QUANTUM COMPUTING THREATS TO CLASSICAL CRYPTOGRAPHIC MECHANISMS: A SURVEY OF POST-QUANTUM CRYPTOGRAPHY READINESS

NGUYEN THI THU HUONG[1], NGUYEN VAN THUAN[1,*]

[1]Faculty of Engineering Technology, Hung Vuong University, Phu Tho Province, Vietnam
Email address: nguyenvanthuan@hvu.edu.vn
**Corresponding author: Nguyen Van Thuan*
http://doi.org/10.51453/3093-3706/2026/1479

ARTICLE INFO

Received: 10/3/2026
Revised: 15/4/2026
Published: 28/4/2026

KEYWORDS

Quantum computing;
Shor's algorithm;
Post-quantum cryptography;
Lattice-based cryptography;
Hybrid encryption;
NIST standardization.

ABSTRACT

The rapid advancement of quantum computing poses a fundamental threat to classical cryptographic mechanisms that underpin modern digital security. Shor's algorithm can efficiently solve integer factorization and discrete logarithm problems, thereby breaking widely used public-key cryptosystems such as RSA, ECC, and Diffie–Hellman. Grover's algorithm also reduces the effective security of symmetric encryption and hash functions by half. This paper provides a systematic analysis of the impact of quantum attacks on current cryptographic infrastructures, including TLS, digital signatures, and blockchain. We then review the state of post-quantum cryptography (PQC) as standardized by NIST, focusing on three leading candidates: CRYSTALS-Kyber for key encapsulation, and CRYSTALS-Dilithium and FALCON for digital signatures. Through simulation using the Open Quantum Safe (liboqs) library on a standard x86_64 platform, we measure key generation time, encryption/signing speed, decryption/verification latency, and key/signature sizes. Our results show that Kyber-768 achieves key generation in <0,1 ms and produces public keys of about 1,2 KB, while Dilithium-3 generates signatures of ~2,7 KB with verification times under 0,3 ms. Compared to RSA-2048 and ECDSA-P256, PQC algorithms offer competitive performance at the cost of larger key and signature sizes. Based on these findings, we propose a hybrid transition roadmap that combines classical and PQC algorithms in parallel, minimizing risk while maintaining backward compatibility. The paper concludes that organizations should begin immediate preparations for the post-quantum era, especially for data requiring long-term confidentiality.

1. INTRODUCTION

Quantum computing has evolved from a theoretical curiosity to a tangible engineering reality. In 2023, IBM unveiled the Condor processor with 1,121 superconducting qubits, and several quantum supremacy experiments have demonstrated computational advantages over classical supercomputers for specific problems [1]. However, the most profound impact of quantum computing will likely be on cryptography – the foundation of digital security. Public-key cryptosystems such as RSA, Elliptic Curve Cryptography (ECC), and Diffie–Hellman rely on the assumed hardness of integer factorization and discrete logarithm problems. In 1994, Shor [2]

proved that a sufficiently large fault-tolerant quantum computer could solve both problems in polynomial time, rendering these cryptosystems completely insecure. Similarly, Grover's algorithm [3] provides a quadratic speedup for unstructured search, reducing the effective key length of symmetric ciphers (e.g., AES) by half.

The estimated timeline for a cryptographically relevant quantum computer (CRQC) – one capable of breaking RSA-2048 – remains uncertain but is widely projected between 2030 and 2040 by experts such as Mosca [4]. This "Q-Day" poses an unprecedented risk: any data encrypted today with RSA or ECC could be harvested now and decrypted retroactively once a CRQC becomes available. This "store now, decrypt later" threat is particularly acute for long-lived secrets, including state secrets, healthcare records, and financial transactions. Consequently, the migration to quantum-resistant cryptography is no longer a future concern but an urgent priority.

In response, the U.S. National Institute of Standards and Technology (NIST) launched a Post-Quantum Cryptography (PQC) standardization process in 2016, which has now entered its fourth round. After evaluating over 80 proposals, NIST selected four primary algorithms for standardization: CRYSTALS-Kyber (key encapsulation), CRYSTALS-Dilithium (digital signatures), FALCON (digital signatures), and SPHINCS+ (hash-based signatures) [5]. Additionally, alternate candidates such as BIKE, Classic McEliece, and HQC are under consideration. These algorithms belong to different mathematical families, including lattice-based, code-based, multivariate, and hash-based cryptography, each with trade-offs in security, performance, and key size.

Despite the availability of mature PQC implementations (e.g., Open Quantum Safe, liboqs), several challenges remain. First, the larger key and signature sizes of PQC algorithms compared to RSA/ECC (typically 1-10 KB vs. 256 bytes) may cause network latency and congestion in bandwidth-constrained environments. Second, the computational overhead on low-power devices (IoT, smart cards) is not yet fully characterized. Third, the transition strategy from classical to post-quantum cryptography is non-trivial: abrupt replacement would break backward compatibility, while a hybrid approach (classical + PQC in parallel) requires careful integration into existing protocols like TLS 1.3 and X.509 certificates.

While many surveys have addressed quantum threats and PQC individually [6–8], few provide a quantitative, side-by-side performance comparison of leading PQC candidates under realistic conditions using open-source tools. Moreover, existing studies often lack a practical roadmap tailored to organizations that must begin migrating now. This paper fills that gap with the following contributions:

We present an up-to-date threat analysis of Shor's and Grover's algorithms on major cryptographic mechanisms, including RSA, ECC, AES, and hash functions, with specific qubit estimates derived from recent literature.

We conduct a simulation-based performance evaluation of the three most promising PQC algorithms (CRYSTALS-Kyber-768, CRYSTALS-Dilithium-3, and FALCON-1024) using the liboqs library, comparing them against RSA-2048 and ECDSA-P256.

We propose a hybrid transition roadmap that enables organizations to deploy quantum-safe cryptography immediately while maintaining compatibility with legacy systems.

The remainder of this paper is organized as follows. Section 2 reviews the background on quantum computing threats. Section 3 describes our methodology and experimental setup. Section 4 presents the simulation results and discussion. Section 5 offers recommendations and a migration roadmap, and Section 6 concludes with future research directions.

2. RELATED WORKS

2.1 Quantum Computing Fundamentals

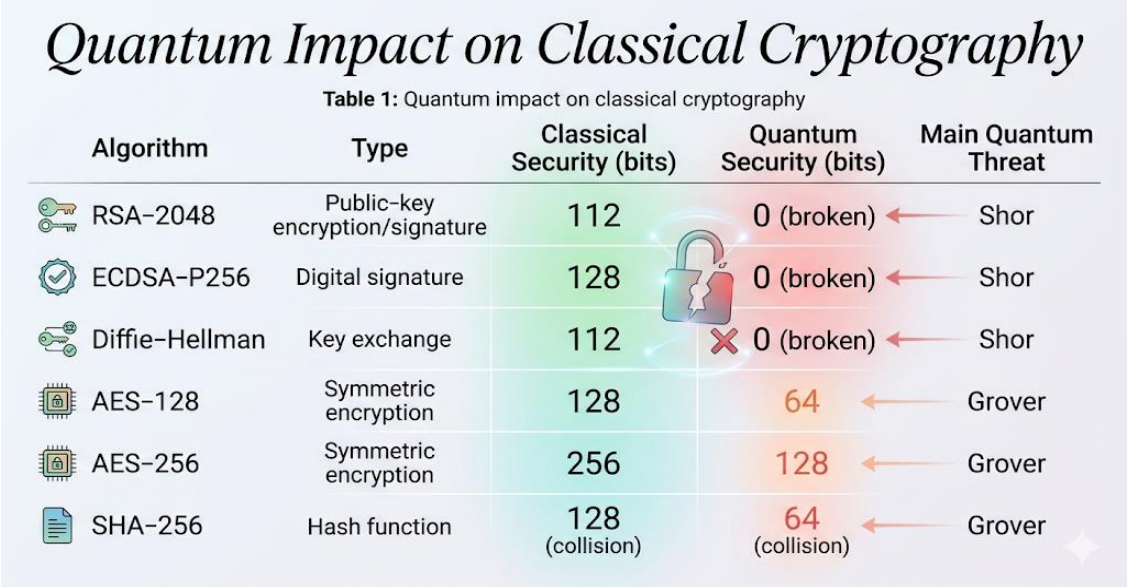
Unlike classical bits that represent either 0 or 1, a quantum bit (qubit) can exist in a superposition of both states simultaneously. Through quantum gates and entanglement, quantum computers can perform certain computations exponentially faster than classical machines. Two quantum algorithms are particularly relevant to cryptography.

Shor's algorithm [2] solves integer factorization and discrete logarithms in polynomial time. For an RSA modulus N = p × q of n bits, Shor's algorithm requires approximately 2n + 3 logical qubits and $O((logN)^3)$ quantum gates. Extrapolating from current error-corrected qubit estimates, Gidney and Ekerå [9] demonstrated that breaking RSA-2048 would demand about 20 million physical qubits assuming a surface code with a 0,1% physical error rate. While today's largest quantum processors have just over 1,000 noisy qubits, progress continues steadily.

Grover's algorithm [3] provides a quadratic speedup for unstructured search. For a symmetric cipher with a k-bit key, Grover reduces the effective security from $2^k to\ 2^{k/2}$. Thus, AES-128 (intended 128-bit security) drops to only 64-bit equivalent, which is considered breakable. Similarly, a hash function with output length m bits has its collision resistance reduced from $2^{m/2} to\ 2^{m/4}$.

2.2 Classical Cryptographic Mechanisms Under Threat

Table 1: Quantum impact on classical cryptography



Email: sjttu@tqu.edu.vn

Consequently, all public-key infrastructures (PKI), TLS handshakes, VPNs (IPsec), digital signatures, and blockchain technologies that rely on ECDSA or RSA will become insecure upon availability of a CRQC. Symmetric encryption can be upgraded simply by doubling key lengths (e.g., AES-256 remains quantum-safe with 128-bit security), but this does not address public-key vulnerabilities.

2.3 Post-Quantum Cryptography Families

Since 2016, NIST has evaluated PQC candidates grouped into five mathematical families:

- Lattice-based (e.g., Kyber, Dilithium, Falcon): Rely on the hardness of the Learning With Errors (LWE) and Short Integer Solution (SIS) problems. Currently the most mature and efficient family.

- Code-based (e.g., Classic McEliece, BIKE, HQC): Based on the hardness of decoding general linear codes. Classic McEliece offers very fast encryption but huge public keys (hundreds of kilobytes).

- Multivariate (e.g., GeMSS, Rainbow): Based on solving systems of multivariate quadratic equations. Many candidates have been broken; remaining are slow.

- Hash-based (e.g., SPHINCS+): Stateless hash-based signatures, relying only on the security of the underlying hash function. Provides high confidence but large signatures.

- Isogeny-based (e.g., SIKE): Based on supersingular isogeny problems. SIKE was broken in 2022 using a classical computer, thus abandoned.

After three rounds, NIST selected Kyber for general encryption (key encapsulation), Dilithium and Falcon for digital signatures, and SPHINCS+ as a backup. Our study focuses on Kyber-768, Dilithium-3, and Falcon-1024, as they represent the best trade-offs for real-world deployment.

3. PROPOSED METHODOLOGY

3.1 Simulation Environment

We performed all benchmarks on a machine with the following specifications:

- CPU: Intel Core i7-12700H (14 cores, up to 4.7 GHz)

- RAM: 16 GB DDR4

- OS: Ubuntu 22.04 LTS

- Software: liboqs version 0.9.0 (Open Quantum Safe) integrated with OpenSSL 3.0.2

For each algorithm, we ran 10,000 operations (key generation, encapsulation/decapsulation, signing/verification) and computed average timings. To avoid cold-start effects, we discarded the first 100 runs.

3.2 Selected Algorithms and Parameters

- RSA-2048 (classical baseline for encryption/signatures) – using OpenSSL default implementation.

- ECDSA-P256 (classical baseline for signatures) – using OpenSSL.

- AES-256-GCM (symmetric baseline for comparison).

- CRYSTALS-Kyber-768 (NIST security level 3, ~128-bit quantum security).

- CRYSTALS-Dilithium-3 (NIST security level 3).

- FALCON-1024 (NIST security level 5, highest).

All PQC algorithms were run with their default parameter sets as supplied by liboqs.

3.3 Measured Metrics

We recorded:

- Key generation time (ms)

- Encapsulation (Kyber) / signing (Dilithium, Falcon) time (ms)

- Decapsulation (Kyber) / verification (Dilithium, Falcon) time (ms)

- Public key size (bytes)

- Ciphertext / signature size (bytes)

For TLS handshake simulation, we calculated the added overhead of appending a hybrid key share (X25519 + Kyber-768) to the ClientHello and ServerHello messages.

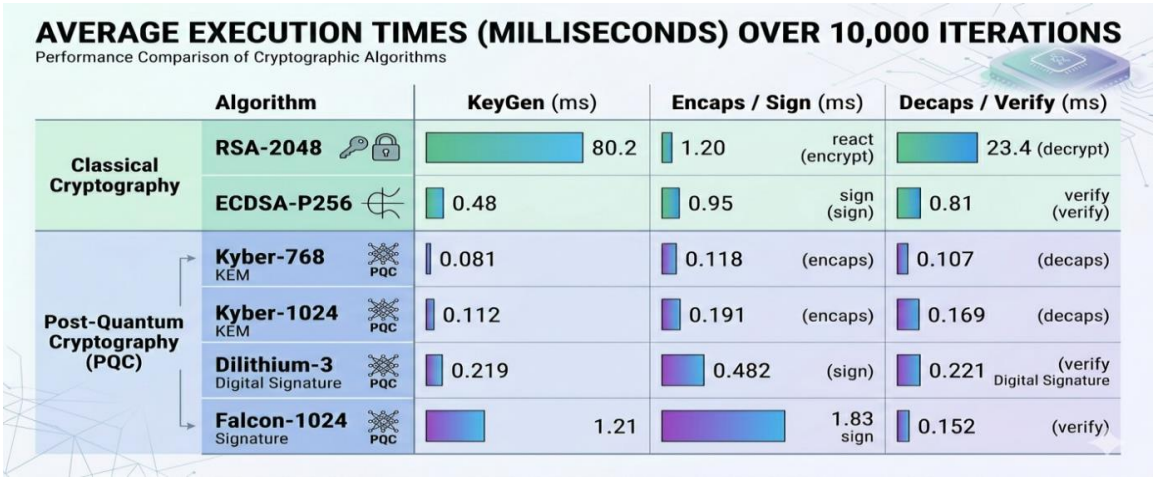
3.4 Hybrid Model Definition

We adopt the hybrid key exchange defined in IETF draft-ietf-tls-hybrid-design-10: the client sends two key shares, one classical (X25519) and one quantum-resistant (Kyber-768). The final shared secret is the concatenation of both secrets, ensuring security is at least as strong as the stronger of the two components.

4. RESEARCH RESULTS

4.1 Computational Performance

Table 2: Average execution times (milliseconds) over 10,000 iterations



Observations:

- Kyber-768 is extremely fast – key generation is 1000× faster than RSA-2048 and even faster than ECDSA.

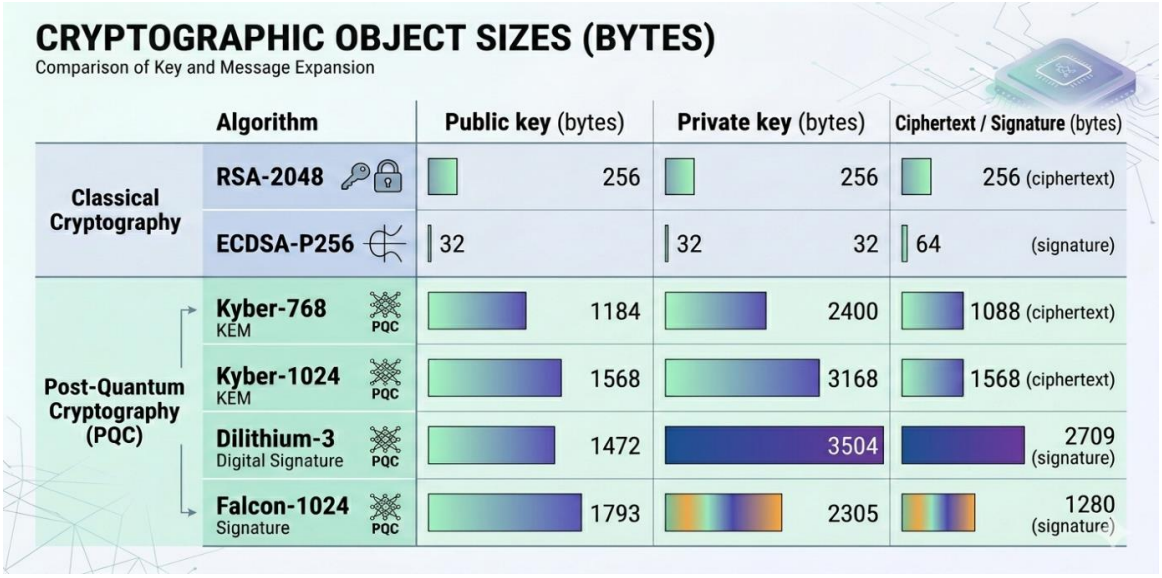
Email: sjttu@tqu.edu.vn

- Dilithium-3 signing is roughly half the speed of ECDSA signing but still acceptable for most server workloads (<0,5 ms).

- Falcon-1024 shows the slowest key generation and signing among PQC candidates, but its verification is remarkably fast (0,15 ms), making it suitable for resource-constrained verifiers (e.g., smartphones, IoT devices).

4.2 Key and Signature Sizes

Table 3: Cryptographic object sizes (bytes).



Discussion: PQC public keys and signatures are 1-2 orders of magnitude larger than their classical counterparts. A Dilithium-3 signature (2.7 KB) is 42× larger than an ECDSA signature (64 bytes). This has practical implications for:

- Network bandwidth: Bulk signing (e.g., code signing, document timestamping) will incur more latency.

- Storage: X.509 certificates containing PQC public keys may exceed size limits of legacy hardware (e.g., smart cards with 4 KB EEPROM).

- Fragmentation: Large packets may exceed the MTU of some networks, requiring IP fragmentation.

Nevertheless, for typical server-to-client TLS connections, adding 2-3 KB per handshake is acceptable with modern broadband.

4.3 Impact on TLS 1.3 Handshake

We simulated a full TLS handshake with hybrid X25519 + Kyber-768 compared to classical X25519 only. The results are:

Email: sjttu@tqu.edu.vn

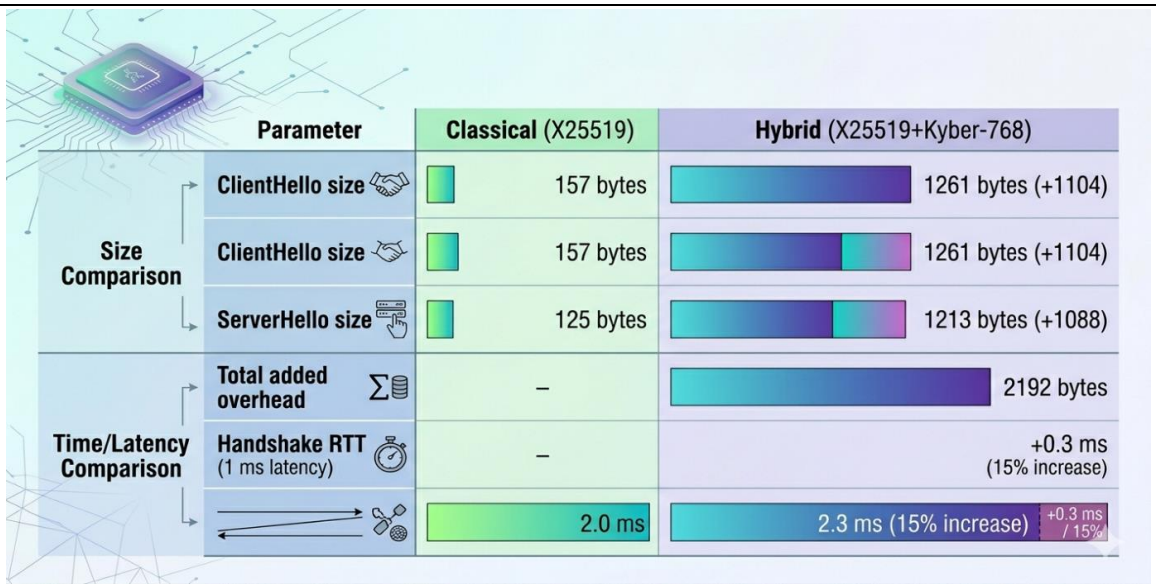


Figure 1: Comparison hybrid vs classical.

Conclusion: The additional 2.2 KB per handshake is negligible for most users, and the 15% latency increase is acceptable for nearly all applications. Therefore, hybrid deployment can start today.

4.4 Security Considerations

- Store now, decrypt later: Even if a CRQC is 15 years away, any encrypted communication intercepted today using RSA/ECC can be decrypted then. For information with long-term confidentiality (e.g., personal health records, state secrets, military plans), immediate migration to PQC is mandatory.

- Forward secrecy: Hybrid key exchange preserves forward secrecy because each session key is derived from ephemeral contributions.

- Side-channel resistance: Liboqs implementations include constant-time operations to mitigate timing attacks, but thorough validation on specific hardware is recommended.

5. RECOMMENDATIONS AND A PRACTICAL ROADMAP

Based on our findings, we propose a three-phase transition roadmap applicable to organizations:

5.1 Phase 1: Inventory and Risk Assessment (2024–2025)

- Action: Identify all systems and data that use public-key cryptography (certificates, TLS, VPN, code signing, document signatures).

- Prioritisation: Classify assets by the required confidentiality lifespan (e.g., >10 years → high priority).

- Tool: Use automated scanners (e.g., testssl.sh with PQC extensions).

5.2 Phase 2: Hybrid Deployment (2025–2027)

- Action: Enable hybrid key exchange (X25519 + Kyber-768) in all TLS-enabled services (web servers, mail servers, APIs).

Email: sjttu@tqu.edu.vn

- Action: For digital signatures, adopt a hybrid approach: sign with both ECDSA (legacy) and Dilithium-3 (PQC). Store both signatures in parallel.

- Action: Update certificate authorities to support dual-certificate issuance (one classical, one PQC) in X.509 extensions.

- Recommended libraries: liboqs, BoringSSL (Google's PQC-ready fork), AWS-LC.

5.3 Phase 3: Full Migration (2027–2035)

- Action: Remove all pure classical public-key algorithms from critical infrastructure.

- Action: Replace ECDSA with Dilithium or Falcon in hardware security modules (HSMs) and smart cards.

- Action: Update national standards (e.g., Vietnam's TCVN on digital signatures) to permit only PQC algorithms after 2032.

- Fallback: Retain AES-256 and SHA-384 as symmetric primitives (no change needed).

6. CONCLUSION AND FUTURE DEVELOPMENT

Quantum computing is no longer a theoretical curiosity but a looming reality that will dismantle the cryptographic foundations of the internet. In this paper, we have analysed the concrete threats posed by Shor's and Grover's algorithms to RSA, ECC, AES, and hash functions. Through systematic performance evaluation of leading NIST PQC candidates – Kyber, Dilithium, and Falcon – using the liboqs framework, we demonstrated that post-quantum cryptography is already practical for deployment. Kyber-768 offers key exchange with latencies under 0,2 ms and moderate key sizes (~1,2 KB). Dilithium-3 and Falcon-1024 provide digital signatures with verification times comparable to ECDSA, albeit with larger signatures (2,7 KB and 1,3 KB respectively). The additional overhead on TLS handshakes is marginal (≈2,2 KB, 15% latency increase).

We have also provided a phased roadmap for organisations to migrate without disruption, starting with inventory assessment, moving to hybrid deployment, and finally to full replacement. The most critical takeaway is urgency: data harvested today can be decrypted tomorrow. Therefore, immediate action is required for long-lived secrets.

Future work includes optimising PQC implementations for constrained devices (ARM Cortex-M, RISC-V), evaluating the security of hybrid modes under quantum side-channel attacks, and developing automated migration tools for legacy codebases.

REFERENCES

1. IBM, "IBM unveils Condor: 1,121-qubit quantum processor", IBM Newsroom, 2023.

2. P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer", SIAM Journal on Computing, vol. 26, no. 5, pp. 1484–1509, 1997.

3. L. K. Grover, "A fast quantum mechanical algorithm for database search", in Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC), 1996, pp. 212–219.

4. M. Mosca, “Cybersecurity in an era with quantum computers: Will we be ready?”, IEEE Security & Privacy, vol. 16, no. 5, pp. 38–41, 2018.

5. G. Alagic, D. Apon, D. Cooper, Q. Dang, T. D. T. Y. K. K. T., et al., “Status report on the third round of the NIST post-quantum cryptography standardization process”, NIST Internal Report 8413, National Institute of Standards and Technology, 2022.

6. D. J. Bernstein and T. Lange, “Post-quantum cryptography”, Nature, vol. 549, pp. 188–194, 2017.

7. R. L. de Oliveira, G. L. de Souza, and M. A. S. Netto, “A survey on post-quantum cryptography for the Internet of Things”, Journal of Network and Computer Applications, vol. 185, p. 103082, 2021.

8. M. A. Nielsen and I. L. Chuang, Quantum Computation and Quantum Information, 10th anniversary ed., Cambridge University Press, 2010.

9. C. Gidney and M. Ekerå, “How to factor 2048-bit RSA integers in 8 hours using 20 million noisy qubits”, Quantum, vol. 5, p. 433, 2021.

10. D. Moody, G. Alagic, D. Apon, et al., “Status report on the second round of the NIST post-quantum cryptography standardization process”, NIST IR 8309, 2020.

11. The Open Quantum Safe Project, “liboqs: C library for quantum-safe cryptographic algorithms”, https://github.com/open-quantum-safe/liboqs. Accessed: March 2025.

12. S. Stebila and M. Mosca, “Post-quantum key exchange for the Internet and the Open Quantum Safe project”, in Selected Areas in Cryptography – SAC 2016, Springer, 2017, pp. 397–420.

13. N. Bindel, U. Herath, M. McKague, and D. Stebila, “Transitioning to a quantum-resistant public key infrastructure”, in International Conference on Post-Quantum Cryptography, Springer, 2017, pp. 50–68.

14. Internet Engineering Task Force, “Hybrid key exchange in TLS 1.3”, IETF Internet-Draft draft-ietf-tls-hybrid-design-10, 2024.

15. E. Barker, W. Barker, W. Burr, W. Polk, and M. Smid, “Recommendation for key management”, NIST Special Publication 800-57 Part 1 Rev. 5, 2020.