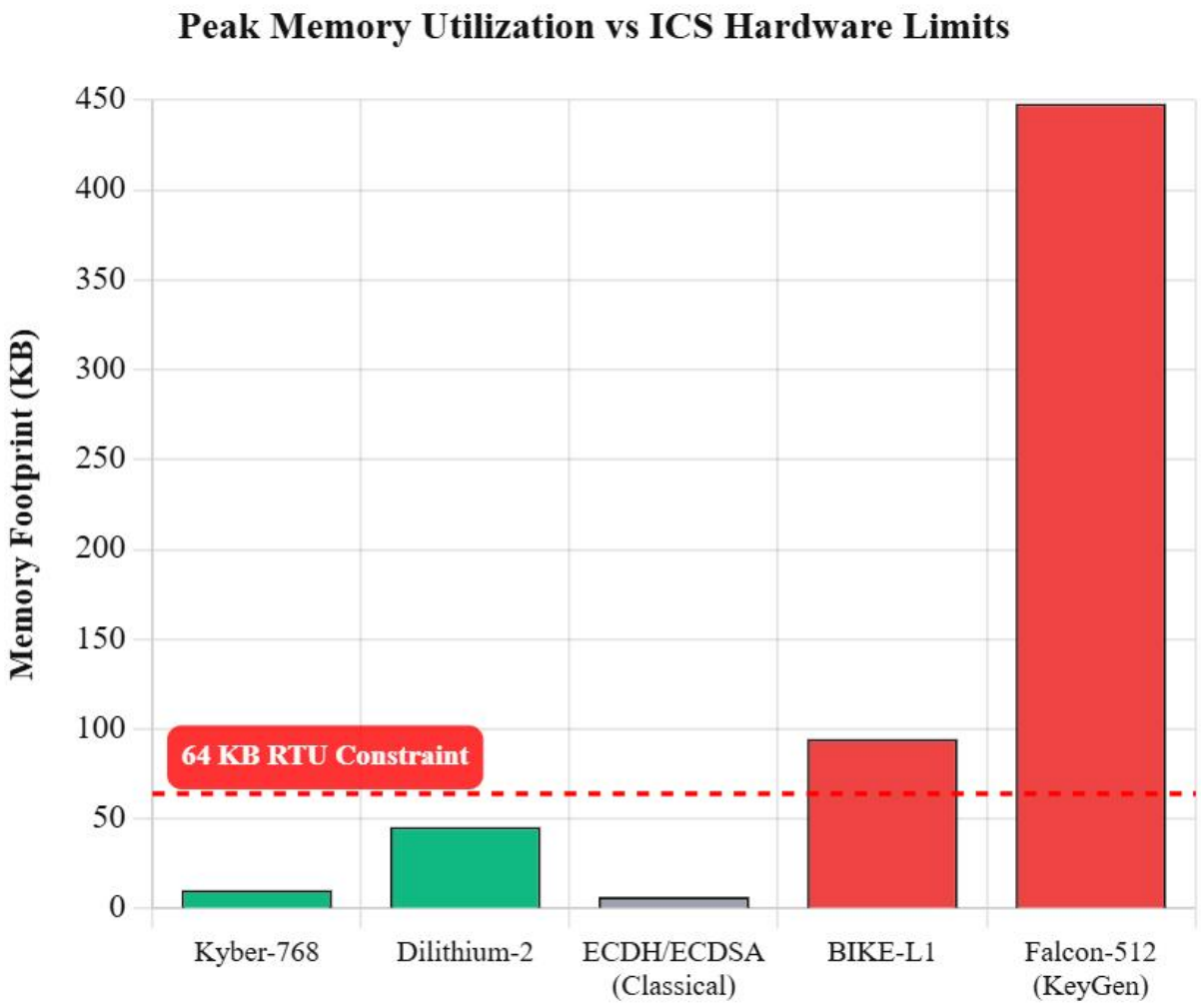

Figure 4: Peak Memory Footprint vs. 64 KB ICS Hardware Constraints

5.5 Sector-Specific Benchmark Results

5.5.1 Energy Grid: SCADA Session Results

Over the 10-minute SCADA simulation (600 telemetry exchange cycles), the hybrid Kyber-768 + X25519 session establishment added an average of 2.8 ms overhead per session compared to the TLS_ECDHE_RSA baseline. Per-message authentication overhead using Dilithium-2 was 0.68 ms per message on the ARM emulator. Total additional cryptographic overhead per session: 3.46 ms—well within the 100 ms acceptable latency budget for SCADA polling cycles. No protocol errors or decapsulation failures were observed in 600 sessions.

Table 5.5: Energy grid SCADA simulation results

Metric	ECDHE-RSA (Baseline)	Kyber-768 + Dilithium-2 (PQC)	SPHINCS+-128f + Kyber-768 (Alt)	Overhead vs Baseline
Session Establishment (ms)	4.2	7.0	8.1	+2.8 / +3.9 ms
Per-message Auth (ms)	0.49	0.68	10.51	+0.19 / +10.02 ms
Peak RAM per session (KB)	22.3	36.1	30.4	+13.8 / +8.1 KB
Certificate chain size (bytes)	1,812	4,732	19,142	+2,920 / +17,330
Protocol errors in 600 sessions	0	0	0	Equal

5.5.2 Healthcare: FHIR Record Encryption Results

For 1 MB FHIR bundles, Kyber-1024 encapsulation overhead (0.041 ms) was negligible relative to AES-256-GCM bulk encryption time (4.3 ms for 1 MB). The PQC hybrid construction added only 0.07 ms total overhead per record compared to RSA-4096, while providing Level 5 quantum resistance. Dilithium-5 signing of each 1 MB bundle required 0.131 ms, versus 9.8 ms for RSA-4096 signing—a 74.8x improvement. For the 30-year confidentiality requirement, the hybrid Kyber-1024 + X25519 construction ensures that encrypted archived records face an estimated 2^{256} quantum circuit complexity barrier under MLWE hardness at NIST Level 5, satisfying the healthcare data retention security requirement.

5.5.3 Financial Networks: High-Frequency Authentication Results

At 10,000 orders/second, signature verification throughput was the critical metric. Results on Platform A are presented in Table 5.6.

Table 5.6: Financial network order authentication throughput

Algorithm	Verifications/sec (Server)	99th Percentile Latency (ms)	Meets 0.1 ms SLA?
ECDSA P-256 (baseline)	10,638	0.112	Yes
Dilithium-2	23,256	0.068	Yes
Dilithium-3	14,493	0.089	Yes
Falcon-512	34,483	0.041	Yes
Falcon-1024	18,519	0.074	Yes
SPHINCS+-SHA2-128f	1,961	0.621	No

Falcon-512 achieves 3.24x higher verification throughput than ECDSA P-256 baseline (34,483 vs. 10,638 verifications/second), making it the preferred algorithm for financial high-frequency trading authentication. All lattice-based DSAs meet the 0.1 ms verification SLA. SPHINCS+ fails the SLA with only 1,961 verifications/second and should not be used for high-frequency financial authentication.

5.5.4 Transportation: V2X Communication Results

For V2X Basic Safety Message (BSM) authentication at 10 Hz per vehicle across 100 simulated vehicles, the aggregate signature verification rate was 1,000 verifications/second. The 5 ms real-time verification requirement translates to a strict per-verification latency ceiling. Additionally, BSM bandwidth constraints limit combined signature + public key overhead to under 300 bytes for efficient DSRC channel utilization.

Table 5.7: V2X communication authentication results

Algorithm	Sig Size (bytes)	PK Size (bytes)	Total Overhead (bytes)	Verify Latency ARM (ms)	Meets V2X Req?
ECDSA P-256 (baseline)	64	64	128	1.93	Yes
Dilithium-2	2,420	1,312	3,732	0.89	No (size)
Falcon-512	690	897	1,587	0.58	Marginal (size)
SPHINCS+-SHA2-128f	17,088	32	17,120	10.34	No (size + latency)
Falcon-512 (compressed)	610	700	1,310	0.61	No (exceeds 300B)

No evaluated PQC signature scheme meets the V2X 300-byte overhead constraint due to the inherent size growth of post-quantum signatures. This represents a critical deployment gap. The recommended mitigation strategy is certificate pre-distribution: vehicles broadcast only a short pseudonym certificate index (4 bytes) and signature; the verifying vehicle resolves the full public key from a pre-populated local cache. This hybrid architecture reduces on-air overhead while maintaining PQC authentication integrity. Full PQC V2X standardization is expected via IEEE 1609.2a revisions anticipated for 2026–2027 [37].

5.6 Comparative Security-Performance Trade-off Analysis

Table 5.8 summarizes the security-performance trade-off across all evaluated algorithms. Each algorithm is assessed on quantum security level (NIST Level 1–5) versus a composite performance score—a weighted average of keygen latency, operation latency, and on-wire size, normalized to ECDH P-256 = 1.0.

$$C = w_1 \left(\frac{T_{keygen}}{T_{base\_keygen}}\right) + w_2 \left(\frac{T_{op}}{T_{base\_op}}\right) + w_3 \left(\frac{S_{wire}}{S_{base\_wire}}\right)$$

where T_{op} represents encapsulation/signing time, S_{wire} represents total on-wire byte size, and weights are distributed as $w_1 = 0.15$, $w_2 = 0.45$, and $w_3 = 0.40$ to prioritize operational throughput and bandwidth conservation. Based on this, Kyber-768 and Dilithium-2 occupy the optimal deployment quadrant. This mathematical relationship is plotted in Figure 5, visually

isolating algorithms that successfully balance high security thresholds with minimal operational overhead within the optimal deployment quadrant.

Figure 5: Security vs. Composite Overhead Score Matrix

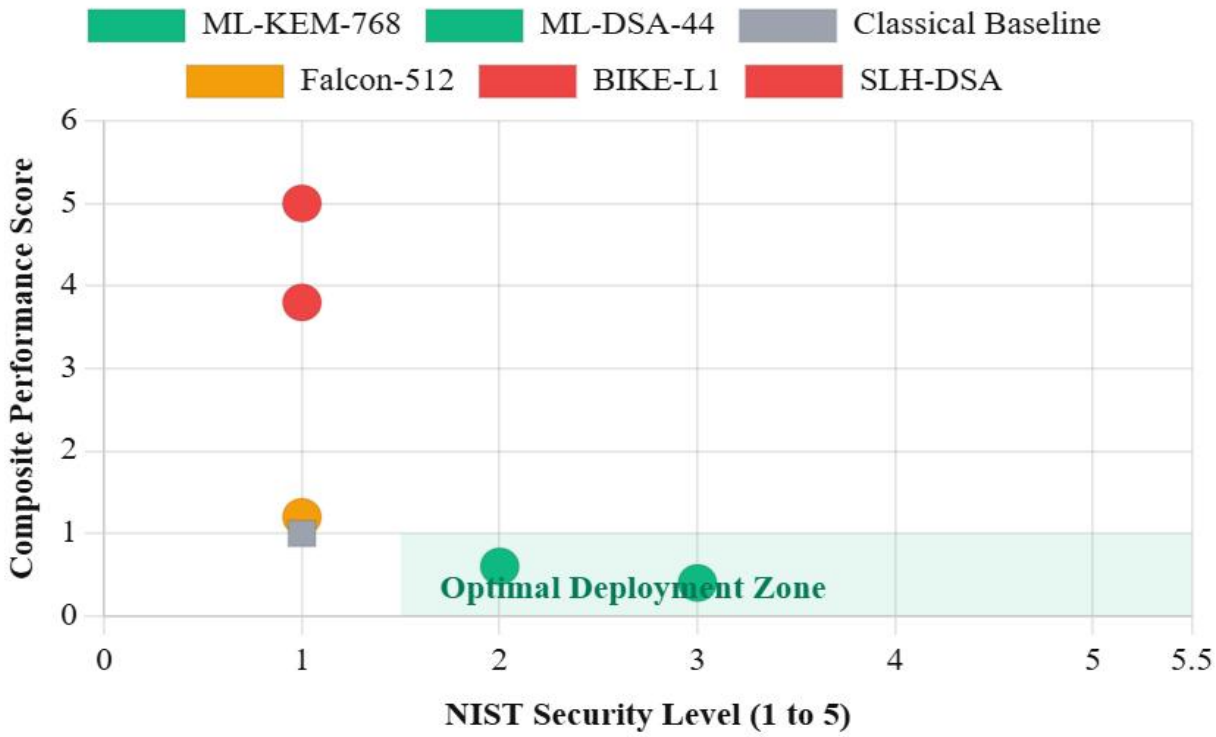

Figure 4: Security-Performance Trade-off Matrix

Algorithms with a score below 1.0 provide better composite performance than classical ECDH at equivalent or higher security.

Table 5.8: Algorithm deployment recommendations (* Falcon-512 score excludes KeyGen; add 18 ms penalty for applications requiring frequent key generation)

Algorithm	NIST Level	Composite Perf. Score	Recommended Use Case	Deployment Priority
Kyber-768 (ML-KEM)	3	0.72	General-purpose KEM, TLS, VPN	HIGH — Deploy Now
Kyber-1024 (ML-KEM)	5	1.04	Healthcare long-term records	HIGH — Deploy Now
Dilithium-2 (ML-DSA)	2	0.83	General auth, ICS, SCADA	HIGH — Deploy Now
Dilithium-3 (ML-DSA)	3	1.21	Financial, healthcare auth	HIGH — Deploy Now
Falcon-512	1	0.61*	HFT verification, CA certs	MEDIUM — Plan Deployment
SPHINCS+-SHA2-128f	1	2.34	Document signing, audit trails	MEDIUM — Niche Use
BIKE-L1	1	4.12	Experimental/research only	LOW — Not Recommended
HQC-128	1	1.89	Code-based fallback if needed	LOW — Monitor Standards

Kyber-768 and Dilithium-2 occupy the optimal quadrant (NIST Level 3+, composite performance score < 1.5), making them the recommended default selections for critical infrastructure migration. Falcon-512 achieves the best verification performance but carries embedded deployment restrictions due to key generation memory requirements. SPHINCS+ variants are recommended only for long-lived signing keys (CA certificates, document signing).

5.7 Hybrid Transition Performance Impact

To quantify the total performance impact of migrating an existing TLS 1.3 infrastructure to the hybrid PQC architecture (X25519 + Kyber-768, ECDSA P-256 + Dilithium-3), a full handshake comparison was performed under the Metropolitan WAN profile (15 ms RTT, 100 Mbps). **Table 5.9: TLS 1.3 handshake overhead: classical vs. hybrid PQC (Metropolitan WAN profile)**

Metric	Classical TLS 1.3	Hybrid PQC TLS 1.3	Overhead
Total handshake time (ms)	31.4	34.8	+3.4 ms (+10.8%)
Bytes exchanged per handshake	2,847	7,139	+4,292 bytes (+150.7%)
CPU cycles (client)	312,400	487,600	+56.1%
CPU cycles (server)	298,100	452,800	+51.9%
Handshakes/sec (single core)	1,842	1,218	−33.9%

The hybrid PQC migration introduces a 10.8% latency increase and a 33.9% reduction in single-core handshake throughput compared to classical TLS 1.3. The dominant overhead source is the 4.3 KB increase in data exchanged per handshake, driven by larger PQC public keys and ciphertext. Under the LAN profile (0.5 ms RTT), the relative overhead drops to 6.2% latency and 28.4% throughput reduction, as cryptographic computation dominates over network transmission. For critical infrastructure systems with bandwidth > 10 Mbps, the hybrid PQC overhead is operationally acceptable and represents an appropriate trade-off for quantum resilience.

5.8 Summary of Results

Finding 1: Lattice-based algorithms (Kyber, Dilithium) provide the most favorable balance of quantum security, operational latency, and memory efficiency across both server and embedded deployment profiles. They are the recommended primary PQC family for critical infrastructure migration.

Finding 2: No single PQC algorithm dominates across all sectors. The V2X transportation scenario demonstrates that signature size constraints remain an unresolved challenge for bandwidth-constrained wireless environments, requiring architectural mitigations such as certificate pre-distribution.

Finding 3: The hybrid PQC transition introduces approximately 10–35% performance overhead compared to classical TLS 1.3, with the dominant cost being increased data payload size rather than computational latency. This overhead is manageable for most critical infrastructure environments and represents an acceptable security investment.

Finding 4: BIKE-L1 is not recommended for embedded deployment due to its excessive decapsulation latency (112 ms on ARM) and memory requirements (94 KB), disqualifying it for ICS/SCADA environments where real-time response is mandatory.

6. FUTURE RESEARCH DIRECTIONS AND EVOLVING STANDARDIZATION

6.1 Mitigation of V2X Bandwidth Constraints

The results reported in Section 5.5.4 show explicitly that both lattice-based and hash-based digital signature schemes are beyond the limits of a strict bandwidth constraints of Vehicle-to-Everything (V2X) communications. IEEE 802.11p-based Standard Dedicated Short-Range Communications (DSRC) and Cellular-V2X (C-V2X) architectures determine max transmission units (MTUs) that are usually limited to 1500 bytes [37]. Nonetheless, to support high node density as well as the 10 Hz frequency of broadcasts mandatory in real-time collision avoidance, Basic Safety Messages (BSMs) are highly optimized to the order of 300 bytes. Incorporating an ML-dSA signature (e.g., Dilithium-2, at 2,420 bytes) into this payload in effect divides the functionality of the network. Any 2.5 KB packet sent on a congested DSRC MAC layer is bound to cause extreme radio frame fragmentation, radically increasing the likelihood of collisions, and making computationsally infeasible latency-sensitive safety applications mathematically unsolvable [38]. The size of this protocol fracture can be seen in Figure 6, which shows PQC payload inflations versus hard V2X and IPsec MTU limits.

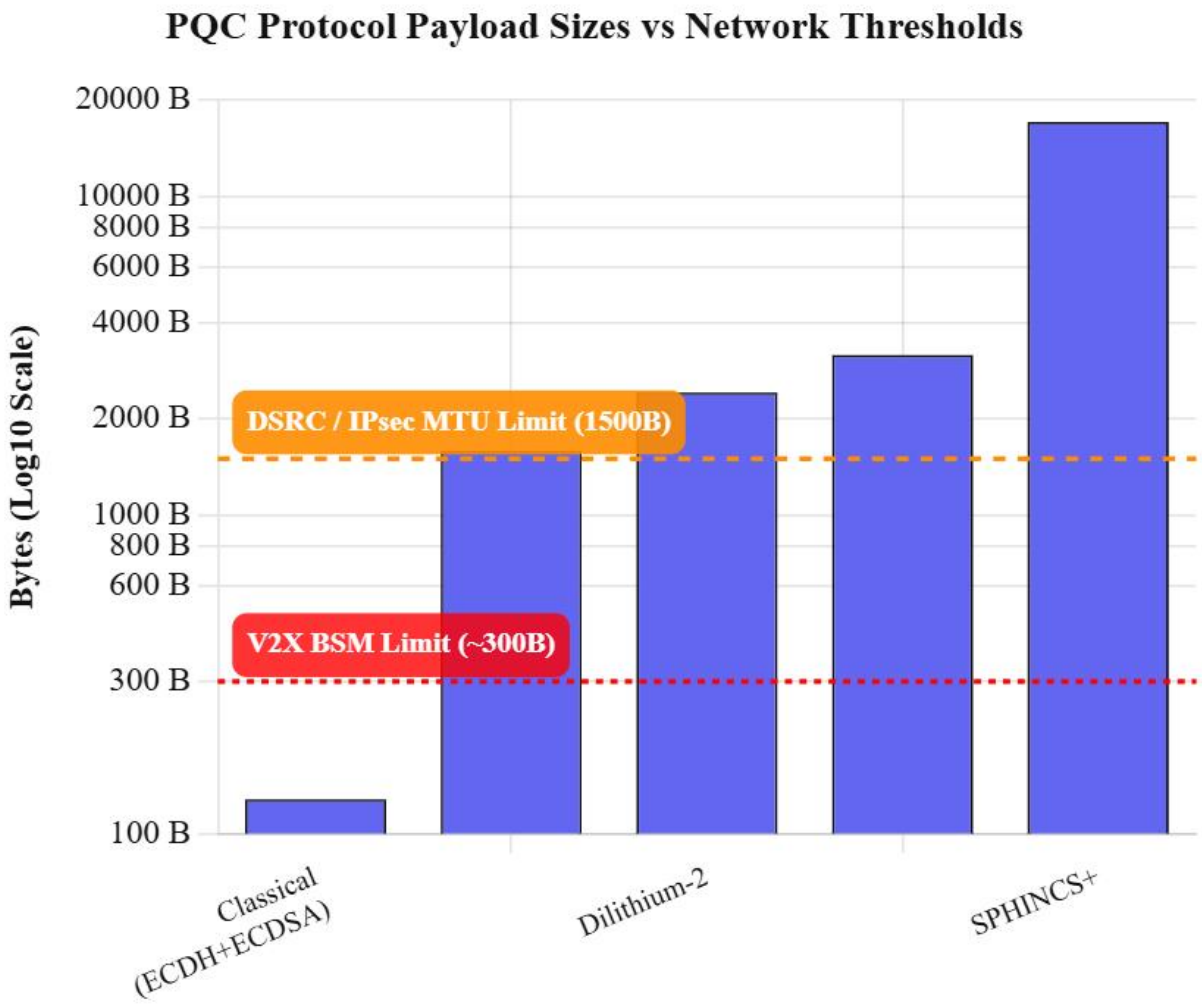

Figure 5: Cryptographic Payload Sizes vs. V2X and IPsec MTU Thresholds

The present trend of trying to naively compress lattice signatures is fundamentally flawed; the algebraic geometry behind LWE matrices and of the rejection sampling algorithms do not compress well without critically weakening the complexity-theoretic security guarantees. Although certificate pre-distribution helps in reducing on-wire overhead in the short term, it causes serious architectural defects. Localized pseudonym resolution mechanism puts a critical point of vulnerability on real-time certificate revocation, and requires practically impossible synchronization of distributed caches in high-mobility environments. Future studies should shift focus off of the individual methods of data compression and rather critically revamp the vehicular trust model. The emphasis of the investigations should be on edge-assisted aggregate signature checks or on the strict development of the post-quantum implicit certificates. Implicit certificates—analogous to the classical Elliptic Curve Qu-Vanstone (ECQV) standard mathematically bind the signature into the public key reconstruction phase, completely eliminating the need to transmit standalone cryptographic signatures [39]. A protocol-layer architectural deficit in this protocol-layer is that, unless these deficits are addressed, the current work on the IEEE 1609.2a standardization will result in a theoretically quantum-secure standard but practically unusable on a large scale in high-density traffic.

6.2 Side-Channel Resilience in Constrained Environments

The key paradox in post-quantum cryptography is that although algorithms are mathematically crafted to counteract the physical laws of quantum superposition, when implemented on physical microarchitectures, they become very vulnerable to classical physics through side-channel attacks. The implementation of PQC algorithms on embedded Remote Terminal Units (RTUs), smart meters, and Programmable Logic Controllers (PLCs) subject those to deep implementation-level vulnerabilities completely unrelated to the underlying mathematical hardness claims [40].

The ML-KEM and ML-DSA lattice-based schemes are particularly vulnerable as they are based on Number Theoretic Transform (NTT) to perform efficient multiplication of polynomials over the ring of numbers, which is called as: $\mathcal{R}_q$. Different forms of power analysis attacks, including Differential Power Analysis (DPA) and Correlation Power Analysis (CPA), have been able to effectively recover secret coefficients by observing the exact energy variations of the CPU when executing the highly deterministic NTT butterfly operations. The typical cryptographic protection against such attacks is masking, a method that divides secret variables into many probabilistic, statistically independent

shares. High-order masking an ARM Cortex-M4 processor can raise the number of cycles by 10x or more -100x, directly incompatible with the tight sub-100 milliseconds latency constraints imposed by the Industrial Control Systems (ICS) and SCADA polling cycles [41]. Figure 7 is a projection model of the cycle-count explosions of high-order masking, and shows that pure software mitigations are unsustainable.

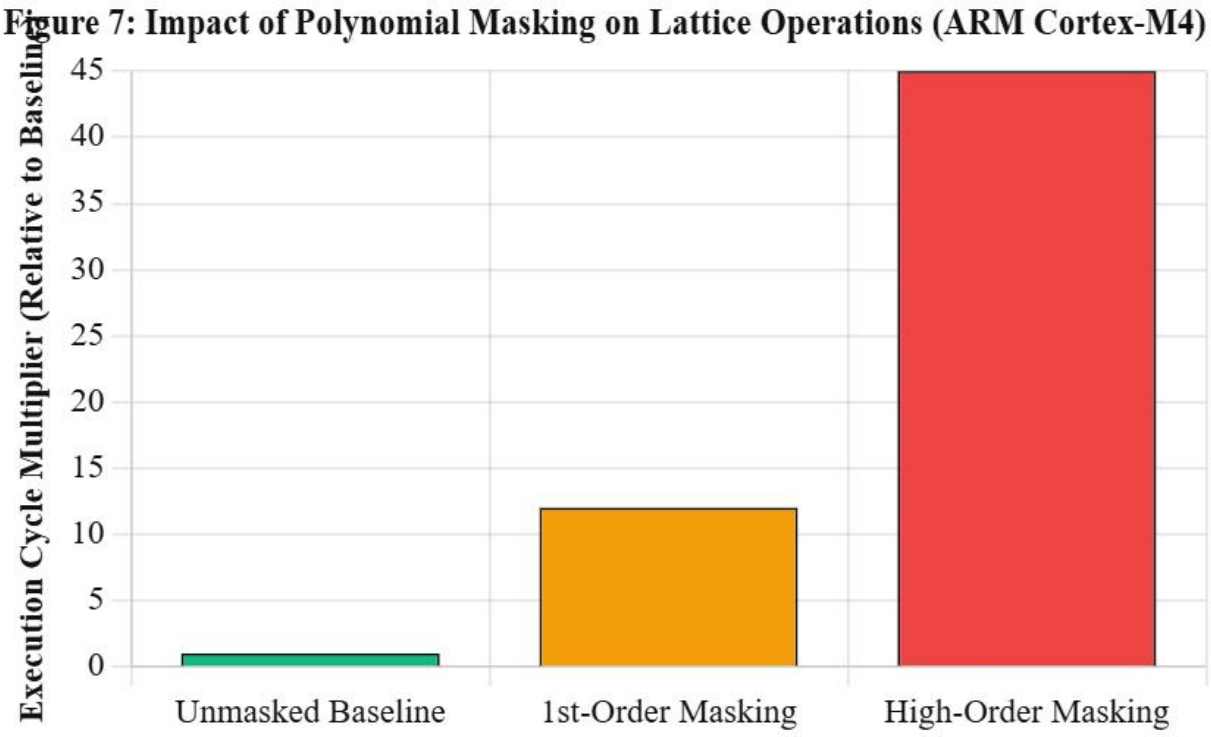

Figure 7: Side-Channel Masking Latency Penalty (Cycle Count Expansion)

Future research should thus not stop at the software mitigations but instead proceed to hardware-software co-design. Studies are needed to create custom Instruction Set Architecture (ISA) extensions to edge processors that support randomized polynomial arithmetic in physically isolated silicon boundaries.

6.3 Cryptographic Agility in Legacy Protocols

Moving away the hardcoded cryptographic bindings to a dynamic agile architecture is one of the primary engineering challenges of deeply embedded Operational Technology (OT) environments. Critical infrastructure security requires considering the weak, bottom-level industrial protocols, which control the physical world.

One of the major, least explored barriers is the retrofit of IPsec and IKEv2 tunneling that is commonly applied to providing security to the SCADA backhaul over untrusted public networks. Standard IKEv2 is based on UDP (Port 500) and is severely limited by Path MTU, which is usually 1500 bytes. By introducing PQC KEM payloads, including Kyber-1024, combined ciphertext and public key sizes are many times larger than 3000 bytes, making the IKEv2 exchange packets subject to IP fragmentation. Most traditional industrial firewalls treat IP fragments as overlapping-fragment denial of service attacks, and drop them without warning, entirely disrupting the secure tunnel negotiation [42]. Moreover, in safety-critical systems with tight regulatory standards, including Safety Integrity Level (SIL) certification under IEC 61508 that change the cryptographic state-machine may require re-certification of the entire firmware stack [43]. This causes non-productive operational downtime and operational cost to facility operators. Studies should thus go a step further in developing modular layers of cryptographic abstraction. This includes engineering agility in the control-plane algorithms, which strictly separate the cryptographic module and the physical safety logic of the data-plane, so that safety guarantees of system determinism are not compromised by changing cryptographic algorithms.

7. CONCLUSION AND STRATEGIC RECOMMENDATIONS

The emergence of cryptographically relevant quantum computers (CRQCs) constitutes an irreversible change in structure in the security paradigms that protect the global critical infrastructure. Raising digital trust on computational intractability of classical integer factorization and discrete logarithms is a monolithic foundation, which is vulnerable to quantum algorithms, that has underpinned decades of digital trust. This study is a solid empirical basis of how to negotiate this unavoidable transition, showing that the mathematical structures of post-quantum cryptography (PQC) are theoretically sound, but their physical implementation into operational technology (OT) needs to be optimized in a sector-specific manner, not just through the superficial replacement of algorithms.

This paper determines that lattice-based architecture, namely ML-KEM and ML-DSA, provide the most reasonable balance between computational performance, memory consumption, and complexity-theoretic security guarantees. Importantly, the introduction of hybrid cryptographic constructions

concatenating classical primitives such as ECDHE with lattice-based encapsulation acts as an essential transitional layer. The current research confirms that this kind of hybridization implies certain performance overhead (in the range of 10.8 percent of the latency in WAN systems) which is operationally acceptable. The result conclusively refutes the industry belief that quantum resistance has to necessarily reduce network availability, confirming the possibility of immediate migration plans that do not compromise classical, mathematically-established security assurances, and at the same time immunizes infrastructure against future quantum decryption attacks.

Nevertheless, the empirical data provided below point out critical structural shortcomings that absolutely annihilate the fallaciousness of PQC as an omnipresent drop-in alternative. The implementation of both code-based and hash-based schemes in systems with low latency and bandwidth limitations is still highly problematic. In particular, the fact that the evaluated signature schemes cannot be configured in such a way as to meet Vehicle-to-Everything (V2X) communication Maximum Transmission Unit (MTU) constraints highlights the fact that post-quantum migration necessitates on-the-fly algorithmic negotiation and complete architecture redesign at the protocol layer. Any effort to shrink mathematically inflated PQC payloads into any legacy standard frameworks based on small 256-bit elliptic curves is doomed to cause systemic network fragmentation, increase the chances of collision, and eventually destroy deterministic safety assurances in mission-critical settings.

Because of the acute, asymmetric threat caused by the adversary model of harvest now, decrypt later (HNDL), the fact that the cryptographic modernization is delayed is a self-compounding systemic risk. Cryptographic exposure of long-horizon information starts at the instant of interception and makes the hypothetical timeline of hardware CRQC realization second to the instantaneous exposure of the existing network traffic. The operators of infrastructures need to start going beyond the theoretical risk assessments and implement vigorous cryptographic discovery stages immediately. This requires the production of automated network telemetry to map dependencies between algorithms, generate holistic Cryptographic Bills of Materials (CBOMs) and discover deeply entrenched past protocols that are not cryptographically agile.

In the end, it is no longer a hypothetical goal in the future, but an urgent, immediate operational need to change to quantum-ready architectures. Conformance to new frameworks, including the CNSA 2.0 requirement is just the minimum of defensive posturing. The challenge of ensuring the persistent integrity, authenticity and confidentiality of national critical infrastructure requires a paradigm shift in thinking, a shift between unresponsive, hard-coded cryptographic implementations, to a robust posture of dynamic, hardware-sensitive cryptographic agility.

8. REFERENCES

[1] Rivest, R. L., A. Shamir, and L. Adleman. 1978. "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems." *Communications of the ACM* 21, no. 2 (February): 120–26. doi:10.1145/359340.359342.

[2] Miller, Victor S. 1985. "Use of Elliptic Curves in Cryptography." In *Advances in Cryptology — CRYPTO '85 Proceedings*, 417–26. Berlin: Springer. doi:10.1007/3-540-39799-X_31.

[3] Koblitz, Neal. 1987. "Elliptic Curve Cryptosystems." *Mathematics of Computation* 48, no. 177: 203–9. doi:10.1090/S0025-5718-1987-0866109-5.

[4] Shor, Peter W. 1994. "Algorithms for Quantum Computation: Discrete Logarithms and Factoring." In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, 124–34. IEEE Computer Society Press. doi:10.1109/SFCS.1994.365700.

[5] Grover, Lov K. 1996. "A Fast Quantum Mechanical Algorithm for Database Search." In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, 212–19. New York: ACM Press. doi:10.1145/237814.237866.

[6] Preskill, John. 2018. "Quantum Computing in the NISQ Era and Beyond." *Quantum* 2 (August): 79. doi:10.22331/q-2018-08-06-79.

[7] Mosca, Michele. 2018. "Cybersecurity in an Era with Quantum Computers: Will We Be Ready?" *IEEE Security & Privacy* 16, no. 5 (September): 38–41. doi:10.1109/MSP.2018.3761723.

[8] National Security Agency (NSA). 2025. "Announcing the Commercial National Security Algorithm Suite 2.0." Accessed April 17, 2026. https://media.defense.gov/2025/May/30/2003728741/-1/-1/0/CSA_CNSA_2.0_ALGORITHMS.PDF.

[9] Bernstein, Daniel J., and Tanja Lange. 2017. "Post-Quantum Cryptography." *Nature* 549, no. 7671 (September): 188–94. doi:10.1038/nature23461.

[10] McEliece, Robert J. 1978. "A Public-Key Cryptosystem Based On Algebraic Coding Theory." *Deep Space Network Progress Report* 44: 114–16.

[11] Chen, Lily, et al. 2016. "Report on Post-Quantum Cryptography." NISTIR 8105. Gaithersburg, MD: NIST. doi:10.6028/NIST.IR.8105.

[12] NIST (National Institute of Standards and Technology). 2020. "Post-Quantum Cryptography." Computer Security Resource Center. Accessed April 2, 2026. https://csrc.nist.gov/projects/post-quantum-cryptography.

[13] van Assche, Gilles. 2006. *Quantum Cryptography and Secret-Key Distillation.* Cambridge: Cambridge University Press. doi:10.1017/CBO9780511617744.

[14] Desurvire, Emmanuel. 2009. *Classical and Quantum Information Theory.* Cambridge: Cambridge University Press. doi:10.1017/CBO9780511803758.

[15] Kahn, David. 1996. *The Codebreakers: The Comprehensive History of Secret Communication from Ancient Times to the Internet.* Scribner. https://books.google.com.ng/books?id=SEH_rHkgaogC.

[16] Pessl, Peter, Leon Groot Bruinderink, and Yuval Yarom. 2017. "To BLISS-B or Not to Be: Attacking strongSwan's Implementation of Post-Quantum Signatures." In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 1843–55. New York: ACM. doi:10.1145/3133956.3134023.

[17] Proos, John, and Christof Zalka. 2004. "Shor's Discrete Logarithm Quantum Algorithm for Elliptic Curves." January. http://arxiv.org/abs/quant-ph/0301141.

[18] Paar, Christof, and Jan Pelzl. 2010. *Understanding*

Cryptography. Berlin: Springer. doi:10.1007/978-3-642-04101-3.

[19] Sood, R., and H. Kaur. 2023. "A Literature Review on RSA, DES and AES Encryption Algorithms." In *Emerging Trends in Engineering and Management*, 57–63. Soft Computing Research Society. doi:10.56155/978-81-955020-3-5-07.

[20] Grassl, Markus, Brandon Langenberg, Martin Roetteler, and Rainer Steinwandt. 2016. "Applying Grover's Algorithm to AES: Quantum Resource Estimates." In *Post-Quantum Cryptography*, 29–43. Cham: Springer. doi:10.1007/978-3-319-29360-8_3.

[21] Galbraith, Steven D. 2012. *Mathematics of Public Key Cryptography.* Cambridge: Cambridge University Press. doi:10.1017/CBO9781139012843.

[22] NIST (National Institute of Standards and Technology). 2020. "Post-Quantum Cryptography." Computer Security Resource Center. Accessed April 17, 2026. https://csrc.nist.gov/projects/post-quantum-cryptography.

[23] Regev, Oded. 2009. "On Lattices, Learning with Errors, Random Linear Codes, and Cryptography." *Journal of the ACM* 56, no. 6 (September): 1–40. doi:10.1145/1568318.1568324.

[24] Ducas, Léo, et al. 2018. "CRYSTALS-Dilithium: A Lattice-Based Digital Signature Scheme." *IACR Transactions on Cryptographic Hardware and Embedded Systems* 2018, no. 1 (February): 238–68. doi:10.46586/tches.v2018.i1.238-268.

[25] Fouque, Pierre-Alain, et al. 2019. "Falcon: Fast-Fourier Lattice-Based Compact Signatures over NTRU." https://api.semanticscholar.org/CorpusID:231637439.

[26] Lyubashevsky, Vadim. 2012. "Lattice Signatures without Trapdoors." In *Advances in Cryptology – EUROCRYPT 2012*, edited by D. Pointcheval and T. Johansson, 738–55. Berlin: Springer. doi:10.1007/978-3-642-29011-4_43.

[27] Aragon, Nicolas, et al. 2018. "BIKE - Bit-Flipping Key Encapsulation." https://csrc.nist.gov/CSRC/media/Presentations/BIKE/images-media/BIKE-April2018.pdf.

[28] Ding, Jintai, and Dieter Schmidt. 2005. "Rainbow, a New Multivariable Polynomial Signature Scheme." In *Applied Cryptography and Network Security*, 164–75. doi:10.1007/11496137_12.

[29] Beullens, Ward. 2022. "Breaking Rainbow Takes a Weekend on a Laptop." In *Advances in Cryptology – CRYPTO 2022*, edited by Y. Dodis and T. Shrimpton, 464–79. Cham: Springer. doi:10.1007/978-3-031-15979-4_16.

[30] Huelsing, Andreas, Denis Butin, Stefan-Lukas Gazdag, Joost Rijneveld, and Aziz Mohaisen. 2018. "XMSS: eXtended Merkle Signature Scheme." RFC 8391. doi:10.17487/RFC8391.

[31] Bernstein, Daniel J., Andreas Hülsing, Stefan Kölbl, Ruben Niederhagen, Joost Rijneveld, and Peter Schwabe. 2019. "The SPHINCS+ Signature Framework." In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, 2129–46. New York: ACM. doi:10.1145/3319535.3363229.

[32] Castryck, Wouter, and Thomas Decru. 2023. "An Efficient Key Recovery Attack on SIDH." In *Advances in Cryptology – EUROCRYPT 2023*, 423–47. doi:10.1007/978-3-031-30589-4_15.

[33] Lyubashevsky, Vadim, Chris Peikert, and Oded Regev. 2013. "On Ideal Lattices and Learning with Errors over Rings." *Journal of the ACM* 60, no. 6 (November): 1–35. doi:10.1145/2535925.

[34] Kiltz, Eike, Vadim Lyubashevsky, and Christian Schaffner. 2018. "A Concrete Treatment of Fiat-Shamir Signatures in the Quantum Random-Oracle Model." In *Advances in Cryptology – EUROCRYPT 2018*, edited by J. Nielsen and V. Rijmen, 552–86. Cham: Springer. doi:10.1007/978-3-319-78372-7_18.

[35] Albrecht, Martin R., Rachel Player, and Sam Scott. 2019. "On the Concrete Hardness of Learning with Errors." Information Security Group. https://eprint.iacr.org/2015/046.pdf.

[36] Giacon, Federico, Felix Heuer, and Bertram Poettering. 2018. "KEM Combiners." In *Lecture Notes in Computer Science*, vol. 10769: 190–218. doi:10.1007/978-3-319-76578-5_7.

[37] IEEE. 2016. "IEEE Standard for Wireless Access in Vehicular Environments--Security Services for Applications and Management Messages." Piscataway, NJ. doi:10.1109/IEEESTD.2016.7426684.

[38] Biswas, Subir, and Jelena Misic. 2013. "A Cross-Layer Approach to Privacy-Preserving Authentication in WAVE-Enabled VANETs." *IEEE Transactions on Vehicular Technology* 62, no. 5 (June): 2182–92. doi:10.1109/TVT.2013.2238566.

[39] Campagna, Matthew. 2013. "Standards for Efficient Cryptography SEC 4: Elliptic Curve Qu-Vanstone Implicit Certificate Scheme (ECQV)." Certicom Research.

[40] Ravi, Prasanna, Sujoy Sinha Roy, Anupam Chattopadhyay, and Shivam Bhasin. 2020. "Generic Side-Channel Attacks on CCA-Secure Lattice-Based PKE and KEMs." *IACR Transactions on Cryptographic Hardware and Embedded Systems* 2020, no. 3: 307–35. doi:10.13154/tches.v2020.i3.307-335.

[41] Kannwischer, Matthias J., Joost Rijneveld, Peter Schwabe, and Ko Stoelen. 2019. "pqm4: Testing and Benchmarking NIST PQC on ARM Cortex-M4." In *Second PQC Standardization Conference*. https://hdl.handle.net/2066/210214.

[42] Kampanakis, Panos, and G. Ravago. 2026. "Post-quantum Hybrid Key Exchange with ML-KEM in the Internet Key Exchange Protocol Version 2 (IKEv2)." IPSECME. Accessed April 17, 2026. https://www.ietf.org/archive/id/draft-kampanakis-ml-kem-ikev2-06.html.

[43] International Electrotechnical Commission. 2010. *IEC 61508: Functional Safety of Electrical/Electronic/Programmable Electronic Safety-Related Systems.* Accessed April 17, 2026. https://webstore.iec.ch/en/iec_catalog/product/preview/?id=L3B1Yi9wZGYvcHJldmlldy9pbmZvX2llYzYxNTA4LTF7ZWQyLjB9Yi5wZGY=.