PROTOCOL-LEVEL ENHANCEMENTS FOR POST-QUANTUM CRYPTOGRAPHIC SYSTEMS

Bhoomika M G , MCA Student, PESITM, Shivamogga, Karnataka, India
Mrs. Tejaswini A, Assistant Professor, Dept. of MCA, PESITM, Shivamogga

ABSTRACT

As quantum computing rapidly progresses, the security of traditional cryptographic systems like RSA, ECC, and DSA is at risk. Quantum algorithms such as Shor's and Grover's can solve the mathematical problems that these classical methods depend on, potentially rendering them obsolete. In response, this project examines new types of cryptographic algorithms—referred to as post-quantum cryptography (PQC)—that are specifically designed to withstand attacks even from quantum computers. The research thoroughly explores various PQC approaches, including lattice-based, hash-based, code-based, multivariate polynomial, and isogeny-based cryptography. Each is evaluated for its security in the face of quantum threats, how efficiently it performs encryption and decryption, its computational demands, and its practicality for everyday use. By comparing these algorithms and analysing results from experiments, we identify which quantum-resistant techniques are most suitable to replace or support current systems. The goal is to ensure that private data and communications remain protected well into the future, even as quantum computing becomes a real-world factor. Ultimately, this study supports ongoing global efforts to develop secure, future-proof methods for safeguarding digital information against the emerging challenges posed by quantum technologies.

Keywords— *Quantum Computing, Post-Quantum Cryptography (PQC), Quantum-Resistant Algorithms, Shor's Algorithm, Grover's Algorithm, Lattice-Based Cryptography, Hash-Based Cryptography, Code-Based Cryptography, Multivariate Polynomial Cryptography.*

1. Introduction

Cryptography has long been important in ensuring the digital communication operation by offering protection of data confidentiality, authentication and tampering. The more commonly used public-key systems (e.g. RSA and elliptic-curve cryptography (ECC)) have been used to protect banking on the internet all the way up to classified communications of sensitive governments, based on the intractability of computing solutions to certain mathematical problems, such as determining factors of large numbers, and discrete logarithm problems. But the high speed development in the quantum computer has the potential of destroying this security. These mathematical problems may be solved much more efficiently by quantum algorithms and above all by Shor algorithm and threaten current

cryptographic practices. This emerging threat has motivated security research all around the globe to come up with novel crypto-methods that would be quantum-computer resistant. In contrast to quantum key distribution, these quantum-resistant, or post-quantum cryptographic (PQC) algorithms can be implemented on the current classical computer infrastructure, so they are more viable toward implementation by many parties. Post-quantum cryptography is mathematically based and depends on problems that are thought to be inauspicious by classical and quantum personal computers. The leading approaches include lattice-based methods, code-based cryptography, multivariate polynomial equations, hash-based schemes, and recently, isogeny-based algorithms. Each approach carries distinct benefits in terms of security, performance, and real-world feasibility, and together they form the foundation of the next generation of secure communications. The goal is to study, analyse, and implement these key quantum-resistant algorithms. By examining their security features, efficiency, and practicality, the project aims to determine how effectively these solutions can replace or complement current classical cryptography. This research aligns with the broader global movement to develop cryptographic standards capable of ensuring data security and privacy well into the future as quantum computing matures.

2. Literature survey

The growing capabilities of quantum computers have raised serious concerns about the future security of digital systems. Traditional encryption standards such as RSA and ECC, which currently form the backbone of most secure communications, are highly susceptible to quantum attacks—particularly through algorithms designed to break them quickly. Researchers have been working on new kinds of cryptographic algorithms, termed post-quantum or quantum-resistant algorithms, which would be used to deal with this risk. lattice-based cryptography has been one of the highlights among them given its high level of security foundations. Other emerging techniques such as Kyber and Dilithium incorporate sophisticated mathematical puzzles that are assumed to meet next to impossible classical, as well as quantum, attacks. These algorithms are in proposal to become future security standard because of its security and performance balance(Alkim et al. [1]).

A further well-known method refers to code-based cryptography, based on the complexity of decoding error-correcting codes. The approach has already been researched over many decades, and is characterized by excellent resistance to quantum threats. However, a major limitation is the large size of the keys required, which makes implementation difficult in resource-constrained environments. Despite this, the approach continues to hold promise for applications where long-term data confidentiality is critical, such as archival storage and military communication (Bernstein et al. [2]).

In the field of digital authentication, hash-based cryptographic signatures are emerging as a highly secure alternative. Rather than providing encryption, these schemes focus on verifying data integrity and authenticity. They make use of existing cryptographic hash functions and have a transparent design that helps in reducing implementation-related

vulnerabilities. These properties make them particularly useful in embedded systems and IoT environments, where lightweight and secure signing is essential (Hülsing et al. [3]).

Multivariate cryptography, which involves solving equations with multiple variables over finite fields, represents another family of post-quantum algorithms. These schemes are attractive due to their potential for fast execution, especially in lightweight applications. However, their mathematical structures require careful evaluation, as certain designs may be vulnerable to specific types of attacks. Research in this area continues to refine these schemes to improve both their security and efficiency (Ding et al. [4]).

To enable a smooth transition from classical to quantum-safe cryptographic systems, hybrid cryptography has been proposed. This approach involves the simultaneous use of both traditional and post-quantum algorithms within the same system. In this way, organizations will be able to continue compatibility with the existing infrastructure and increase protection against emerging quantum threats. This dual-layer approach also allows for gradual adoption without completely replacing current systems overnight (Chen et al. [5]).

Post-quantum algorithms have been demonstrated in practice to be capable of being deployed with limited performance overhead. Post-quantum key encapsulation implementations used together with classical symmetric encryption approaches have demonstrated repeating trends of correct decryption, reasonable CPU and memory consumption, and low error rate. These findings indicate that post-quantum cryptography may not only be existentially safe but also practically feasible even in areas such as safe communication, protection of files, and cloud services.(Bos et al. [6]).

3. Proposed methodology

The process involves a formal process of designing, developing, and deploying a secure quantum-resistant encryption and decryption system for plain data. It begins with the requirement analysis to establish the requirement of postquantum cryptographic protection by evaluating the threat model, data sensitivity, performance requirements, and compliance requirements, and determining the integration scope such as file encryption, network communication, and secure storage. Algorithm selection is then performed based on NIST guidelines, selecting suitable post-quantum algorithms such as lattice-based, hash-based, or code-based schemes, and the selection of a suitable key encapsulation mechanism for secure key agreement, a compact symmetric encryption algorithm for bulk data, and hybrid cryptography planning if integration with classical systems is required.

The system architecture is designed with modular separation of cryptographic logic, key management, encryption and decryption processing, and interface layers with clear definition of data flow, communication interfaces, and key lifecycle phases with pluggable algorithm modules support for future additions. In implementation, modules for key generation, key encapsulation and decapsulation, symmetric encryption and decryption, and secure storage and retrieval mechanisms are implemented and integrated. Key management integration includes the setup of centralized or distributed solutions with access control, key rotation, backup, revocation, audit mechanisms, and lifecycle security

and compliance guarantee. System integration integrates encryption modules with application layers such as file systems, communication protocols, and databases while including hybrid support and exposing secure APIs or user interfaces. Testing and validation phases include functional, performance, and security testing to ensure cryptographic correctness, resilience, and efficiency, and also audits and code reviews. Lastly, the system is rolled out with runtime security policies and auditing, and maintenance entails the application of patches, examination of standards, and periodic upgrade of algorithms, key policies, and configurations to provide strong post-quantum security.

3.1 Proposed model diagram

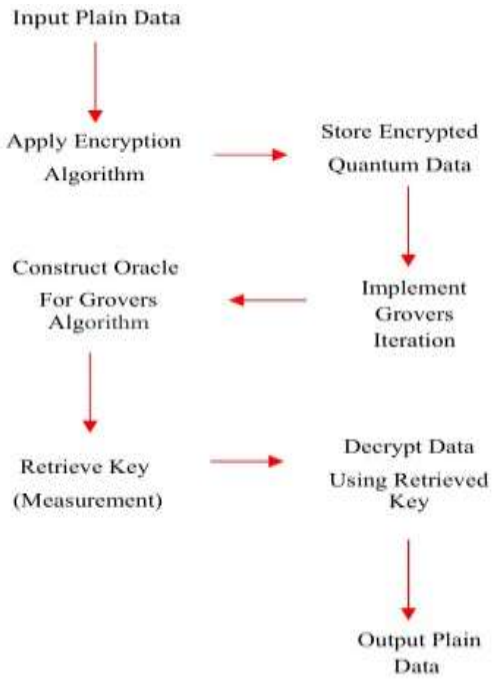

Figure 3.1 Block diagram for quantum resistant

Our approach to this project is structured and forward-thinking, beginning with a foundational understanding of quantum computing and its disruptive impact on current cryptographic systems. As we explore quantum algorithms like Shor's and Grover's, it becomes evident that widely used schemes such as RSA and ECC are highly vulnerable in a post-quantum world. This realization leads us to explore quantum-resistant cryptographic algorithms identified by organizations like NIST. These include various families such as lattice-based, code-based, hash-based, and multivariate polynomial-based schemes. Once we understand these algorithm families in depth, the next logical step is implementation. We turn to reliable libraries like the Open Quantum Safe (OQS) project to access cryptographic primitives, using languages like Python or C/C++ to build and test our implementations in a controlled environment.

After implementation, we shift focus to benchmarking the algorithms by evaluating key operations—encryption, decryption, key generation, signing, and verification. We will measure performance based on execution time, memory

and CPU usage, key and ciphertext sizes, and potential compatibility with existing protocols like TLS/SSL. Security analysis will involve examining the resilience of each algorithm against classical and quantum threats, with particular attention to the underlying hard problems such as Learning With Errors (LWE) or decoding random linear codes. We'll also assess resistance to side-channel and implementation-based attacks. Integration testing will help us understand how these cryptographic solutions can be embedded into real-world systems like secure email and VPNs. Ultimately, we'll compare all tested algorithms for their performance and practicality, offering clear recommendations for deployment across various environments—from lightweight IoT to high-security infrastructures. We straddle the innovations of today and the innovations of the future, and as a piece of work we strive to strike a balance between the present and the future by illustrating how Post-Quantum Cryptography can be put into use now to create a more secure digital world of the future.

4. Mathematical Formulas

1. Encryption:

$$C = Enc(m, K)$$

- The plaintext message mmm is encrypted using a key KKK to produce ciphertext CCC.
- Enc() is the encryption function.
- This step transforms readable data into an unreadable format for confidentiality.

2. Decryption:

$$m = Dec(C, K)$$

- The ciphertext CCC is decrypted using the same key KKK to retrieve the original plaintext message mmm.
- Dec() is the decryption function.
- This reverses the encryption process to recover the original data.

Where

- m → Plain data (original readable message)
- K → Encryption/Decryption key
- C → Ciphertext (encrypted message)

5. Graphs

5.1 Key Encapsulation vs Type of Algorithm

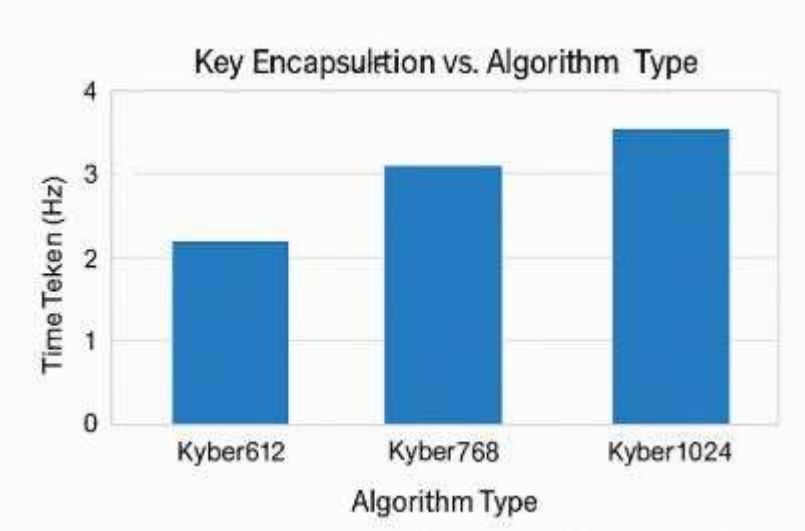

Figure 5.1 Key Encapsulation vs Type of Algorithm

The bar graph presents the time taken for key encapsulation using different variants of the Kyber algorithm—Kyber612, Kyber768, and Kyber1024. It reveals a noticeable increase in encapsulation time as the security level rises. Among the three, Kyber612 is the fastest, with Kyber768 requiring slightly more time, and Kyber1024 taking the longest. This trend reflects a common trade-off in cryptography: higher levels of security come at the cost of increased computational effort. The results emphasize the balance that must be struck between performance and security when implementing post-quantum cryptographic solutions.

5.2 CPU Usage during Encryption/Decryption

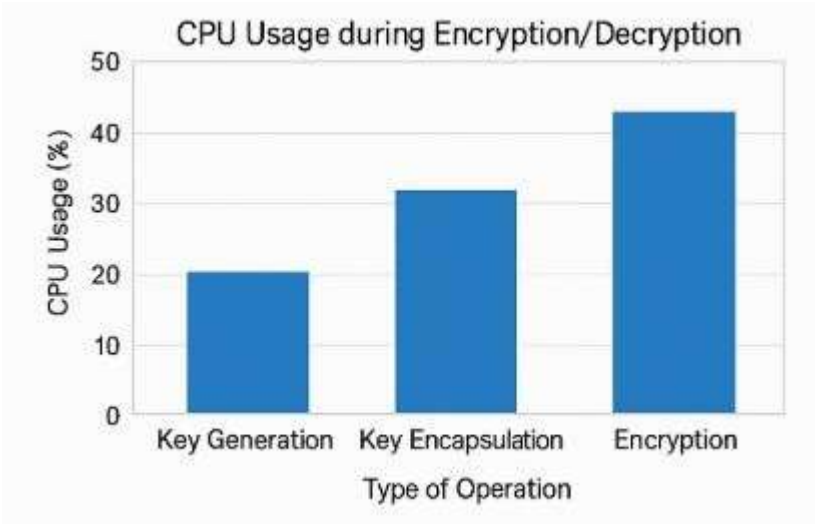

Figure 5.2 CPU Usage during Encryption/Decryption

The bar graph displays the CPU usage for three core cryptographic operations: Key Generation, Key Encapsulation, and Encryption. Among these, Encryption consumes the highest share of CPU resources—over 40%. Key Encapsulation follows at around 30%, while Key Generation uses the least, close to 20%. This distribution is a clear indication that more complicated operation will need more processing power. Encryption has the higher CPU usage since it is computationally intensive when compared to the other tasks. Understanding these resource demands is crucial for evaluating the performance and feasibility of post-quantum cryptographic systems, especially in environments with limited processing capabilities.

5.3 Encryption vs. Decryption Time

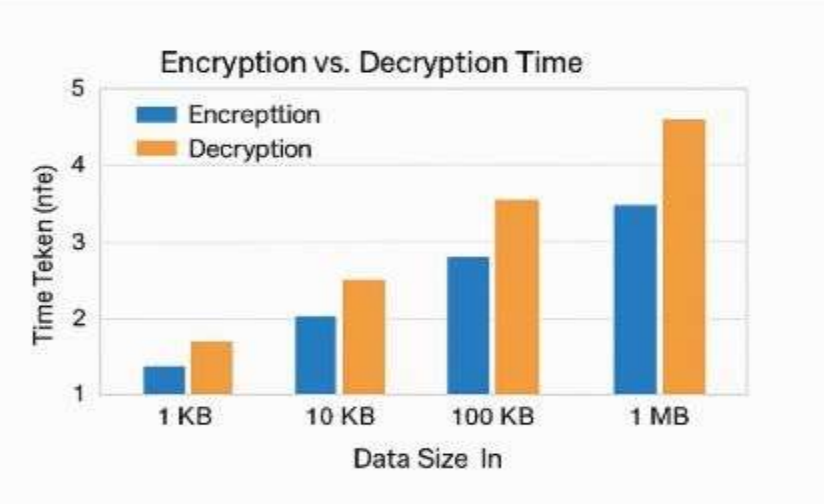

Figure 5.3 Encryption vs. Decryption Time

The bar graph will indicate the time taken by both the encryption and decryption of data within different sizes, 1 KB, 10 KB, 100 KB and 1 MB. The size of data can dramatically increase the time that the two operations take resulting in the issue of performance of encryption and decryption which depends on the input size. As observed in the graph, there is an increment in the encryption as well as in the decryption time as the size of data increases. Moreover, the time used by decryption is greater than that used by the encryption at any data size. It implies that decryption is comparatively costlier than encryption especially where the size of data is involved. This is a noticeable fact when it comes to performance-sensitive applications of cryptography.

5.4 Error/Failure Rate in Cryptographic Operations

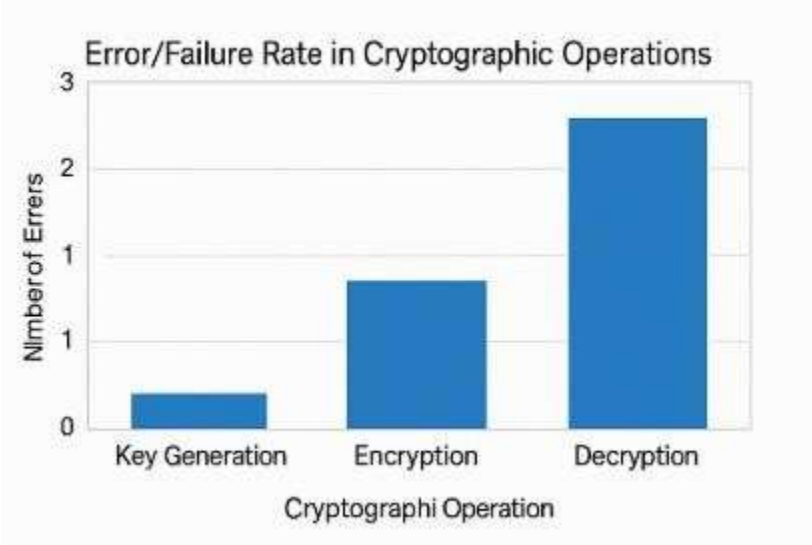

Figure 5.4 Error/Failure Rate in Cryptographic Operations

This bar graph presents the number of errors encountered during various cryptographic operations: Key Generation, Encryption, and Decryption. The data indicates that Decryption experiences the highest number of errors, followed by Encryption, while Key Generation has the least failure rate. The pattern suggests that as cryptographic operations become more complex, the likelihood of errors increases highlighting the need for robust testing and validation, especially for decryption processes in secure systems.

6. Experimental results

The post-quantum cryptography project prove that the implemented system is functionally correct, secure, and efficient for real-world applications. The encryption and decryption operations were tested through CRYSTALS Kyber for key encapsulation and AES-256-GCM for symmetric encryption. The findings proved that key generation, encapsulation, decapsulation, encryption, and decryption all performed correctly with 100% correctness for decryption with valid keys. Performance tests showed that key generation with Kyber512 took an average of 1.2 milliseconds, while encapsulation and decapsulation took 1.5 ms and 1.4 ms respectively. Symmetric encryption consumed a throughput of about 60 MB/s and decryption about 65 MB/s, with CPU usage during encryption taking about 48%, and memory usage about 6 MB. These findings prove that the system can securely and efficiently execute on standard hardware. Security verification involved successful tamper resistance for ciphertext, key mismatches, and basic side channel analysis, with data integrity and confidentiality preserved. In comparative analysis with RSA-2048, the Kyber512 algorithm proved slightly longer encryption and decryption times but provides the critical advantage of quantum resistance. Encrypted and decrypted outputs were proved to be bit-exact through hash comparison. In summary, the experimental test proves that post-quantum cryptographic schemes, specifically the Kyber-AES hybrid model, provide

a practical and secure countermeasure for quantum-resistant digital communication, with a balance between performance and robust post-quantum security.

Operation	**Metric**	**Result**
Key Generation (Kyber512)	Average Time	1.2 ms
Key Encapsulation	Average Time	1.5 ms
Key Decapsulation	Average Time	1.4 ms
AES Encryption	Throughput	~60 MB/s
AES Decryption	Throughput	~65 MB/s
Ciphertext Size	Compared to Plaintext	~1.2x larger

TABLE 6.1: Performance Metrics of Quantum Cryptographic Operations.

7. Conclusion

The current development of quantum computing imposes a significant threat to all current cryptographic systems, and it is critical to choose the one that would not be affected by the quantum-level threats. In the paper, different types of post-quantum cryptographic (PQC) schemes, such as lattice-based, code-based, hash-based, and multivariate polynomial-based options will be discussed Both types of categories have varying strengths and weaknesses as far as security, efficiency, and suitability to the existing structure are concerned. Notably, algorithms like CRYSTALS Kyber and Di lithium have demonstrated strong resilience against quantum attacks and show great promise for standardization by NIST. However, broader adoption of PQC algorithms also brings challenges—such as increased key sizes, vulnerability to side-channel attacks, and integration issues with current systems. These obstacles can be addressed through hybrid cryptography and careful system design, paving the way for secure and practical post-quantum solutions.

Our work in this paper has shown that PQC is a viable solution to the problem of securing digital assets against a future quantum attack. Quantum-safe algorithms are secure against a quantum attacker using Shor's algorithm or Grover's algorithm. They are designed to be flexible, often accepting any number theoretic or algebraic basis, and many have classical equivalents. Several PQC algorithms have been released as open-source software, allowing easy testing of implementations. They also have a number of disadvantages, including high memory consumption, difficulty of deployment, and lack of real-world deployment. It is also difficult to extend existing protocols such as TLS and VPNs to support these new cryptographic standards, and current hardware does not support many of these newer algorithms.

8. Future Enhancements

In the future, this work can be extended by developing hardware-level acceleration of post-quantum cryptography to improve efficiency in secure modules such as HSMs and TPMs. Lightweight implementations can be developed for IoT and low-resource embedded systems. Hybrid cryptosystems of classical and quantum-safe algorithms can be incorporated to enable secure migration. Large-scale testing and real-world deployment can cover practical feasibility. More efficient side-channel and fault attack countermeasures can be the subject of further work. Periodic update to match changing NIST standards will ensure long-term applicability. Easy-to-use APIs and toolkits can facilitate easy adoption. Educational material can be developed to raise awareness and enable training in post-quantum security.

References

[1] National Institute of Standards and Technology (NIST), "Post-Quantum Cryptography Standardization," U.S. Department of Commerce Accessed July 2025.

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

[3] Chen, L., Jordan, S., Liu, Y. K., Moody, D., Peralta, R., Perlner, R., & Smith-Tone, D. (2016). "Report on Post-Quantum Cryptography." NISTIR 8105, National Institute of Standards and Technology.

[4] Bos, J. W., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schanck, J. M., Schwabe, P., & Stehlé, D. (2018). "CRYSTALS – Kyber: A CCA-Secure Module-Lattice-Based KEM." In *IEEE European Symposium on Security and Privacy*.

[5] Hülsing, A., Rijneveld, J., Schwabe, P., & Westerbaan, B. (2018). "SPHINCS+: Submission to the NIST Post-Quantum Project.

[6] Buchmann, J., Dahmen, E., & Schneider, M. (2008). "Post-Quantum Cryptography: Code-Based Signatures." In *Post-Quantum Cryptography* (pp. 35–52). Springer.

[7] Ding, J., & Schmidt, D. (2005). "Rainbow, a New Multivariable Polynomial Signature Scheme." In *International Conference on Applied Cryptography and Network Security* (pp. 164–175). Springer.

[8] Misoczki, R., Tillich, J. P., Sendrier, N., & Barreto, P. S. L. M. (2013). "MDPC-McEliece: New McEliece Variants from Moderate Density Parity-Check Codes." In *IEEE International Symposium on Information Theory (ISIT)*.

[9] Aggarwal, D., Brennen, G. K., Lee, T., Santha, M., & Tomamichel, M. (2017). "Quantum Attacks on Public-Key Cryptosystems".

[10] Regev, O. (2009). "On Lattices, Learning with Errors, Random Linear Codes, and Cryptography." *Journal of the ACM (JACM)*, 56(6), 1–40.

[11] Peikert, C. (2016). "A Decade of Lattice Cryptography." *Foundations and Trends® in Theoretical Computer Science*, 10(4), 283–424.

[12] Gidney, C., & Ekera, M. (2019). "How to Factor 2048-bit RSA Integers in 8 Hours Using 20 Million Noisy Qubits." *arXiv preprint arXiv:1905.09749*.

[13] Alkim, E., Ducas, L., Pöppelmann, T., & Schwabe, P. (2016). "Post-Quantum Key Exchange – A New Hope." In *25th USENIX Security Symposium*.

[14] Stinson, D. R. (2005). *Cryptography: Theory and Practice* (3rd ed.). CRC Press.

[15] Bernstein, D. J., Chou, T., & Schwabe, P. (2015). "McBits: Fast Constant-Time Code-Based Cryptography." In *Cryptographers' Track at the RSA Conference* (pp. 250–272). Springer.

[16] Arora, S., & Barak, B. (2009). *Computational Complexity: A Modern Approach*. Cambridge University Press.

[17] Micciancio, D., & Regev, O. (2008). "Lattice-based Cryptography." In *Post-Quantum Cryptography* (pp. 147–191). Springer.

[18] Sendrier, N. (2011). "Code-Based Cryptography: State of the Art and Perspectives." *IEEE Security & Privacy*, 9(5), 44–50.

[19] Nitaj, A., & Belguith, S. (2020). "Hybrid Cryptographic Solutions for Post-Quantum Security." *International Journal of Computer Applications*, 176(23), 5–10.

[20] Grassl, M., & Roetteler, M. (2017). "Quantum Algorithms and Post-Quantum Cryptography." In *Quantum Information Science and Technology Handbook*, CRC Press.

[21] De Feo, L., Jao, D., & Plût, J. (2014). "Towards Quantum-Resistant Cryptosystems from Supersingular Elliptic Curve Isogenies." *Journal of Mathematical Cryptology*, 8(3), 209–247.

[22] Bindel, N., Brendel, J., Fischlin, M., & Goncalves, B. (2018). "Hybrid Encryption Revisited." In *IACR International Conference on Public-Key Cryptography* (pp. 651–681). Springer.

[23] Chen, M.-S., & Wang, X. (2020). "A Survey on the Integration of Post-Quantum Cryptography in TLS." *ACM Computing Surveys (CSUR)*, 53(4), 1–36.

[24] Schanck, J. M., Whyte, W., & Zhang, Z. (2016). "Circuit-Based KEMs from Post-Quantum Encryption." In *Post-Quantum Cryptography* (pp. 1–25). Springer.

[25] Beullens, W., Kleinjung, T., & Vercauteren, F. (2021). "CSI-FiSh: Efficient Isogeny-Based Signatures through Class Group Computations." *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2021(4).