Computer Networks 217 (2022) 109327

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

A post-quantum lattice based lightweight authentication and code-based hybrid encryption scheme for IoT devices

Swati Kumari [a,*], Maninder Singh [b], Raman Singh [c], Hitesh Tewari [d]

[a] Assistant Professor, Department of Computer Science and Engineering, Thapar Institute of Engineering and Technology, Patiala, Punjab, India
[b] Professor and Dean of Academics, Department of Computer Science and Engineering, Thapar Institute of Engineering and Technology, Patiala, Punjab, India
[c] Lecturer in Cyber Security, School of Computing, Engineering & Physical Sciences, University of the West of Scotland, Lanarkshire Campus, Scotland, G72 0LH
[d] Assistant Professor, School of Computer Science and Statistics, Trinity College Dublin, Ireland

ARTICLE INFO

Keywords:
Post-quantum cryptography
Ring learning with errors (Ring-LWE)
Internet of things (IoT)
Polynomial multiplication
QC-LDPC

ABSTRACT

The Internet of Things (IoT) introduces an active connection between smart devices for revolutionizing our modern lives in this world. But, IoT devices often exhibit several security issues, so transmission between the nodes should be protected using cryptographic approaches. However, the complexity of conventional cryptographic approaches is very high and is vulnerable to quantum attacks. This paper presents a robust and lightweight post-quantum lattice-based authentication and code-based hybrid encryption scheme for resource-constrained IoT devices. The proposed Ring-Learning with Errors (Ring-LWE) based authentication scheme introduces Bernstein reconstruction in polynomial multiplication to achieve minimal computation cost; hence, resource-limited IoT devices are viable to use the reliable authentication mutually. This approach offers indefinite identity privacy and location privacy. Hence, the proposed signature generation and verification process are highly efficient compared to the existing ring signature systems. Also, the proposed post-quantum hybrid code-based encryption scheme follows Diagonal Structure Based QC-LDPC Codes with column loop optimization and Simplified Log Domain Sum-Product Algorithm (SLDSPA) to provide the function of light weight encryption with minimum hardware requirements. The total authentication delay of the proposed authentication scheme is 23% less than the authentication scheme that is considered conventional polynomial multiplication. Also, the optimized design of the proposed code based HE uses only 64 slices and 640 slices on Xilinx Virtex-6 FPGA for encoding and decoding processes, respectively. These simulation results prove the effectiveness of the proposed cryptographic scheme against other competitive systems in terms of its functionality and hardware complexities.

1. Introduction

Currently, the Internet of Things (IoT) is extended to different day-to-day applications due to its smartness and intelligence [1]. An IoT application environment is linked to different physical world systems, infrastructure and protocols to build practical connected smart device networks. As manifold devices integrate into the IoT system, an excessive level of vulnerabilities occurs that can violate the security of device-to-device communication in the application environment [2, 3]. Therefore, it is essential to establish a secure transmission medium for IoT devices to conserve the privacy and confidentiality of the application user and data integrity against vulnerabilities. The security loophole occurs when an important part of confidential data can be exchanged between devices that should be considered when developing reliable IoT protocols to guarantee the security of IoT devices [4, 5].

In the last few decades, the security problems of IoT have been tackled by developing different solutions for authentication, data encryption, hardware encryption, network segmentation, and location-based services [6]. The most significant aspect of IoT security is authentication. The IoT network is susceptible to security issues for several reasons, including its dynamic nature as a result of mobility, availability of inadequate resources, and heterogeneous nature related to protocols, devices, and platforms. The main limiting factor of IoT networks is that IoT devices have constrained storage, power supply, and bandwidth. The conventional authentication approaches are not applicable for IoT networks because of the resource restraint nature of IoT devices. Hence, there is a requirement for lightweight and privacy-preserving authentication approaches for IoTs [7].

* Corresponding author.
E-mail addresses: swati.kumari@thapar.edu (S. Kumari), msingh@thapar.edu (M. Singh), raman.singh@uws.ac.uk (R. Singh), htewari@cs.tcd.ie (H. Tewari).

https://doi.org/10.1016/j.comnet.2022.109327
Received 21 December 2021; Received in revised form 18 June 2022; Accepted 25 August 2022
Available online 28 August 2022

In order to set up reliable communication among two devices and furnish secure information exchange, different cryptographic schemes are presented in various layers of the network. Certain schemes are implemented in physical-layer that gets advantage of low complexity and overhead [8, 9]. The application-layer protocols followed public key cryptography (PKE) for secure key exchange and communication establishment. Though, conventional PKEs, such as Rivest–Shamir–Adleman (RSA) and elliptic curve cryptography (ECC), have excessive computation and communication costs to be successfully developed on resource-limited IoT devices [10, 11]. At the same time, with recent improvements in quantum computers and the vulnerability of classic PKEs against quantum attacks based on Shor's algorithm, it is essential to address alternate quantum-resistant cryptosystems for secure device-to-device communication [12].

In quantum computing, a great extent of research has been carried out, not only to enhance Shor and Grover's algorithms but also to build post-quantum cryptographic algorithms. These post-quantum algorithms are secure against quantum attacks, but while runtime occasionally, they can be slower and requires larger keys. In order to mitigate the problems engendered by quantum computing, it is vital to discover robust and lightweight post-quantum cryptographic algorithms that can resist quantum attacks [13, 14]. General post-quantum algorithms follow lattices, multivariate quadratics, hash functions, linear codes and isogeny assumptions for cryptographic computation. Particularly, lattice-based post-quantum cryptography is the most promising research hotspot due to the faster computational efficiency and ability to resist quantum computing attacks [15]. Code-based cryptography is a promising alternative to post-quantum cryptography by employing various codes and unique structures to accomplish security against quantum attacks. Particularly in code-based cryptography, certain cryptographic schemes follow Hamming distance codes which makes the public-key size large and achieves low efficiency. Still, recently Quasi-Cyclic (QC) codes and low-density parity-check (LDPC) codes have been employed to minimize the public key size and enhance the efficiency of the cryptosystem [16, 17].

One of the significant security issues in IoT is Identity privacy. The crucial information regarding the user's identity may leak by the properties around us connected to the IoT [18]. IoT devices require authentication before offering any service to them. The identity of the IoT device should be passed through the network to get authentication from other devices. The IoT devices are usually communicated via a wireless medium. Hence, the device's identity privacy may be compromised while transmitting the identity via this medium in the form of plain text.

Furthermore, the installation of IoT in smart environments can embrace cameras or other systems. These devices may determine the user's activities directly or indirectly and reveal their location data at specified times [19]. Thus, there is a need for securing algorithms in IoT to preserve the identity and the physical location of the IoT users. The conventional approaches apply pseudonyms to preserve IoT devices' identity and location information. In this approach, a user and/or a device is authenticated without requiring the device holders' identity or other personal information [20]. But the computational complexity of this approach is high, and it needs efficient algorithms for managing the keys and pseudonyms. Thus, an efficient authentication approach should provide identity and location privacy in IoT nodes without increasing the complexity/overhead. The proposed post-quantum cryptographic scheme combines the benefits of lattice-based cryptography, code-based cryptography, and key derivation functions for developing robust authentication and lightweight encryption for resource-constrained IoT devices.

The main motivation of the present research work is to develop secure mutual authentication and data transmission schemes for IoT devices based on post-quantum Cryptography Algorithms. This work aimed to enhance the earlier lattice-based and code-based cryptosystems by introducing a lightweight security approach based on an IoT system. The main challenging block in lattice-based cryptography is polynomial multiplication. This work aims to introduce delay-efficient polynomial multiplication to improve the effectiveness of lattice-based cryptography. In this work, the idea of Bernstein is extended to the recursion of the Karatsuba formula for reducing the space complexity or the delay. Also, the existing code-based cryptography, such as QC-LDPC codes, increased the code's sparse behavior and computational complexity of the design. Also, their security has been decreased to the conventional linear code decoding issue. The proposed code-based data security algorithm introduced a new diagonal structure-based parity check matrix of QC-LDPC code with column-wise looping to provide the function of lightweight encryption with minimum hardware requirements. The main contributions of this research work are listed as follows:

- A new Lattice-based authentication scheme (LR-IoTA) is proposed by introducing a new polynomial multiplication with Bernstein reconstruction to realize high-speed authentication with minimum message exchanges and fewer hardware components.
- A novel code-based hybrid encryption scheme is proposed to improve data security with fewer computational/hardware requirements. It optimizes the diagonally structured QC-LDPC code generation and decoding processes based on column loop optimization and the simplified log domain sum-product algorithm (SLDPA).
- The proposed cryptographic scheme is validated using certain security and performance analyses. In security analysis, different attack models such as Replay Attack, Man-in-the-Middle (MITM) attack, Key-Compromise Impersonation (KCI) attack and Ephemeral Secret Leakage (ESL) attack are considered. In performance analysis, the performance of the proposed protocols is evaluated in terms of communication cost, computation cost and hardware requirements.

The rest of the paper is organized as follows: Section 2 reviews recent related works on post-quantum cryptography schemes, and Section 3 provides the preliminaries related to the proposed methodologies. Section 4 illustrates the generic system model of the IoT network. Section 5 explains the proposed protocols. Section 6 validates the performance of the proposed protocol in terms of security and performance analysis. Finally, the paper is concluded in Section 7.

2. Related works

Some of the recent research works related to the post-quantum cryptography scheme for the Internet of Things are listed below:

Recently, the National Institute of Standards and Technology advanced the security threat of the post-quantum public-key cryptographic scheme by launching a novel standardization method. Some of the digital signature candidates launched by NIST are Dilithium and qTESLA [21]. The qTESLA method is a new signature generation method that is considered the Ring-LWE problem. Different algorithms have been developed for securing IoT networks in the last few decades. The public key infrastructure-based (PKI) algorithms use the node's public key and digital signature to generate the certificate. Li et al. [22] proposed smooth projective hash functions (SPHF) and commitment-based password-hashing schemes (PHS) over lattice-based cryptography. Also, they proposed an asymmetric Password Authenticated Key Exchange (PAKE) protocol for securing the network against quantum attacks. But, this approach did not give full privacy because the authority may reveal the original identity of the node. Furthermore, it did not give security poof for complete anonymity.

Cheng et al. [23] proposed a privacy-preserving authentication approach based on certificateless cryptography, elliptic curve cryptography (ECC), and pseudonym-based cryptography (PBC). This approach used certificateless cryptography for the protection of the private key of the user. Also, the identity of IoT devices has been hidden using pseudonym-based cryptography. In addition, trustless cooperation, tamper-proof recording, and a secure running environment for

authentication were provided by the blockchain technique. However, the ring signature-based approach's security is high when compared with the state-of-the-art pseudonym-based methods. Shim et al. [24] provided cryptanalysis of various lattice-based blind signatures and proposed a general construction based on secure homomorphic encryption that has been utilized for hiding the blinding parameters. The blind signature method must be unforgeable. But, the length of this blind signature is too long, and it did not provide detailed proof for unforgeability.

In [25], Wang et al. improved the security of the smart-card-based password authentication protocol using the problem of Ring-LWE. In this work, a new quantum two-factor authentication (2FA) has been introduced using the idea of Alkim et al.'s [26] lattice-based key exchange and Wang-Wang's [27] "fuzzy-verifier + honeywords" approach. But, Wang-Wang's "fuzzy-verifier + honeywords" method has been constructed using group-based cryptosystems, and hence it is susceptible to the postquantum era. Also, it is more difficult to issue the private keys securely among the group members. Lee et al. [17] have proposed an IoT device key encapsulation technique based on post-quantum cryptography called RLizard. The proposed scheme mainly relied on errors and rounding problems of ring learning. This approach considered the Ring-LWE problem to reduce the key generation time and space requirements of the Lizard method. This Lizard scheme used rounding methods in the encryption process instead of adding some errors. Here, the key generation phase needed polynomial multiplication and was designed using conventional polynomial multiplication. The performance of this polynomial multiplier has been degraded because of the space complexity or the delay.

Chaudhary et al. [28] reviewed the most recent lattice-based cryptography scheme by considering the IoT environments. They identified that the LWE or Ring-LWE methods could provide secure authentication and encryption/decryption in resource constrained IoT devices. Aujla et al. [29] proposed a new cryptography scheme based on lattice-based Ring-LWE for health care systems, which considered the existence of 5 G, SDN, edge, and cloud in the generic system model of IoT network. Although the evolution of lattice-based ring-LWE cryptography has been confirmed on IoT networks, the selected algorithms have not been developed in the hardware platform. Hence, Wang et al. [30] provided FPGA-based implementation for lattice-based cryptography by analysing the hardware complexities of different polynomial multiplications. Conventional lattice-based ring-LWE cryptography schemes used Number Theoretic Transform (NTT) based polynomial multiplier [31] with fixed parameters. Hence, they were not reusable while arriving at new parameters. Hence, a new adaptive NTT-based polynomial multiplier has been developed [30] with a parameterizable design. In addition, sparse polynomial multiplication (SPM) has also been used in lattice-based cryptography schemes to multiply the indeterminate polynomials. However, this design increased the overhead while increasing the parallelism of the SPM design.

To simplify the design complexity of Ring-LWE-based cryptography, Buchmann et al. [32] replaced the Gaussian distribution with a binary distribution. This method is suitable for transmitting short-length messages, not for arbitrary-length messages, due to the consideration of Lattice-Based Public-Key Encryption. Ebrahimi et al. [33] analysed the susceptibility of binary Ring-LWE for fault attacks in IoT devices. This scheme used a shift and added an algorithm to multiply the polynomials. Here, the software solution of binary Ring-LWE considered 8-bit and 32-bit lightweight microcontrollers. The strength of the lattice-based cryptosystem that used Ring-LWE has been further improved by introducing double authentication [34]. However, this approach doubles the computation time of the authentication phase. To improve the encryption efficiency, Norah et al. [35] have presented a light-weight SIMON block cipher-based cryptography algorithm for IoT-equipped devices supported for healthcare applications. The modification is performed on digital round operations of SIMONs cipher, especially on several shift operations affecting the algorithm's speed. Hence optimization is required to regulate the number of shift operations.

Alternatively, Chikouche et al. [36] have proposed an authentication protocol for IoT users' privacy preservation based on code-based cryptography. This protocol follows code-based cryptography, one of the promising post-quantum cryptography techniques able to resist quantum attacks. Moreover, this protocol intended to enhance the variant of the McEliece cryptosystem using Quasi-Cyclic Moderate-Density Parity-Check (QC-MDPC) to resist various typical attacks. The hardware requirements of the lightweight post-quantum cryptography scheme based on QC-LDPC codes have been presented in [37]. This approach focused on the cryptosystem's speed, area and power efficiency because the low-cost and power-constrained IoT applications require hardware analyses. Phoon et al. [38] implemented the QC-MDPC in FPGA by optimizing it for IoT applications. Also, a newly customized rotation engine (CRE) has been introduced along with different recent methodologies, including adaptive threshold and hamming weight estimation, to increase the speed of the decryption algorithm. This approach consumed more area for storing the longer key.

The concept of IoT can be extended to smart applications, including E-health, E-homes, transportation, or energy production. One of the common application domains in IoT is vehicular networks. HAN et al. [39] proposed a Traceable Ring Signature (TRS) approach to protect the security of vehicular networks. This approach added certain information to the ring signature to improve its traceability of the ring signature. Here, the master key has been generated by selecting the algorithm on the ideal lattice to make them resistant to quantum attacks. This scheme has solved the issue of privacy protection of user identity. But, location privacy has not been discussed in this scheme. The identity privacy-preserving approach should preserve location privacy. Mundhe et al. [40] satisfied the privacy requirements of the vehicular network in terms of identity privacy and location privacy using the ring Signature-Based Conditional Privacy-Preserving Authentication method. This point motivates us to use the ring signature-based authentication scheme in an IoT network. However, the hardware complexity of the existing ring signature-based authentication scheme is increased due to the use of conventional NTT and SPM-based polynomial multiplier.

In literature, the polynomial multiplication is developed using either the schoolbook polynomial multiplication approach (SPMA) or Number Theoretic Transform (NTT). SPMA is an efficient approach for hardware implementation, whereas NTT needs more complex pre-computation and array re-ordering to improve the performance. Karatsuba algorithm is an alternative entrant and is not extensively deliberated for Ring-LWE execution in FPGA [47]. Liu et al. [48] proposed an optimized SPMA (oSPMA) for Lattice-Based Cryptography. This approach decreased the processing delay and saved adder, subtractor and multiplexer units. It utilized the DSP slice completely through the adoption of clever stacking in multiplicands and the allowance of two multiplications inside a single DSP slice per cycle. Zhang et al.'s [49] extended the oSPMA by adding extra DSP slices to enable four multiplications per cycle. This approach's throughput has been increased compared to Liu et al. [48]'s Ring- LWE crypto processor. Even though the hardware structure of the oSPMA is so simple, its throughput is very low. But the NTT structure has higher throughput than that of oSPMA. Thus, NTT has been widely used in Ring-LWE to perform highly efficiently and robustly.

Liu et al. [50] proposed a resource-efficient, side-channel secure Ring-LWE cryptographic processor using an NTT multiplier. Feng et al. [51] used the idea of Stockham's fast Fourier transform in an NTT-based polynomial multiplier to tackle the delay problem in the Ring-LWE cryptosystem. But, the hardware requirement for an NTT-based polynomial multiplier is very high, whereas it increases the system's throughput. Wong et al. [47] developed a Karatsuba Algorithm in FPGA to support polynomial multiplication in a Lattice-based cryptosystem. Here, a one-level Karatsuba structure has been introduced to improve the performance of polynomial multiplication in Ring-LWE. It simplified

Table 1
Notations and Abbreviations.

Symbol	Description	Symbol	Description
Ψ	Lattice structure	E	Bound to sample uniform random value forY_n
Λ_i	Linearly independent vectors	q	Polynomial bounded modulus
Z^j	j -dimensional space with integers	ν_n	A sample vector of length n
G_σ	Discrete Gaussian distribution	$\lfloor\nu\rfloor_{f,q}$	The polynomial with $\lfloor\nu \bmod \rfloor q_f$ operations applied to all coefficients.
Z_q	Finite filed over q	ς	Hashed code word
$\Re^+$	Positive real numbers	$\bar{\varsigma}$	Encoded output of ς
$\aleph$	Natural number	S_{se}	Signature of sender node
χ	probability distribution used to specify error vector	V	Bound to check uniform distribution of signature
R_q	Quotient ring	ℓ	Number of polynomial elements (PEs)
i	Degree of polynomial	u	Variables of linear polynomial equation
N	Number of IoT members in a ring	a, b	Coefficients of linear polynomial equation
pk_n	Public key of IoT device n	$\dot{R}_0$	First reconstruction in Bernstein method
sk_n	Secret key of IoT device n	$\dot{R}_1$	Second level of reconstruction in Bernstein method
sk_{se}	Secret key of sender device	C	Final recursive product
P	Public key set	H_{qc}	Parity check matrix of QC-LDPC code
K	Keyword message	$X \times Y$	Size of H_{qc}
S_n	Signature of all ring members	H_L	Lower decomposition of H_{qc}
pk_{ds}	Required Public key during data sharing	H_U	Upper decomposition of H_{qc}
sk_{ds}	Required secret key during data sharing	z	Number of submatrices
m	Message	P_Y	Column vector used to construct permutation matrix
ssk	Session key	W	Dense matrix in code based HE
C_T	Cipher text	n_0	Number of circulant matrices
R_n	Random matrix	G	Sparse transformation matrix
δ_n,ε_n	Secret matrices	$\widehat{e}$	Random error vector
ϖ	Weight value to check short signature	Lc_y	The prior data of variable node y
σ	Standard deviation	$L\bar{c}_y$	The posterior data of variable node y
M	Probability of acceptance of key generation	$L_{Rx,y}$	Extrinsic check to variable message from x to y
Y_n	Polynomial whose coefficients are in $[-E,E]$	$L_{Q_{y,x}}$	extrinsic variable node to check message from y to x

the negative cyclic operations while recombining the sub-polynomials.

In the literature, the security challenges in IoT have been improved using different cryptographic approaches. However, some methods are not employed for IoT-constrained devices because of the higher demand for computation execution time. Moreover, most existing authentication approaches exposed the node's identity to the adversary. Also, the authentication approach should give location privacy by preventing an attacker from detecting a device by chasing its route or guessing its location from the broadcasted data. This location privacy has not been considered in most existing methods. Furthermore, the post-quantum cryptography scheme should consider authentication and data sharing schemes. All the existing methods used either Ring-LWE or code-based cryptography in the authentication or encryption scheme.

However, the complexity of the lattice-based Ring-LWE schemes depends on the complex sparse polynomial multiplications. Similarly, the complexity of code-based cryptography schemes depends on the code generation and decoding processes. The existing NTT multipliers need more complex pre-calculation and array re-ordering to improve the throughput. The pre-calculation part contains exponential computation, increasing the hardware complexity compared to ordinary arithmetic units. Likewise, the SPM included a dense and a sparse polynomial. It requires more memory chunks for storing the data of the sparse polynomial its sparsity. Hence, it increases the area overhead while increasing the parallelism of the design. Also, the space complexity or the delay of the Karatsuba multiplier is very high. In addition, the existing QC-LDPC encoder uses dense structures to generate the parity check matrix, increasing the decoding complexity. To tackle the above issues, the proposed work combines both Ring-LWE and code-based cryptography to give network security and data security. In addition, the complexity of both algorithms is decreased by constructing alternative polynomial multiplication and encoding/decoding processes, respectively.

3. Preliminaries

This section describes some essential preliminaries related to the proposed cryptographic scheme. For ease of presentation, a few instinctual notations are provided in Table 1 and will be utilized throughout this paper. The notation Z denotes the integers throughout the rest of the paper.

The basic definitions related to lattice-based cryptography and code-based encryption scheme are listed as follows:

Lattices: Lattices can be described by the collection of points with the periodic arrangement in i -dimensional space. A general mathematical format of lattice $\Psi$is defined in (1)

$$\Psi = \{a_1\Lambda_1 + a_2\Lambda_2 + ...a_i\Lambda_i | a_i \in Z\} \tag{1}$$

Where, $\Lambda_1, \Lambda_2, ...\Lambda_i \in Z^j$denote$i$linearly independent vectors, Z^jsignifies, j -dimensional space with integers, i and j represent rank and dimension, respectively. A lattice is called a full rank lattice only when $i = j$.

Gaussian distribution: The centered discrete Gaussian distribution G_σ for $\sigma > 0$ relates the probability $p_\sigma(u)/p_\sigma(Z)$ to $u \in Z$ for $p_\sigma(u) = e^{(-u^2/2\sigma^2)}$and $p_\sigma(Z) = 1 + 2\sum_{u=1}^{\infty} p_\sigma(u)$. Hence, $G_\sigma^i = p_\sigma(u)/p_\sigma(Z^i)$. Also, the proposed scheme uses $\delta \leftarrow G_\sigma^n$ to represent a matrix δ with$n$elements that are equally and self-sufficiently sampled from G_σ^n.

Learning with Errors (LWE): The vectors constructed from an error set of linear equations can be differentiated using the LWE problem. Consider modulus $q = \text{poly(i)}$ be random and $\delta_n \in Z_q^i$ be vector. The distribution of LWE distribution for the provided vector $\delta_n \in Z_q^i$and a random matrix $R_n \in Z_q^i$is $T_n = R_n\delta_n + \varepsilon_n(\bmod q)$. Here, a probability distribution $\chi : Z_p \rightarrow \Re^+$ is used to specify every coordinate of the error vector $\varepsilon_n \in Z_q$ on Z_q.

The computational-LWE problem: For a vector $\delta \leftarrow \chi^i$ and given arbitrarily many samples from the LWE distribution to compute δ.

The decisional-LWE problem: several samples are provided arbitrarily from Z_q^{i+1} for distinguishing the samples distributed uniformly from the samples that distributed as the LWE distribution for some fixed vector$\delta \leftarrow \chi^i$.

***Ring LWE*:** The proper definition for Ring-LWE is Learning with Errors over Rings. This Ring LWE is specifically developed to denote the LWE problem on polynomial rings over finite fields. Lyubashevsky et al. [41, 42] proposed the problem of ring learning with errors (Ring LWE). Here, the formal definition of the ring-LWE problem is provided. A (rational) integer modulus $q \geq 2$ parameterizes the Ring-LWE. Let R_q be the quotient ring of $(R\ mod\ q)$ i.e., $R_q := R/qR$. Ring-LWE problems selects the polynomials $R_n(u)$ and $\delta_n(u)$ from a ring $R_q = Z_q(u)/f(u)$ in a uniform manner. Here, $f(u)$denotes the irreducible polynomial of degree i. The samples are collected from error distribution χ, which is generally a discrete Gaussian distribution $G_\sigma^i$with standard deviation σ to construct an error polynomials $\varepsilon_n(u)$of degree i. The ring-LWE distribution contains tuples(A, T), where $T = R\delta + \varepsilon(\bmod q)$.

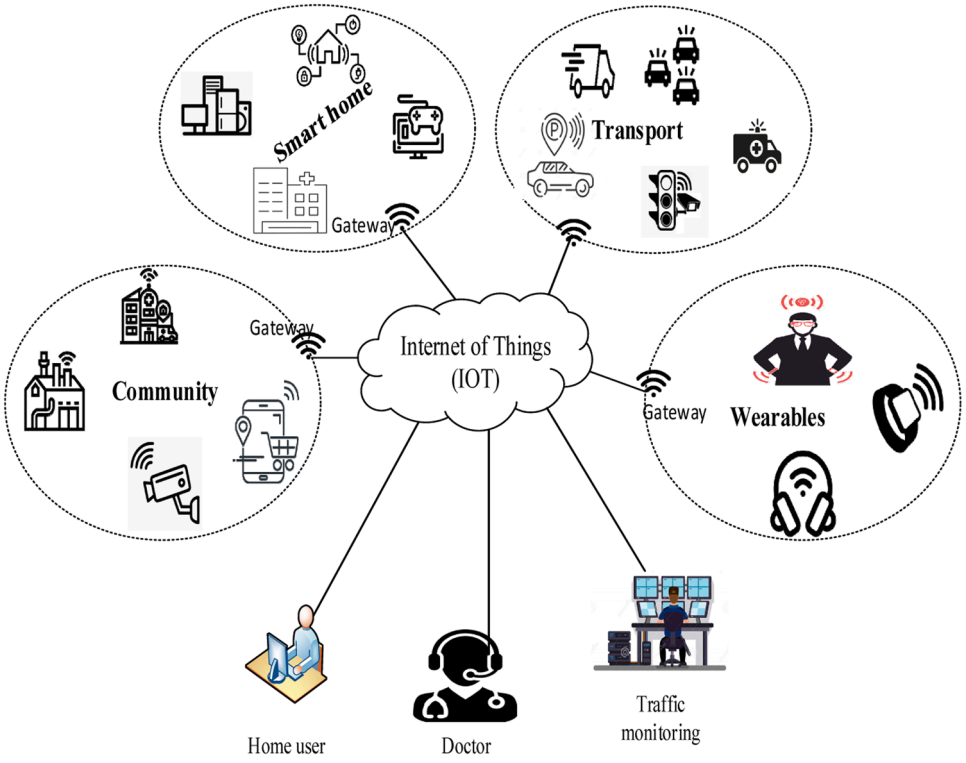

Fig. 1. Generic Structure of IoT Network.

Hardness assumption

The authentication method's security depends on the Ring-LWE problem's hardness. One can define this problem as detecting the value of a private key *sk* is a hard problem.

Definition 1: Assum1er $i \in N$, prime number $i \in N$ access oracle O^i_{sk} for the generation of an output pair (R,T). Here, $R \in R^i_q$ is the polynomial generated uniformly, and $T = R{\cdot}sk + \varepsilon(\bmod q)$. In the same way, private key *sk* is selected from R^i_q and ε is obtained from G^i_σ. The two deviations of the LWE problem are search-LWE and decisional-LWE. The search-LWE problem defines the hard problem of detecting the value of a private key *sk* from(R,T). The decisional-LWE problem defines the probability of distinguishing the $O^i_{sk}$and R is negligible. Here, $O^i_{sk}$and R denotes the oracles that are utilized for the generation of polynomials (R, u)Where $R,u \in R^i_q$. The size of the public key can be reduced using this Ring-LWE to increase the speed of the operation.

Ring Signature

The user anonymity can be achieved using the ring signatures in contradiction to the group signature. Besides, the users of the ring signatures are not fixed to a group. Here, an ad-hoc group can be formed with certain users by the signer while broadcasting a message. The signer used other users' public keys without their knowledge to hide his identity. The main security requirement of the ring signature is Anonymity and unforgeability. Thus, a ring signature is used in this paper to implement the authentication approach. It provides high-speed authentication and anonymity to nodes. Furthermore, the nodes are prevented from the adversary that tries to obtain the node's location through the analysis of broadcasted messages. Ring-LWE signature is used to verify the nodes using three algorithms as given below:

- *Key generation*: It receives security parameters as input to give public key pk_n and private key sk_n as output. Here,n denotes the IoT members of ring structure$n = 1,2,...N$
- *Signature generation with keyword*: It produces a ring signature S_n by taking a keyword message$K$, the secret key of sender $sk_{se}$and public keys of all IoT members P. Here $se \in \{1,2,...N\}$.
- *Signed keyword verification:* It uses public keys of all the IoT devices P, a keyword message K, and signature S_n to verify whether the signature is valid or not. The respective IoT device is authenticated to start data sharing when it returns the valid output.

***Hybrid encryption (HE)*:** A HE method combines the Key encapsulation process (KEP) and a Data encapsulation process (DEP). This HE process includes three algorithms: key generation, encryption and decryption. Here, KEP and DEP are used to encrypt the session key and message *m*respectively. To differentiate the security key of the authentication and data sharing parts, the public/secret key pair of the data-sharing part is denoted as(pk_{ds}, sk_{ds}).

- Key generation: The key generation process of KEP is executed by considering the security parameters to return public/secret key pair(pk_{ds}, sk_{ds}).
- Encryption: A public key pk_{ds} and a message*m*are considered as inputs. Initially, the encryption process of KEP is executed to return a session key *ssk*and a ciphertext C_{T0}. Subsequently, it executes the encryption algorithm of DEP to return a ciphertext C_{T1}. The final output of this algorithm is $C_T = (C_{T0}, C_{T1})$.
- Decryption: Inputs are a secret key sk_{ds} and a ciphertext C_T, executes the decryption algorithm of KEP to return a session key *ssk*. Also, executes the decryption algorithm of DEP to return a message*m*.

4. System model for authentication and data sharing

Internet of Things (IoT) introduces an active connection between smart devices for exchanging and collecting data over the Internet. Fig. 1 illustrates the generic structure for IoT networks with four circumstances: smart home, transportation, wearable and community. Here, the smart devices presented in each circumstance are linked to the Internet via a neighbouring gateway. Similarly, the users such as home

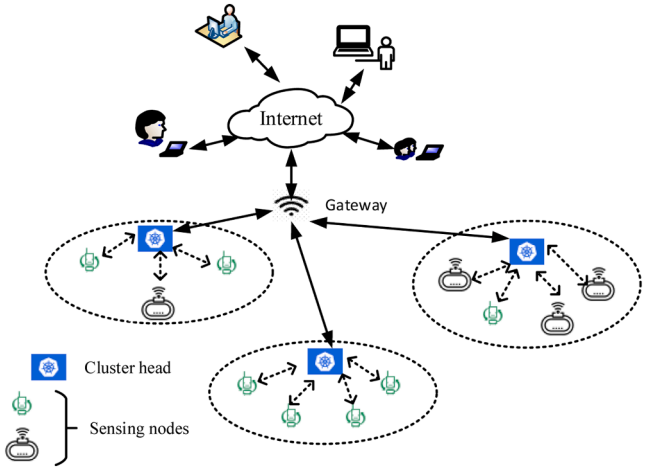

Fig. 2. Hierarchical IoT network.

users, doctors, and traffic monitoring users use the gateway to access the real-time data from the authorized IoT devices.

The hierarchical IoT network is a specific type of generic IoT network. This network contains several nodes, including gateway nodes, cluster head nodes and sensing nodes, as illustrated in Fig. 2. Every application in a hierarchical IoT network has a single gateway node. This network model has a hierarchy between gateway nodes, cluster head nodes and sensing nodes. Here, the sensing node transmits its sensed data towards its cluster head node, and then the cluster head node transmits the data to the gateway node. Every entity transmits its data via a wireless link in a hierarchical IoT network. Nowadays, emerging applications in the area of IoT have been developed to interconnect numerous digital devices to the internet to exchange information. However, while exchanging information from one device to another, security is the major concern because of authentication and privacy. Here, the entities users, sensing nodes and cluster heads presented in the hierarchical IoT are not trustworthy. The most significant entity of the hierarchical IoTs is the gateway node. Hence, it is considered a trusted node and is not compromised in any situation.

This paper introduces a new ring network structure with local authentication and data sharing entities. Here, the Ring structures deliver authorization services for locally registered entities (IoT devices), and the same structures can be extended to accomplish trust relations with other rings globally. Hence, smart IoT devices such as sensing nodes and gateway nodes form a ring in the proposed authentication method. After that, public and private keys are generated by every ring member. Then, the sender node generates a lattice-based ring signature. This signature is transmitted along with the keyword to the receiver for verification. The signature is accepted after the successful verification at the receiver node.

In this paper, a new authentication and data sharing scheme is proposed. The overall research work has two main phases: Authentication and data sharing, as shown in Fig. 3. These significant measures are required to achieve secure IoT communication. The proposed method

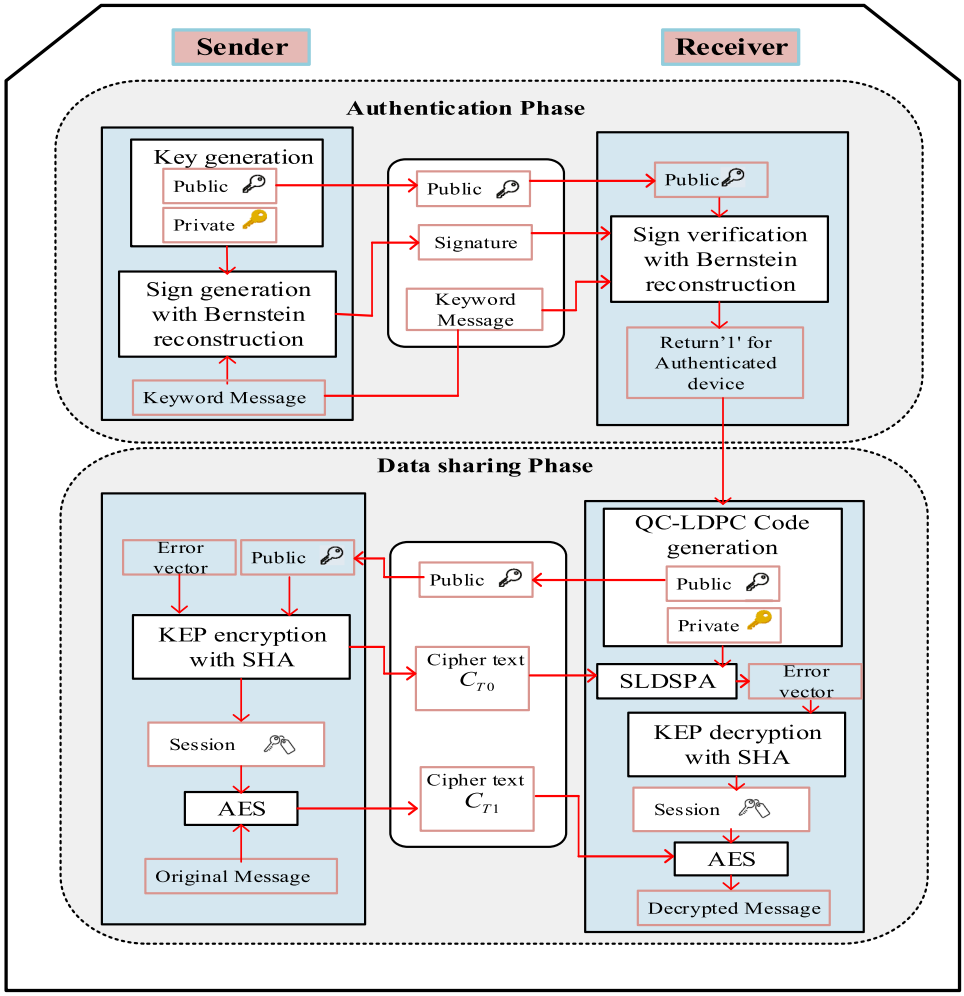

Fig. 3. Overall structure of the proposed post-quantum based authentication and data sharing scheme.

increases security by combining the benefits of lattice-based cryptography and code-based cryptography. Here, lattice-based cryptography is used to increase network security and code-based cryptography is used to improve data security. Hence, every IoT node in the ring executes different key generation algorithms during the authentication and data sharing phases. Here, the authentication phase generates the public and private key pairs by sampling the data from discrete Gaussian distribution G_σ^i. Instead, the data sharing phase generates the public and private key pairs using QC-LDPC codes.

In the first phase, a mutual authentication scheme is developed for IoT devices based on a robust post-quantum Lattice-Based Ring Signature. Here, a ring is formed in which each member is an IoT device. If one node wishes to transmit secured data to others, it generates a post-quantum lattice-based ring signature. This signature is verified by the receiver device to authenticate the sender for data sharing. In this work, the Ring-LWE-based cryptographic scheme is modified by employing a sparse polynomial multiplication with Bernstein reconstruction to increase computation speed.

In the second phase, a code-based cryptography scheme is proposed for the security necessities of the authenticity and confidentiality of any data exchanged between the devices. The proposed Hybrid encryption (HE) secured the data sharing by introducing a Diagonal Structure-Based QC-LDPC Codes in the key and data encapsulation mechanism. The proposed QC-LDPC Code secure KEM is optimized by considering both *column-wise loop optimization* and Simplified Log Domain Sum-Product Algorithm (SLDSPA). Here, column-wise loop optimization is utilized to minimize the execution time of the constructed code, and SLDSPA is responsible for decoding the constructed code.

4.1. Lattice-based ring signature for IoT device authentication (LR-IoTA)

In this scheme, a ring structure is formed using IoT devices. Subsequently, a public-private key pair is generated by every IoT member. If one node wishes to transmit secured data to others, it generates a post-quantum lattice-based ring signature for the specified keyword based on the private and public keys of the remaining devices. The security of such lattice depends on the average case hardness problem named learning with errors (LWE). This Ring-LWE-based cryptographic method utilized Finite fields precisely. The finite filed addition is simple to implement. However, the multiplication unit increases the cost in terms of time and hardware requirements. Because finite filed multiplication includes sparse and dense polynomial multiplication, the hardware complexity of the polynomial multiplication can be analysed by considering the space complexity in terms of XOR gate, AND gate and delay units. In this work, the Ring-LWE-based cryptographic scheme is modified by employing a polynomial multiplication with Bernstein reconstruction to increase computation speed. In order to reduce the space complexity and delay of polynomial multiplication, the reconstruction process of sparse polynomial multiplication is optimized using the Bernstein approach. The proposed LR-IOTA is based on four algorithms, i.e., setup, key generation, signature generation for keyword and signed keyword verification. During the set-up stage, the system parameters are provided with the knowledge of security parameters. Here, LR-IOTA is considered a random matrix $R_n \in Z_q^i$where $n = 1, 2, \ldots N$represents the quantity of public polynomial constructed uniformly. A detailed description of the remaining stages of LR-IOTA is provided in the following subsections.

4.1.1. Key generation

During this stage, all the IoT devices $n = 1,2,\ldots N$of the ring produce their public-private key pair. This key generation process samples data from discrete Gaussian distribution G_σ^i to generate two secret matrices $\delta_n \in Z^i$ and $\varepsilon_n \in Z^i$. Also, the matrix $\varepsilon_n$is checked by some rejection conditions to ensure exactness and short signatures. The conventional signature generation method [30] restarts the key generation if $|\varepsilon_n| > 7\varpi\sigma$ for any value(n). The proposed key generation process computes ϖ number of massive entries by considering the vector ε_n . Then, all these entries are added together, and the resultant value is compared with $M = 7\varpi\sigma$ to introduce rejection sampling.

At last, the key generation process returns the public key $T_n = R_n\delta_n + \varepsilon_n$ and the secret key matrices $sk_n = (\delta_n, \varepsilon_n)$. Here, $R_n\delta_n$ needs matrix-matrix multiplication. In this scheme, $R_n \in Z_q^i$is chosen as a global constant, so it doesn't need sampling. The key generation process is clearly explained in Algorithm 1.

4.1.2. Signature generation

This algorithm assists the sender device (a member of the ring) in generating a signature by considering a keyword message K. The sender needs the global constant R_n, secret keys $\delta_n$and $\varepsilon_n$to generate the signature.

Initially, the sender device generates polynomials Y_n by sampling the values randomly from $[-E, E]^i$. Then, polynomials $\nu_n, \forall n \in (1, 2\ldots N)$are generated and are added together to get ν. Then the result of ν is concatenated with Kafter performing a rounding operation. In general, $\lfloor\nu\rceil_f$ denotes ν modulo f. Hence, $\lfloor\nu \bmod q\rceil_f$ can be abbreviated as $\lfloor\nu\rceil_{f,q}$. Then Secure Hash Algorithm (SHA) is used to hash the resultant value to ς. After that, the binary outputs of SHA are processed using the encoding function $F(\varsigma)$to give vector $\bar{\varsigma}$. Then, the signs for the sender node S_{se} and the remaining members of the ring structures S_n are generated using the sender's private key sk_{se} and public key of the remaining members of the ring structures pk_n as expressed in (2) and (3):

$$S_{se} = (Y_{se} + sk_{se}\varsigma)R_{se} \tag{2}$$

$$S_n = R_nY_n + pk_n\varsigma; n \neq se \tag{3}$$

After computing the signatures, the sender device computes $\varpi$and executes rejection sampling to ensure that any malicious IoT devices can't excerpt the key of the signer device from the constructed signature. The rejection condition is $|\lfloor\varpi_n\rceil_{2^f}| > 2^{f-1} - M$. This condition also confirms that the verification works even for a short signature. Also, it checks the uniform distribution of the signature within the allowable range $[-E + V, -E + V]$for $V = 14\sigma\sqrt{\varpi}$. When all the above conditions are satisfied, it returns the signature $(S_n, \bar{\varsigma})$ .When the conditions are not satisfied, the signature generation algorithm is restarted to generate another signature over K. Finally, the sender IoT device sends this signature and keyword message K to the receiver. The key generation process is clearly explained in Algorithm 2. Here, steps 9, 11, and 15 need sparse polynomial multiplication to get the values of R_nY_n,$sk_{se}\varsigma$, and $\varepsilon_n\varsigma_n$respectively. Here, the polynomial multiplication is optimized using the Bernstein approach.

4.1.3. Bernstein reconstruction in polynomial multiplication

Consider two in determinates polynomials of degree $i$and these polynomials are expressed in linear form as: $\varepsilon(u) = \sum_{x=0}^{i-1} a_x u^x$and $\varsigma(u) = \sum_{x=0}^{i-1} b_x u^x$in $F_2[u]$with the power of 2 for i. The proposed scheme takes $i$inputs for each polynomial and generatesℓ polynomial elements (PE). Hence, the linear form of PEs of $\varepsilon$and ςcan be denoted as $A_0(\varepsilon), A_1(\varepsilon), \ldots A_{\ell-1}(\varepsilon)$and $B_0(\varsigma), B_1(\varsigma), \ldots B_{\ell-1}(\varsigma)$ respectively. The proposed polynomial multiplication obtains $C = \varepsilon \times \varsigma$based on three multiplications of half-sized polynomials. To generate the PEs for the given $i$andε, ς, two parameters $\lambda = \lfloor.(i+k-1)/k\rfloor$ and $\lambda' = i - (k-1)\lambda$are determined. It determines the slices of $\varepsilon$and $\varsigma$of length $i$at specified index. For $k = 2$and $\ell = 3$, the polynomial elements (PE) can be simply obtained by splitting the polynomials into two halves as mentioned in (4):

$$\varepsilon(u) = \sum_{x=0}^{i/2-1} a_x u^x + u^{i/2} \sum_{x=0}^{i/2-1} a_{x+i/2} u^x;\ \varsigma(u) = \sum_{x=0}^{i/2-1} b_x u^x + u^{i/2} \sum_{x=0}^{i/2-1} b_{x+i/2} u^x \tag{4}$$

From these expressions, two halves of the polynomials $\varepsilon$and ςcan be defined as $\varepsilon_L = \sum_{x=0}^{i/2-1} a_x u^x$; $\varepsilon_H = \sum_{x=0}^{i/2-1} a_{x+i/2} u^xand\varsigma_L = \sum_{x=0}^{i/2-1} b_x u^x$;

$\varsigma_H = \sum_{x=0}^{i/2-1} b_{x+i/2} u^x$respectively. Then, three polynomials are obtained from εas: $A_0(\varepsilon) = \varepsilon_L$, $A_1(\varepsilon) = \varepsilon_L + \varepsilon_H$and $A_2(\varepsilon) = \varepsilon_H$.Similarly, three polynomials are obtained from ς as: $B_0(\varsigma) = \varsigma_L$, $B_1(\varsigma) = \varsigma_L + \varsigma_H$and $B_2(\varsigma) = \varsigma_H$. After that, a pairwise multiplication is done on the PE of $\varepsilon$and ς to get recursive products using (5)

$$C_0 = A_0 B_0;\ C_1 = A_1 B_1;\ C_2 = A_2 B_2 \tag{5}$$

Then, the recursive products should be reconstructed to obtain $C = \varepsilon \times \varsigma$. Here, Bernstein's reconstruction introduces recursive calculations for executing parallel multiplications with a sub quadratic space complexity and a logarithmic delay. Specifically, the number of XOR gates, AND gates and the delay of Bernstein optimized reconstruction are much lesser than conventional sparse polynomial multiplication. Algorithm 3 shows the reconstruction process of sparse polynomial multiplication using the Bernstein approach. The optimized procedure of Bernstein reconstruction is provided in (6):

$$\overline{\mathrm{R}}_0 = C_0 + u^{i/2} C_1; \overline{\mathrm{R}}_1 = C_0\left(1 + u^{1/2}\right); C = \overline{\mathrm{R}}_1 + u^{i/2} C_2 \tag{6}$$

The above formula represents Bernstein's Reconstruction for a single recursion in polynomial multiplication. This approach also applies to computing three half-size products in (4) recursively. Here, parallel computation is introduced to execute the recursive computations. This realization reduced the space complexity and delay of the polynomial multiplications compared to conventional non-recursive multipliers.

4.1.4. Signed keyword verification

It validates the signature by taking the keyword message, signature $(S_n, \bar{\varsigma})$ and public keys P as input. The receiver IoT device runs this algorithm to verify the signature sent by the sender device. Initially, the receiver uses the same encoding function to calculate ς from $(\bar{\varsigma})$. After that, $\overline{\varpi}$ is computed, and it is rounded to ϖ for concatenating it with K. The receiver employs the SHA algorithm on the results of the concatenation process to get $\bar{\bar{\varsigma}}$. When $\bar{\bar{\varsigma}}$ is identical to ς, this algorithm output as logic "1". If the output value is 1, then it indicates that the sender is a valid device and allows data sharing. If the output is 0, it indicates that data sharing should not be allowed for such a device. The verification process is clearly explained in Algorithm 4. In step 4 of the algorithm 4, $T_n\varsigma$requires sparse polynomial multiplication with Bernstein reconstruction.

4.2. Data sharing based on the code-based hybrid encryption scheme

In this section, the proposed code-based hybrid encryption is described in detail for data sharing. The proposed data sharing phase combines the advantages of both KEP and DEP. The KEP module is responsible for making a pair of public/private keys and encrypting a session key using a public key to the end IoT device. Only the end IoT device can decapsulate the session key using its secret key. A DEM phase is responsible for plaintext encryption using the session key to preserve the confidentiality of the data.

Moreover, the key derivation function (KDF) is incorporated in the proposed encryption scheme, which follows the hash function to resist Quantum attacks. Here, KEP generates the key using Diagonal Structure-Based QC-LDPC Codes, which minimizes the sparsity features of QC-LDPC codes. The column-wise loop optimization method can achieve the Optimization of Diagonal Structure-Based QC-LDPC Codes. Hence, the building block that presents the concept of Diagonal Structure-Based QC-LDPC Code generation is initially described. Then the code-based HE is constructed from QC-LDPC codes. In addition, the proposed code-based HE scheme introduced Simplified Log Domain Sum-Product Algorithm (SLDSPA) to decode the QC-LDPC code with less computational complexity.

4.2.1. Generation of diagonally structured QC-LDPC codes

This work uses the lower–upper decomposition of the diagonally structured parity check matrix (PCM) of H_d to construct sparse QC-LDPC Code H_{qc} of size $X \times Y$. Initially, a random parity check matrix$H$is constructed with the size of $Y - X \times X$. Then, $H_L$and $H_U$matrices are obtained by decomposing the Has mentioned in (7) and (8):

$$H = H_U \times H_L \tag{7}$$

$$H_L = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ L_{2,1} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ L_{Y,1} & L_{Y,2} & \cdots & 1 \end{bmatrix}; H_U = \begin{bmatrix} U_{1,1} & U_{1,2} & \cdots & U_{1,X} \\ 0 & U_{2,2} & \cdots & U_{2,X} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & U_{YX} \end{bmatrix} \tag{8}$$

After decomposing the PCM, a diagonal shifting is performed to get the diagonal structure of PCM H. Subsequently, the number of non-zero diagonal elements is determined. Then, the PCM matrix is rearranged using the columns re-ordering strategy. The QC-LDPC codes H_{qc} is constructed by applying cyclic shifting on the sub-matrices whose row and columns are identical. To get the QC-LDPC codes H_{qc}, the code $H$is separated into z no. of sub-matrices with $Y = column\ weight \times z$ and $X = row\ weight \times z$. Then, a circulant shifting is applied to each submatrix.

The execution speed of the code generation process can be increased by optimizing the programming loops. It improves the cache performance using the capabilities of parallel processing efficiently. Here, a contiguous block of memory is used to save each column. Hence, it needs column-wise traversing to process the elements of matrices rapidly. Hence, a column-wise loop Optimization is introduced to generate QC-LDPC code. The algorithmic steps of sample column-wise loop optimization to perform circulant shifting of sub-matrices are provided as follows:

Step 1: Obtain the submatrices $H\_sub(i,j)$ where, $i = 1, 2, \ldots X/row\ weight$ and $j = 1, 2, \ldots Y/column\ weight$ from diagonally structured PCM

Step 2: Determine the number of sub-matrices along the column direction of H using the MATLAB function $size(H\_sub, 2)$

Step 3: Determine the number of sub-matrices along the row direction of H using the function $size(H\_sub, 1)$

Step 4: Iterate the value from $j = 1$to $size(H_sub, 2)$and $i = 1$to $size(H_sub, 1)$to perform the circulant shifting based on column-wise loop Optimization as mentioned in (9):

$$H\_cir(i,j) = circshift(H\_sub\{i,j\}, 1) \tag{9}$$

Step 5: End the program.

If the column-wise traversing time is displayed in seconds, it gives quicker traversing than the row-wise traversing time. After performing the circulant shifting, a column-wise circulant permutation of sub-matrices is performed. To do this, a column vector $P_Y$is considered with random integer values ranging from $[1 : Y]$to construct a random permutation matrix of size z. After performing the circulant permutation, the elements of each row and column are shifted using XOR operation to get H_{qc}. The generated $H_{qc}$of size $X \times Y$ consists of n_0 circulant matrices. Algorithm 5 provides the step-by-step procedure for the Diagonally Structured QC-LDPC Codes generation process.

4.2.2. Hybrid encryption based on QC-LDPC codes

It comprises three steps: key generation, encryption and decryption. It uses a KEP to encrypt a session key and a DEP to encrypt a message m. The algorithms are defined as given below:

***Key generation*:** Obtain the input parameters of QC-LDPC codes, including X, Y, *row weight*, *column weight* and returns a public/private key pair specifically for data sharing(pk_{ds}, sk_{ds}). Here, the sparse matrices generated from the diagonally structured QC-LDPC codes are considered as PCM of private codes, and then a transformation matrix G is selected randomly to obtain a dense matrix $W = H_{qc}G$. The sparse circulant matrices generated from the QC-LDPC codes are expressed in (10)

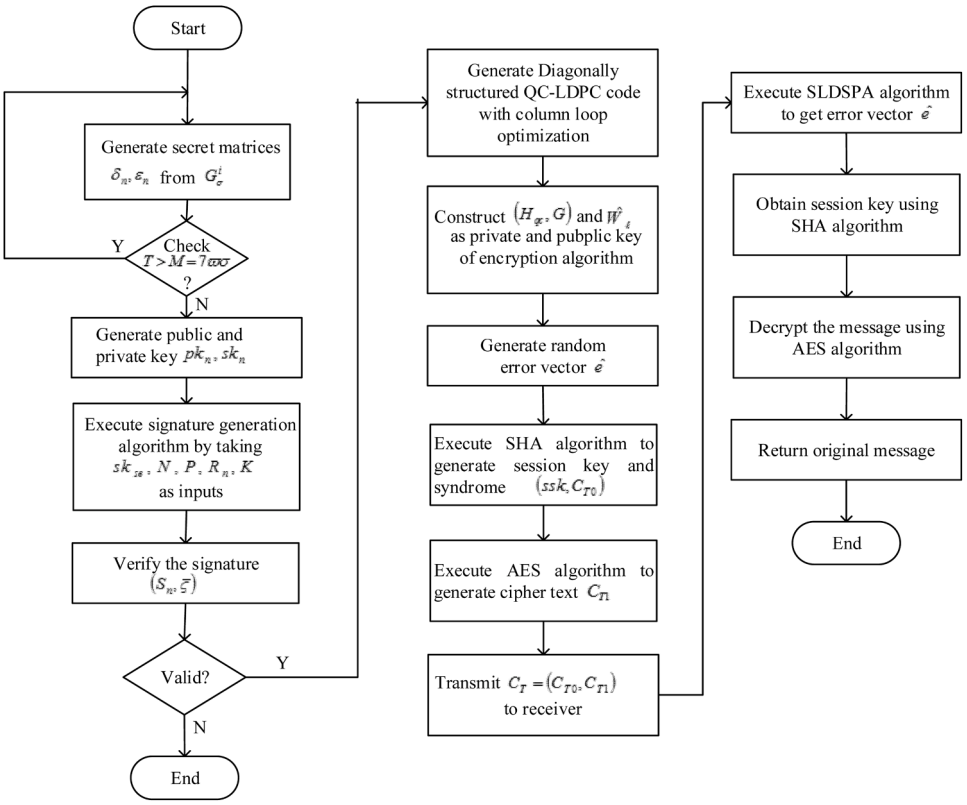

Fig. 4. Flow diagram of proposed authentication and data sharing scheme.

$$H_{qc} = \left[H_{qc}^{0}|H_{qc}^{1}|H_{qc}^{2}|\cdots|H_{qc}^{n_0-1}\right] \tag{10}$$

Where,$n_0$represents the number of circulant matrices. The randomly selected sparse transformation matrix $G$is defined in (11):

$$G = \begin{bmatrix} G_{0,0} & G_{0,1} & \cdots & G_{0,n_0-1} \\ G_{1,0} & G_{1,1} & \cdots & G_{1,n_0-1} \\ \vdots & \vdots & \ddots & \vdots \\ G_{n_0-1,0} & G_{n_0-1,1} & \cdots & G_{n_0-1,n-1} \end{bmatrix} \tag{11}$$

Here, the weight of every row/column of $G$is fixed as$\omega = \sum_{n=0}^{n_0-1}\omega_n$. Then, the matrix $W$is constructed using (12)

$$W = H_{qc}G = \left[W_0|W_1|W_2|\cdots W_{n_0-1}\right] \tag{12}$$

The size of generated matrix W is $X \times Y$. Here, $W_{n_0-1}$is inverted to obtain the matrix $\widehat{W}$using (13)

$$\widehat{W} = W_{n_0-1}^{-1}W = [\widehat{W}_0|\widehat{W}_1|...|\widehat{W}_{n_{o-2}}|I] = [\widehat{W}_{\nearrow}|I] \tag{13}$$

The key generation algorithm of the proposed code-based HE outputs$(H_{qc},\ G)$as private key$sk_{ds}$and $\widehat{W}_{\nearrow} = [\widehat{W}_0|\widehat{W}_1|...|\widehat{W}_{n_{o-2}}]$as public key$pk_{ds}$.

***Encryption*:** This algorithm outputs a cipher text $C_T = (C_{T0}, C_{T1})$using the public key pk_{ds} and a message m. This algorithm first uses the encryption process of KEP that outputs a session key and cipher text pair (ssk, C_{T0}) by taking a public key $pk_{ds} = \widehat{W}_{\nearrow}$ as input. Similarly, the encryption process of DEP outputs a cipher text $C_{T1}$by taking the session key $ssk$and a message m as input. Hence, the output of this encryption module is $C_T = (C_{T0}, C_{T1})$.

KEP encryption: It creates a random error $\widehat{e} \in F_2^n$of weight $wei(\widehat{e}) = y \in Y$and calculates its syndrome $C_{T0} = [\widehat{W}_l|I] \times \widehat{e}^T$.Then, it uses SHA in MAC-mode to generate ssk of length $AptCommandmathcall_k$from $\widehat{e}$. Hence, this algorithm returns(ssk, C_{T0}).

DEP encryption: It uses AES to encrypt the message$m$using$ssk$to generate $C_{T1} = AES(ssk, m)$

***Decryption*:** It is used to obtain original messages by taking a secret key $sk_{ds}$and a cipher text $C_T$as input, this algorithm first uses the decryption process of KEP that outputs a session key $ssk$by taking the secret key sk_{ds} and C_{T0} as input. Similarly, the decryption process of DEP considered the session key $ssk$and C_{T1}as input to return the decrypted message.

KEP decryption: It receives$C_{T0} = [\widehat{W}_l|I] \times \widehat{e}^T = W_{n_0-1}^{-1}W\widehat{e}^T = W_{n_0-1}^{-1}H_{qc}G\widehat{e}^T$. Here, the private key $(H_{qc}, G)$and the decoding algorithm of QC-LDPC code is used to return a random error $\widehat{e}$. Here, SLDSPA is introduced to decode QC-LDPC code. Initially, the syndrome $C_{T0}^{y} = H_{qc}\widehat{e}^T$is obtained with the knowledge of the transformation matrix $G$and $W_{n_0-1}^{-1}$ from the private key. Thus, the proposed QC-LDPC decoding is functioned based on H_{qc} and the syndrome$H_{qc}\widehat{e}^T$.After succeeding in the decoding process, it uses SHA in MAC-mode to generate $ssk$of length $\nearrow k$ from $\widehat{e}$. Here, a bipartite graph is used to represent the input matrix H_{qc}. Also, the terms variable node and check nodes are used to represent the Columns and rows of the matrix H_{qc}respectively. Here, the syndrome of QC-LDPC is of the form $C_{T0}^{y} = [C_{T0}^{0}, C_{T0}^{1}, ..., C_{T0}^{Y}]$.The decoding process of QC-LDPC depends on check node and variable node processing.

The proposed SLDSPA algorithm modifies the check node processing of the conventional sum-product algorithm using a min-sum algorithm. Here, the SLDSPA algorithm is explained by considering the following notations:Lc_ydenotes the prior data of variable node$y$, $L\bar{c}_y$ denotes the posterior data of variable node Y, $L_{Rx,y}$and $L_{Q_{y,x}}$define the extrinsic check to variable message from$x$to $y$and the extrinsic variable node to check message from$y$to$x$respectively. The decoding process can be defined in four steps: Initialization, Check node processing, variable node processing and decoding. During the initialization process, the prior data $Lc_y = -C_{T0}^{y}$and the variable to check the message $L_{Q_{y,x}} = LC_y$are initialized. The check node and bit node processing of the sum-product algorithm are expressed in (14) and (15), respectively.

Table 2
Parameter settings.

LR-IOTA authentication			Data sharing with code-based HE		
Parameter	Definition	Value	Parameter	Definition	Value
i	The degree of the polynomial	512	X	Number of rows in PCM	408
σ	Standard deviation	43	Y	Code length	816
ϖ	Weight value to check the short signature	18	*row weight*	Row weight of circulant matrix	6
E	Bound to sample uniform random value forY_n	$2^{21}-1$	*column weight*	column weight of circulant matrix	3
q	Polynomial bounded modulus	$2^{29}-3$	n_0	Number of a circulant matrix	4
V	Bound to check the uniform distribution of signature	2554.069			
M	Probability of acceptance of key generation	5418			

$$L_{Rx,y} = \log \frac{1+\prod_{y' \in Y(x)\backslash y} \tanh\left(L_{Q_{y'x}}/2\right)}{1-\prod_{y' \in Y(x)\backslash y} \tanh\left(L_{Q_{y'x}}/2\right)}. \tag{14}$$

$$L_{Q_{y,x}} = L\bar{c}_y - L_{Rx,y} \tag{15}$$

Where, $L\bar{c}_y = Lc_y + \sum_{x \in X(n)} L_{Rx,y}$. The proposed simplified algorithm associates the $L_{Q_{y,x}} = LC_y$with the non-zero element ofH_{qc}. Also, it simplifies the check node processing expression by using the relation $2\tanh^{-1}A = \log\frac{1+A}{1-A}$and by considering the sign and magnitude of $L_{Q_{y,x}}$as provided in (16):

$$Lc_{X,Y} = 2\tanh^{-1} \prod_{y' \in Y(x)\backslash y} sign\left(L_{Q_{y',x}}\right) \prod_{y' \in Y(x)\backslash y} \tanh\left(L_{Q_{y',x}}/2\right) \tag{16}$$

Also, it recognizes that the term related to the least$L_{Q_{y',x}}$rules the product term, and hence the product is approached to a minimum value as mentioned in (17):

$$Lc_{X,Y} = \prod_{y' \in Y(x)\backslash y} sign\left(L_{Q_{y',x}}\right) \min_{y' \in Y(x)\backslash y} \left|L_{Q_{y',x}}\right| \tag{17}$$

To simplify the computational complexity, the check node processing of the proposed decoding algorithm considers the non-zero element of H_{qc}in the check node and variable node processing steps. Thus, it determines the non-zeros in the column and row of H_{qc} and are represented as $c1$ and $r1$respectively. Hence, the equation of check node processing and variable node processing is further simplified in (18) and (19):

$$Lc_{x,y} = \prod_{y' \in Y(x)\backslash y} sign\left(L_{Q_{y',x}}(x,c1)\right) \min_{y' \in Y(x)\backslash y} \left|L_{Q_{y',x}}(x,c1)\right| \tag{18}$$

$$L_{Q_{y,x}} = Lc_y + \sum_{x \in X(n)} L_{Rx,y}(r1,y) - L_{Rx,y}(r1,y) \tag{19}$$

Finally, the decoding step determines the decoded result $\hat{e}$using the condition provided in (20):

$$\hat{e} = \begin{cases} 1; if\ L\bar{c}_y < 0 \\ 0; L\bar{c}_y > 0 \end{cases} \tag{20}$$

$\hat{e}$is considered as a valid decoding result. Then, the SHA algorithm is used in MAC-mode to generate *ssk*of length $\ell_k$from $\hat{e}$. The decoding process of the proposed code-based HE using SLDSPA is explained in Algorithm 6.

DEP decryption: It uses the session key *ssk* and a ciphertext C_{T1} to extract the original message as:$m = AES(ssk, C_{T1})$. The complete process of the proposed authentication and data sharing scheme is illustrated in Fig. 4.

5. Simulation results

The proposed LR-IOTA authentication and code-based data sharing approaches are simulated in MATLAB 2018a and Xilinx working platform with an intel 8 GB RAM under the Windows 10 operating system. The proposed LR-IOTA authentication scheme considers the parameter set of 128 bits of security as listed in Table 2. Similarly, the proposed data sharing part that uses code-based HE considers the parameters of QC-LDPC codes.

Here, the LR-IOTA authentication scheme parameters have been chosen by considering the following bound for 128-bit security [43].

a The value of $\varpi$is chosen using the bound $2^{\varpi}\binom{i}{\varpi} \geq 2^{128}$
b The value of $E$and $V$are used in algorithm 2 to distribute the signature within the allowed range. The value of Eshould be represented in the form of power of 2, and it should be greater than $14\sqrt{\varpi}\sigma(i-1)$. Also, the bound for $V = 14\sigma\sqrt{\varpi}$
c The prime number $q$should exceed $2^{2f+1+(\tau/n)}/E$ where, $\tau = 160$and the value of $f$is subjected to $(1-14\sigma\varpi/2^f)^j \geq 1/3$. Here,$j = 1024$and it is chosen using the totient function of Euler $i = \varphi(j)$.

5.1. Adversary model and evaluation criteria

In this section, an adversary model is clearly defined, and evaluation criteria are presented for filling the breach in fairly assessing the authentication method. In this paper, a probabilistic polynomial time (PPT) adversary p_{adv} is assumed. An authentication approach is more secure when each PPT adversary succeeds in breaking the approach with only negligible probability. The abilities of adversaries in our model are listed as follows:

(A1) Adversary p_{adv}can excerpt the private key from the provided set of public keys
(A2) Adversary p'_{adv}can forge the signature through the detection of a collision in the collision-resistant hash function
(A3) p''_{adv} can break the authentication method's security assumption.

The security of the proposed method can be validated through the formulation of certain evaluation criteria as provided follows:

(E1) *Anonymity:* The signer node's private key should not be shown to the adversary.
(E2) *Unforgeability:* The signature generated by the ring members should not be generated by the adversary.
(E3) *Resistance to known attacks:* The scheme should resist various kinds of basic and sophisticated attacks, including Replay Attacks, Man-in-the-Middle (MITM) attacks, Key-Compromise Impersonation (KCI) attacks and Ephemeral Secret Leakage (ESL) attacks.

5.2. Formal security model

In this section, the security model is defined for formally capturing the adversary's abilities and possible attack processes.

Participants. LR-IOTA protocol Φ contains n number of IoT devices, and they are provided with public-private key pairs (pk_n, sk_n) where$n = 1,2,...N$. Here, the IoT node that wants to transmit the data generates a signature utilizing its private and public keys of all other members in a ring. All the ring members can verify this signature without requiring the original identity of the node.

Execution of the protocol: The interaction among an adversary p_{adv} and the protocol participants happen only through oracle queries. It models the abilities of the adversary in a real attack. The execution of the protocol can be defined with the help of a challenge between the challenger (CG) and an adversary nodep_{adv}. CG executes $KG(1^{\lambda})$ algorithm for getting the public-private key pairs (pk_n, sk_n). Then, p_{adv}receives all the public keys$P$. After that, p_{adv}sends many ring signing queries (δ, P, K) to CG for various values of K and δ). CG answers with $S_{\delta} \leftarrow SG(sk_{\delta}, P, K)$. Subsequently, p_{adv} selects random indexes $\mathscr{i}_0, \mathscr{i}_1 \in (1, 2, \cdots n)$ and transmits the query $(\mathscr{i}_0, \mathscr{i}_1 P, K)$ to CG. Here, $\mathscr{i}_0, \mathscr{i}_1$are indexes with $pk_{\mathscr{i}_0} pk_{\mathscr{i}_1} \in P$. CG arbitrarily chooses a bit $\mathfrak{b} \leftarrow \{0,1\}$ for computing the signature $S_{\delta a i_b} \leftarrow SG(sk_{a i_b}, P, K)$ and transmits it to p_{adv}. Finally, the value $\mathfrak{b}' = \{0,1\}$ is guessed by $p_{adv}$and it succeeds in the challenge when $\mathfrak{b} = \mathfrak{b}'$.

Freshness: One of the key points in the description of protocol security is the freshness notion. It portrays the instinctive fact that a session key can't be shown to the p_{adv} inconsequentially. When the session key *ssk* between the nodes is not revealed by p_{adv}, then it is named as fresh.

Notion of security: In general, p_{adv}aimed to predict the signature as same as that of the signature produced by every ring member. The attacker node extracts a private key from P or from a device that signed the keyword at an earlier time to forge the signature. At this condition, the security assumption of protocol Φ is broken byp_{adv}. Hence, the notion of the unforgeability of a signature should be defined. When p_{adv} desires to attack Φ, it should detect the conflict in the collision-resistant hash function or break the security hypothesis of Φ for forging the proposed LR-IoTA protocol.

5.3. Formal security analysis

In this section, a brief security analysis is demonstrated to prove the security necessities of the proposed method. Here, the security of the proposed model is proven under the random oracle model (ROM). One can prove the security of the generated signature by playing a game between an unauthorised node and a challenger.

Game 1 (G1): Consider$\mathscr{A}$as an attacker of the authentication protocol. The public key *pk*is generated by the challenger for the proposed LR-IoTA authentication protocol at a provided security level for running the adversary algorithm. After that, $\mathscr{A}$receive *pk*as input for making h_q oracle hash queries and s_q sign queries randomly to provide a signature with probabilityρ. Thus, the random oracle has $h_q + s_q$quantity of requests. Here, $\mathscr{A}$is considered for solving LWE. G0 represents the running of attacker $\mathscr{A}$on the actual LR-IoTA authentication protocol. In G1, the ROM simulation replaces the sign queries of G0. Initially, the sign query handler of G1 selects the binary string $\overline{\varsigma}$ uniformly with τ -bit security. Then the signature is randomly sampled as $S_n \leftarrow G_{\sigma}^{i}$and determine $\varpi \leftarrow S_n - T_n \varsigma (\bmod q)$and check whether hash (i.e., SHA) has been defined on $(\lfloor \varpi \rfloor_{f,q}, K)$ or not. If the condition is true means, the game is aborted. Otherwise, the hash $SHA(\lfloor \omega \rfloor_{f,q}, K) = \varsigma$ is programmed to return $(S_n, \overline{\varsigma})$. If the attacker can distinguish playing G0 from G1, it can solve the LWE problem. Hence, they should be undifferentiated, as mentioned in Lemma 1.

Lemma 1. *When q be prime, and it exceeds* $((2^{(f+1)j+\tau})/(2E)^{g})^{1/j-i}$*and all other conditions are held as in [45], then these G0 and G1 are not distinguishable.*

Proof. In the ROM, the $\overline{\varsigma}$ is considered as independent of Y. Hence, $\overline{\varsigma}$ is decoupled from Yfor showing the changes as statistically negligible. By Lemma 3 of [45], adequate entropy is available in the distributed values $\lfloor RY (\bmod q) \rfloor_f$ because $\overline{\varsigma}$ has uniform distribution in the sequence of τ -bit. Thus, the reliability of our actual signature generation LR-IOTA approach is closer to the initial step of the sign query handler in G1. Also, $\lfloor \varpi \rfloor_{f,q}$ has adequate distribution. Thus, one can neglect the probability of the game being abandoned in G1, as stated in [32]. Hence, the bound for the distinguished probability of G0 and G1 is$s_q(s_q + h_q) \max((2^{f+1}/q)^{j}, 2^{-\tau})$. Hence, the differentiators can't differentiate these two games.

5.3.1. Security proof

The security of the LR-IOTA authentication protocol Φcan also be proven by considering the following theorems:

Theorem 1. *Believe that, Search-LWE and decisional-LWE are hard problems to solve, and hence the proposed protocol Φ satisfies the security aspects of anonymity. In this security aspect, the private key of the sender node should not be extracted by* p_{adv}*utilizing the public keys of all the ring members.*

Proof. Consider p_{adv} for extracting the private key from set P. Here, an alternative PPT algorithm CG is constructed to solve *search*-LWE from the provided pairs (R_n, T_n).

(i) CG receipts the security parameter 1^{λ} as input for the generation of (T_n, δ_n). Then, it generates S_z overK_z utilizing $\delta_z$and all T_n. Here, $z$is selected arbitrarily from $\{1, 2, \ldots, N\}$
(ii) CG transmits1^{λ}, collection of public keys and $S_z$together withK_z to p_{adv}.
(iii) p_{adv} employs a tactic on recipients and produces z' for which $(S_{z'}, K_{z'}) \notin (S_z, K_z)$.

The output distribution of the algorithm CG is the same as solving the *search*-LWE problem. The *search*-LWE problem is hard to solve, and hence the private key cannot be extracted from the provided set T_n. It proves the security of Φin terms of anonymity.

Theorem 2. *An adversary node can't produce a valid signature* S_n*concerning*$SV(K, S_n) = 1$*. Consider* $\rho_r[f_{A,\Phi}^{sign}(\lambda) = 1]$*to define the probability of falsifying the signature.* $coll_{A,\Phi}(\lambda)$*denotes the probability of detecting a collision in the hash function by* p'_{adv}*on the execution of Φ. This kind of attack can be mathematically expressed as:*

$$\rho_r\left[f_{A,\Phi}^{sign}(\lambda) = 1\right] = \rho_r\left[f_{A,\Phi}^{sign}(\lambda) = 1 \cap coll_{A,\Phi}(\lambda)\right] + \rho_r\left[f_{A,\Phi}^{sign}(\lambda) = 1 \cap \overline{coll}_{A,\Phi}(\lambda)\right] \tag{21}$$

To prove the proposed LR-IoTA protocolΦas unforgeable, the phrases on the right side of (21) should be negligible.

Proof. For finding a collision, we need to play game 2 (G2) among the adversary and a challenger. Here, the challenger receives a security parameter to generate $(T_n, sk_n)$and transmit $P = \{T_1, T_{2,\ldots T_N}\}top'_{adv}$. Then, p'_{adv}send a request to get signatures over the messages k_j from a challenger. The challenger produces signatures $S_j$on entire messages $k_j$and transmit to p_{adv}. Then, p_{adv}determine

$$\varsigma = SHA\left(\left\lfloor \sum_{n=1}^{N} R_n S_n^j - T_n \varsigma \right\rfloor_{f,q} \| k_j\right) \text{for some } k_j; K \notin k_j. \tag{22}$$

If the challenger algorithm executes p'_{adv}as a subroutine then, the result of the challenger is similar to the result of p'_{adv}while p'_{adv} executes

the proposed algorithms one by one because the output distribution of the challenger algorithm is equal to finding a collision in the collision-resistant hash function, which is insignificant. Hence, the first phase of (21) is proved.

The next phrase of (21) can be proven by considering an Adversary p'_{adv} that breaks the security statement of LWE. Here, p''_{adv}*transmitsP* to p'_{adv}. p'_{adv} sends request to p''_{adv} for getting certain signatures over the messages k_j. Then, p''_{adv} used signing oracle to generate signatures. Finally, p'_{adv} generates an alternative signature over the message k_j, here $K \notin k_j$. In this case, a collision does not happen, and $\Phi$is attacked. This means that p''_{adv} generates the fake signature. Instead, p''_{adv} determines a valid signature only when p''_{adv} knows the private key of the challenger $s_{ch} \in sk_n$from P. But, the search LWE statement proves that the private key cannot be determined from the public key. That is,

$$\begin{aligned}\rho_r\left[f_{P'_{adv},\Phi}(\lambda)=1\right] &= \rho_r\left[f_{P'_{adv},\Phi}(\lambda)=1 \cap \overline{coll}_{P''_{adv},\Phi}(\lambda)\right] \\ &= \rho_r\left[findsearchLWE_{P''_{adv}}(\lambda)=S_{ch}\right] \end{aligned} \quad (23)$$

where $\rho_r[findsearchLWE_{P''_{adv}}(\lambda) = S_{ch}]$represents the probability of detecting a secret key from a public key set. It is negligible. It proves the second phrase of (21) and unforgettable of LR-IoTA.

5.4. Attack model

In this section, different attack models are considered to prove the Resistance of the proposed protocol against known attacks such as Replay Attacks, Man-in-the-Middle (MITM) attacks, Key-Compromise Impersonation (KCI) attacks and Ephemeral Secret Leakage (ESL) attacks.

5.4.1. Replay attack

In the proposed authentication scheme, every node generates the signature based on private key sk_n and public keys P. Also, it updates the private/public key pairs periodically; hence, the pairs are inoperable while changing the ring members. Hence, the attacker will be stated as invalid even if it transmits a valid message–signature pair. Hence, the data will not be shared with the attacker's device. Hence, the proposed authentication scheme is resistant to replay attacks.

5.4.2. Man-in-the-Middle (MITM) attack

Every IoT device produces its public-private key pair in the proposed authentication scheme. Suppose an IoT device wishes to transmit secure data to others. In that case, it generates a lattice-based ring signature based on a keyword message K based on its private key and public keys of the remaining ring members to get authentication from the receiver node. Consider an attacker attempts to capture the keyword message and tried to fabricate them. However, the fabrication is not possible. Because the sender node executes a key generation algorithm to generate public-private key pairs. An attacker receives the public keys and attempts to excerpt the secret key of the sender node for generating a tuple $S_{se} = (Y_{se} + sk_{se}\varsigma)R_{se}$. To do this, it should resolve the search-LWE problem. But the search-LWE problem is so difficult to solve. Hence, the attacker can't get the private key from the public key set easily. Therefore, the attacker makes some random ring signatures using $(se,P,K)$and transmit them to the receiver for different values of message K and index se of sender node having $pk_{se} \in P$. But the signatures generated by the attacker node are non-uniform. As an alternative, the signature generated by the sender node is uniform. It shows the presence of an attack. Accordingly, our authentication protocol withstands the Man-in-the-Middle (MITM) attack.

5.4.3. Key-Compromise impersonation (KCI) attack

In this attack, the attacker knows the secret key of the signer node. But the attacker can't impersonate the receiver to the signer. Because the proposed method executes two levels of security. During the authentication phase, the attacker makes several ring signing queries $(se, P, K)$to sender. It responds to the attacker with a challenging signature. Then, the attacker guesses the code $\bar{\varsigma} = \bar{\bar{\varsigma}}$ due to the knowledge of the private key of the signer node and authenticating the sender to start data sharing. If the attacker tries to impersonate the receiver to sender during data sharing, the sender generates a private key using QC-LDPC code. Then, it computes its public syndrome (CT) and symmetric key by selecting a random variable $\hat{e}$. This session key encrypts the message and sends it to the responder. Consider the attacker knows the private key of the initiator $(H_{qc}, G)$and captures the message. The attacker should know the session key for impersonating the receiver. However, the attacker can't compute the session key because the decryption algorithm recovers $\hat{e}$ from C_{T0}using the private key of the receiver. But the private key is sparse, and an attacker can't be separate $H_{qc}$from Gusing $W = H_{qc}G$ to recover the secret key of the receiver with the knowledge of the public key and secret key of the sender. Accordingly, the proposed protocol withstands the KCI attack.

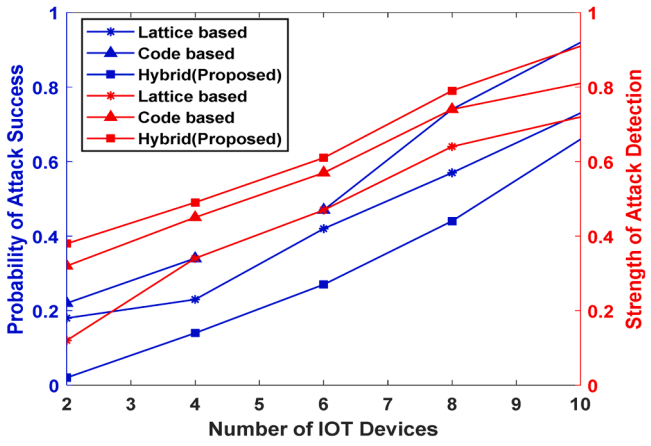

Fig. 5. Probability of attack success over small size IoT network.

5.4.4. Ephemeral secret leakage (ESL) attack

In this protocol, the sender and receiver nodes periodically update the private and public keys. Thus, it becomes operable when the ring members are changed. Here, the key generation process depends on the random samples from a discrete Gaussian distribution G^i_σ. An attacker can't generate the ring signature even if the random samples G^i_σ selected by the sender and receiver is known to him/her due to their periodic updating process. Also, the sender and receiver verify the ring signature to generate the session key in each data sharing session. As a result of this, an attacker is unable to generate the session key even if the random samples $\hat{e}$ selected by the initiator is known to him/her during data sharing. Thus, this protocol withstands the ESL attack.

The proposed method increases security by combining the benefits of lattice-based cryptography and code-based cryptography. Here, lattice-based cryptography is used to increase network security and code-based cryptography is used to improve data security. However, the existing methods used either lattice-based cryptography or code-based cryptography to achieve network and data security. As a result, they decrease the level of security in IoT networks regarding the possibility of attack detection. Fig. 5 compares two performance measures in a single graph, such as the probability of attack success and strength of attack detection. Here, the probability of attack success represents how efficiently an attacker can hack the devices and force them to behave maliciously. But, the strength of attack detection refers to how efficiently the cryptosystem can discriminate the authenticated devices and malicious ones.

Here, a series of experiments is conducted by considering different attack models and a different number of IoT members in a ring. Initially,

Table 3
Comparative analysis of polynomial multiplications.

Multipliers	Slices	LUTs	Flip Flops	Time cost τ_{mul} (ms)
NTT [31]	251	–	–	4.14
Adaptive NTT [30]	545	1576	361	11.11
SPM [40]	127	393	240	7.40
Proposed with Bernstein reconstruction	72	72	64	0.811

nine separate ring structure is formed by considering each with a different number of IoT devices (2 to 10). Here, the possibility of attack success and the strength of attack detection of each ring member are analysed by considering three experiments with three different attacker nodes to execute the MITM attack, KCL attack and ESL attack separately. Here, the first attacker node tried to fabricate the hash value $\bar{\varsigma}$ by sending 100 random ring signatures using (se, P, K). The second attacker tried to decrypt the message using the knowledge of the private key of the signer node and 100 random values of $\hat{e}$. Also, the third attacker made 100 trials to get authentication from the signer node over certain time intervals using the knowledge of a random sample G_{σ}^{i} and tried to get original data using the knowledge of $\hat{e}$. Hence, a total of 300 random attacks determines the average performance of the possibility of attack success and the strength of attack detection on each ring structure. However, the proposed scheme outperforms other methods because of the usage of both lattice-based cryptography and code-based cryptography. This analysis proves the requirement of hybrid methods in IoT networks.

5.5. Performance analysis

This section presents the performance analysis of our LR-IOTA authentication protocol and data sharing with code-based HE in terms of the efficiency measurement features (i) communication cost, ii) hardware requirement and iii) computation cost. Here, the performance of the proposed authentication and data sharing parts are validated separately in the measurement above attributes.

5.5.1. Communication cost

The total communication cost of the LR-IOTA authentication protocol depends on the size of the public key p_k, secret key p_k and signature S. The size of p_k is $i \times j * \lfloor log_2(q) \rfloor$. The secret key size is $(i^2 + i \times j) \times \lceil \log_2(14\sigma) \rceil$. Also, the signature size is $i \times \lceil \log_2(2E) \rceil + 256$. Similarly, the communication cost of data sharing with code-based HE is the size of p_{kds} + size of s_{kds} + size of cipher text C_T. Because the initiator sends the public key p_{kds} and cipher text C_T. Then the responder decrypts the message using a secret key s_{kds}. Here, QC-LDPC codes with length Y is adopted with $n_0 = 4$. Hence, $Y = n_o \times p$, where, p represents the size of the circulant matrix. The proposed data sharing unit designed a public key $\widehat{W}_{\nearrow} = [\widehat{W}_0 | \widehat{W}_1 | ... | \widehat{W}_{n_{o-2}}]$. Hence, bit-size of the p_{kds} is $(n_o - 1) \times p$. The size of the secret key in bits is $n_o(\kappa + Y \log(p))$; where $\kappa = 18$ signifies *row weight* $\times$ *column weight*. Hence, total communication cost is the size of p_k + size of s_k + size of cipher text C_T.

5.5.2. Hardware requirement and computation cost

In this paper, FPGA-based implementation is considered to evaluate different design characteristics of the proposed models, including slice, LUT, Flip flops, and computation time. The costliest part of the ring-LWE-based authentication scheme is polynomial multiplication. A Bernstein reconstruction is introduced to reduce the hardware complexity of the polynomial multiplication. Table 3 validates the FPGA design characteristics of the proposed multiplier design by comparing it with the existing polynomial multiplications such as Number Theoretic Transform (NTT) based polynomial multiplier [31], adaptive NTT-based polynomial multiplier [30] and sparse polynomial multiplication [40].

Table 4
Comparative analysis with existing Ring-LWE schemes.

Method	Multipliers	LUTs	Flip Flops	Slices	Delay (ms)
Liu et al. [48]	oSPMA	317	198	103	3.00
Zhang et al.'s [49]	Extended oSPMA	699	705	265	3.33
Liu et al. [50]	NTT	–	–	8680	4.25
Feng et al. [51]	Stockham NTT	1307	889	406	12.50
Wong et al. [47]	Karatsuba Algorithm	1125	1034	394	2.97
Proposed	Bernstein reconstruction	486	235	124	2.43

Table 5
Computation cost of LR-IOTA in terms of sparse polynomial multiplication.

Method	Signature generation cost	Verification cost
Tian et al. [46]	$j\tau_s + j(N+1)\tau_{mul}$	$j(N+2)\tau_{mul}$
Liu et al. [34]	$2N\tau_{mul}$	$2N\tau_{mul}$
Mundhe et al. [40]	$N\tau_{mul}$	τ_{mul}
LR-IOTA	$N\tau_{mul}$	τ_{mul}

Here, NTT is a Fast Fourier transform theorem generalization. It needs exponential factors to determine the multiplication of two polynomials. However, the exponential computation's hardware complexity is much higher than ordinary arithmetic units. Also, the existing approaches did not optimize the reconstruction process of polynomial multiplication. Instead, the proposed multiplier design used simple arithmetic computations using XOR and AND gates. Also, the delay factor has been reduced by introducing Bernstein reconstruction recursively. Hence, the hardware requirement of the proposed multiplier design is much better than the existing designs.

In Table 4, the proposed LR-IOTA with Bernstein reconstruction is compared with earlier implementations of the Ring-LWE crypto processors have been implemented using different polynomial multipliers. Simulation results describe that the structure from [48] is an area-efficient design while its throughput is very low. Instead, the hardware complexity of NTT is high but produces high throughput. Alternatively, Stockham NTT [51] consumed a small area, whereas its speed is very low. The delay of the Karatsuba Algorithm [47] is much lower than all other methods. The proposed model determines the best trade-off between area and delay due to considering Bernstein's reconstruction in polynomial multiplication. Because it used simple arithmetic computations using XOR & AND gates and reduced the number of bit additions of one recursion in the Karatsuba Algorithm.

The proposed polynomial multiplier design with Bernstein reconstruction plays an important role in the LR-IOTA authentication protocol's signature generation and verification processes. Hence, the computation cost of the authentication protocol is analysed in terms of sparse polynomial multiplication in Table 5. Here, τ_{mul} and τ_s denotes the time cost and sampling algorithms of polynomial multiplication. In Table 4, the time required for the computation of addition, subtraction, or hash operations is neglected. Because the computation overhead of sparse multiplication is very high compared to other addition, subtraction and hashing functions. Here, the Signature generation requires only one τ_{mul} for each ring member. That is, the sender node computes its signature using $S_n = R_n Y_n + pk_n \varsigma$ and signature of remaining members using $S_{se} = (Y_{se} + sk_{se}\varsigma)R_{se}$. Hence, a total of $N\tau_{mul}$ are required for a single ring structure with N members. Similarly, the receiver node verifies the signature using $\overline{\varpi} \leftarrow S_n - T_n \varsigma (\bmod q)$. Hence, it requires only one τ_{mul}. The Signature generation and verification cost of the authentication scheme proposed by Mundhe et al. [40] and the proposed LR-IOTA are equal. However, Mundhe et al. [40] developed the authentication scheme based on NTT and SPM approaches. The computation time τ_{mul} in Table 3 proves the effectiveness of the time cost

Table 6
Comparative analysis of total computation cost for authentication schemes.

Method	Device/Environment	Key generation (ms)	Sign generation(ms)	Verification (ms)
Wang et al. [30]	Intel core i5–4570S, 2.90 GHz	0.492	16.812	1.327
Mundhe et al. [40]	Intel core i5–4570S, 2.90 GHz	0.347	15.679	0.981
HAN et al. [39]	Intel core i5–4570S, 2.90 GHz	0.508	17.567	1.629
Shim et al. [24]	Intel core i5–4570S, 2.90 GHz	0.542	16.375	1.428
LR-IOTA (Proposed)	Intel core i5–4570S, 2.90 GHz	0.288	13.299	0.735

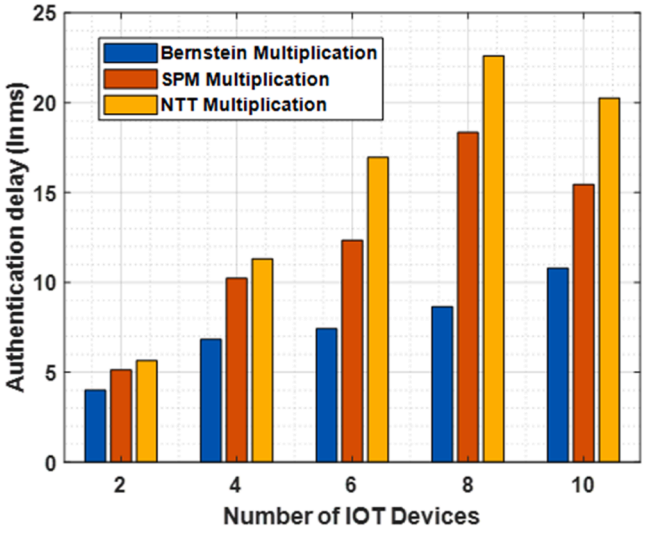

Fig. 6. Total authentication delay of LR-IoTA.

of the proposed Bernstein reconstruction against the conventional NTT and SPM approach.

However, the total computation cost of the proposed LR-IOTA authentication protocol is compared with the existing methods in Table 6. It does not feel fair the comparison against methods that are not specially designed on the same equipment and environment. Hence, the most popular authentication schemes proposed by Wang et al. [30], Mundhe et al. [40] and HAN et al. [39] are all implemented by choosing the same Intel Core i5–4570S, 2.90 GHz Environment for a fair comparison. The total computation cost of Shim et al. is high compared to other methods because this scheme increases the timing overhead due to the requirement of homomorphic encryption for hiding the blinding factors. HAN et al. [39] did not introduce any efficient polynomial multiplication scheme for the generation of traceable Ring Signatures. Hence, it increases the delay of the sign generation algorithm. Instead, Wang et al. [30] and Mundhe et al. [40] optimized the authentication scheme using NTT and SPM. However, the time cost of these multipliers is high compared to the proposed Bernstein reconstruction. Also, the proposed scheme generates a shorter signature by satisfying certain properties. Thus, it reduces the verification time.

The computation cost of the proposed key generation algorithm depends on the time cost of the Gaussian sampler ($\tau_{GS} = 0.0093$), check time for secret matrix ε ($\tau_{\varepsilon} = 0.0954$)and the time cost of public-key construction using LWE distribution ($\tau_T = 0.1842$). Hence, the total time cost of Key generation is the addition of $\tau_{KG} = \tau_{GS} + \tau_T + \tau_{\varepsilon} \approx 0.3$. Similarly, the signature generation algorithm depends on the time cost of ν polynomial generation using Y sampler ($\tau_{YS} = 0.0054$), hashing time of SHA $\tau_{SHA} = 0.1162$, encoding time $\tau_{enc} = 0.0078$,time required to generate tuple for sender ($\tau_{S_{se}} = 0.6130$), the time required to generate tuple for remaining ring members ($\tau_{S_n} = 5.8079$) and the rejection sampling time for S_n ($\tau_{RS} = 6.7507$). Hence, the total time cost of the signature generation algorithm is$\tau_{SG} = \tau_{YS} + \tau_{SHA} + \tau_{enc} + \tau_{S_{se}} + \tau_{S_n} + \tau_{RS} \approx 13.3$. Likewise, the verification algorithm depends on the cost of $\overline{\omega}$ computation $\tau_{\overline{\omega}} = 0.6124$, encoding time$\tau_{enc} = 0.0078$, and hashing time of SHA$\tau_{SHA} = 0.1162$. Hence, the total time cost of the signature verification algorithm is $\tau_V = \tau_{\overline{\omega}} + \tau_{enc} + \tau_{SHA} \approx 0.7$. The total authentication delay (τ_{AD}) can be computed by adding the Key generation (τ_{KG}), signature generation (τ_{SG}) and verification times (τ_V). Thus,$\tau_{AD} = \tau_{KG} + \tau_{SG} + \tau_V$.

Table 7
Comparative analysis of total computation cost for data sharing schemes.

Method	Device/ Environment	Key generation (ms)	Encryption time(ms)	Decryption time (ms)
Aujla et al. [29]	Intel core i5–4570S, 2.90 GHz	1.8153	3.7484	5.7591
Ebrahimi et al. [33]	Intel core i5–4570S, 2.90 GHz	1.4897	3.4187	6.1913
Buchmann [32]	Intel core i5–4570S, 2.90 GHz	1.1932	2.4394	5..9612
Phoon et al. [38]	Intel core i5–4570S, 2.90 GHz	0.9761	1.8924	6.6423
Code based HE (Proposed)	Intel core i5–4570S, 2.90 GHz	0.8549	1.5298	5.8430

Fig. 6 compares the total authentication delay of the proposed LR-IOTA with different multiplier units such as Bernstein reconstruction, NTT and SPM on MATLAB working platform with Intel core i5–4570S, 2.90 GHz environment. However, the LR-IOTA protocol with Bernstein reconstruction reduced the authentication delay due to the reduction of some bit additions in one recursion of polynomial multiplication.

Table 7 proves that the proposed data sharing with code-based HE achieved good results compared to the existing encryption methods. Here, the most popular data sharing schemes proposed by Aujla et al. [29], Ebrahimi et al. [33], Buchmann [32] and Phoon et al. [38] are all implemented by choosing the same Intel Core i5–4570S, 2.90 GHz Environment for a fair comparison. Aujla et al. [29] use the same Ring-LWE for authentication and data encryption. Instead, the proposed scheme used Ring-LWE only for authentication, introducing code-based HE to encrypt the data. Hence, the proposed method achieves two-level of security. This simple code-based encryption requires a sparse parity check matrix instead of a lattice structure. It neglects the requirement of polynomial multiplications during data sharing. As a result of this, the key generation and encryption time have been reduced as compared to the existing Ring-LWE-based encryption scheme proposed by Aujla et al. [29], Ebrahimi et al. [33], and Buchmann [32]. Phoon et al. [38] used QCMDPC-based encapsulation, and its computational complexity is high compared to QC-LDPC. Hence, the total computation cost of this approach is increased.

The total computation cost of the proposed data sharing scheme depends on the time cost of encryption key generation (τ_{eKG}), encryption

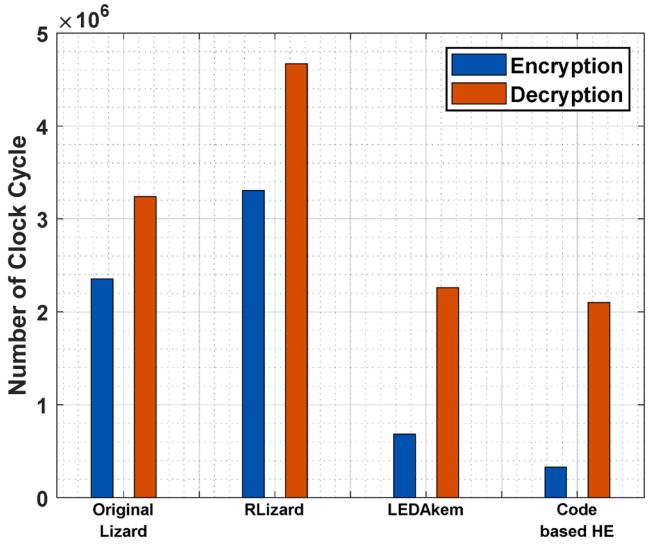

Fig. 7. Comparative analysis of clock cycles required for code-based HE.

time (τ_{enc}) and decryption time (τ_{dec}). However, the computation cost of encryption key generation depends on the time cost of QC-LDPC ($\tau_{H_{qc}} = 0.8522$), the time cost of private key construction ($\tau_{sk_{ds}} = 0.0013$) and the time cost of public-key construction ($\tau_{pk_{ds}} = 0.0013$). Hence, the total time cost of encryption key generation is $\tau_{eKG} = \tau_{H_{qc}} + \tau_{sk_{ds}} + \tau_{pk_{ds}} \approx 0.8549$. Similarly, the computation cost of τ_{enc} depends on the time cost of error vector construction ($\tau_{\hat{e}} = 0.0001$), the time cost of syndrome computation ($\tau_{C_{T0}} = 0.0020$), hashing time of SHA to generate *ssk* ($\tau_{SHA-ssk} = 0.0726$) and the time cost of AES ($\tau_{AES} = 1.4549$). Hence, the total time cost of the encryption algorithm is $\tau_{enc} = \tau_{\hat{e}} + \tau_{C_{T0}} + \tau_{SHA-ssk} + \tau_{AES} \approx 1.5298$. The computation cost of decryption time (τ_{dec}) depends on the time cost of SLDSPA ($\tau_{SLDSPA} = 5.2002$), the time cost of SHA to generate *ssk* ($\tau_{SHA-ssk} = 0.0726$) and the time cost of AES ($\tau_{AES} = 0.570213$). Hence, the total time cost of the decryption algorithm is $\tau_{dec} = \tau_{SLDSPA} + \tau_{SHA-ssk} + \tau_{AES} \approx 5.8430$.

The proposed code-based HE's performance has also been measured regarding the number of clock cycles required for encryption and decryption operations. Here, the speed at which the CPU can carry out instructions is considered to measure the CPU cycles. The CPU with a clock speed of 2.5 GHz can carry out 2.5 billion cycles per second. In Fig. 7, the clock cycles needed for both encryption and decryption on code-based HE are compared with existing Ring-LWE methods (e.g., original Lizard [17], RLizard [17]) and code-based method (e.g., LEDAkem [37]). From this analysis, one can understand that LEDAkem [37] outperforms the Ring-LWE methods such as the original Lizard [17], RLizard [17] while considering the encryption and decryption operations. This LEDAkem [37] requires 6.82×10^5 clock cycles to encrypt the data and 2.26×10^6 clock cycles to decrypt the data. However, the proposed code-based HE requires only 3.27×10^5 clock cycles for encryption and 2.0982×10^6 clock cycles for decryption. The proposed code-based HE requires a smaller number of clock cycles due to the inclusion of column loop optimization and simplified decoding processes.

The performance of the data sharing phase mainly depends on the complexity of QC-LDPC code generation and decoding processes. The proposed data sharing unit constructs the diagonally structured QC-LDPC code with column loop optimization and decodes the error vector using SLDSPA. This key encapsulation and decapsulation scheme reduced the hardware complexity of the data sharing unit, as discussed in Section 4.2. Table 8 compares the proposed key encapsulation and decapsulation scheme with the conventional QC-LDPC code-based encapsulation and decapsulation. The encoding process of the proposed optimized QC-LDPC generation scheme requires 38.46% fewer LUTs than the existing method. However, the proposed decoding process requires 640 slices, 635 LUTs and 646 Flip flops. Hence, the number of slices, LUTs and Flip Flops are 26.43%, 71.42% and 1.82% less compared to the existing method. The proposed scheme reduced the hardware complexity and was delayed due to the consideration of

Table 8
Comparative analysis of key encapsulation and decapsulation processes.

Method	Stage	Slices	LUTs	Flip Flops	Time cost (ms)
Hu et al. [37]	Encoding	33	104	53	2.9
	Decoding	870	2222	658	16.1
Optimized QC-LDPC generation (Proposed)	Encoding	64	64	64	0.317
	Decoding	640	635	646	1.427

Algorithm 1
Key generation $KG(1^{\lambda}, R_n, N)$

Input: public random polynomial matrix R_n, Ring IoT members$N$
Input: Public key pk_n, private key sk_n
1.**for**$n = 1$to N
2. $\delta_n, \varepsilon_n \leftarrow G_{\sigma}^{i}$
3. **for**$n = 1$to i
4. $value[n] \leftarrow absolute(\varepsilon(n))$
5. **end for**
6. Initialize$T \leftarrow 0$
7. **for**$n = 1$to ϖ
8. Initialize maximum$\leftarrow 0$
9. Initialize position$\leftarrow 0$
10. **for** $k \leftarrow 1$to ϖ
11. **if** $value[k] >$ maximumthen
12. maximum$\leftarrow value[k]$
13. $position \leftarrow k$
14. **end if**
15. **end for**
16. $value[position] \leftarrow 0$
17. $T = T +$ maximum
18. **end for**
19. **if** $T > M = 7\varpi\sigma$ then
20. restart
21. **end if**
22. $T_n = R_n\delta_n + \varepsilon_n (\bmod q)$
23. $pk_n \leftarrow T_n$
24. $sk_n = (\delta_n, \varepsilon_n)$
24. **end for**
25. return pk_n,sk_n

Algorithm 2
Signature generation with keyword $SG(sk_{se}, P, K, N)$

Input: sender's private keysk_{se},Ring IoT members N, public key set P, public random polynomial matrix R_n, and keyword message$K$
Output: signature $(S_n, \bar{\varsigma})$
1. **for** $n = 1$to N
2. $Y_n \leftarrow [-\mathrm{E}, \mathrm{E}]^i$
3. $\nu_n \leftarrow R_n Y_n (\bmod q)$
4. **end for**
5. $\nu_n \leftarrow add(\nu_1, \nu_2, \ldots \nu_N)$
6. $\varsigma \leftarrow SHA(\lfloor \nu \rceil_{f,q} K)$
7. $\bar{\varsigma} \leftarrow encode(\varsigma)$
8. **if** $n \neq se$**then**
9. $S_n = R_n Y_n + pk_n\varsigma$
10. **else if**$n = se$**then**
11. $S_{se} = (Y_{se} + sk_{se}\varsigma)R_{se}$
12. **end if**
13. **end if**
14. **end for**
15. $\varpi \leftarrow \nu_n - \varepsilon_n\bar{\varsigma} (\bmod q)$
16.**if** $|\lfloor \varpi_n \rceil_{2^f}| > 2^{f-1} - M \& S_n \leq E - V$ **then**
17. Restart
18. **end if**
19. **return** $(S_n, \bar{\varsigma})$

Algorithm 3
Berns-Mul

Input: ε,ς, k
Output: $C = \varepsilon \times \varsigma$
1. if $i \leq (k-1)^2$then
2. return $\varepsilon \times \varsigma$
3. $\lambda = \lfloor .(i+k-1)/k \rfloor$, $\lambda' = i - (k-1)\lambda$
3. **for**$n = 0$to $k-2$**do**
4. $\varepsilon_n = slice(\varepsilon, n\lambda, \lambda)$; $\varepsilon_L \leftarrow \varepsilon_n$
6. $\varsigma_n = slice(\varsigma, n\lambda, \lambda)$; $\varsigma_L \leftarrow \varsigma_n$
8.$\varepsilon_{n-1} = slice(\varepsilon, (k-1)\lambda, \lambda')$; $\varepsilon_H \leftarrow \varepsilon_{n-1}$
6. $\varsigma_{n-1} = slice(\varsigma, (k-1)\lambda, \lambda')$;$\varsigma_H \leftarrow \varsigma_{n-1}$
8. Determine $A_0, A_1, \ldots A_{\swarrow -1}$,$B_0, B_1, \ldots B_{\swarrow -1}$
10. **end for**
11. **for**$n = 0$to $\swarrow - 1$**do**
12. $C_n = Berns\_Mul(A_n, B_n, e_n)$
13. **end for**
14. **for** $n = 0$ to $\swarrow - 1$**do**
15. Determine C by applying (5) recursively
16. **end for**
17. Return C

Algorithm 4
Signed keyword verification $SV(S_n, \overline{\varsigma}, \mathrm{P}, \mathrm{K}, \mathrm{N})$

Input: signature $(S_n, \overline{\varsigma})$, public key set P, keyword message$K$, Ring IoT members N
Output: Valid 1 or Invalid 0
1. $\varsigma \leftarrow encode(\overline{\varsigma})$
2. Initialize $\varpi \leftarrow 0$
3. **for**$n = 1$to N**do**
4. $\overline{\varpi} \leftarrow S_n - T_n\varsigma(\bmod q)$
5. $\varpi \leftarrow \varpi + \overline{\varpi}$
6. **end for**
7. $\overline{\varsigma} \leftarrow SHA((\lfloor \varpi \rfloor_{f,q} K))$
8. if $\overline{\varsigma} = \overline{\varsigma}$ then return 1
10. else return 0
11. end if

Algorithm 5
Generation of Diagonally Structured QC-LDPC

1. Initialize PCM with random binary values, the size of PCM is $X \times Y$
2. Perform lower-upper (LU) decomposition to determine a new matrix as in (6)
3. Construct diagonal matrix of $i = 1, 2, \ldots X$and $j = 1, 2, \ldots Y$
4. Determine the number of non-zero diagonal components
5. Reorganize PCM of Hcolumn by column
6. Decompose Hinto$z$ no. of sub-matrices with$Y = column\ weight \times z$ and $X = row\ weight \times z$
7. Perform column-wise circulant shifting on sub-matrices
8. Permute the sub-matrices by constructing a random permutation matrix with a column vector P_Y.
9. Execute XOR operation to shift the elements of each row and column
10. Return H_{qc}

Algorithm 6
Decoding process of code-based HE using SLDSPA

1. Initialize$Lc_Y = -C_{T0}^Y$
2. Associate the $LC_y$with non-zero elements of $H_{qc}$to get $L_{Q_{y,x}}$
3. Process the check nodes using the non-zeros in column of H_{qc}
4. Obtain $Lc_{x,y}$using (18)
5. Determine the posterior data of variable node $L\bar{c}_y$ using (15)
6. Process the bit nodes using the non-zeros in row of H_{qc}
7. Obtain $L_{Q_{y,x}}$using (19)
8. Perform decoding process using (20)
9. Return $\widehat{e}$

diagonal structures PCM, simplified decoding structure and close loop optimization, respectively.

6. Conclusion

In this paper, post-quantum cryptographic schemes combined the benefits of lattice-based cryptography, code-based cryptography, and key derivation functions for developing robust authentication and lightweight encryption for resource-constrained IoT devices. The computational complexity of the proposed Ring-Learning with Errors (Ring-LWE) based cryptographic scheme has been reduced by introducing a polynomial multiplication with Bernstein reconstruction. In addition, the Optimization of Diagonal Structure-Based QC-LDPC Codes along with Simplified Log Domain Sum-Product Algorithm (SLDSPA) helps to minimize the execution time of the constructed code and simplify the hardware complexities of the decoder designs respectively. The security analysis of the proposed method validated the security and privacy of the proposed method in IOTs against certain attacks. The simulation results and its comparison with other systems verify the security and cost-efficiency of the proposed post-quantum cryptographic method and show its feasibility for IoT devices. The proposed authentication scheme reduced the signature generation and verification time to 13.299 ms and 0.735 ms due to the consideration of Bernstein's reconstruction in sparse polynomial multiplication. Also, it improved the security level of the encryption scheme using optimized QC-LDPC encoding and SLDSPA decoding. As a result, the encryption and decryption times have been reduced to 1.5298 ms and 5.8430 ms, respectively. Also, the proposed SLDSPA decoding scheme used only640 slices on Xilinx Virtex-6 FPGA. Hence, the proposed cryptographic method can be embraced in the IoT networks where authenticity, security, and lightweight are the crucial features.

Author statement

The submission of work has not been previously published. The content of the work has not been submitted or published anywhere else.

Compliance with ethical Standards

Funding

No funding is provided for the preparation of manuscript.

Conflict of interest

Authors declare that they have no conflict of interest.

Ethical approval

This article does not contain any studies with human participants or animals performed by any of the authors.

Data availability statement

Data sharing not applicable to this article.

Declaration of Competing Interest

No conflict of Interest

Data Availability

No data was used for the research described in the article.

References

[1] T. Fernandez-Carames, From Pre-quantum to Post-quantum IoT security: a survey on quantum-resistant cryptosystems for the internet of things, IEEE IoT J. 7 (2020) 6457–6480, https://doi.org/10.1109/jiot.2019.2958788.

[2] Y. Yang, L. Wu, G. Yin, L. Li, H. Zhao, A survey on security and privacy issues in Internet-of-Things, IEEE IoT J. 4 (2017) 1250–1258, https://doi.org/10.1109/jiot.2017.2694844.

[3] A. Mukherjee, Physical-layer security in the internet of things: sensing and communication confidentiality under resource constraints, Proc. IEEE 103 (2015) 1747–1761, https://doi.org/10.1109/jproc.2015.2466548.

[4] S. Gupta, C. Dutta, Internet of things security analysis of networks using quantum key distribution, Indian J. Sci. Technol. 9 (2016), https://doi.org/10.17485/ijst/2016/v9i48/105551.

[5] M. Aman, K. Chua, B. Sikdar, Mutual authentication in IoT systems using physical Unclonable functions, IEEE IoT J. 4 (2017) 1327–1340, https://doi.org/10.1109/jiot.2017.2703088.

[6] A. Adeel, M. Ali, A.N. Khan, T. Khalid, F. Rehman, Y. Jararweh, J. Shuja, A multi-attack resilient lightweight IoT authentication scheme, Trans. Emerg. Telecommun. Technol. 33 (3) (2022) e3676.

[7] M.F. Aziz, A.N. Khan, J. Shuja, I.A. Khan, F.G. Khan, A.U.R. Khan, A lightweight and compromise-resilient authentication scheme for IoTs, Trans. Emerg. Telecommun. Technol. 33 (3) (2022) e3813.

[8] A. Broadbent, C. Schaffner, Quantum cryptography beyond quantum key distribution, Designs Codes Cryptogr. 78 (2016) 351–382, https://doi.org/10.1007/s10623-015-0157-4.

[9] J. Zhang, S. Rajendran, Z. Sun, R. Woods, L. Hanzo, Physical layer security for the Internet of Things: authentication and key generation, IEEE Wirel. Commun. 26 (5) (2019) 92–98.

[10] O. Althobaiti, M. Dohler, Cybersecurity challenges associated with the internet of things in a Post-quantum world, IEEE Access (2020).

[11] R. Xie, C. He, C. Xu, C. Gao, Lattice-based dynamic group signature for anonymous authentication in IoT, Ann. Telecommun. 74 (2019) 531–542, https://doi.org/10.1007/s12243-019-00705-x.

[12] S. Sciancalepore, G. Piro, G. Boggia, G. Bianchi, Public key authentication and key agreement in IoT devices with minimal airtime consumption, IEEE Embed. Syst. Lett. 9 (2017) 1–4, https://doi.org/10.1109/les.2016.2630729.

[13] S. Ghosh, R. Misoczki, M.R. Sastry, Lightweight post-quantum-secure digital signature approach for IoT motes, Cryptol. ePrint Archive (2019).

[14] A. Lohachab, A. Lohachab, A. Jangra, A comprehensive survey of prominent cryptographic aspects for securing communication in post-quantum IoT networks, Internet of Things 9 (2020), 100174, https://doi.org/10.1016/j.iot.2020.100174.

[15] W. Liu, J. Ni, Z. Liu, C. Liu, M. O'Neill, Optimized modular multiplication for Supersingular isogeny Diffie-Hellman, IEEE Trans. Comput. 68 (2019) 1249–1255, https://doi.org/10.1109/tc.2019.2899847.

[16] M.H. Farzam, S. Bayat-Sarmadi, H. Mosanaei-Boorani, Implementation of super singular isogeny-Based Diffie-Hellman and key encapsulation using an efficient scheduling, IEEE Trans. Circuit. Syst. Regul. Pap. (2020).

[17] J. Lee, D. Kim, H. Lee, Y. Lee, J. Cheon, RLizard: post-quantum key encapsulation mechanism for IoT devices, IEEE Access 7 (2019) 2080–2091, https://doi.org/10.1109/access.2018.2884084.

[18] H. Choudhury, HashXor: a lightweight scheme for identity privacy of IoT devices in 5G mobile network, Comput. Netw. 186 (2021), 107753.

[19] I. Butun, M. Gidlund, Location privacy assured Internet of Things, ICISSP 19 (2019) 1–8.

[20] M. Akil, L. Islami, S. Fischer-Hübner, L.A. Martucci, A. Zuccato, Privacy-preserving identifiers for IoT: a systematic literature review, IEEE Access 8 (2020) 168470–168485.

[21] M.J. Kannwischer, J. Rijneveld, P. Schwabe and K. Stoffelen, pqm4: testing and benchmarking NIST PQC on ARM Cortex-M4. (2019).

[22] Z. Li, D. Wang, E. Morais, Quantum-safe round-optimal password authentication for mobile devices, IEEE Trans. Dependable Secure Comput. (2020).

[23] G. Cheng, Y. Chen, S. Deng, H. Gao, J. Yin, A blockchain-based mutual authentication scheme for collaborative edge computing, IEEE Trans. Comput. Soc. Syst. (2021).

[24] K.-A. Shim, Y. An, Cryptanalysis of lattice-based blind signature and blind ring signature schemes, IEEE Access 9 (2021) 134427–134434.

[25] Q. Wang, D. Wang, C. Cheng, D. He, Quantum2fa: efficient quantum-resistant two-factor authentication scheme for mobile devices, IEEE Trans. Dependable Secure. Comput. (2021).

[26] E. Alkim, L. Ducas, T. Pöppelmann and P. Schwabe, NewHope without reconciliation. Cryptology ePrint Archive (2016).

[27] D. Wang, P. Wang, Two birds with one stone: two-factor authentication with security beyond conventional bound, IEEE Trans. Depend. Sec. Comput. 15 (4) (2018) 708–722.

[28] R. Chaudhary, G. Aujla, N. Kumar, S. Zeadally, Lattice-based public key cryptosystem for internet of things environment: challenges and solutions, IEEE IoT J. 6 (2018) 4897–4909, https://doi.org/10.1109/jiot.2018.2878707.

[29] G. Aujla, R. Chaudhary, K. Kaur, S. Garg, N. Kumar, R. Ranjan, SAFE: sDN-assisted framework for edge–cloud interplay in secure healthcare ecosystem, IEEE Trans. Ind. Inf. 15 (2019) 469–480, https://doi.org/10.1109/tii.2018.2866917.

[30] W. Wang, S. Tian, B. Jungk, N. Bindel, P. Longa, J. Szefer, Parameterized hardware accelerators for lattice-based cryptography and their application to the HW/SW co-design of qtesla, IACR Trans. Cryptogr. Hardware Embedd. Syst. (2020) 3.

[31] C. Du, G. Bai, Efficient polynomial multiplier architecture for ring-LWE based public key cryptosystems. IEEE ISCAS, 2016, pp. 1162–1165, pages.

[32] J. Buchmann, F. Gopfert, T. Guneysu, T. Oder, T.P. oppelmann, High-performance and lightweight lattice-based public-key encryption, in: Proceedings of the 2nd International Workshop on IoT Privacy, Trust, and Security, ACM, USA, 2016, pp. 2–9.

[33] S. Ebrahimi, S. Bayat-Sarmadi, Lightweight and Fault-Resilient Implementations of Binary Ring-LWE for IoT Devices, IEEE IoT J. 7 (2020) 6970–6978, https://doi.org/10.1109/jiot.2020.2979318.

[34] J. Liu, Y. Yu, J. Jia, S. Wang, P. Fan, H. Wang, et al., Lattice-based double-authentication-preventing ring signature for security and privacy in vehicular Ad-Hoc networks, Tsinghua Sci. Technol. 24 (2019) 575–584, https://doi.org/10.26599/tst.2018.9010131.

[35] N. Alassaf, A. Gutub, S. Parah, M. Al Ghamdi, Enhancing speed of SIMON: a light-weight-cryptographic algorithm for IoT applications, Multimed. Tools Appl. 78 (2019) 32633–32657, https://doi.org/10.1007/s11042-018-6801-z.

[36] N. Chikouche, P. Cayrel, E. Mboup, B. Boidje, A privacy-preserving code-based authentication protocol for Internet of Things, J. Supercomput 75 (2019) 8231–8261, https://doi.org/10.1007/s11227-019-03003-4.

[37] J. Hu, M. Baldi, P. Santini, N. Zeng, S. Ling, H. Wang, Lightweight key encapsulation using LDPC codes on FPGAs, IEEE Trans. Comput. 69 (3) (2019) 327–341.

[38] J.-.H. Phoon, W.-.K. Lee, D.C-K. Wong, W.-.S. Yap, B.-.M. Goi, R.C-W. Phan, Optimized IoT cryptoprocessor based on QC-MPDC key encapsulation mechanism, IEEE IoT J. 7 (9) (2020) 8513–8524.

[39] L. Han, S. Cao, X. Yang, Z. Zhang, Privacy protection of VANET based on traceable ring signature on ideal lattice, IEEE Access 8 (2020) 206581–206591.

[40] P. Mundhe, V.K. Yadav, A. Singh, S. Verma, S. Venkatesan, Ring signature-based conditional privacy-preserving authentication in VANETs, Wirel. Pers. Commun. 114 (1) (2020) 853–881.

[41] V. Lyubashevsky, C. Peikert, O. Regev, On ideal lattices and learning with errors over rings, in: Annual international conference on the theory and applications of cryptographic techniques, Berlin, Heidelberg, Springer, 2010, pp. 1–23.

[42] V. Lyubashevsky, C. Peikert, O. Regev, A toolkit for ring-LWE cryptography, in: Annual international conference on the theory and applications of cryptographic techniques, Berlin, Heidelberg, Springer, 2013, pp. 35–54.

[43] Ö. Dagdelen, R. El Bansarkhani, F. Göpfert, T. Güneysu, T. Oder, T. Pöppelmann, A. H. Sánchez, P. Schwabe, High-speed signatures from standard lattices, in: International Conference on Cryptology and Information Security in Latin America, Cham, Springer, 2014, pp. 84–103.

[45] S. Bai, S.D. Galbraith, An improved compression technique for signatures based on learning with errors, in: Cryptographers' Track at the RSA Conference, Cham, Springer, 2014, pp. 28–47.

[46] M. Tian, L. Huang, W. Yang, Efficient lattice-based ring signature scheme, Chinese J. Comput. 35 (2016) 712–718, https://doi.org/10.3724/sp.j.1016.2012.00712.

[47] Z.-.Y. Wong, D.C-K. Wong, W.-.K. Lee, K.-.M. Mok, High-Speed RLWE-Oriented Polynomial Multiplier Utilizing Karatsuba Algorithm, IEEE Trans. Circuits Syst. Express Briefs 68 (6) (2021) 2157–2161.

[48] W. Liu, S. Fan, A. Khalid, C. Rafferty, M. O'Neill, Optimized schoolbook polynomial multiplication for compact lattice-based cryptography on FPGA, IEEE Trans. Very Large Scale Integr. VLSI Syst. 27 (10) (2019) 2459–2463.

[49] Y. Zhang, C. Wang, D.E.S. Kundi, A. Khalid, M. O'Neill, W. Liu, An efficient and parallel R-LWE cryptoprocessor, IEEE Trans. Circuits Syst. Express Briefs 67 (5) (2020) 886–890.

[50] D. Liu, C. Zhang, H. Lin, Y. Chen, M. Zhang, A resource-efficient and side-channel secure hardware implementation of ring-LWE cryptographic processor, IEEE Trans. Circuits Syst. Regul. Pap. 66 (4) (2018) 1474–1483.

[51] X. Feng, S. Li, S. Xu, RLWE-oriented high-speed polynomial multiplier utilizing multi-lane stockham NTT algorithm, IEEE Trans. Circuits Syst. Express Briefs 67 (3) (2019) 556–559.

Swati Kumari is currently pursuing PhD as well as working as an Assistant Professor Computer Science & Engineering Department since January 2017, at Thapar Institute of Engineering & Technology Patiala. Her research area includes post quantum cryptography , Iot Security. She has worked as an Assistant Professor at Jharkhand Rai University . She has published more than 14 research papers in different International refereed journals... She has worked as a lecturer at Cambridge Institute of Technology. she has qualified gate in the year 2012.She has worked as a software developer at Wipro Technology, New Delhi from September 2010-november 2011.

Dr. Maninder Singh is Ph.D. in Computer Science with specialization in Network Security and M.E. in Software Engineering from Thapar Institute of Engineering & Technology, Patiala and B.E. in Computer Engineering. He joined Computer Science and Engineering Department of Thapar Institute of Engineering & Technology, Patiala (India) in 1996 and is presently serving as Professor & Head of the department. His-research interests include Network Security, Cyber Physical Systems & Security and IoT. He has more than 60 research publications in reputed Journals and Conferences. Under his supervision, 7 PhD theses have been awarded and 3 PhD. theses are on-going. He also has more than 46 Master's theses to his credit. He has been Principal Investigator & Consultant in the area of Cyber Physical Systems and Security funded by TCS, DRDO and Corporate houses (details below). His-publication in Springer have been declared by respective journal as one of most cited publication for year 2016. Dr. Singh has authored multiple MOOC sessions and video lecture series for EMRC, CEC India. He has been volunteering his service being on-board member for NBA accreditations. He is heading Computer Science and Engineering Department and centre of Information and Technology Management (CITM), responsible for campus wide Network, ERP, IT Strategy: planning, deployment and management.

Raman Singh received his Master of Engineering in information technology in 2010 and a Ph.D. degree in computer science & engineering in 2016 from Panjab University, Chandigarh India. He is an IEEE member since 2012. He is working as a Lecturer in the School of Computing, Engineering, and Physical Sciences, the University of the West of Scotland, Lanarkshire, the United Kingdom since November 2021. He has worked as a Post-Doctoral Fellow at the School of Computer Science and Statistics, Trinity College Dublin, The University of Dublin, Ireland from February 2020 to February 2021. He also worked as an Assistant Professor in the Computer Science and Engineering Department of Thapar Institute of Engineering and Technology, Patiala (India) from June 2016 to November 2021. His primary research interest is in the area of cyber and network security. He has completed Post-Doc Fellowship in Blockchain and Applied Cryptography and a Ph.D. in Machine learning-enabled Intrusion Detection systems. He has working experience in the following technologies: ns-3 simulator, ndnSIM, Dockers Containers, Python programming language, Blockchain technology: Multichain and Ethereum, Matlab, and C++.

Hitesh Tewari received his BA(Mod), MSc and PhD degrees in computer science from Trinity College Dublin, and is a Fellow of the College. His research interests are in the areas of Network Security, Applied Cryptography, Mobile Communications, and Connected and Autonomous Vehicles. He has co-authored a best-selling book on Electronic Payments. Over the past number of years, he has been very active in the blockchain space, with particular emphasis on decentralized networks and privacy-preserving protocols.