Iraqi Journal of Information and Communications Technology(IJICT)
Vol. 8, Issue 1, April 2025
www.ijict.edu.iq
ISSN: 2222-758X
e-ISSN: 2789-7362

QUANTUM-SECURE KEY DISTRIBUTION IN A RESOURCE-CONSTRAINED ENVIRONMENT

Mohammed AJ. Hammed [1], **M. F. Al-Gailani** [2]
[1,2] Department of Cyber Security Engineering, College of Information Engineering, Al-Nahrain University, Jadriya, Baghdad, Iraq
mohammed.j@nahrainuniv.edu.iq[1], m.falih@nahrainuniv.edu.iq[2]
Corresponding Author: **M. F. Al-Gailani**
Received:24/07/2023; Revised:04/10/2023; Accepted:21/10/2023
DOI:10.31987/ijict.8.1.258

***Abstract*- In the evolving landscape of quantum computing, the security of conventional cryptosystems is increasingly compromised. This requires innovative data protection strategies, leading to the exploration of Post-Quantum Cryptography (PQC) techniques. This research aims to address these challenges, focusing on the integration of Nth-Degree Truncated Polynomial Ring (NTRU) encryption - known for its quantum resistivity - and Shamir's Secret Sharing (SSS) for secure and quantum-secured key distribution, particularly in resource-constrained environments. Despite their effectiveness, NTRU ciphers face obstacles in public key distribution due to significant computational and memory requirements. Leveraging SSS, a scheme to distribute the NTRU public key to multiple shares, providing enhanced security while mitigating resource limitations is proposed. Recognizing the potential for security breaches like Man-in-the-Middle (MitM) and Distributed Denial of Service (DDoS) attacks, the proposed work seeks to fortify defenses against these threats. The findings of the proposed method are intended to offer a significant contribution to strengthening cryptographic practices in the face of emerging quantum technologies.**

keywords: **Key distribution, Quantum-secure, Secret sharing, NTRU.**

I. INTRODUCTION

With the advent of quantum computing on the horizon, it is more important than ever to develop Post-Quantum Cryptography (PQC) approaches, which will make it easier to shield sensitive data from the risks posed by quantum computing [1]. Due to the intrinsic resilience of the Nth-Degree Truncated Polynomial Ring (NTRU) encryption scheme against quantum assaults, it has recently emerged at the forefront of discussion as a potential PQC solution [2]. Nevertheless, the difficulty of NTRU public key distribution in situations with limited resources still remains due to the high computing and memory requirements [3]. This hurdle can be overcome by integrating the Shamir's Secret Sharing (SSS), a method that splits up the encryption key into numerous shares, each of which is worthless on its own but is valuable when taken together [4]. This provides a solution to resource restrictions while ensuring that the distribution of NTRU public keys is appropriately secured. Despite this, cryptographic key attacks such as Man-in-the-Middle (MitM) and Distributed Denial of Service (DDoS) attacks still seriously threaten information security [5] and attempt to damage the normal traffic of a targeted server, service, or network by overwhelming them with huge packets [6].

In view of this, this study aims to investigate whether the NTRU cipher can be combined with SSS to achieve a secure key distribution that is both quantum-secure and resistant to the aforementioned risks. The findings of this study can significantly improve the security protocols used in cryptographic techniques for use in the quantum era.

Existing cryptographic systems face two major challenges: vulnerability to quantum computing threats and resource constraints in practical deployment. Conventional key distribution schemes like RSA and ECC are susceptible to quantum

ISSN: 2222-758X
e-ISSN: 2789-7362

attacks [7], particularly those leveraging Shor's algorithm [8]. Furthermore, the implementing of quantum-secure alternatives such as NTRU cipher in resource-constrained environments, such as IoT devices, presents significant hurdles due to their high computational and memory demands [9]. Thus, the urgent need to develop efficient, quantum-secure key distribution techniques suited for resource-limited settings forms the core of this research.

II. RELATED WORKS

In the current era of quantum computing development, conventional cryptographic schemes are in the severe risk of becoming obsolete. This fact makes the study of quantum-secure key distribution schemes increasingly significant as they intend to safeguard the secure exchange of keys against quantum computing threats, thereby preserving confidentiality in the forthcoming cryptographic infrastructures [10].

The authors in [11] suggested a secure data exchange protocol for cryptographic key generation, renewal, and distribution KGR system. The proposed protocol message exchanges are implemented via MQTT, an Internet of Things (IoT) application layer protocol that supports constrained environments. However, the proposed protocol supports classical symmetric cryptography that could not stand against quantum attacks, and is designed to operate in a client-server model, which is deprecated with the advent of the peer-to-peer system. The authors in [12] proposed an asymmetric computing key exchange protocol, through the use of the Diffie-Hellman key exchange protocol and the Subset Product problem. They claimed that their protocol reduces the computational complexity of one party while allowing an appropriate rise in the computational complexity of the other party. However, the proposed protocol is based on the Diffie-Hellman key exchange algorithm, which is vulnerable to quantum attacks. Recent literatures have begun to migrate their research interests from classical cryptography like Advance Encryption Standard (AES), Rivest-Shamir-Adleman (RSA) and ECC to Quantum Safe (Post Quantum) cryptography. For instance, the authors in [13] studied quantum-secure cryptography approaches and conducted a survey of the various quantum key distribution protocols, their simulation tools, and challenges in adopting them. The authors in [14] made an evaluation of the quantum key distribution using BB84 protocol. However, the evaluated protocol lacks real implementation in resource-constrained environment. The authors in [15] recommended NTRU quantum secure authentication for IoT devices. The intermediary gateway node receives the public key of each participating side. Thus, revealing the privacy of the public key allows the adversary to forge the communication, opening the door to a MitM attack. Wireless Sensor Network (WSN), which is inherited in nature, is one of the resource-constrained environments, where sensor nodes and sink nodes come with limited computational resources. Some recent literature experiments showed the usability of quantum key distribution in WSN era. The authors in [16] proposed a distribution scheme based on the BB84 protocol which is a quantum key distribution scheme developed by Charles Bennett and Gilles Brassard in 1984, with the aid of AES standard. While the proposed scheme indicates a high level of security, the overhead negotiation among the sink node and the sensor nodes is high.

Based on the above, the current study proposes a method that addresses two challenges: security challenges, represented by finding a secure key distribution method resilient to quantum attacks, and performance challenges, represented by adapting this quantum secure key distribution to be applicable in a resource-constrained environment.

ISSN: 2222-758X
e-ISSN: 2789-7362

III. METHODS

A. *NTRU Cryptosystem*

The NTRU cryptographic algorithm, founded in 1998 [17], is a lattice-based cryptosystem that has been widely recognized as a potential candidate for post-quantum cryptographic applications. The basis of NTRU lies in the hardness of the shortest vector problem in a high-dimensional lattice, which is believed to be resistant to attacks from quantum computers. NTRU's primary strength is its quantum resistance. As compared to classical public-key cryptographic schemes, such as RSA and ECC. NTRU is believed to provide security against quantum computing threats. Furthermore, it has been observed that NTRU operates faster than RSA for equivalent security levels, which adds to its advantages [18]. The NTRU public-key cryptosystem employs an $N-1$ degree polynomial as its object's guiding principle. If a and b are two polynomials in ring R, then they can be defined by equations (1) & (2).

$$a = a_0 + a_1x + a_2x^2 + a_3x^3 + \cdots + a_{N-2}x^{N-2} + a_{N-1}x^{N-1} \tag{1}$$

$$b = b_0 + b_1x + b_2x^2 + b_3x^3 + \cdots + b_{N-2}x^{N-2} + b_{N-1}x^{N-1} \tag{2}$$

The vector representation of coefficient a is given by equation (3).

$$a = (a_0, a_1, a_2, a_3, a_{N-2}, a_{N-1}) \tag{3}$$

The vector representation of coefficient b is given by equation (4).

$$b = (b_0, b_1, b_2, b_3, b_{N-2}, b_{N-1}) \tag{4}$$

In convoluted ring polynomials, the fundamental operations are addition, subtraction, and convolution multiplication. The coefficient of the polynomial $(a_0, a_1, a_2, \ldots, a_{N-1})$ i is an integer. Some coefficient values are 0. This collection of polynomials is known as R.

The NTRU cryptosystem consists of three phases [26]:

1) Key Generation: NTRU Key Generation begins with the selection of two polynomials $f \in L_f$ and $g \in L_g$, provided that f has an inverse modulo p and q, so f_p and f_q can be written in equations (5) & (6).

$$f \times f_p \equiv 1 \pmod{p} \tag{5}$$

$$f \times f_q \equiv 1 \pmod{q} \tag{6}$$

The private key is composed of the polynomial expressions f and f_p. Once the polynomials f and g are determined, the public key h can be calculated using equation (7).

$$h \equiv p \cdot f_q \cdot g \pmod{q} \tag{7}$$

ISSN: 2222-758X
e-ISSN: 2789-7362

Fig. 1 illustrates the steps of this phase.

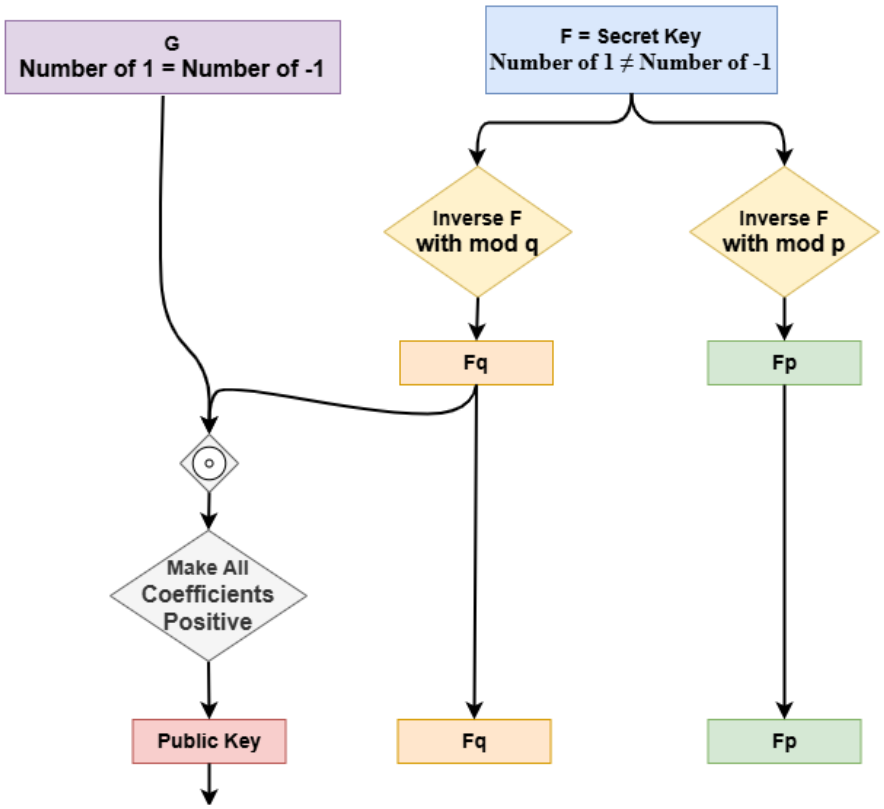

Figure 1: Key-Pair generation.

2) Encryption: To carry out the encryption procedure, two polynomials m and r are chosen, such that both are $\in L_r$. The input message is mapped into polynomial m, and the random polynomial r is used to shuffle the messages. The ciphertext e is calculated by equation (8).

$$e \equiv r \times h + m \pmod{q} \tag{8}$$

Fig. 2 shows these steps.

3) Decryption: The decryption process begins by calculating the polynomial $a = f \times e \pmod{q}$, then define the coefficient a between $-\frac{q}{2}$ and $\frac{q}{2}$, then calculating the polynomial $b = a \pmod{q}$, so that the private key f_p is obtained to

ISSN: 2222-758X
e-ISSN: 2789-7362

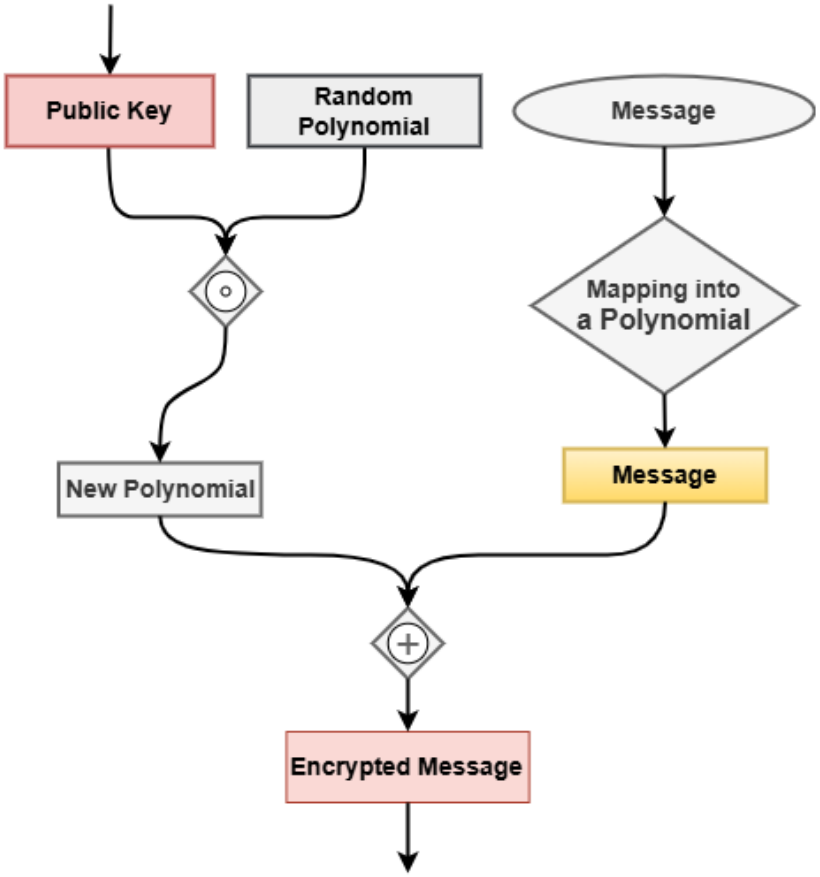

Figure 2: Message encryption.

calculate the value of d as shown in equation (9).

$$d = f_p \cdot b \pmod{p} \tag{9}$$

Fig. 3 presents the steps of the decryption phase.

B. Shamir's Secret Sharing (SSS)

This scheme is proposed by Adi Shamir in 1979. It is an algorithm that allows a secret to be divided into multiple parts, or "shares" [19]. The intriguing part of this scheme is that reconstructing the original secret requires a minimum threshold of these shares, thus ensuring security while facilitating flexibility [20]. SSS cryptosystem is a method of securely

ISSN: 2222-758X
e-ISSN: 2789-7362

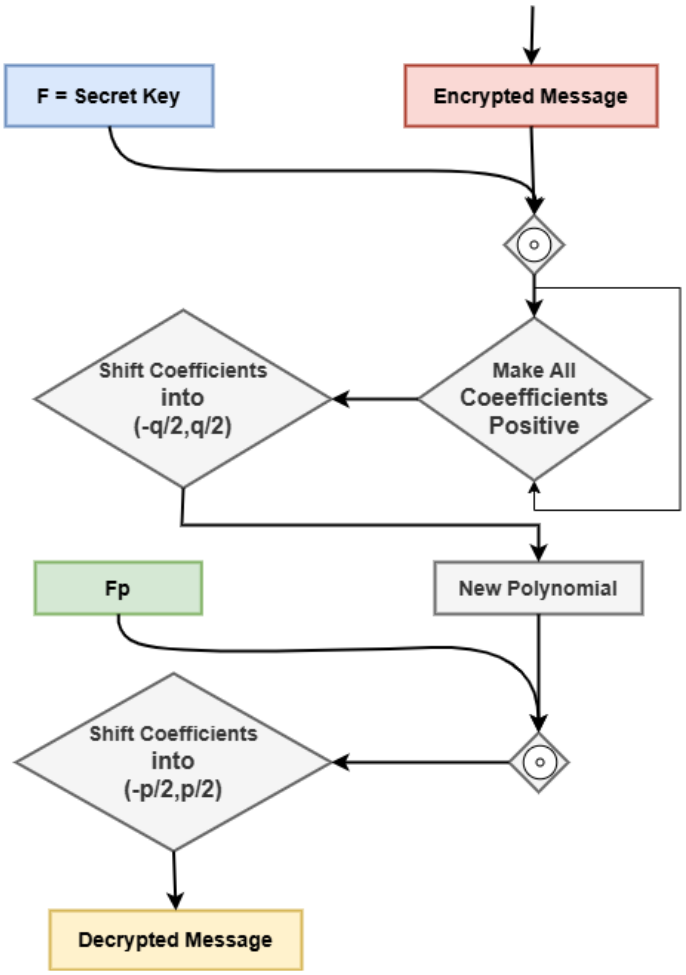

Figure 3: Message decryption.

splitting a secret into multiple shares, distributing them among the participants, and reconstructing the original secret is only possible after combining a sufficient number of shares [21].

Divide the selected data S into multiple sub-data s_i and distribute them into multiple sides, rendering the original data S unreconstructible unless there are T pieces that can be disclose to find again the original data S, so $T-1$ pieces cannot be located [22]. Such a concept is contingent upon determining the threshold T, i.e., a value that specifies the number of pieces s_i required to reconstruct S after it has been divided and disseminated across multiple sides. The SSS algorithm consists of two phases [20]:

- *Distribution phase*: This phase involves taking some secret data denoted by S, it determining the number of sites

NS that will receive the secret parts s_i. A threshold value T, based on these data, must be determined in order to reconstruct S. The next step is to construct a polynomial function $f(x_i)$ with degree $T-1$, to calculate the secret pieces NS. The $T-1$ coefficients are random integers chosen from $GF(S)$ as shown in equation (10):

$$f(x_i) = \sum_{j=0}^{T-1} (a_j \times x^j) \pmod{p} \tag{10}$$

For $i = \{1, ..., NS\}, a_0 = S, \{a_1, ..., a_{T-1}\} \in GF(S)$.

The final step is to distribute the pieces $NS = \{s_1, s_2, ..., s_{NS}\}$ to the sites.

- *Reconstruction phase*: This phase involves reconstructing S from T pieces of data s_i. This requires constructing the original $f(x)$, thanks to LaGrange interpolation, that renders this using equations (11) & (12):

$$l_i = \prod_{M \neq j} \left(\frac{x - x_M}{x_j - x_M} \right) \tag{11}$$

$$f(x) = \sum_{i=0}^{k-1} (y_i \times l_i) \tag{12}$$

IV. PROPOSED SYSTEM

A. *System Architecture*

As shown in Fig. 4, the architecture of the proposed system for a quantum-secure key distribution scheme mainly consists of five sensor nodes and one sink node, which are constructed to represent a typical resource-constrained environment. The sensor nodes in the proposed system are represented by Raspberry Pi 3 development boards. These nodes are responsible for collecting various environmental data. Choosing the Raspberry Pi 3 for these nodes provides an affordable, powerful, and versatile platform, which also offers practical limitations on computational resources, and is therefore in line with the scope of the proposed study focused on resource-constrained environments.

B. *Proposed System Phases*

The proposed system involves the following phases:

1) Shares Generation Phase: This phase involves generating s_i shares from the secret S, which is represented by the NTRU public key h, using the SSS algorithm. Algorithm 1 illustrates the steps of the current phase.

First, the sender generates a pair of keys; public key h and private key (f, f_p) in polynomial form. These polynomials must meet the requirements of the NTRU algorithm, in terms of the number of coefficients $(N-1)$. Since data is sent and received between constrained devices in binary form, this requires converting these data from polynomial-based format to binary digits. In baseline NTRU implementation, each coefficient within the polynomial that represents this data is treated as a separate message. This means that a polynomial with degree 167 resulted in 167 separate messages, which were in turn converted into 167 ciphertexts as the output of the NTRU encryption phase. In the case of constrained devices, there are no resources available to accomplish these intensive calculations.

The proposed system overcomes this limitation by processing the data in a different way. After extracting the coefficients

ISSN: 2222-758X
e-ISSN: 2789-7362

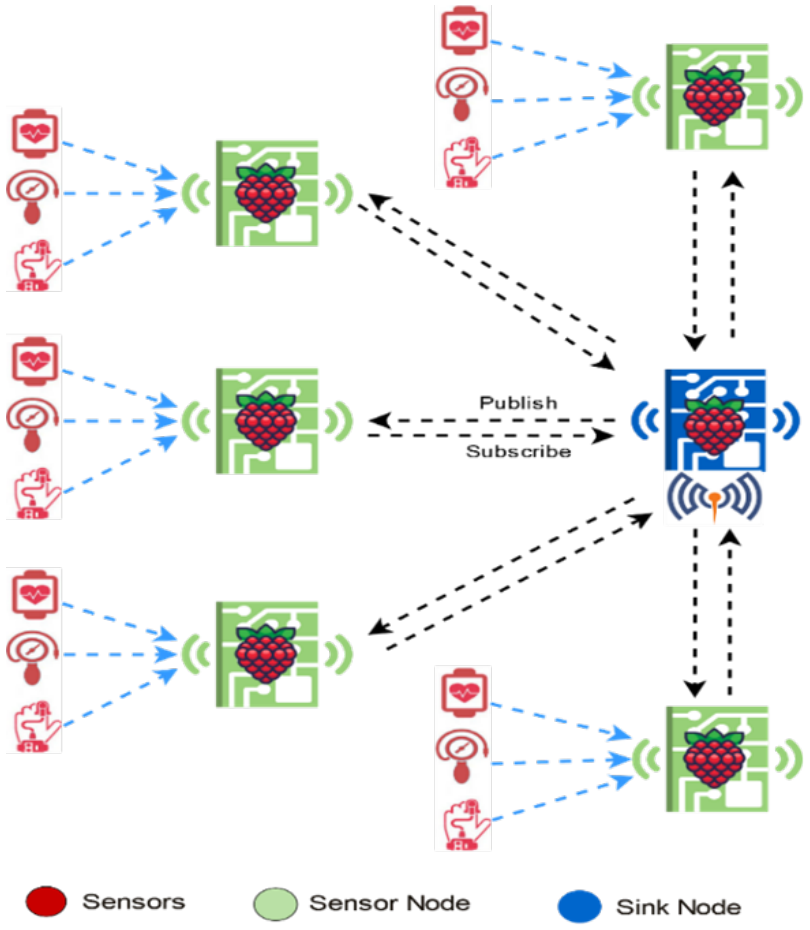

Figure 4: System architecture.

Algorithm 1: Shares Generation

Input: h, n, t, p
Output: shares $[1, \ldots, n]$

Extract Coefficients
Concatenate Coefficients
Choose a finite field for the polynomial coefficients
Generate polynomial $f(x)$ *of degree* $(t-1)$
Set $f(0) = h$
for $i = 1$ *to* n ***do***
    Calculate $y = f(x) \pmod{p}$
    Calculate share $S_i = (x, y_j)$
    Append S_i *to shares*
End for
Return *shares*

ISSN: 2222-758X
e-ISSN: 2789-7362

of the h polynomial, the next step is to concatenate them to construct the secret S. This secret is set to the coefficient a_0 of the polynomial $f(x)$ generated by the sender. Based on the total number of constrained devices n, the sender chooses a threshold t to reconstruct the secret. The degree of the generated secret polynomial must meet the requirement of $t-1$. In the end, each device receives one share, represented by a point coordinate, (x, y). Fig. 5 explains these steps mathematically, for generating 5 shares, with a threshold of 3.

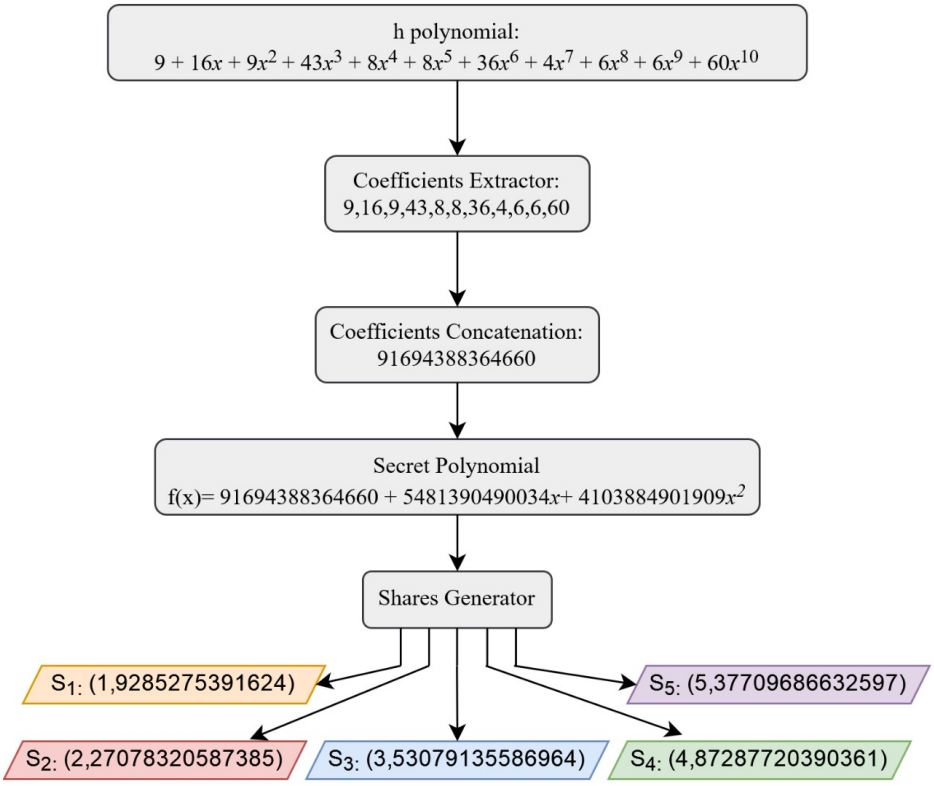

Figure 5: Shares generation.

2) Public Key Reconstruction Phase: After receiving its corresponding share, each participating device has to communicate with the other participants in order to gather t shares, to reconstruct the secret S. Algorithm 2 shows the steps of the current phase. After gathering t shares, the participating device calculates the LaGrange factor for each share gathered. The next step is to calculate the sum of the product of the y-coordinate value by the LaGrange factor. The result is an interpolation of the polynomial $f'(x)$ which its coefficient is a_0. $f'(0)$ is then extracted and separated, in ordered to construct S' polynomial that matches the polynomial of the public key, h. Fig. 6 explains the completion of the mathematical example starting with the first phase.

ISSN: 2222-758X
e-ISSN: 2789-7362

Algorithm 2: Public Key Reconstruction

Input: n, p, t_shares
Output: h
1 ***for*** $i = 1$ *to* t ***do***
2 *Calculate* $l(i)$
3 *Calculate* $\prod l(i \text{ to } t)$
4 *Calculate* $y_j \cdot \prod l(i \text{ to } t)$
5 ***End for***
6 *Calculate polynomial* $f(x)$ *of degree* $(t-1) \pmod p$
7 *Extract* $S = f(0)$
8 *Split* S *into Coefficients*
9 *Convert Coefficients into polynomial* h
10 ***Return*** h

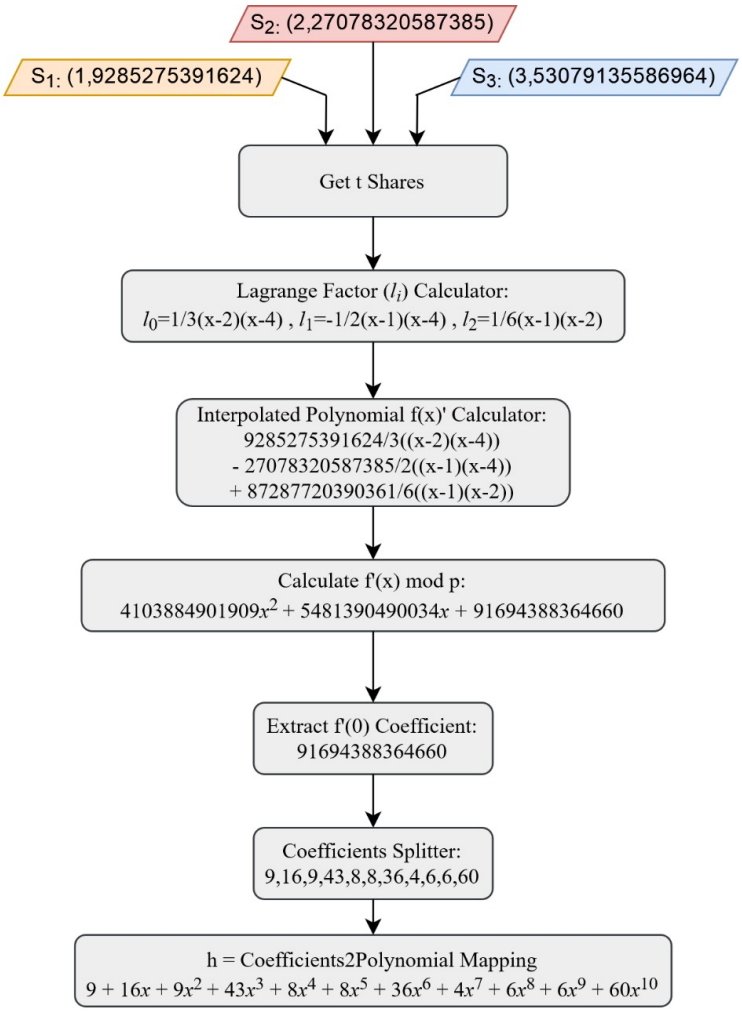

Figure 6: Public key reconstruction.

ISSN: 2222-758X
e-ISSN: 2789-7362

V. RESULTS

Table I explains a comparative analysis for the proposed system with other related literature, in terms of secure key distribution, resilience to classical and quantum attacks, implementation method, Resource-Constrained Environment (RCE) suitability, and support peer-to-peer system.

TABLE I
Comparative Analysis with Other Related Works

Work	[11]	[12]	[15]	[16]	Proposed Work
Secure Key Distribution	✗	✓	✗	✓	✓
Resilience to Classical Attacks	✓	✓	✓	✓	✓
Resilience to Quantum Attacks	✗	✗	✓	✓	✓
Practical Implementation	✓	✗	✗	✗	✓
RCE Suitability	✓	✓	✓	✗	✓
Peer-to-Peer Support	✗	✓	✓	✓	✓

The proposed system combines NTRU encryption with SSS scheme and exhibits substantial resilience against two major forms of attacks in cryptographic communication: MitM and DDoS attacks.

MitM attacks pose significant threats in cryptographic communication as the attacker intercepts and potentially alters the communication between two parties without their knowledge. In the context of the proposed system, an attacker might aim to compromise the key shares that are transmitted from the sensor nodes to the sink node. However, the integration of SSS with NTRU encryption offers robust protection against such attacks. SSS ensures that even if only a few shares of the key are intercepted, the attacker cannot reconstruct the full key without having a threshold number of shares. The quantum-resistant nature of the NTRU encryption algorithm further enhances this defense, making it extremely resource-intensive for a quantum-enabled attacker to break the encryption in [1] &[2].

As for DDoS attacks, they aim to overwhelm the system with more network traffic than it can handle, thus denying service to legitimate users. Due to the resource-constrained nature of the proposed system, resilience to DDoS attacks becomes significant consideration. To mitigate this, two-pronged strategy is employed. First, the distributed nature of key shares across multiple sensor nodes aids diffuses network traffic, making the entire system harder to overwhelm. Second, the Raspberry Pi 4 used for the sink node possesses superior computational capabilities that provide additional resistance against high network loads. Table II explains the development boards used in the proposed system as an experimental environment.

TABLE II
Experimental Environment

Device Type	No. of Devices	Development Board	CPU		RAM	Storage	OS
			Family	Freq.			
Sensor Node	4	Raspberry Pi 3B	Cortex-A53	1.2 GHz	1GB	32GB	Raspbian
Sink Node	1	Raspberry Pi 4B	Cortex-A72	1.4 GHz	8GB	32GB	Raspbian

VI. CONCLUSION

In conclusion, this research presents a robust system that integrates NTRU cipher with SSS for secure and quantum-safe key distribution in resource-constrained environments. Through rigorous analysis and simulation, the system has demonstrated resilience against common cryptographic attacks like MitM and DDoS. The implementation of the scheme on Raspberry Pi platforms has effectively illustrated its practical applicability and efficiency under realistic computational and memory limitations.

Despite the promising results, there are several directions for future work. First, additional security measures can be explored to further enhance the system's resilience against other potential quantum-enabled attacks. Moreover, while the proposed system is designed for a specific number of sensor nodes, future studies can focus on scalability and improving system performance for larger networks.

FUNDING

None.

ACKNOWLEDGEMENT

The authors would like to thank the reviewers for their valuable contribution in the publication of this paper.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

REFERENCES

[1] M. Kumar, "Post-quantum cryptography Algorithm's standardization and performance analysis," Array, vol. 15, p. 100242, Sep. 2022, doi: 10.1016/J.ARRAY.2022.100242.
[2] L. Cherckesova, O. Safaryan, P. Razumov, V. Kravchenko, S. Morozov, and A. Popov, "Post-Quantum Cryptosystem NTRUEnCrypt and Its Advantage over Pre – Quantum Cryptosystem RSA," E3S Web Conf., vol. 224, p. 01037, Dec. 2020, doi: 10.1051/E3SCONF/202022401037.
[3] R. Alléaume et al., "Using quantum key distribution for cryptographic purposes: A survey," Theor. Comput. Sci., vol. 560, no. P1, pp. 62–81, Dec. 2014, doi: 10.1016/J.TCS.2014.09.018.
[4] W. J. Buchanan, D. Lanc, E. Ukwandu, L. Fan, G. Russell, and O. Lo, "The Future Internet: A World of Secret Shares," Futur. Internet 2015, Vol. 7, Pages 445-464, vol. 7, no. 4, pp. 445–464, Nov. 2015, doi: 10.3390/FI7040445.
[5] M. Shafiq, Z. Gu, O. Cheikhrouhou, W. Alhakami, and H. Hamam, "The Rise of 'internet of Things': Review and Open Research Issues Related to Detection and Prevention of IoT-Based Security Attacks," Wirel. Commun. Mob. Comput., vol. 2022, 2022, doi: 10.1155/2022/8669348.
[6] A. A. Abdulrahman and M. K. Ibrahem, "Evaluation of DDoS attacks Detection in a New Intrusion Dataset Based on Classification Algorithms," Iraqi J. Inf. Commun. Technol., vol. 1, no. 3, pp. 49–55, Feb. 2018, doi: 10.31987/IJICT.1.3.40.
[7] J. Suo, L. Wang, S. Yang, W. Zheng, and J. Zhang, "Quantum algorithms for typical hard problems: a perspective of cryptanalysis," Quantum Inf. Process., vol. 19, no. 6, pp. 1–26, Jun. 2020, doi: 10.1007/S11128-020-02673-X/FIGURES/7.
[8] H. T. Larasati and H. Kim, "Quantum Cryptanalysis Landscape of Shor's Algorithm for Elliptic Curve Discrete Logarithm Problem," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 13009 LNCS, pp. 91–104, 2021, doi: 10.1007/978-3-030-89432-0_8.
[9] O. M. Guillen, T. Poppelmann, J. M. Bermudo Mera, E. F. Bongenaar, G. Sigl, and J. Sepulveda, "Towards post-quantum security for IoT endpoints with NTRU," Proc. 2017 Des. Autom. Test Eur. DATE 2017, pp. 698–703, May 2017, doi: 10.23919/DATE.2017.7927079.
[10] G. N. Brijwani, P. E. Ajmire, and P. V. Thawani, "Future of Quantum Computing in Cyber Security," in Handbook of Research on Quantum Computing for Smart Environments, IGI Global, 2023, pp. 267–298. doi: 10.4018/978-1-6684-6697-1.CH016.
[11] J. Furtak, "Data Exchange Protocol for Cryptographic Key Distribution System Using MQTT Service," Proc. 17th Conf. Comput. Sci. Intell. Syst. FedCSIS 2022, pp. 611–615, 2022, doi: 10.15439/2022F260.
[12] H. Wang, J. Wen, J. Liu, and H. Zhang, "ACKE: Asymmetric Computing Key Exchange Protocol for IoT Environments," IEEE Internet Things J., 2023, doi: 10.1109/JIOT.2023.3279283.
[13] G. Xu, J. Mao, E. Sakk, and S. P. Wang, "An Overview of Quantum-Safe Approaches: Quantum Key Distribution and Post-Quantum Cryptography," 2023 57th Annu. Conf. Inf. Sci. Syst. CISS 2023, 2023, doi: 10.1109/CISS56502.2023.10089619.
[14] H. A. Qaisi and M. F. Al-Gailani, "EVALUATION OF QUANTUM KEY DISTRIBUTION BY SIMULATION," Iraqi J. Inf. Commun. Technol., vol. 5, no. 3, pp. 15–22, Dec. 2022, doi: 10.31987/IJICT.5.3.157.
[15] S. H. Jeong, K. S. Park, Y. H. Park, and Y. H. Park, "An efficient NTRU-based authentication protocol in IoT environment," Adv. Intell. Syst. Comput., vol. 857, pp. 1262–1268, 2019, doi: 10.1007/978-3-030-01177-2_91/COVER.
[16] L. H. Alhasnawy and A. K. AL-Mashanji, "Improving Wireless Sensor Network Security Using Quantum Key Distribution," Baghdad Sci. J., Mar. 2023, doi: 10.21123/BSJ.2023.7460.

ISSN: 2222-758X
e-ISSN: 2789-7362

[17] J. Hoffstein, J. Pipher, and J. H. Silverman, "NTRU: A ring-based public key cryptosystem," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 1423, pp. 267–288, 1998, doi: 10.1007/BFB0054868/COVER.
[18] B. Harjito, H. N. Tyas, E. Suryani, and D. W. Wardani, "Comparative Analysis of RSA and NTRU Algorithms and Implementation in the Cloud," Int. J. Adv. Comput. Sci. Appl., vol. 13, no. 3, pp. 157–164, 2022, doi: 10.14569/IJACSA.2022.0130321.
[19] A. Shamir, "How to share a secret," Commun. ACM, vol. 22, no. 11, pp. 612–613, Nov. 1979, doi: 10.1145/359168.359176.
[20] E. Dawson and D. Donovan, "The breadth of Shamir's secret-sharing scheme," Comput. Secur., vol. 13, no. 1, pp. 69–78, Feb. 1994, doi: 10.1016/0167-4048(94)90097-3.
[21] P. Sarosh, S. A. Parah, and G. M. Bhat, "Utilization of secret sharing technology for secure communication: a state-of-the-art review," Multimed. Tools Appl., vol. 80, no. 1, pp. 517–541, Jan. 2021, doi: 10.1007/S11042-020-09723-7/METRICS.
[22] L. J. Pang and Y. M. Wang, "A new (t, n) multi-secret sharing scheme based on Shamir's secret sharing," Appl. Math. Comput., vol. 167, no. 2, pp. 840–848, Aug. 2005, doi: 10.1016/J.AMC.2004.06.120.