

Mathematical Hardness Assumptions in Modern Cryptography: A Comparative Survey from Classical Number Theory to Post-Quantum Security

Yash Soni

Abstract

Modern cryptography is based on the premise that certain mathematical problems are extremely difficult for computers to solve. This premise - the "hardness assumptions" - underpins secure communication on the internet, in banking, and the authentication of our digital selves. Traditional public-key cryptographic techniques such as RSA and Diffie-Hellman are built on number-theoretic problems such as factoring numbers into primes and discrete logarithms. They are considered to be "hard" for ordinary - "classical" - computers to solve. However, strong quantum algorithms reveal these to be easy problems for a sufficiently powerful large-scale quantum computer, thereby jeopardizing these systems.

The secondary study examined principal mathematical hardness assumptions that underpin modern cryptography. We compared classical assumptions with post-quantum ones in post-quantum signature, key establishment, and encryption schemes such as code-based, hash-based, isogeny-based, lattice-based, and multivariate. We studied their theoretical and mathematical bases, security strengths, computational efficiencies, and defenses against quantum attacks, with specific attention devoted to NIST selection (Kyber, Dilithium, Falcon, and SPHINCS+). This analysis found that although classical assumptions are mature and widely used, they were susceptible to quantum attacks, while several of the post-quantum assumptions seem resistant but are not so mature and less well explored.

We also discussed challenges that must be addressed when deploying and using the chosen post-quantum systems for infrastructure migration.

Keywords: Cryptography, Mathematical Hardness Assumptions, Computational Complexity, RSA, Discrete Logarithm Problem, Lattice-Based Cryptography, Post-Quantum Cryptography.

Chapter 1. Introduction

1.1 Background

Cryptography is the study of concealing and protecting data from unauthorized use, alteration, and interception. This field, primarily developed in and after the 1970s, makes use of both mathematics and computer science to create schemes for encoding messages, signing digital documents, and securely distributing cryptographic keys.

1.2 Research Context

Public-key cryptography enables two parties who are separated by an insecure channel (network, channel, etc.) to communicate securely and reliably without first exchanging any secrets. This is only possible thanks to hardness assumptions which can be efficiently computed in one direction, but for which

computing in the reverse direction without the proper secret is very computationally difficult (hard). The advent of quantum computers represents an even greater security concern: quantum algorithms like Shor's algorithm are able to break such computationally hard problems (integer factorization and discrete logarithms), meaning they can break RSA and many other popular cryptographic constructions, given access to a sufficiently powerful quantum computer.

1.3 Problem Statement

In the case of traditional hardness assumptions such as discrete logarithms and factorization, we believe that these hardness assumptions are safe against actual attackers. However, quantum computers pose a problem for many crypto-systems and, should it become possible to build quantum computers at scale, security for both the government, enterprises, and individuals may well come crashing down as much crypto may well become useless. Thus, a systematic comparative review, analyzing the mathematical foundations of hardness assumptions from classical and postquantum, and examining their security against attacks from classical and quantum will come in handy.

1.4 Research Objectives

In summary, this secondary research has four main objectives: review the main mathematical hardness assumptions behind modern cryptosystems. Compare the mathematical and security foundations of these assumptions. Assess the strength of each assumption under classical and quantum adversarial power; and Research current, newly proposed postquantum mathematical hardness assumptions and their standardisation procedures.

1.5 Research Questions

The study focuses on three central research questions (RQs):

RQ1: What mathematical hardness assumptions form the basis of modern public-key cryptography?

RQ2: How do classical hardness assumptions compare with post-quantum alternatives?

RQ3: Which assumptions currently offer the strongest long-term security guarantees against both classical and quantum attackers?

1.6 Scope and Delimitations

Here, we survey schemes based on public keys and signatures that rely on a hardness assumption in one of the following areas: number theory, lattices, error-correcting codes, hash functions, multivariate polynomials, or isogenies. We do not discuss any symmetric cryptography unless relevant for context. We cover the literature and standards from around 1976 up to 2026, including significant postquantum standardization efforts. Proof techniques and very specific cryptanalytic attacks are skimmed down so that the content should be understandable for students in the 12th grade.

1.7 Organization of the Paper

The paper is structured in nine chapters as follows: Chapter 2 describes the methodology applied in this paper. Chapter 3 explains the mathematical preliminaries and general principles such as complexity classes, one-way functions, etc. Chapter 4 states traditional hardness assumptions of problems such as the discrete logarithm and integer factorization. Chapter 5 enumerates classic cryptanalytic algorithms and discusses their limitations. Chapter 6 proposes postquantum hard problems (lattice and code-based) as building blocks of cryptographic systems. Chapter 7 presents a multidimensional comparative review. Chapter 8 provides information on standardization activities and industrial considerations. Chapter 9 points out the research gaps and suggests possible future research directions, and Chapter 10 gives the conclusion.

Chapter 2. Methodology

2.1 Nature of the Study

This is a secondary research paper that does not suggest any new algorithms or experiments but gathers, summarizes, and compares numerous works. The paper approach uses a narrative and comparative survey method, where we explain in words and we systematically compare different hypotheses based on standard parameters.

2.2 Literature Search Strategy

Relevant literature is drawn from major digital libraries and repositories:

IEEE Xplore, ACM Digital Library, SpringerLink, ScienceDirect, and Wiley for journal and conference papers.

arXiv and the IACR ePrint Archive for preprints and cryptography research reports.

NIST publications and status reports for post-quantum standardization.

Search terms include “hardness assumption”, “RSA”, “discrete logarithm”, “lattice-based cryptography”, “Learning With Errors”, “post-quantum cryptography”, and “quantum attacks”.

2.3 Inclusion Criteria

Sources included in this survey are:

Peer-reviewed journal articles and conference proceedings in cryptography and information security.

Standardization documents from NIST and related bodies.

Seminal cryptographic papers and books describing core assumptions and schemes.

Publications roughly between 1976–2026, covering both classical and post-quantum perspectives.

2.4 Exclusion Criteria

The study excludes:

Non-technical blog posts and marketing materials unless used only to illustrate deployment trends.

Very short notes without enough technical content.

Systems that do not clearly state the underlying hardness assumption.

2.5 Data Extraction Strategy

From each selected source, the following information is extracted:

Mathematical basis (number theory, lattice theory, coding theory, etc.).

Specific security assumption (e.g., “integer factorization is hard”).

Cryptographic application (encryption, signatures, key exchange).

Known classical and quantum attacks.

Quantum vulnerability and current belief about resistance.

Current adoption and standardization status (e.g., NIST-selected algorithms).

2.6 Framework for Comparative Analysis

To compare assumptions, the paper uses parameters such as:

Computational hardness (how difficult the underlying problem is believed to be).

Existence and quality of formal security proofs and reductions.

Efficiency (runtime, memory) and scalability in practice.

Key and signature sizes, which affect bandwidth and storage.

Quantum resistance (how vulnerable or resistant they are to known quantum algorithms).

Standardization and deployment status (e.g., PQC standards by NIST).

Chapter 3. Mathematical Foundations of Cryptographic Hardness

3.1 Computational Complexity Theory

Complexity theory is the study of how much time and space an algorithm needs as a function of input size. Problems that can be solved in polynomial time (time proportional to some polynomial in the size of the input) are called efficiently solvable problems.

Hardness assumptions often involve problems that are believed to be NP-hard or at least not solvable in polynomial time by any known algorithm. Cryptographers care about average-case difficulty (random instances are hard) and not just worst-case difficulty.

3.2 One-Way Functions

A one-way function is easy to compute in one direction, but is believed to be very hard to invert without special information. For example, it is easy to multiply two large prime numbers together but hard to factor back into primes.

Most public-key schemes are based on the construction of one-way functions from some underlying mathematical problems and the assumption that there is no efficient algorithm for inverting the functions for typical inputs.

3.3 Trapdoor Functions

A trapdoor function is a special kind of one-way function that is easy to invert if you know a secret piece of information called the “trapdoor”. RSA is an example: anybody can encrypt with the public key, but only the one having the private key (trapdoor) can decrypt efficiently.

Trapdoor functions enable authorised parties to invert public-key encryption and signatures, while preventing unauthorised from doing so without the hard problem.

3.4 Security Reductions

A security reduction is a mathematical proof that shows that if you can break a cryptographic scheme then you can solve some underlying hard problem. This provides a formal reduction of the security of the scheme to the hardness assumption.

Good reductions boost confidence in the scheme. If we believe that the hard problem cannot be solved efficiently, then we believe that the scheme is secure against that class of attacks.

3.5 Mathematical Properties Required for Secure Cryptography

Secure cryptographic constructions need problems that are:

- Hard on average, not just in rare pathological cases.

- Structurally rich enough to build encryption and signatures.

- Stable under small perturbations (e.g., adding noise, as in lattice-based schemes).

They must also resist the best known classical and quantum algorithms and be amenable to efficient implementations.

Chapter 4. Classical Mathematical Hardness Assumptions

4.1 Integer Factorization Problem

The integer factorisation problem is as follows: given a large integer NN that is the product of two large primes pp and qq , can we efficiently recover pp and qq ? For general large integers, no classical algorithm is known that runs in polynomial time.

Applications: The basic security assumption of the RSA cryptosystem is based on the difficulty of factoring. This is the problem at the heart of RSA encryption, signatures, and key transport in many TLS deployments.

Pros: Factorisation has been studied for centuries, which gives strong confidence in the hardness of the problem on classical computers.

Limitations: Shor's quantum algorithm can factorise large integers in polynomial time on a scalable quantum computer, thus breaking RSA and related schemes.

4.2 Discrete Logarithm Problem (DLP)

The discrete logarithm problem is: given a cyclic group with generator g and element h , find the exponent x such that $g^x = h$. In many groups used in cryptography (for example, multiplicative groups modulo a prime), no efficient classical algorithm is known for large parameters.

Uses:

Diffie-Hellman key exchange.

ElGamal Cryptosystem.

DSA (Digital Signature Algorithm).

In addition, Shor's algorithm can efficiently solve DLP on a quantum computer, which is a threat to these systems.

4.3 Elliptic Curve Discrete Logarithm Problem (ECDLP)

Elliptic curves define groups of points with a special addition law. The elliptic curve discrete logarithm problem is similar to DLP but in elliptic curve groups: given points P and $Q = kP$, find the scalar k .

Applications: Elliptic Curve Cryptography (ECC), including ECDH (Elliptic Curve Diffie-Hellman) and ECDSA (Elliptic Curve Digital Signature Algorithm).

ECC offers similar security with much smaller key sizes compared to RSA, because the best known classical attacks are slower for ECDLP than for integer factorization. However, quantum algorithms also threaten ECDLP in principle.

4.4 Diffie-Hellman Assumptions

Several assumptions are related to Diffie-Hellman:

Computational Diffie-Hellman (CDH): Given g^a and g^b , compute g^{ab} .

Decisional Diffie-Hellman (DDH): Distinguish between random quadruples and ones of the form (g, g^a, g^b, g^{ab}) .

Gap Diffie-Hellman (GDH): Situations where DDH is easy but CDH remains hard.

These assumptions support many key exchange and encryption schemes, but are again vulnerable to Shor-type quantum attacks on the underlying DLP.

4.5 Comparative Summary of Classical Assumptions

Mathematical properties like factorizations and discrete logarithms, which can be solved by quantum computers very efficiently, are considered a weakness of classical assumptions, which are mostly based on number theory and group theory with a lot of practical usage.

Chapter 5. Cryptanalysis of Classical Hardness Assumptions

5.1 Classical Cryptanalytic Algorithms

Several algorithms deal with the factorization and discrete logarithm problem:

- Pollard's Rho and Baby-Step Giant-Step discrete logarithm algorithms that work with some groups.

- Quadratic sieve and Number Field Sieve factorization techniques, and the latter (NFS) is the most efficient classical algorithm for factoring a large integer.

Typically, parameter values make those methods not work out in polynomial time at all, but the time factor stays between those of sub-exponential algorithms. This way, RSA key lengths as well as DLP key sizes are based on these estimates.

5.2 Advances in Computational Power

Classical computing power has increased a lot over the years thanks mostly to advancements in semiconductor technologies like Moore's law and parallel computing as well as GPU and cloud usage. With these developments, adversaries can now perform very elaborate and complex crypto-breaking attacks even on a large scale, which will soon reduce the effectiveness of legacy algorithms that use small keys.

Due to this, a reasonable level of security demands bigger keys (e.g., RSA 3072 bits for about 128-bit security these days) - that's going to be more expensive with both computations and data transfer.

5.3 Security Parameter Evolution

Over the years, security parameters like modulus sizes and group orders have also changed to ensure adequate protection against ever-evolving attacks. The standards organizations are continuously reviewing and changing key size recommendations based on the recent cryptographic knowledge and the available computing power.

The increase in key size is not sufficient against quantum attacks; these problems can be decomposed by quantum computers within polynomial time independent of the parameter sizes, given that the quantum computer is powerful enough.

5.4 Limitations of Classical Assumptions

The main limitations of classical assumptions are:

- Vulnerability to quantum algorithms like Shor's.

- Growing computational and bandwidth costs as key sizes increase.

- Dependence on specific number-theoretic structures that may hide unforeseen weaknesses.

These limitations motivate the exploration of alternative assumptions that are believed to resist both classical and quantum attacks.

Chapter 6. Post-Quantum Mathematical Hardness Assumptions

6.1 Quantum Threat Model

The quantum threat model says that attackers might one day have access to quantum computers of a scale big enough, capable of running Shor's algorithm and other quantum algorithms. Based on that scenario, the cryptosystems relying on factorization and discrete logarithm problems are not secure, so new assumptions have to be made so that the problems behind them will not be vulnerable to any quantum algorithm that is currently known to be efficient.

6.2 Lattice-Based Hardness Assumptions

Lattices are essentially regularly spaced points in multidimensional space. Experts say lattice problems like the Shortest Vector Problem (SVP) and Closest Vector Problem (CVP) are still difficult for quantum computers.

Learning With Errors and Ring-LWE form fundamental assumptions: given linear equations with random noise, the task of finding the hidden variables is considered hard. Lattice-based post-quantum schemes

such as Kyber (for key encapsulation) and Dilithium and Falcon (for digital signatures), which rely on LWE-type assumptions.

These schemes have good speed and reasonably small key and signature size for achieving post-quantum security.

6.3 Code-Based Hardness Assumptions

Code-based cryptography involves the use of various error-correcting codes and the decoding problem, such as the Syndrome Decoding Problem, which involves decoding a random, linear code that is heavily corrupted by errors. The well-known McEliece cryptosystem is based on such ideas and is currently thought not to be susceptible to quantum computing.

Code-based systems normally feature very big public keys, but their running time may be quite fast.

6.4 Hash-Based Assumptions

Hash-based signing technology uses one-way cryptographic hash functions and common structures (e.g., a Merkle tree) in its design. For example, the cryptographic signature scheme SPHINCS+ considers a cryptographic hash only secure (in the sense that one party cannot guess the other party's message or sign it without being detected) if it meets certain properties like the "random oracle" model or being "collision resistant" to the fullest extent possible.

Stateless (i.e., you can sign and verify messages without worrying at all about the state of the protocol), Hash-based signatures provide strong quantum resistance against quantum attacks, but they result in longer signatures and slow sign/verify operations that lattice-based schemes can handle more efficiently.

6.5 Multivariate Polynomial Assumptions

Multivariate cryptography works with systems of multivariate polynomial equations over finite fields. Solving a random set of multivariate quadratic equations (MQ problem) is a notoriously difficult problem. It is this hardness that has given it a prominent role in the construction of signature and encryption schemes.

Although these schemes can be quite fast and efficient, one of the drawbacks is the size of the public keys, which can be enormous. Besides, some versions have been cracked by very intelligent algebraic attacks, and this shows that these schemes require good care and attention in their design, just like any good system.

6.6 Isogeny-Based Assumptions

Isogeny-based cryptography draws on the mathematical challenge of discovering isogenies, which are kinds of transformations that preserve the structure between elliptic curves. This kind of problem is considered difficult enough to keep the confidentiality of data even against powerful adversaries or quantum computing, but there have been instances where specific schemes like SIKE were compromised by such new attacks.

The reason why isogeny-based cryptosystems are considered is that they are capable of delivering the encryption of high level security with very small sizes, but in reality they are relatively new; people do not believe that their security is at that level, as for example the lattice-based systems.

6.7 Comparative Summary of Post-Quantum Assumptions

PQ assumptions rely on diverse mathematical domains and are selected to prevent efficient quantum algorithms from being applied. Among the schemes based on lattices and codes are the leading candidates, which are going to be standardized by NIST. Kyber, Dilithium, Falcon, SPHINCS+, and HQC have either been selected or are currently under consideration for standardization. Their chief worry is that long-term trust shall be assured and that the world's people have them at hand in daily use.

Chapter 7. Comparative Analysis

7.1 Comparison Based on Mathematical Foundations

To put it differently, classical assumptions are based on integer (number theory), modular arithmetic, and elliptic curves.

If one family of algorithms in post-quantum cryptography is broken, for instance, in the future, then other families may keep security. This is a kind of risk spreading.

7.2 Comparison Based on Security Guarantees

Many traditional cryptographic schemes have experienced decades of analyses of their properties and weaknesses by cryptanalysts and have stood up to classical attacks with confidence in their soundness, but at the heart of their structure, there is Shor's algorithm, which can break them efficiently. However, post-quantum schemes are less mature, but some have proofs of security which assume the difficulty of solving certain problems at the very worst, as well as have a solid foundation of hardness evidence coming from complexity theory (in particular, lattice-based systems). It is also thought that these schemes could be resilient even against quantum computers.

7.3 Comparison Based on Computational Efficiency

In terms of real examples, those lattice-based designs named Kyber and Dilithium are really fast, and their efficiency is comparable with or even sometimes better than RSA when they offer the same or highest level of security. On the contrary, Code-based and hash-based schemes might not be as efficient; they either can be slower or can require more bandwidth, but probably they will still be good enough for a large variety of applications.

Picking up on performance and efficiency, you can see that it is a fact that it largely depends on the hardware being used, how they are implemented, and finally, what they are being used for. For example, embedded devices versus servers.

7.4 Comparison Based on Quantum Resistance

Roughly, schemes factor numbers or calculate logs (discrete logarithms) to work. With quantum computers, Shor's algorithm allows factoring and solving discrete logarithms efficiently. The reason why post-quantum schemes are taken into consideration is that no efficient quantum algorithm is known for solving the problems behind each one of the schemes. That is, while the old schemes rely upon either factoring or discrete logarithms, a fact which puts their security in jeopardy in a context of a quantum computer capable of large-scale operations, as Shor's algorithm is a quantum algorithm which can do factoring and discrete logarithms efficiently. Post-quantum schemes, then again, are chosen with the sole consideration of the underlying problems being such that their solution cannot be found efficiently even by a quantum computer.

7.5 Comparison Based on Practical Deployment

RSA encryption and Elliptic Curve Cryptography (ECC) are extensively used in the present time for secure network communication using Transport Layer Security (TLS), Virtual Private Networks (VPNs), and other data transfer protocols. Yet, there are plans to transition from the use of these classical cryptographic algorithms to post-quantum cryptographic algorithms. In fact, quantum computers capable of breaking these algorithms are in the near term. NIST has set out its guidelines and standards for the post-quantum algorithms that are the basis of Kyber, Dilithium, and SPHINCS+, among others. The cryptographic organization is currently working towards Falcon and HQC, and these are going to be an integral part of future cryptographic deployments.

The adoption of these technologies is happening sooner in government entities, financial services companies, as well as certain cloud computing providers that often make use of schemes that blend classical and post-quantum technologies.

7.6 Comparative Synthesis

The following table summarizes key aspects:

Hardness Assumption	Mathematical Domain	Classical Security	Quantum Security	Typical Applications	Current Status
Integer Factorization (RSA)	Number theory	Strong, well-studied	Broken by Shor's algorithm	Encryption, signatures, key transport	Widely deployed, being phased out.
Discrete Logarithm (DLP/ECDLP)	Groups, elliptic curves	Strong, well-studied	Broken by Shor's algorithm	Key exchange, signatures	Widely deployed, quantum-vulnerable.
LWE / Ring-LWE (Kyber, Dilithium)	Lattice theory	Strong evidence, reductions	Believed resistant, no known attacks	PQC encryption, signatures	NIST-standardized or selected by CSRC.
Syndrome Decoding (McEliece, HQC)	Coding theory	Long-studied, robust	Believed resistant	PQC encryption	Selected; large keys.
Hash-based (SPHINCS+)	Hash functions, trees	Strong if hash is secure	Believed resistant	PQC signatures	NIST-standardized; larger signatures.
Multivariate MQ	Multivariate polynomials	Mixed; some schemes broken	Believed resistant for good designs	Signatures, encryption (research)	Active research; limited standardization
Isogeny problems	Elliptic curve isogenies	Emerging, some breaks	Unclear; some schemes attacked	Experimental key exchange	Research stage after SIKE break.

Chapter 8. Standardization and Emerging Trends

8.1 NIST Post-Quantum Cryptography Standardization

NIST spearheaded a multi-stage international effort for the selection and standardization of post-quantum algorithms. NIST released in 2024 standards for algorithms based on CRYSTALS-Kyber, CRYSTALS-Dilithium, and SPHINCS+, with Falcon and HQC as the other two candidates.

Such standards provide the much-needed clarity to industry and government sectors on post-quantum schemes.

8.2 Industry Adoption

Governments, financial institutions, cloud providers, and secure communication services are starting to experiment and implement quantum-resistant post-quantum algorithms, in most cases side by side with traditional cryptographic mechanisms.

Blockchain and IoT are also areas where researchers are looking into post-quantum cryptography as potential tools for protecting valuable information stored on the blockchain and on devices.

8.3 Migration Challenges

Challenges include:

- Upgrading protocols and software stacks (like TLS) worldwide.
- Handling large keys and signatures in code-based and hash-based schemes.
- Ensuring interoperability and avoiding performance regressions.

Organizations must inventory where classical cryptography is used and plan gradual migration strategies.

8.4 Hybrid Cryptographic Systems

Sure, a practical way of protecting confidentiality in the quantum era could be to rely on cryptography of classical algorithms together with post-quantum ones. This way, if at least one component remains secure, the encryption is safe. Earlier incarnations of "post-quantum TLS" adopted a hybrid approach consisting of both key exchange and digital signatures to help in the transition.

Chapter 9. Research Gaps and Future Directions

9.1 Current Research Gaps

Despite rapid progress, there are gaps:

- Limited long-term empirical data on post-quantum schemes in large-scale deployments.
- Need for more refined analyses of side-channel and implementation attacks on new schemes.

9.2 Open Mathematical Problems

Several mathematical questions remain open:

- Tight hardness results for average-case lattice problems.
- Better understanding of multivariate and isogeny problems under both classical and quantum attacks.

Solving these could strengthen or weaken current assumptions.

9.3 Challenges in Practical Deployment

Real-world deployment must balance security, efficiency, usability, and compatibility:

- Some schemes have big keys or signatures, stressing bandwidth and storage.
- Legacy hardware and protocols may not easily support new operations.
- Training engineers and updating standards takes time.

9.4 Future Directions in Cryptographic Hardness Assumptions

Future work may include:

Designing new assumptions with stronger complexity-theoretic foundations.
Combining multiple assumptions in layered or hybrid designs.
Continuous monitoring and updating of recommended algorithms as new attacks and hardware appear.

Chapter 10. Conclusion

10.1 Summary of Findings

This study demonstrated a heavy dependence of modern cryptographic solutions on mathematics in the form of hardness assumptions. Public-key cryptography Mainly has been mainly built upon these assumptions like the difficulty of factoring integers and solving discrete logarithms. Even though these classical assumptions have allowed a secure channel of communication for a long time, their use is being questioned in the quantum computing era.

Quantum-resistant cryptographic problems like lattice-, code-, hash-, multivariate-, and isogeny-based problems can act as excellent substitutes and are currently being standardized through the effort of crypto communities and even implemented by them.

10.2 Comparative Insights

Classical assumptions are mature and widely deployed but offer weak long-term guarantees in a world with quantum adversaries. Post-quantum assumptions are younger but designed to resist known quantum algorithms, with lattice- and hash-based schemes currently leading standardization.

10.3 Implications for Future Cryptographic Systems

Future cryptographic solutions should no longer be only based on factorization and discrete logarithms but should also include post-quantum algorithms, which help in the protection of long-lived data and infrastructure. Standards such as NIST provide us with a set of the strongest encryption schemes.

Advances in mathematics as a field, the creation of quantum algorithms, the security of their implementation, and the study thereof will remain important research topics.

10.4 Concluding Remarks

Mathematical hardness assumptions are like the invisible legs upon which digital security has been standing all this time. With the rapid development in technologies, mainly the advent of quantum computing, the cryptographic basis will need to be totally reshaped for continuing trust. The main thing will be to have a range of assumptions as a safeguard and move forward with post-quantum cryptography for the long run of secure data.

References

1. Rivest, R. L., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2), 120–126.
2. Diffie, W., & Hellman, M. (1976). New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6), 644–654.
3. Koblitz, N. (1987). Elliptic curve cryptosystems. *Mathematics of Computation*, 48(177), 203–209.
4. Miller, V. S. (1986). Use of elliptic curves in cryptography. In *Advances in Cryptology – CRYPTO '85* (pp. 417–426). Springer.
5. Shor, P. W. (1997). Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5), 1484–1509.

6. Ajtai, M. (1996). Generating hard instances of lattice problems. In *Proceedings of the 28th ACM Symposium on Theory of Computing* (pp. 99–108). ACM.
7. Regev, O. (2009). On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM*, 56(6), 34.
8. NIST. (2024). *PQC Standardization Process: FIPS 203, FIPS 204, and FIPS 205*. National Institute of Standards and Technology.
9. NIST. (2022). *Status report on the third round of the NIST Post-Quantum Cryptography standardization process*(NIST IR 8413).
10. Bos, J. W., Costello, C., Ducas, L., et al. (2018). CRYSTALS-Kyber: A CCA-secure module-lattice-based KEM. In *Proceedings of the 2018 IEEE European Symposium on Security and Privacy*.
11. Ducas, L., Durmus, A., Lepoint, T., & Lyubashevsky, V. (2013). Lattice signatures and Bimodal Gaussians. In *Advances in Cryptology – CRYPTO 2013*.
12. Bernstein, D. J., Hülsing, A., et al. (2019). SPHINCS+: Submission to the NIST post-quantum cryptography project.
13. Bernstein, D. J., Lange, T. (2017). Post-quantum cryptography. *Nature*, 549(7671), 188–194.
14. Mosca, M. (2018). Cybersecurity in an era with quantum computers: Will we be ready? *IEEE Security & Privacy*, 16(5), 38–41.
15. Fluhrer, S., et al. (2020). A comprehensive survey on post-quantum TLS. *IACR Cryptology ePrint Archive*.
16. Rosenthal, E., & Schanck, J. M. (2021). Post-quantum cryptography: Computational-hardness assumptions and beyond. *IACR ePrint Archive 2021/571*.
17. IET. (2022). Quantum algorithms for attacking hardness assumptions in classical and post-quantum cryptography. *IET Information Security*, 16(4), 258–274.
18. McEliece, R. J. (1978). A public-key cryptosystem based on algebraic coding theory. *DSN Progress Report*, 42(44), 114–116.
19. Castryck, W., & Decru, T. (2023). An efficient key recovery attack on SIDH. *Advances in Cryptology – EUROCRYPT 2023*, 423–447.
20. Peikert, C. (2016). A decade of lattice cryptography. *Foundations and Trends® in Theoretical Computer Science*, 10(4), 283–400.
21. Merkle, R. C. (1989). A certified digital signature. *Conference on the Theory and Application of Cryptology*, 218–238.