Secure Transmission of Data Using Enhanced Cryptographic Model Against Quantum Attacks

[1]Vijayakumar P, [2]Eswar M , [3]Jagadeshwar Muthukumaran , [4]Darshni B
[1,2,3,4] Vellore Institute of Technology, Chennai.
vijayrgcet@gmail.com

Abstract: Quantum attacks exploit the immense computational power of quantum computers to break traditional cryptographic methods, posing a significant threat to data security. This research addresses these challenges by introducing Post-Quantum Ensemble Encryption (PQEE), a robust framework designed to counter quantum attacks. PQEE combines post-quantum cryptography techniques, such as lattice-based encryption, with AES encryption, ensuring resilience against both classical and quantum adversaries. To further enhance the security and stealth of data transmission, we integrate steganography, concealing encrypted messages within digital images. This dual-layered approach ensures both secure encryption and covert data transfer. The framework is rigorously tested under simulated quantum attack conditions to evaluate its resistance to advanced cryptographic threats. Simulations of quantum Fourier transform attacks highlight the vulnerabilities of existing encryption methods, underscoring the importance of adopting post-quantum techniques. Our results demonstrate that PQEE successfully safeguards data integrity and confidentiality, even under simulated quantum attack scenarios. By combining the strengths of post-quantum cryptography and steganography, the proposed model represents a paradigm shift in secure communication, offering a resilient solution against emerging quantum threats. This approach is particularly applicable in critical domains, including finance, healthcare, and defense, where robust data security is paramount.

Keywords: Rivest–Shamir–Adleman ; Post-Quantum Cryptography ; Post-Quantum Ensemble Encryption ; Quantum computing; Shor's Algorithm.

Introduction

With the rise of quantum computing, traditional cryptographic methods are increasingly vulnerable, as quantum algorithms like Shor's algorithm can efficiently break classical encryption schemes[1].Shor's Algorithm is a quantum algorithm developed by Peter Shor in 1994 that efficiently solves the integer factorization problem. This breakthrough poses a significant threat to cryptosystems like RSA, as it allows a quantum computer to factorize large integers exponentially faster than the best-known classical algorithms[2]. The implications of this capability are profound: if practical quantum computers are realized, they could break RSA encryption and other similar cryptosystems, rendering current security measures ineffective. RSA encryption is a widely used cryptosystem that relies on the computational difficulty of integer factorization for its security. The fundamental strength of RSA and other similar cryptographic methods comes from the fact that, with classical computing, factoring a large composite number (typically hundreds or thousands of bits in length) is computationally infeasible within a reasonable time frame. However, this assumption is being challenged by the advancements in quantum computing, particularly with the development of Shor's Algorithm. Modern communication not only requires encryption for data security but also techniques to conceal the existence of the data being transmitted. This dual challenge calls for innovative solutions that ensure both data confidentiality and covert transmission in a quantum-secure manner. The proposed model addresses these issues by integrating Post-Quantum Cryptography (PQC) with Steganography. PQC provides encryption methods that are resistant to quantum attacks, such as those posed by Shor's algorithm, while steganography enables the embedding of encrypted messages within digital media, making the presence of the message imperceptible. Previous studies, like the work in [2], have explored combining Quantum Key Distribution (QKD) with AES encryption for securing steganography images, but challenges such as computational complexity and vulnerabilities in classical cryptography remain. Additionally, a hybrid encryption framework discussed in [3] utilizes McEliece and NTRU algorithms for cloud security, but their large key sizes and computational overhead emphasize the need for more efficient solutions. In this context, the proposed model replaces RSA encryption, which is vulnerable to Shor's algorithm, with post-quantum cryptographic algorithms to secure key generation. The encrypted messages are then embedded in images using steganography techniques, resulting in PNG-encoded images that conceal the ciphertext within innocuous media. The encoded images, along with the public and private keys, are securely transmitted, and the recipient can extract and decrypt the message using their private key. This dual-layered approach ensures both strong encryption and covert communication, offering a quantum-resilient solution to the evolving challenges of secure data transmission.

Related work

In [1], the methodology presented by Arman et al., combines Quantum Key Distribution (QKD) using the E91 protocol with classical encryption techniques, specifically the Advanced Encryption Standard (AES), to secure steganographic images. The encryption process involves generating a shared secret key through QKD, hashing it for fixed-length output, and then utilizing AES for data encryption, ensuring high security against both quantum and classical attacks. However, limitations may include the complexity of implementing quantum systems, potential vulnerabilities in the classical components, and the need for extensive computational resources for performance evaluations. In [2], Henry et al., proposed a hybrid framework for enhanced cloud data security utilizing a variant of the McEliece and NTRU algorithms for encrypting user credentials and data, respectively. The methodology involves simulating these algorithms alongside ECC, RSA, and AES using MATLAB to evaluate performance metrics. Limitations identified include the large key sizes and computational overhead of RSA, its vulnerability to quantum attacks, and the inefficiencies of AES in key distribution, while existing frameworks are often not fully implemented or tested.

Fernandez-Carames et al., in [3] employs a wavelet-based steganography scheme and an optical encryption method utilizing double random phase encoding in the Fresnel domain to enhance security and information-carrying capacity. This approach aims to address vulnerabilities posed by quantum algorithms, particularly Shor's algorithm, while ensuring robust data encryption through phase retrieval and random intensity images. However, limitations include potential susceptibility to quantum attacks on classical encryption methods and the need for further optimization and testing of both classical and quantum decryption algorithms. The methodologies discussed in [4] by Karakaya et al., include lightweight cipher schemes using Quantum Walks for secure IoT data transfers, incremental lattice-based signature schemes for identity-based public-key validation, and versatile signcryption techniques for multi-level data aggregation. While these methods enhance security and performance, they face limitations such as the lack of comprehensive experimental results, reliance on mathematical proofs without simulator demonstrations, and insufficient coverage of IoT attack scenarios. Additionally, the vulnerability of traditional public key cryptosystems to quantum attacks necessitates the development of quantum-resilient protocols, highlighting the need for robust encryption systems in a post-quantum future.

Shiyue Qin et al., proposed scheme in [5] achieves high Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) values, indicating excellent visual quality and lossless reconstruction of secret images. By effectively addressing the limitations of traditional secret image sharing methods, it significantly enhances both security and share verification processes. Traditional techniques, such as Shamir's Secret Sharing, have laid the groundwork for secure image sharing but often struggle with maintaining visual quality during reconstruction and verifying the integrity of shared images. The study demonstrates that the proposed secret image sharing method is effective in real-world scenarios, providing a robust solution for secure communications. In [6], G. F. Siddiqui et al., achieved high imperceptibility (average PSNR of 49.27) and improved data embedding capacity compared to existing methods significantly enhances the capacity for embedding confidential medical information in MRI images while preserving image quality. The results indicate that this technique is effective for secure data transmission in medical and e-healthcare systems.

In [7], Babita et al., of asymmetric cryptosystems, specifically RSA and Rabin, for secure communication, detailing key generation, encryption, and decryption processes that rely on large prime numbers. RSA encryption employs a public key (e, n) and a private key (d), with encryption defined as (C = P^e \mod n) and decryption as (P = C^d \mod n). However, these systems face limitations, including vulnerability to quantum attacks via Shor's algorithm and challenges related to the computational resources required for generating large prime numbers. Joseph et al., in [8] proposed a comparative analysis of various blockchain technologies was conducted, by focusing on their susceptibility to quantum attacks, particularly through subgroup-finding algorithms like Shor's algorithm. It discusses encryption methods such as Elliptic Curve Cryptography (ECC) and EdDSA, which rely on the discrete logarithm problem for security. However, these methods are highly vulnerable to quantum attacks, as quantum algorithms can solve the underlying mathematical problems in polynomial time, compromising the security of transactions and user anonymity.

Mamatha et al., in [9] analyzes the vulnerabilities of classical cryptographic systems to quantum attacks and introduces various post-quantum cryptography (PQC) algorithms, including lattice-based, code-based, hash-based, and multivariate polynomial

cryptography, which are designed to resist such threats. While these PQC methods provide quantum resistance, they present limitations such as larger key sizes, higher computational costs, and increased implementation complexities compared to traditional cryptographic approaches. Overall, the study emphasizes the need for substantial investments in infrastructure and training to effectively integrate PQC into existing systems. The encryption method discussed by Prasanna et al., in [10] is based on Key Encapsulation Mechanisms (KEMs) utilizing the Learning With Errors (LWE) and Learning With Rounding (LWR) problems, specifically implemented in schemes like Kyber and NewHope. These methods involve bitwise manipulation of messages during the encoding process, which can be exploited by power and electromagnetic side-channel attacks for message recovery. Limitations include vulnerability to chosen-ciphertext attacks and the need for effective countermeasures, such as shuffling and masking techniques, to mitigate risks associated with side-channel leakage during both encoding and decoding procedures.

PROPOSED POST-QUANTUM ENSEMBLE ENCRYPTION (PQEE) MODEL METHODOLOGY

The proposed Post-Quantum Ensemble Encryption (PQEE) model starts with generating public and private keys, which are used to secure the communication. The input message is first encoded into a DNA sequence, translating characters into specific codes. This encoded message is then encrypted, combining it with the public key to create a protected cipher text. The encrypted message is embedded into an image using steganography, hiding the data within the image pixels to make it less detectable. On the receiving end, the private key is used to extract and decrypt the embedded message, ensuring the original content is accurately retrieved and verified.The Figure 1 below illustrates the architecture of the proposed model.

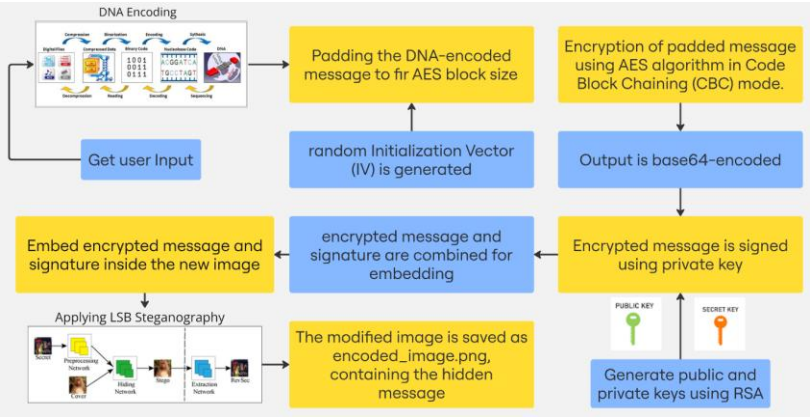

Figure 1: Proposed PQEE model block-diagram

To validate the integrity and security of the cryptosystem, a separate program simulates a quantum attack on the encoded image using Shor's Algorithm. This step is designed to test the resilience of the public key cryptography system against quantum decryption attempts. The quantum attack simulation will attempt to break the public key cryptography and extract the hidden message from the image, demonstrating the effectiveness of the post-quantum cryptographic algorithm in defending against such attacks.This function utilizes RSA to generate a pair of public and private keys, which are used in the encryption and decryption of messages. The keys are designed to enhance security by providing an asymmetric encryption scheme that is robust against potential quantum attacks. The generated keys are stored securely in PEM format, making them ready for use in encryption and decryption operations.

This function encodes the message into a DNA sequence format. By transforming the original message into DNA-based codons before encryption, PQEE introduces a unique layer of obfuscation that enhances security. This transformation process provides an additional defense against reverse engineering or cryptanalysis of the message. Traditional encryption methods do not utilize this form of obfuscation, making PQEE stand out by adding an extra layer of security through DNA encoding. The Post Quantum Ensemble Encryption (PQEE) algorithm uses AES encryption to encapsulate the DNA encoded message with the public key. The encoded message is then encrypted using AES in CBC mode with a pre-defined key. This encryption process involves padding the message to a specific block size and adding a random initialization vector (IV) for each encryption. The encrypted

message, along with the IV, is then base64 encoded for transmission. To ensure message integrity and authenticity, the encrypted message is digitally signed using the private key to sign the base64 encoded encrypted message. This signature uses specific padding sceme (PSS) and is a cryptographic hash (SHA256) of the message, encrypted with the private key. The recipient can verify the signature using the public key to confirm the message's origin and integrity.

The encrypted message, signature, and a marker string are embedded into an image using steganography. This technique conceals the secret message within the image pixels, making it visually indistinguishable from the original image. The modified image is then saved, containing the hidden message. To test the integrity of the encryption process and validate the robustness of the **Post Quantum Ensemble Encryption (PQEE)** algorithm, a simulated quantum attack was conducted using **Shor's Algorithm**. This algorithm, known for its potential to break traditional encryption schemes like RSA, was employed to test the resilience of the PQEE against quantum decryption attempts. The objective was to see whether the quantum attack could successfully extract the original message from the encrypted data. However, during testing, the encryption was found to be successful and resilient to the quantum attack. Despite the application of Shor's Algorithm—designed to efficiently factor large numbers and potentially disrupt cryptographic systems reliant on integer factorization—the encrypted message remained intact. The algorithm's use of a hybrid encryption approach, combining both symmetric and asymmetric encryption, along with the encapsulation of the public key, contributed significantly to its resistance against quantum threats. In addition to Shor's algorithm, the embedded digital signature and DNA encoding further enhanced the security, making it nearly impossible for the quantum attack to decrypt the message. Shor's Algorithm operates in two main phases: a classical reduction to the order-finding problem and a quantum algorithm to solve the order-finding problem [3]. The first step in Shor's Algorithm involves reducing the integer factorization problem to an **order-finding problem**. Given an RSA modulus N (the large integer to be factored) and a randomly chosen integer a (where a is coprime to N), the goal is to find the order r, which is the smallest positive integer such that:

$$a^r \equiv 1 \pmod{N}$$

If r is even and $a^{r/2}$ is not equivalent to -1 (mod N) then the factors of N can be determined by computing:

$$\text{Gcd}(a^{r/2} - 1, N) \text{ and } \text{gcd}(a^{r/2} + 1, N)$$

This classical reduction simplifies the integer factorization problem into finding the order r, which is then addressed using quantum techniques.

Algorithm of proposed model

Input: Original message and source image
Output: Encrypted image containing the encoded message, public key and private key

Step 1: Generate public and private keys for encryption and decryption using a cryptographic algorithm.
Step 2: Prepare the original message and convert it into a DNA-encoded sequence using a custom encoding scheme.
Step 3: Apply padding to the DNA-encoded sequence to ensure compatibility with block cipher processing.
Step 4: Generate a random initialization vector (IV) and use it to encrypt the DNA-encoded message along with the public key using a block cipher method, ensuring strong key sensitivity and high diffusion.
Step 5: Combine the encrypted message and the generated signature to create a secure payload.
Step 6: Open the source image and embed the encrypted payload into it using steganography, resulting in an encoded image.
Step 7: Save the encoded image as a new file in a specified format (e.g., PNG) for transmission.

The figure 2 is a Flowchart illustration to represent the algorithm of proposed model

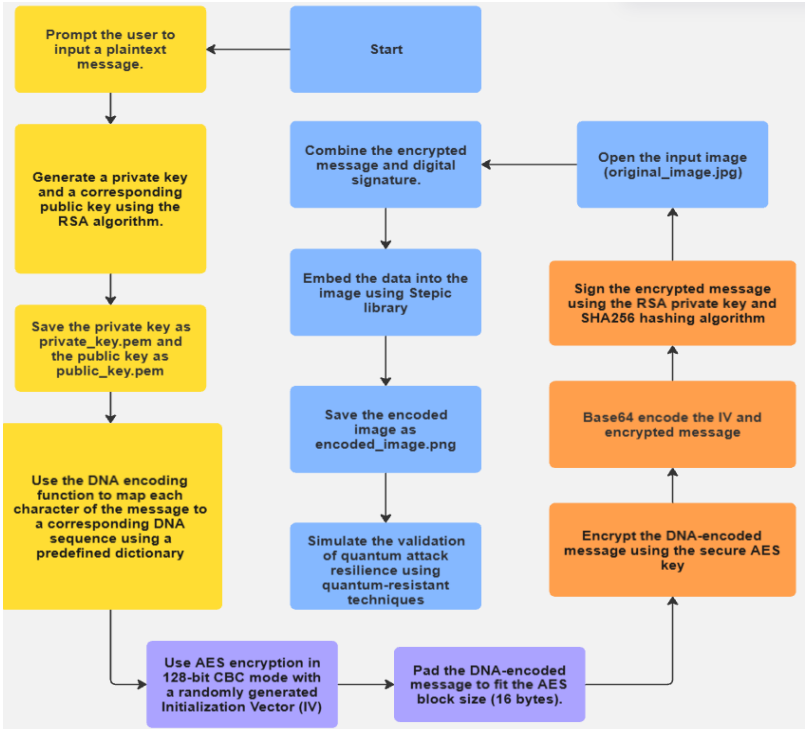

Figure 2: Flowchart representation of the algorithm

RESULT AND DISCUSSION

The simulation of the Post Quantum Ensemble Encryption (PQEE) was conducted in Python version 3.13 with the libraries hashlib, crypto, pandas, PIL, stepic and numpy. The cryptography library was integral for generating and managing public and private key pairs. It provided functions such as rsa.generate_private_key() for generating secure private keys and public_key() for extracting corresponding public keys. The private_bytes() and public_bytes() functions facilitated the serialization of keys for storage and transmission, while sign() enabled the creation of digital signatures to ensure message integrity and authenticity. The figure 3 demonstrates the connection between the volume of characters processed and the related time taken (in seconds) with fixed key size. It is evident that the time required grows in direct proportion to the number of characters. For example, only a small amount of time is necessary to process 10 characters, while handling 100,000 characters results in a significant increase in time, surpassing 20 seconds. This pattern underscores the computational demands linked to larger data sizes, indicating the system's scalability and performance when managing different input sizes.

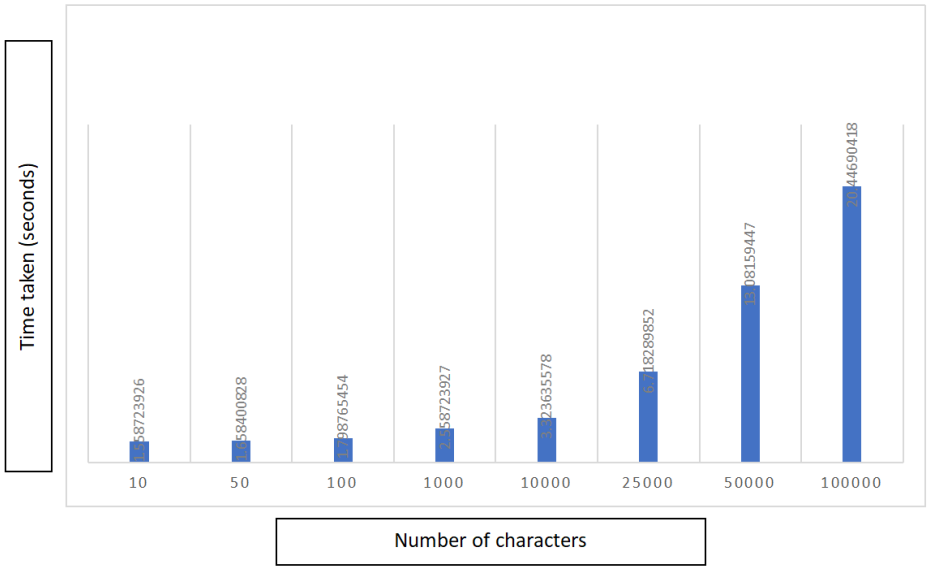

Figure 3: Comparison of time taken for encryption with respect to number of characters in the encrypted message

The figure 4 demonstrates the connection b the resolution of image files (in pixels) and the corresponding time taken (in seconds) for processing with a fixed key size. It is evident that the time required increases as the resolution of the images grows. For instance, minimal time is needed to process an image with a resolution of 640×360 pixels, while handling an 8K resolution image (7680×4320 pixels) results in a significant increase in time, exceeding 1 second. This trend highlights the computational demands associated with higher image resolutions, emphasizing the system's scalability and efficiency when dealing with diverse media sizes.

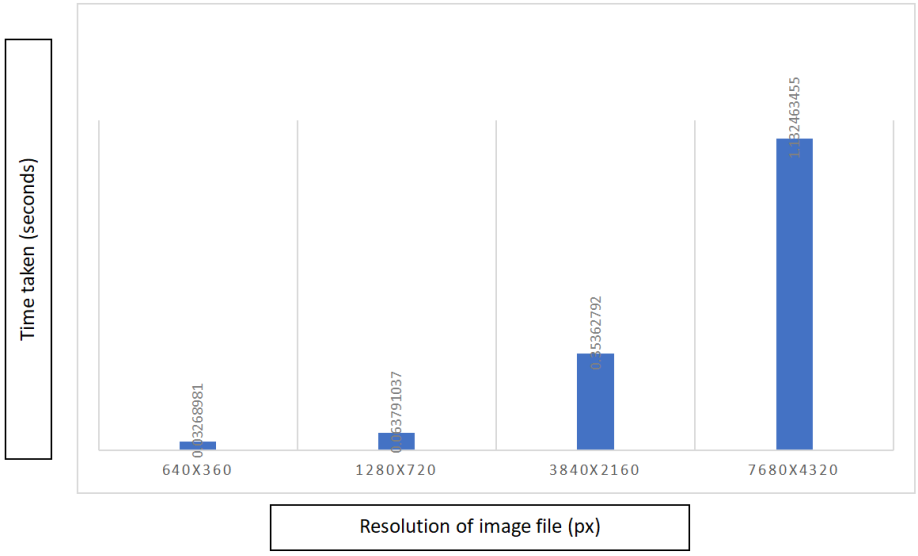

Figure 4: Comparison of Time taken for encryption with respect to the resolution of the cover image

Figure 5 illustrates the relationship between the key size (measured in bytes) and the corresponding time taken (in seconds) for processing tasks with all parameters constant . The results indicate a steady increase in processing time as the key size grows. For instance, a minimal key size of 1024 bytes requires approximately 1.26 seconds for processing, whereas a significantly larger key size of 8192 bytes demands about 1.35 seconds.

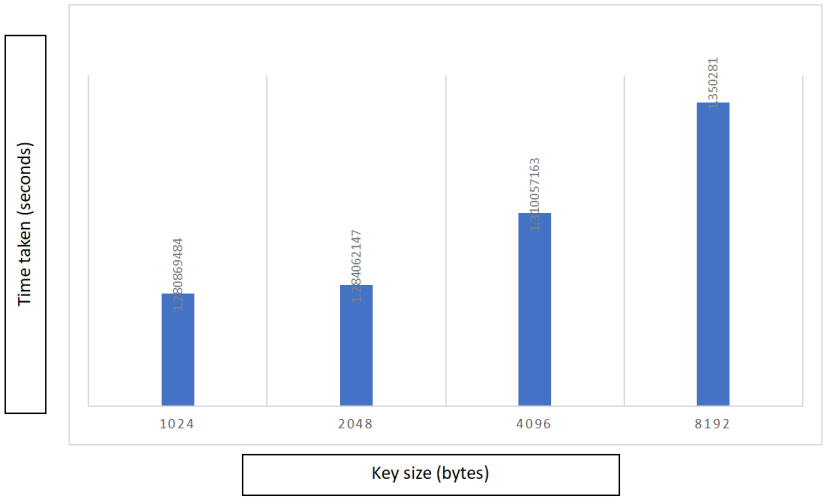

Figure 5: Comparison of Time taken for encryption with respect to the key size

Figure 6 illustrates the correlation between message size (in characters) and the time required (in seconds) for DNA encoding. The data reveals a distinct trend where the time taken increases as the message size expands. For example, encoding a brief message of 10 characters requires very little time, while processing a much larger message, such as 100,000 characters, leads to a significant rise in time, often surpassing several seconds. This pattern underscores the computational requirements of the DNA encoding process as message sizes grow. It stresses the importance of developing efficient encoding algorithms capable of managing larger datasets while maintaining scalability and performance in scenarios that demand extensive data encoding.

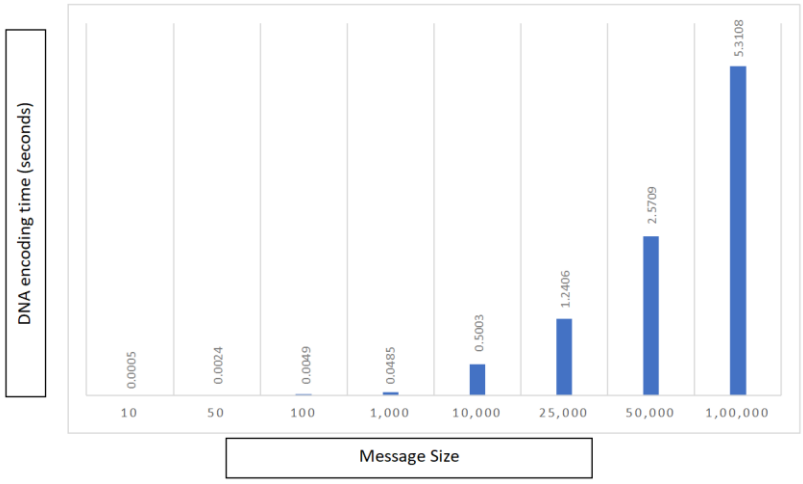

Figure 6: Comparison of Time taken for DNA encoding with respect to message size

Conclusion

The results affirm that the PQEE approach, which integrates classical and innovative cryptographic methods, can effectively counteract potential quantum threats. The combination of DNA-based encoding, symmetric block encryption, and public key encapsulation proved to be robust against advanced attack simulations. This showcases the system's potential as a foundational strategy for future encryption protocols that aim to address post-quantum security needs. In practice, while quantum computers capable of breaking current encryption standards are still in development, the proactive measures outlined in PQEE indicate that a combination of novel encoding and layered cryptography can offer significant advantages. Future work could involve expanding the system to incorporate other post-quantum cryptographic algorithms and testing the protocol with real-world data under diverse conditions to confirm its practicality and efficiency.

References

[1] Sykot, Arman & Azad, Md Shawmoon & Tanha, Wahida & Morshed, B M Monjur & Shubha, Syed & Mahdy, Mahdy Rahman Chowdhury, Multi-Layered Security System: Integrating Quantum Key Distribution with Classical Cryptography to Enhance Steganographic Security, Vol. 45, Issue 67, pp. 78-89, May 2024.

[2] Ukwuoma, Henry & Gabriel, Arome & Thompson, Aderonke & Alese, Boniface. (2022). Post- quantum cryptography-driven security framework for cloud computing. Open Computer Science. 12. 142-153. 10.1515/comp-2022-0235.

[3] T. M. Fernandez-Carames and P. Fraga-Lamas, "Towards Post-Quantum Blockchain: A Review on Blockchain Cryptography Resistant to Quantum Computing Attacks," in IEEE Access, vol. 8, pp. 21091-21116, 2020.

[4] Karakaya, A., & Ulu, A. (2024). A survey on post-quantum-based approaches for edge computing security. WIREs Computational Statistics, 16(1), e1644.

[5] Shiyue Qin, Zhenhua Tan, Fucai Zhou, Jian Xu, Zongye Zhang, and Jinwei Wang. 2021. A Verifiable Steganography-Based Secret Image Sharing Scheme in 5G Networks. Sec. and Commun. Netw. 2021 (2021). 10.1155/2021/6629726.

[6] G. F. Siddiqui et al., "A Dynamic Three-Bit Image Steganography Algorithm for Medical and e-Healthcare Systems," in IEEE Access, vol. 8, pp. 181893-181903, 2020, 10.1109/ACCESS.2020.3028315.

[7] Jajodia, Babita & Thombre, Ritu. (2021). Experimental Analysis of Attacks on RSA & Rabin Cryptosystems using Quantum Shor's Algorithm. 10.21467/proceedings.114.74.

[8] Joseph J. Kearney, Carlos A. Perez-Delgado,Vulnerability of blockchain technologies to quantum attacks, Array, Volume 10, 2021, 100065, ISSN 2590-0056, 10.1016/j.array.2021.100065.

[9] Mamatha, D.G., Dimri, N., & Sinha, R. (2024). Post-Quantum Cryptography: Securing Digital Communication in the Quantum Era. ArXiv, abs/2403.11741.

[10] Prasanna Ravi, Anupam Chattopadhyay, Jan Pieter D'Anvers, and Anubhab Baksi. 2024. Side-channel and Fault-injection attacks over Lattice-based Post-quantum Schemes (Kyber, Dilithium): Survey and New Results. ACM Trans. Embed. Comput. Syst. 23, 2, Article 35 (March 2024), 54 pages.