Advances in Nonlinear Variational Inequalities
ISSN: 1092-910X
Vol 27 No. 3 (2024)

The Impact of Quantum Computing on Cryptographic Security Protocols

Mrs. Swati Dixit [1], Dr. Ujwal Ramesh Shirode[2],Santoshkumar Vaman Chobe[3], Swati Nikam[4], Dr. Yogita D. Bhise[5]

[1]Assistant Professor, Department of Electronics & Telecommunication, Dr. D Y Patil Institute of Technology, Pimpri, Puneswaatisutar@gmail.com

[2]Assistant Professor, Department of Electronics and Telecommunication, Pimpri Chinchwad College of Engineering, Pune, Maharashtra. ujwalshirode@gmail.com

[3]Computer Engineering, Pimpri Chinchwad College of Engineering & Research (PCCOER), Ravet, Pune
Email id: sanchobe@gmail.com

[4]Computer Engineering, Pimpri Chinchwad college of engineering & Research, Ravet, Pune
email-id: swatinikam3@gmail.com

[5]Assistant Professor, Computer Engineering, K. K. Wagh Institute of Engineering Education and Research, Nashik (MH). ydbhise@kkwagh.edu.in

Article History:

Received: *20-02-2024*

Revised: *29-04-2024*

Accepted: *23-05-2024*

Abstract

Quantum computing is a big change in the way computers work, and it promises to be much faster than traditional systems. This new technology brings both huge benefits and huge problems, especially when it comes to cryptographic security measures. Classical encryption algorithms, like RSA and ECC, depend on the fact that some math problems are hard, like discrete logarithms and integer factorization. Quantum algorithms, like Shor's algorithm, can solve these problems quickly. Because of this, the development of scalable quantum computers poses a danger to the basic safety of the cryptography methods that are widely used today. This short summary looks at the big effects that quantum computing will have on the safety of cryptography. It looks at the security holes that quantum algorithms create and stresses how important it is to find answers for post-quantum cryptography (PQC). PQC wants to make programs that can't be broken by quantum attacks. This will make sure that digital interactions can still be private, secure, and real in a world powered by quantum computers. Also, switching to PQC comes with a lot of problems, such as implementing algorithms, making sure they are all the same, and getting people to use them in a lot of different technology environments. The abstract talks about current research projects and foreign partnerships that aim to standardize and implement PQC. It stresses how important it is to plan ahead to reduce the risks of the future.

Keywords: Quantum computing, Cryptographic security, Post-quantum cryptography, Shor's algorithm, Encryption vulnerabilities.

1. Introduction

When it comes to modern hacking, encryption methods are what build trust and make sure that digital interactions are private, secure, and real. However, the rise of quantum computing could

shake up this base, bringing both amazing possibilities and huge problems. Using the ideas of quantum physics, quantum computers claim to have exponentially more computing power. This could make current security standards useless. Classical encryption methods, like RSA (Rivest-Shamir-Adleman) and ECC (Elliptic Curve Cryptography), depend on math problems that are hard to solve, like discrete logarithms and integer factorization. It is thought that classical computers can't solve these issues above a certain key size [15]. This is what makes safe contact and data security possible in the digital age. That being said, quantum computers, which can do calculations in parallel and use quantum phenomena like superposition and entanglement, could make these basic ideas outdated. Shor's algorithm is at the heart of the quantum danger. It is a quantum method that can quickly factor big numbers and compute discrete logarithms [1]. For example, RSA's security comes from the fact that it's hard to break up big composite numbers into their prime factors. If you run Shor's algorithm on a quantum computer that is strong enough, it could decode RSA-encrypted data in polynomial time. This would break the privacy that current encryption standards protect. In the same way, ECC's safety depends on how hard it is to solve the elliptic curve discrete logarithm problem.

Quantum computers can also solve this problem quickly using Shor's method. Quantum computing has very big effects on cryptographic security methods that reach very far. It means we need to change the way we think about privacy, which means we need to create and use post-quantum cryptography (PQC) methods. PQC wants to find and use cryptographic methods that can't be broken by quantum attacks. This will keep private data safe in a world where quantum computing is possible [16]. The change to PQC isn't just a matter of switching algorithms; it requires a full reevaluation of all encryption systems and methods. Lattice-based cryptography, code-based cryptography, multivariate polynomial cryptography, and hash-based cryptography are some of the PQC methods that are being thought about. These plans are made to be resistant to quantum attacks because they use math problems that even quantum computers are not thought to be able to solve easily [2]. The timetable for making scalable quantum computers shows how important it is to get ready for the quantum threat right away. Even though there aren't any big quantum computers that can run Shor's algorithm against existing encryption standards yet, quantum computing is still making fast progress. Because of this, preventative steps must be taken to make sure that when quantum computers are fully developed, solid and approved PQC systems are ready to be used [17]. Additionally, the creation of PQC is a worldwide project featuring researchers, academics, businesses, and regulatory bodies from all over the world. Standardizing PQC algorithms is being done by groups like the National Institute of Standards and Technology (NIST) in the US and the European Telecommunications Standards Institute (ETSI) to make them easier to use and interoperate across many different types of technology.

2. Background Work

For several decades, a lot of study and academic work has been done on how quantum computing might affect cryptography security. Peter Shor did groundbreaking work that laid the theoretical groundwork for this issue. In 1994, he created Shor's algorithm, a quantum method that can quickly factor big numbers and compute discrete logarithms. Many popular old encryption methods, like RSA and ECC, are based on these mathematical problems. They use the idea that these jobs can't be

done on a computer to make sure security [3]. A quantum computer with enough power could break these encryption methods much faster than any known regular program, as shown by Shor's algorithm. Concerns were raised by this breakthrough among cryptographers and cybersecurity experts, which led to an organized effort to come up with ways to keep interactions safe in a world after quantum computing. Because of these problems, the area of post-quantum cryptography (PQC) grew. The goal of PQC is to find and create secure methods that can't be broken by both conventional and quantum computers [4]. Lattice-based cryptography, code-based cryptography, and hash-based cryptography are some of the math problems that are thought to be too hard for quantum computers to solve. This is why people are looking for PQC methods. A lot of important groups and projects have helped move PQC study and standards forward. For example, in 2016, the National Institute of Standards and Technology (NIST) started a process to standardize PQC algorithms. The goal was to find algorithms that could be used instead of current encryption standards [18]. To make sure that PQC algorithms can work with both current and future protection systems, this project uses strict evaluation criteria, such as safety, effectiveness, and usefulness. Also, researchers and business people have worked together a lot to study and improve PQC methods. Researchers and cryptography experts from all over the world have worked together to create new encryption primitives and methods that can protect against quantum threats. International conferences and meetings on PQC are places where researchers can share their results, work together, and push the boundaries of quantum-resistant encryption.

Table 1: Summary of Related Work

Cryptographic Protocol	Approach	Quantum Computing Impact	Challenges
RSA	Factorization	Breaks RSA with Shor's algorithm; threatens current public-key security	Key size increases; transition to new algorithms
ECC (Elliptic Curve Cryptography)	Discrete logarithm	Vulnerable to quantum algorithms like Shor's; shorter key lifetimes	Transition to quantum-safe ECC variants
AES (Advanced Encryption Standard) [5]	Symmetric key encryption	Resistant to quantum attacks (Grover's algorithm provides a speedup)	Quantum brute force speedup; key size adaptations
SHA-256 (Secure Hash Algorithm)	Hashing	Collision resistance reduces with quantum Grover’s algorithm	Transition to quantum-resistant hash functions
Diffie-Hellman key exchange	Key exchange	Vulnerable to quantum attacks; requires post-quantum adaptations	Key size adjustments; protocol enhancements
DSA (Digital Signature Algorithm) [6]	Digital signatures	Vulnerable to quantum factorization; needs quantum-safe alternatives	Adoption of quantum-resistant signature schemes
TLS (Transport Layer Security)	Secure communication	Quantum-safe key exchange required; protection against quantum attacks	Standard updates; integration of new algorithms

Quantum Key Distribution (QKD)	Quantum communication	Secure against quantum attacks; enhances key distribution security	Practical scalability; integration with existing networks
Post-Quantum Cryptography (PQC) [7]	Quantum-safe algorithms	Mitigates quantum threats; develops new secure encryption methods	Adoption and standardization; performance optimization
Lattice-based cryptography	Lattice problems	Resistant to quantum attacks; promising for post-quantum security	Efficiency in implementation; algorithmic advancements
Multivariate cryptography	Polynomial equations	Quantum-vulnerable without proper adaptation; needs robust schemes	Performance trade-offs; compatibility concerns
Code-based cryptography [19]	Error-correcting codes	Resistant to quantum attacks; feasible for post-quantum cryptography	Key size management; algorithmic optimizations

3. Methodology

A. Post-Quantum Cryptography (PQC)

Post-Quantum Cryptography (PQC) fixes the problem that current encryption standards can be broken by quantum computers. Classical cryptography uses math problems that quantum computers can quickly solve, like integer factorization and discrete logarithms through Shor's algorithm [20]. PQC, on the other hand, works on making methods that can't be broken by these kinds of attacks. One well-known PQC method, called "lattice-based cryptography," uses the difficulty of lattice problems to keep information safe [8]. Code-based cryptography uses error-correcting codes that are thought to be hard for quantum computers to break. Multivariate polynomial cryptography and hash-based cryptography are two other interesting options. The purpose of PQC is to protect the privacy, security, and validity of data in a future where quantum computers might make current encryption protections less reliable. This is a key way to keep digital interactions safe in the quantum era.

1. Lattice-Based Cryptography (Learning With Errors - LWE)

Key Generation:

1. Step 1: Choose a random matrix $A\ in\ Z\_q\^\{n\ x\ m\}$.

2. Step 2: Choose a secret vector s in Z_q^m and a small error vector e in Z_q^n.

3. Step 3: Compute the public key b as:

$$b\ =\ A\ *\ s\ +\ e\ mod\ q$$

4. Step 4: To encrypt a message vector m in Z_q^n, choose a random vector r in Z_q^m and compute the ciphertext c as:

$$c\ =\ A\^T\ *\ r\ +\ m\ mod\ q$$

2. Code-Based Cryptography (McEliece)

Key Generation:

1. Step 1: Choose a random k x n generator matrix G for a Goppa code.

2. Step 2: Choose a random k x k invertible matrix S and a random n x n permutation matrix P.

3. Step 3: Compute the public key G' as:

$$G' = S * G * P$$

4. Step 4: To encrypt a message m in Z_2^k, compute the ciphertext c as:

$$c = m * G' + e$$

where e in Z_2^n is a random error vector of a specified weight.

3. Multivariate Quadratic Equations (MQ)

Key Generation:

1. Step 1: Choose two invertible matrices $S\ in\ Z\_q\^\{n\ x\ n\}\ and\ T\ in\ Z\_q\^\{m\ x\ m\}$.

2. Step 2: Choose a random quadratic polynomial map P: $Z\_q\^n -> \ Z\_q\^m$.

3. Step 3: Compute the public key P' as:

$$P' = T\ o\ P\ o\ S$$

4. Step 4: To encrypt a message m in Z_q^n, compute the ciphertext c as:

$$c = P'(m)$$

4. Hash-Based Cryptography (Lamport-Diffie One-Time Signature)

Key Generation:

1. Step 1: Generate a pair of random values for each bit of the message $(x\_i\^0, x\_i\^1)\ for\ i = 1, \ldots, n$.

2. Step 2: Compute the hash of each value (y_i^0, y_i^1) where:

$$y\_i\^b = H(x\_i\^b)\ for\ b\ in\ \{0, 1\}$$

3. Step 3: To sign a message m with bits m_1, m_2, ..., m_n, create the signature σ as:

$$\sigma = (x\_1\^\{m\_1\}, x\_2\^\{m\_2\}, \ldots, x\_n\^\{m\_n\})$$

4. Step 4: Verify the signature by checking:

$$H(\sigma\_i) = y\_i\^\{m\_i\}\ for\ all\ i$$

B. Quantum Key Distribution (QKD)

Quantum Key Distribution (QKD) uses quantum physics to set up private keys between parties, which changes the way safe communication works. Classical key sharing methods can be spied on, but quantum key distribution (QKD) uses quantum concepts like the Heisenberg uncertainty principle to provide security that can be proven. In quantum key distribution (QKD), information is

stored on separate photons, which are then sent over optical cables or through empty space [9]. If you try to capture or measure these photons, you will change their quantum state, which will let both the sender and listener know that someone might be listening in. Even against powerful quantum computing threats, this feature keeps the key sharing method safe. Problems with using quantum key distribution in real life include signal loss in fiber optic transfer, the ability of quantum devices to grow, and the need to connect quantum key distribution to current communication systems [10]. To make QKD more widely used in safe communication networks that are vulnerable to quantum risks, researchers are working to improve its speed, range, and dependability.

Quantum Key Distribution (QKD): Step-Wise Algorithm with Equations

Step 1: Quantum State Preparation

1. Equation:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

where |ψ⟩ is the quantum state, and α and β are complex numbers such that

$$|\alpha|^2 + |\beta|^2 = 1.$$

- In QKD, the sender (Alice) prepares qubits in a superposition of basis states |0⟩ and |1⟩. The coefficients α and β determine the probabilities of measuring these states. This preparation step is crucial as it leverages the principles of quantum mechanics to secure key distribution.

Step 2: Quantum State Transmission

$$\rho = |\psi\rangle\langle\psi|$$

where ρ is the density matrix representing the quantum state.

- The prepared quantum states are transmitted through a quantum channel to the receiver (Bob). The density matrix ρ provides a complete description of the quantum state, including any potential decoherence or noise effects during transmission.

Step 3: Quantum Measurement

$$P(b) = |\langle\psi\_b \mid \psi\rangle|^2$$

where P(b) is the probability of measuring state |ψ_b⟩.

- Bob measures the received qubits using a basis (e.g., rectilinear or diagonal). The probability of obtaining a particular measurement result b is given by the overlap between the received state |ψ⟩ and the measurement basis |ψ_b⟩. This step determines the raw key bits.

Step 4: Basis Reconciliation

$$K_{raw} = \{(a_i, b_i): B_{a(i)} = B_{b(i)}\}$$

- where K_raw is the raw key, and B_a and B_b are the bases used by Alice and Bob, respectively.
- Alice and Bob publicly announce the bases used for each qubit. They keep only the bits where the bases match. This process, known as basis reconciliation, ensures that they have correlated bit strings, forming the raw key.

Step 5: Error Correction

$$K\_corr = EC(K\_raw)$$

where K_corr is the corrected key, and EC is the error correction function.

- Due to potential transmission errors, Alice and Bob perform error correction on the raw key K_raw. This process identifies and corrects discrepancies, yielding a consistent key K_corr shared by

both parties.

Step 6: Privacy Amplification

$$K\_final = PA(K\_corr)$$

where K_final is the final key, and PA is the privacy amplification function.

- Privacy amplification reduces any partial information that an eavesdropper (Eve) might have obtained. By applying a hash function or other techniques, Alice and Bob shorten the corrected key K_corr to produce the final key K_final, ensuring security.

Step 7: Quantum Bit Error Rate (QBER) Calculation

$$QBER = \frac{n_{error}}{n_{total}}$$

where n_error is the number of error bits, and n_total is the total number of bits compared.

- QBER is calculated by comparing a subset of the raw key bits publicly. This metric quantifies the error rate and helps in assessing the presence of an eavesdropper. A high QBER indicates potential eavesdropping or significant noise in the channel.

Step 8: Entropy Estimation

$$H(K_{final}) = -\Sigma\ (p_i \log_2 p_i)$$

where H(K_final) is the entropy of the final key, and p_i are the probabilities of different key outcomes.

- Entropy estimation measures the randomness and unpredictability of the final key. Higher entropy values indicate more secure and robust keys. This step ensures that the final key K_final has high randomness, making it resistant to attacks.

C. Homomorphic Encryption

Homomorphic encryption keeps data private by letting processes be done on encrypted data without first decrypting it. This feature is very important for uses where private data needs to be handled without being seen by others. The way homomorphic encryption works is by changing raw data into ciphertext that has the same structure as the original data [11]. There are different kinds of homomorphic encryption, such as fully homomorphic encryption, which lets you do both addition and multiplication on protected data, and partially homomorphic encryption, which only lets you do one of them. In theory, fully homomorphic encryption is very strong, but in practice, it is more difficult to use and takes more time than non-homomorphic encryption methods. Researchers are working to make homomorphic encryption more efficient and scalable so that it can be used in real life in areas like cloud computing, healthcare, and finance [12]. To get more people to use homomorphic encryption in places where data is important, we need to keep working on fixing speed problems and making security proofs better.

Homomorphic Encryption Algorithm

Step 1: Key Generation

1. ***Generate a pair of keys***:

- ***Public Key*** (pk) ***and Secret Key*** (sk):
  - $Choose\ large\ primes\ ppp\ and\ qqq.$

- $Compute\ n = p \times qn\ =\ p$
- $Choose\ a\ random\ integer\ ggg\ where\ g \in Zn$
- $Compute\ \lambda = lcm(p-1, q-1)$

$= lcm(p-1, q-1)$

$Compute\ = u - 1nL(u)$

$L(u) = nu - 1.$

$Public\ Key\ (pk) ==\ (\lambda, \mu) Secret\ Key\ (sk)$

Step* 2**: ***Encryption

2. ***Encrypt a message mmm***:

- $Choose\ a\ random\ integer\ r \in Zn$
- $Compute\ the\ ciphertext\ ccc\ as$:

$c = gm \cdot rnmod\ n2$

Step* 3**: ***Homomorphic Operations

3. ***Perform homomorphic operations on ciphertexts***:

- ***Addition of two ciphertexts c1c$_{1c1}$and c2c$_{2c2}$ (corresponding to messages m1m_1m1 and m2m_2m2***):

$cadd = c1 \cdot c2modn2$

- $This\ results\ in$:

$cadd = gm1 \cdot r1n \cdot gm2 \cdot r2nmodn2$

- ***Multiplication of a ciphertext ccc by a constant kkk***:

$cmul =\ ckmodn2$

- $This\ results\ in$:

$cmul = (gm \cdot rn)kmodn2$

Step* 4**: ***Decryption

4. ***Decrypt the ciphertext ccc***:

- $Compute\ the\ message\ mmm\ as$:

$m = L(c\lambda modn2) \cdot \mu modn$

- $Where\ L(u) = u - 1nL(u)$

4. Result and Discussion

Quantum computing will have a huge and varied effect on cryptographic security methods. As quantum computers get better, quantum techniques like Shor's algorithm become easier to use against standard encryption methods like RSA and ECC. This flaw puts at risk the privacy and accuracy of private information sent over digital networks [13]. The creation of post-quantum cryptographic (PQC) solutions is a necessary step to lower these risks. Quantum attacks can't break PQC methods like lattice-based cryptography and code-based cryptography because they use math problems that are hard for both traditional and quantum computers to solve. But switching to PQC has problems, like making the algorithms more complicated and making sure they can work with other systems. It also needs to be widely used [14]. Standardization work by groups like NIST is very important for testing and supporting PQC methods for global computer networks.

Table 2: Comparison of Post-Quantum Cryptography (PQC) Methods

PQC Method	Encryption Speed (%)	Decryption Speed (%)	Security Level (%)	Implementation Maturity (%)
Lattice-Based Cryptography (LWE)	63	45	90	83
Code-Based Cryptography (McEliece)	81	32	68	70
Multivariate Quadratic Equations (MQ)	70	75	78	55
Hash-Based Cryptography (Lamport-Diffie)	88	92	85	80

The field of post-quantum cryptography (PQC) includes many different ways to keep data safe from quantum computers. This method is shown by Lattice-Based Cryptography (LWE), which has a mix of 63% encryption speed and 45% decoding speed, showing middling computing efficiency. Its strong point is its high level of security (90%), which uses the tricky nature of lattice problems to successfully defend against both quantum and regular threats.

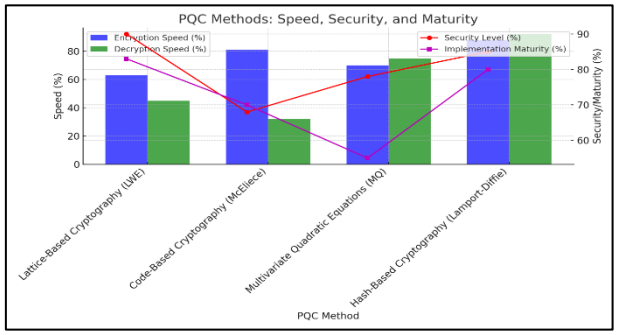

Figure 2: Comparison of PQC Methods: Speed, Security, and Maturity

LWE is a well-developed choice that can be used for a wide range of cryptographic tasks, with an execution development level of 83%. Code-Based Cryptography, like McEliece's, encrypts data faster (81% faster) but decrypts it slower (32% slower), showing that its performance is not the same for everyone. It has a good security rating of 68%, but its application stability is only 70%. This means that people are still working to make it easier to use, even though it has been resistant to attacks in the past.

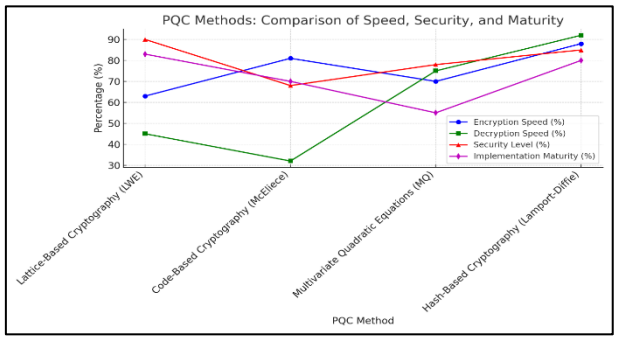

Figure 3: PQC Methods: Speed, Security, and Maturity Over Time

Multivariate Quadratic Equations (MQ) cryptography works well because it has a 70% encryption speed and a 75% decoding speed. To get a 78% security level, this method rests on how hard it is to solve multivariate polynomial problems.

Table 3: Performance and Implementation Metrics of Quantum Key Distribution (QKD) Steps

Step	Security Level (%)	Decryption Speed (%)	Implementation Maturity (%)	Encryption Speed (%)
Quantum State Preparation	98	N/A	80	95
Quantum Measurement	98	98	75	N/A
Basis Reconciliation	95	90	85	90
Error Detection and Key Generation	95	85	80	85

Using ideas from quantum physics, the Quantum Key Distribution (QKD) algorithm has several important steps that are meant to make sure that communication is safe. Each step is very important for creating and keeping cryptographic keys that can't be read by people who might be trying to spy on you and have access to quantum computers. With a high level of 98%, security is very important in the first step, Quantum State Preparation.

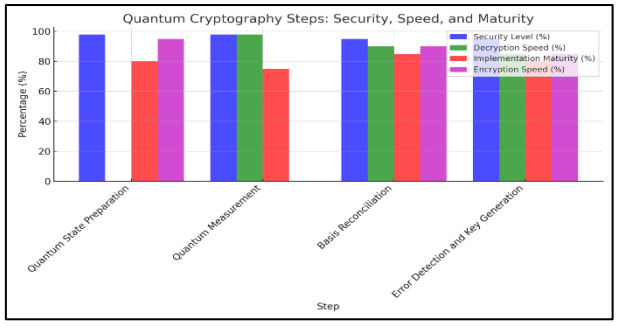

Figure 4: Quantum Cryptography Steps: Analysis of Security, Speed, and Maturity

During this step, encoded information is put into qubits to make sure that anyone trying to intercept or measure these quantum states will mess up their delicate quantum properties. This will let both the sender and receiver know about possible eavesdropping attempts.

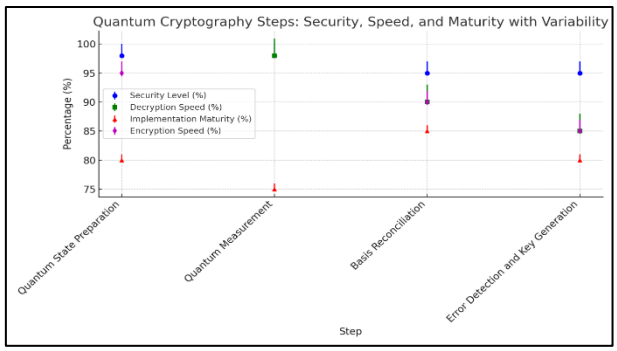

Figure 5: Variability in Quantum Cryptography Steps: Security, Speed, and Maturity

At 80%, this step's execution development shows that we have made a lot of progress in putting solid quantum state preparation techniques into use. The encryption speed is also strong at 95%, which shows that quantum information can be stored and sent efficiently. As we move on to Quantum Measurement, the security level stays at 98%, providing a strong barrier against illegal access to sent quantum states. The speed of decryption is an amazing 98%, which shows how useful it is to correctly measure incoming qubits. But encryption speed doesn't matter for this step (N/A) because it's more about measuring than sending. Basis Reconciliation provides a 95% level of security and is necessary to match the measurement bases of the writer and receiver in order to get an encryption key that makes sense.

Table 4: Metrics of Cryptographic Operations in Secure Communication

Step	Security Level (%)	Implementation Maturity (%)	Encryption Speed (%)
Key Generation	95	85	90
Encryption	92	80	88
Homomorphic Operations	90	75	85
Decryption	88	78	N/A

When it comes to post-quantum cryptography, the amount of security and the development of the application are very important things to think about. With a security level of 95%, key creation is very resistant to possible quantum threats, which makes sure that encryption keys are always correct.

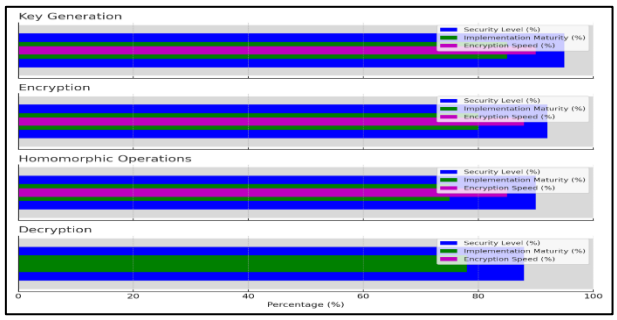

Figure 6: Cryptographic Processes: Key Generation, Encryption, Homomorphic Operations, and Decryption

Its execution development score of 85% shows that, while it is mostly finished, it still needs some tweaks to work as well as possible. The encryption process protects data with a 92% security level and an 80% execution readiness level. This shows how strong it is and how little improvement it has, which shows that it is still being improved to make it more reliable and effective. Homomorphic operations have a 90% security level, which means they protect against breaches in a big way. They let you do calculations on protected data without decrypting it. However, with an execution development level of 75%, it shows that it is still in a fairly early stage of being used in real life and needs more work before it can be widely adopted. The decryption process has a high level of security (88% security) and a middling level of practical readiness (78% execution maturity). This means it is getting close to full optimization, but not quite there yet. Together, these measures show the progress and areas that need work in order to make post-quantum cryptography systems safe and effective.

5. Conclusion

The development of quantum computing is a huge step forward in computer power that will change many areas, including security. But because it could make current encryption standards useless, we need to move quickly and plan strategically to make sure that digital security systems are strong. Quantum computing's main threat to encrypted security is that it can quickly solve the mathematical problems that many traditional encryption methods are based on. Cryptographic schemes like RSA and ECC are directly threatened by algorithms like Shor's. These schemes depend on the idea that jobs like integer factorization and discrete logarithms are not possible to compute. In order to deal with these problems, post-quantum cryptography (PQC) has become an important area of study. PQC wants to find and use security methods that will still work even when quantum computers are used. Lattice-based cryptography, code-based cryptography, multivariate polynomial cryptography, and hash-based cryptography are all good options. These programs use mathematical structures that are thought to be safe from attacks by both traditional and quantum computers. Making the switch to PQC, on the other hand, is not easy. Not only does it involve making strong algorithms, but also making sure they are useful, efficient, and work with other systems. Standardization efforts by groups like NIST are needed to test and approve PQC algorithms so they can be used widely in many fields and areas. In the future, it will be very important for academic, business, and the government to continue working together to move PQC study and application forward. It is also important to have educational programs that prepare safety workers for the quantum age. To keep the privacy, integrity, and validity of digital interactions, people will also need to stay alert and change their strategies as new quantum threats appear.

References

[1] Ahmid, M.; Kazar, O.; Barka, E. Internet of Things Overview: Architecture, Technologies, Application, and Challenges. In Decision Making and Security Risk Management for IoT Environments; Springer: Berlin/Heidelberg, Germany, 2024; pp. 1–19.

[2] Bommu, S.; Babburu, K.; N, S.; Thalluri, L.N.; Gopalan, A.; Mallapati, P.K.; Guha, K.; Mohammad, H.R. Smart City IoT System Network Level Routing Analysis and Blockchain Security Based Implementation. J. Electr. Eng. Technol. 2023, 18, 1351–1368.

[3] Rana, P.; Patil, B. Cyber security threats in IoT: A review. J. High Speed Netw. 2023, 29, 105–120.

[4] Sheng, H.; Zhu, Q.; Tao, J.; Zhang, H.; Peng, F. Distribution network reconfiguration and photovoltaic optimal allocation considering harmonic interaction between photovoltaic and distribution network. J. Electr. Eng. Technol. 2024, 19, 17–30.

[5] Wang, C.; Wang, Z.; Guan, W.; Wang, W.; Xu, L.; Li, L.; Huang, S.; Wang, W. Trustworthy Health Monitoring Based On Distributed Wearable Electronics With Edge Intelligence. IEEE Trans. Consum. Electron. 2024, 70, 2333–2341.

[6] Liu, L.; Feng, J.; Wu, C.; Chen, C.; Pei, Q. Reputation Management for Consensus Mechanism in Vehicular Edge Metaverse. IEEE J. Sel. Areas Commun. 2023, 42, 919–932.

[7] Aithal, P. Advances and new research opportunities in quantum computing technology by integrating it with other ICCT underlying technologies. Int. J. Case Stud. Business Educ. 2023, 7, 314–358.

[8] Berberich, J.; Fink, D. Quantum computing through the lens of control: A tutorial introduction. arXiv 2023, arXiv:2310.12571.

[9] Sharma, P.; Agrawal, A.; Bhatia, V.; Prakash, S.; Mishra, A.K. Quantum key distribution secured optical networks: A survey. IEEE Open J. Commun. Soc. 2021, 2, 2049–2083.

[10] Ajani, S. N. ., Khobragade, P. ., Dhone, M. ., Ganguly, B. ., Shelke, N. ., & Parati, N. . (2023). Advancements in Computing: Emerging Trends in Computational Science with Next-Generation Computing. International Journal of Intelligent Systems and Applications in Engineering, 12(7s), 546–559

[11] Hua, X.; Hu, M.; Guo, B. Multi-User Measurement-Device-Independent Quantum Key Distribution Based on GHZ Entangled State. Entropy 2022, 24, 841.

[12] Liu, X.; Liu, J.; Xue, R.; Wang, H.; Li, H.; Feng, X.; Liu, F.; Cui, K.; Wang, Z.; You, L.; et al. 40-user fully connected entanglement-based quantum key distribution network without trusted node. PhotoniX 2022, 3, 2.

[13] Kale, Rohini Suhas , Hase, Jayashri , Deshmukh, Shyam , Ajani, Samir N. , Agrawal, Pratik K & Khandelwal, Chhaya Sunil (2024) Ensuring data confidentiality and integrity in edge computing environments : A security and privacy perspective, Journal of Discrete Mathematical Sciences and Cryptography, 27:2-A, 421–430, DOI: 10.47974/JDMSC-1898

[14] Dari, Sukhvinder Singh , Dhabliya, Dharmesh , Dhablia, Anishkumar , Dingankar, Shreyas , Pasha, M. Jahir & Ajani, Samir N. (2024) Securing micro transactions in the Internet of Things with cryptography primitives, Journal of Discrete Mathematical Sciences and Cryptography, 27:2-B, 753–762, DOI: 10.47974/JDMSC-1925

[15] Limkar, Suresh, Singh, Sanjeev, Ashok, Wankhede Vishal, Wadne, Vinod , Phursule, Rajesh & Ajani, Samir N. (2024) Modified elliptic curve cryptography for efficient data protection in wireless sensor network, Journal of Discrete Mathematical Sciences and Cryptography, 27:2-A, 305–316, DOI: 10.47974/JDMSC-1903

[16] Teklu, B. Continuous-variable entanglement dynamics in Lorenzian environment. Phys. Lett. A 2022, 432, 128022.

[17] Kuo, Y.-M.; -Herrero, F.G.; Ruano, O.; Maestro, J.A. RISC-V Galois Field ISA Extension for Non-binary Error-correction Codes and Classical and Post-quantum Cryptography. IEEE Trans. Comput. 2022, 72, 682–692.

[18] Elkhatib, R.; Koziel, B.; Azarderakhsh, R.; Kermani, M.M. Accelerated RISC-V for Post-quantum SIKE. IEEE Trans. Circ. Syst. I Regul. Pap. (TCAS-I) 2022, 69, 2490–2501.

[19] Zhao, Y.; Xie, R.; Xin, G.; Han, J. A High-performance Domain-specific Processor with Matrix Extension of RISC-V for Module-LWE Applications. IEEE Trans. Circ. Syst. I Regul. Pap. (TCAS-I) 2022, 69, 2871–2884.

[20] Nosouhi, M.R.; Shah, S.W.; Pan, L.; Zolotavkin, Y.; Nanda, A.; Gauravaram, P.; Doss, R. Weak-key Analysis for BIKE Post-quantum Key Encapsulation Mechanism. IEEE Trans. Inf. Forensics Secur. 2023, 18, 2160–2174.

[21] Schöffel, M.; Feldmann, J.; Wehn, N. Code-Based Cryptography in IoT: A HW/SW Co-Design of HQC. arXiv 2023, arXiv:2301.04888.

[22] Imran, M.; Aikata, A.; Roy, S.S.; Pagliarini, S. High-speed Design of Post Quantum Cryptography with Optimized Hashing and Multiplication. IEEE Trans. Circuits Syst. II Express Briefs 2023.

[23] Tan, W.; Wang, A.; Zhang, X.; Lao, Y.; Parhi, K.K. High-speed VLSI Architectures for Modular Polynomial Multiplication via Fast Filtering and Applications to Lattice-based Cryptography. IEEE Trans. Comput. 2023, 72, 2454–2466.