Light Weight Circular Error Learning Algorithm (CELA) for Secure Data Communication Protocol in IoT-Cloud Systems

Mangala N
Senior Director, CDAC
Research Scholar, Dept. of CSE
JNTU Anantapur
Anantapuramu, India

Eswara Reddy B
Director of Admissions, JNTUA
Professor, Dept. of CSE
JNTU Anantapur
Anantapuramu, India

Venugopal K R
Former Vice-Chancellor, Bangalore University
Honorary Professor, Dept. of CSE
University Visvesvaraya College of Engineering
Bangalore, India

***Abstract*—The data driven smart applications, utilize the IoT, Cloud Computing, AI and other digital technologies to create, curate and operate on large amounts of data to provide intelligent solutions for day-to-day problems. Security of Data in the IoT-Cloud systems has become very crucial as there are several attacks such as ransomware, data thieving, and data corruption, causing huge loss to the application users. The basic impediment in providing strong security solutions for the IoT systems, is due the resource limitations of IoT devices. Recently, there is an additional threat of quantum computing being able to break the traditional cryptographic techniques. The objective of this research is to address the bifold challenge and design a light weight quantum secure communication protocol for the IoT Cloud ecosystem. The Ring Learning With Errors (RLWE) lattice based cryptography has emerged as the most popular in the NIST PQC Standardization Program. A light weight Circular Learning Error Algorithm (CELA) has been proposed by optimizing RLWE to make it suitable for IoT-Cloud environment. The CELA inherits the advantages of quantum security and homomorphic encryption from RLWE. It is observed that CELA is light weight in terms of execution time and a slightly bigger cipher text size provides higher security as compared to RLWE. The paper also offers plausible solutions for future quantum secure cryptographic protocols.**

Keywords*—Quantum secure cryptography; homomorphic encryption; lattice-based cryptography; Learning With Errors (LWE); Ring Learning With Errors (RLWE); Circular Error Learning Algorithm (CELA)

I. Introduction

Smart applications, such as smart traffic management, analyzing customer spending habits, smart homes surveillance, or emergency response for remotely monitored patients, provide intelligent solutions by analysing data from multiple information sources. All these applications are dependent on data. Data is essential for evaluation of performance, prediction of future trends, intelligent decision making and controlling risks. The smart application ecosystem has emerged as a natural confluence of IoT, Cloud Computing, Big Data and AI which provide intelligent automation for various real-life problems [1].

The IoT devices are generating huge amounts of data, but since the sensor-actuator IoT devices have low compute capabilities, the data captured by IoT is sent to Cloud for storage and processing. The IoT-Cloud ecosystem is providing a strong foundation to develop smart applications, in almost all possible domain. Real time processing, underlying infrastructure of servers, storage and communication, and operations, are some key aspects of the IoT-Cloud ecosystem [2].

The amount of data in the world is expanding at a very rapid pace. The survey by Statista [3] shows that the quantity of data created, captured or consumed worldwide in previous decade was at 2 zetabytes (ZB) as on 2010, and in this decade it crossed 64ZB in 2020; and is predicted to reach 180ZB by 2025. And nearly 49% of the world's stored data shall be hosted in public cloud environments. Corrupting the data, data theft and denial of access etc. have become a means of livelihood for hackers. Several attacks are being reported on the data driven systems [4]. Concerns about the safety of data were brought up in [5] and [6]. Data has to be protected against access by any unauthorized persons both during transmission over network and while being stored in any devices. An infrastructure built on the Internet of Things must include standards and services that are necessary for protecting, managing, and connecting the various Internet of Things devices and applications.

Data Security, as defined by NIST (National Institute of Standards and Technology, USA), is the process of maintaining confidentiality, integrity, and availability of an organization's data in a manner consistent with the organization's risk strategy [7]. According to IBM [8], Data Security is the technique to protect digital information from unauthorized access, corruption, or theft, throughout its entire life-cycle from data generation, data transportation over network, data storage and retrieval.

Cryptographic techniques are well established methods for securing data both during transit as well as storage, for a wide range of applications over traditional computers, IoT devices and the Internet. However, there are several challenges encountered while securing the IoT-Cloud ecosystem, such as: (i) resource constraints of IoT devices, (ii) size of crypto keys, i.e. the strength of cryptograhic scheme, (iii) shared access by multiple users, (iv) real time response for the applications, (v) increased threats due to development of Quantum Computing etc. These aspects make traditional protocols vulnerable and impracticable, prodding researchers to develop improved cryp-

tographic techniques.

The most important aspect of any cryptographic technique is to make sure that the encryption cannot be cracked by adversaries. At present, the public key cryptographic algorithms are based on mathematical methods such as discrete logarithm problem (ECC, DHKE) and prime factorization of large integers (RSA), which are difficult to solve in traditional computers.

The RSA public key cryptography is based on modulus arithmetic using very large integers and cycling the data in multiple rounds with the public key; and the decryption is based on finding the prime factors of the very large number. Breaking the RSA-1024 by brute-force would take approximately 2^{1024} comparisons, i.e. a total compute operations of about 10^{264} years, on a single CPU system. However, in 2010, the 768-bit RSA, was cracked in about a week by using the large computational capacity available in a supercomputer cluster [9]. Now, with the emergence of quantum computing Shor's algorithm (which is capable of finding the prime factors of an integer, at an exponentially lesser time), the need to develop quantum resistant cryptographic algorithms has been felt [10].

While several strong cryptographic methods are envisaged, the IoT devices are resource-constrained with limited compute and storage making it impractical to perform encryption and decryption in real-time and also for storing the encrypted data. Hence, light weight cryptographic methods are required. Though conventional encryption techniques, would serve the purpose of protecting the data, the modern IoT-Cloud applications require a multi-user shared, searchable and processable data for quick analytics and control. To cater to these requirements, new methods for encrypted data query and processing have been proposed for the IoT-Cloud systems [11].

Recently, the National Institute of Standards and Technology (NIST) announced the Post Quantum Cyptography (PQC) Standardization Program open call to evaluate and standardize public-key cryptographic algorithms for securing against the emerging quantum attacks. Among the 50 cryptographic proposals that were short-listed, 21 were based on lattice techniques; and in that, the RLWE algorithm turned out to be the most popular with a count of 15 entries. The lattice based crytpo techniques are prime candidates as they have strong security, compute efficiency, wide applicability, homomorphic encryption, and quantum resistance. The NewHope [12] is a fast, memory efficient, and simple scheme that is a suitable replacement of RSA and ECC. It provides homomorphic encryption, reasonable sized key and ciphertexts and significantly resistant to side-channel attacks. Internally it uses the Ring-LWE lattice-based technique.

1) Contribution of this paper: The aim of this research is to propose a highly secure yet light weight protocol for the IoT-Cloud ecosystem. The primary contributions of this paper include:

(i) Lightweight Cryptography: The proposed CELA has lower time complexity on account of modulus switching technique which reduces the polynomial ring order in intermediate computations.

(ii) Improved Security: Enhanced security is provided by increasing the key size to twice that of the base RLWE algorithm and using random noise vector. The ciphertext is of bigger size and more complex.

(iii) Plausible Solutions for Quantum Secure Cryptography: Since, the number theory based cryptographic algorithms are susceptible to be cracked by quantum computers, robust security schemes are suggested towards quantum secure cryptography.

2) Organization of the Paper: Organization of the paper is as follows. The related work is summarized in Section II. A brief description of various cryptographic techniques is presented in Section III. The IoT-Cloud architecture and smart applications are described in Section IV. Section V presents the security challenges in IoT-Cloud ecosystem. The design and implementation of the proposed CELA is presented in Section VI. Plausible solutions for quantum secure cryptography in resource constrained IoT-Cloud ecosystem are provided in Section VII and the conclusion is contained in Section VIII.

II. Related Work

A summary of research papers published over the past five years which focus on the consumer-oriented IoTs clouds applications for the purpose of understanding intelligent IoT-Cloud systems is provided in reference [13], [14]. The authors of [15] investigated the factors that influence the acceptance of Cloud Computing, examined the types of attacks that may be launched against cloud-based environments, and proposed solutions for making cloud-based environments more private and secure. The authors of [16] identified privacy schemes that are usable in cloud-based systems that are based on the internet of things to protect data more thoroughly. In [17], a methodology is presented for the investigation of the privacy and security issues that are prevalent in social networks that are based on cloud platforms. The author in [18] investigates both under-explored and common security vulnerabilities associated with cloud systems from a technical point of view, looking into several types of cyberattacks that could occur in cloud environments.

Improvisations to the Attribute Based Encryption (ABE) schemes have been demonstrated in the following papers by - Li Jiguo et al. [19] based on hidden access structure, Zhao et al. [20] based on hidden policies, Han et al. [21] for Traceable and revocable CP-ABE, Das et al. [22] for Multi-Authority CP-ABE and, Zhang et al.[23] based on lightweight Searchable Attribute Based Encryption(LS-ABE).

Wang et al. [24] proposed FABRIC - a fast and secure unbounded cross-domain Proxy Re-Encryption (PRE) scheme for data sharing in multi-user cloud and Deng et al. [25] proposed transforming identity-based encryption (IBE) ciphertext into an identity-based broadcast encryption (IBBE) ciphertext.

Other recent cryptographic efforts include - Logistics Chaos Model theory based encryption for Big data in cloud environment by Abdel-Kader et al. [26]; Catalan number crypto keys for stronger encryption in IoT systems by Saracevic MH et al. [27]; Boneh-Goh-Nissim (BGN) homomorphic encryption for Shared IoT-Cloud environment by S Halder et al. [11]; Differential levels of encryption security for sensitive and non-sensitive data by Atiewi et al. [28]; and CMAP

(Cryptography and Machine Learning based Authentication protocol) by Singh and Saxena [29].

The more recent efforts in lattice based and LWE schemes provide strong security for light weight devices and PQC. Adaptations and implementation in both hardware and software for the Lattice-based Cryptographic schemes, have been demonstrated by AK Sahu et al. [30], N Hamid et al. [31] and Wai-Kong Lee et al. [32]. While these schemes offered homomorphic encryption resistant to post quantum cryptographic attacks, there is scope for improving their performance for using in real-life IoT-Cloud applications.

A summary of latest research in Learning With Errors Crypto Schemes is available in - Xie et al. [33], Kundi et al. [34], Jose [35], Xiaodan Xi et al. [36], Pengzhou He et al. [37], Shahriar Hadayeghparast et al. [38], and Dongdong Xu et al. [39], have optimized the LWE encryption algorithms and implemented them in hardware. The main aim of these efforts is to develop light weight, low power, low silicon-area * delay, i.e. compact and high speed, yet stronger encryption hardware which can be used inside context devices.

Optimizations and implementation of the RLWE scheme on IoT compatible low power RISC and ARM processors, are presented in Ebrahimi et al. [40], Zhe Liu et al. [41] and Zhang et al. [42].

Various quantum resistant security schemes based on XMSS, HBS, and Quantum-AES for IoT-Cloud ecosystems are offered in the works of WK Lee et al. [43], Santosh Ghosh et al. [44] and Roopa Golchha et al. [45]. The qualifying PKE algorithms of the NIST PQC Standardization program are described in [46].

The comparison of research articles published in the last five years in the area of lattice-based homomorphic post-quantum resistant cryptographic schemes is captured in Table I.

While lattice based schemes like RLWE are found to be quantum resistant and homomorphic, they require to be optimized to be practicable for compact resource-constrained IoT devices. Improving the computational speed and key size may make them amenable for real-world IoT-Cloud applications.

III. Preliminaries

Protecting the data from malicious and inadvertent problems is an important task. The various methods used to protect the Confidentiality, Integrity and Privacy of data include - Authentication and access control, Data Masking, Cryptographic techniques, Steganographic techniques, Differential Privacy techniques and Blockchain.

A. Classification of Cryptographic Schemes

Cryptography is a mature and popular data security mechanism employed for several Internet applications including IoT and Cloud. The variations in cryptography are revealed in the classification [49] shown in Fig. 1.

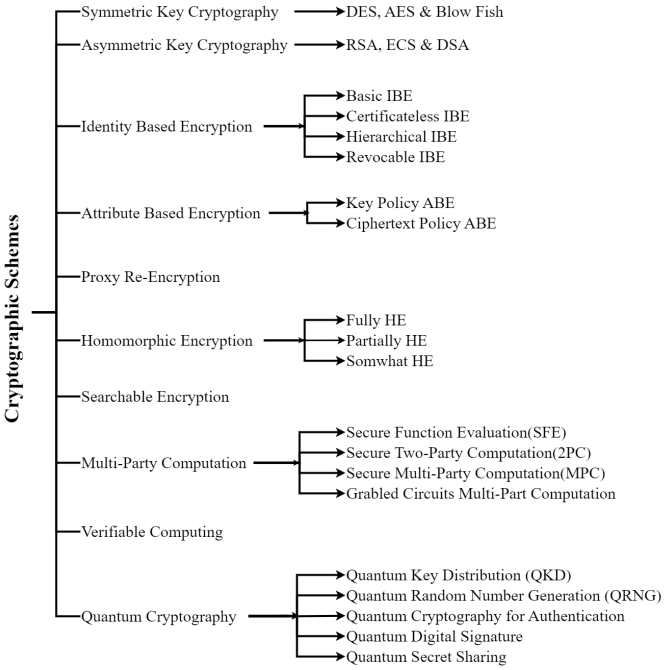

Fig. 1. Classification of cryptographic schemes.

B. Cryptographic Schemes

(i) Symmetric Key Cryptography: This is an encryption system where the sender and receiver have a single common key for enciphering and deciphering the messages. Symmetric Key Systems are simple and fast, but the problem is that sender and receiver have to somehow exchange the key in a secure way. E.g. Data Encryption System (DES).

(ii) Asymmetric Key Cryptography: Under this system the public key of the receiver is used to encrypt the data and send on the channel; and the receiver can only decrypt with his private key. Hence there is no need for key transfer. E.g. RSA, ECC and ElGamal Cryptosystem.

(iii) Key Exchange: The Diffie Hellman Key Exchange (DHKE) provides a mathematical protocol for exchanging the secret key between the sender and the receiver over a potentially non-safe communication channel. The two parties involved in communication, Alice and Bob, start by agreeing on a finite field F_q and a primitive root g, i.e. a generator of the cyclic multiplicative group F_{*_q} where the field F is an algebraic set with two operations (addition and multiplication) and q is the modulus. The pair (q, g) is made public. To obtain the private key, Alice selects an integer a and Bob selects an integer b. Then, Alice sends $g_a \ modulo \ q$ to Bob, who on receiving it, raises it to $b \ modulo \ q$, getting $g_{ab} modulo \ q$. Next, Bob sends $g_b \ modulo \ q$ to Alice, who raises it to a, also obtaining $g_{ab} \ modulo \ q$. Now, even if g is publicly known, the eavesdropper (Eve) cannot obtain $g_{ab} \ modulo \ q$, and hence cannot get a or b secret keys to crack the cipher text.

(iv) Identity Based Encryption: The parties involved in communication identify themselves using a unique identifier such as an email address or user name as their public key. The data is encrypted using the recipient's email address or username. Some popular Identity Based Encryption (IBE)

TABLE I. SUMMARY OF QUANTUM SECURE CRYPTOGRAPHIC RESEARCH WORK

Author Concept/ Model	Algorithm Implementation	Performance Advantages	Research Gaps Future Challenges
AK Sahu et al. [30], 2021, Lightweight Multi-Party Authentication and Key Agreement for IoT e-Healthcare	Lightweight Multi-Party Authentication and Key Agreement (LMPAKA) based on Lattice-Based IBE	- Lowered Power consumption - Resistant to Replay Attacks, Mutual Authentication attacks, Impersonation, Forward Secrecy, Quantum attacks	Include Privacy of the user data in Cloud server
N Hamid et al. [31], 2020, FPGA implementation of Ring LWE based crypto techniques	NewHope, Kyber, Dilithium, and R.EMBLEM	- Hardware solutions are twice faster and energy efficient. - Generic FPGA kernels are usable by any lattice cryptosystems	Evaluate the accelerator for lattice-based post-quantum algorithms
Wai-Kong Lee et al. [32], 2022, Implemented Lattice NTRU polynomial convolution in GPU Tensor cores	Parallelized algorithm of NTRU polynomial convolution	Tensor-core-based polynomial convolution on NVIDIA RTX3080 GPU improved by over 3 times compared to CPU	Implement on gateway device in IoT
Xie et al. [33], 2020, Hardware implementation for BRLWE	Designed and implemented finite field arithmetic AB + C on Xilinx Vivado series Virtex-7 (XC7V2000) and Kintex-7 (XC7K325) devices	The area-time complexities and area-delay product of this design is superior compared to the existing designs, namely #LUT, #FF, #slice.	Extend the hardware for future applications
Y Chen et al. [47], 2021, On the fly keys generation for LWE based Multi-Key FHE	Dynamic Multi-Key FHE algorithm	- Beforehand fixing of number of keys is not required - Distribution helped to reduce encryption workload on cloud	Presently the MK-FHE is single-bit encryption, plan to extend to multi-bit
Yang Su et al. [48], 2020, FPGA-Based Hardware Accelerator for Leveled RLWE FHE	- BGV FHE - NTT-based modular polynomial multiplier unit - LUT Modular Reduction unit homomorphic evaluation function	- Hardware Area saving of 50% - And 20% speed up compared to existing solutions - Implemented on Virtex UltraScale FPGA	Extend to support wider range of security parameters
Kundi et al. [34], 2022, AxRLWE: Approximate RLWE implementation on FPGA for IoT	Using Approximate Computing (AxC), R-LWE is approximated in a multilevel fashion by replacing the normal Discrete Gaussian (DG) with Binomial Distribution in RLWE multiplier module	- The Xilinx Kintex-7 FPGA based AxRLWE designs provide up to $\approx$ 64% area saving compared to the accurate R-LWE - And $\approx$ 3$\times$ lower energy consumption against the R-BLWE	Implement PQC within the tight area/energy budgets of the highly resource-constrained IoT devices
S. Ebrahimi et al. [40], 2020, Fault-Resilient Software implementations of binary Ring-LWE on lightweight IoT microcontrollers	Implemented Fault-Resilient Binary Ring-LWE on 8- and 32-b AVR ATxmega128A1 and ARM Cortex-M0 microcontrollers	- 80ms for encryption and 120ms for Decryption - Resistance to first order attacks- randomization, zeroing, skipping faults - Implementation on tiny microcontrollers suitable for IoT	Manage larger key and ciphertext sizes
Zhe Liu et al. [41], 2020, Software Implementation of RLWE on ARM NEON and MSP430	- Vectorized Iterative NTT for ARM NEON - Optimized SWAMS2 reduction technique for MSP430	- The implementation is 7x faster than ECC of the same security level	Practical application in IoT devices
Jose L et al. [35], 2022, Hardware implementation with optimized arithmetic for InvBRLWE	- Inverted Binary Ring-LWE (InvBRLWE) - Linear-Feedback Shift Register (LFSR)	- LFSR optimizes the resource usage for heavy computations - 71.23% less (area x delay) product than recent designs	Deploy for real world application environments
Xiaodan Xi et al. [36], 2023, Hardware implementation of physical unclonable function (PUF)	- ML Resistant LWE decryption module Lattice PUF - implemented on Xilinx Spartan-6 FPGA	- Resistant to ML attacks on both classical and quantum computers - Prototyped with 2136 challenge- response pairs (CRPs)	Validate the prototype Lattice PUF hardware in field applications
Pengzhou He et al. [37], 2022, RBLWE-based PQC accelerators on the FPGA	- RBLWE Encryption - Implemented on Xlinix Virtex-7	Design is balanced between speed and hardware area usage for compact IoT devices	Enhance for side-channel attack resistance
Shahriar Hadayeghparast et al. [38], 2022, Lightweight cryptoprocessor based on RISC-V for InvRBLWE	Optimized Inv Ring-BinLWE	- 51% faster than RBLWE - Implemented on Xilinx FPGA	Prevent side channel attacks like intentional fault injections and Differential Power Analysis
Dongdong Xu et al. [39], 2022, New scheme of ring-BinLWE based on 2's complement ring	Ring-BinLWE based on the 2's complement	- Lightweight implementation on Spartan 6 FPGA - Compact design; less hardware area	Improve resistance to side channel attacks such as Differential Power
Fan Zhang et al.[42], 2020, Side-Channel Analysis and Countermeasure Design on ARM-Based Quantum-Resistant SIKE	Supersingular Isogeny Key Encapsulation(SIKE)	Achieved higher security along with lower time and memory	Eliminate vertical leakages and resist SCA Attacks
WK Lee et al.[43], 2022, The DPCrypto accelerates PQC using Dot-Product Instructions on GPUs	Lattice based KEM FrodoKEM and SaberKEM implemented on NVIDIA V100 and T4 GPUs using Dot-Product instructions for matrix, hash and other operations in the algorithms	- Implementation very useful for Secure Online Transactions and IoT communications - Executing compute-intensive KEM on GPUs reduces the burden for cloud	Execution speed on T4 is less compared to V100 because of the inherent number of cores in each type of GPU
Santosh Ghosh et al. [44], 2018, A Lightweight Post-Quantum Secure Digital Signature Approach for IoT Motes	Hash Based Signature (HBS), XMSS Scheme, Keccak-400 hash function, WOTS+, SHA-3	- Novel XMSS signature scheme has a small memory footprint - FPGA implementation takes 4.8 million clock cycles; 5x faster than software	Extend the novel signature approach as a ultra light weight end-to-end IoT security
Roopa Golchha et al. [45], 2023, Quantum AES for Fog Enabled Cyber-Physical systems	Quantum AES	- Improved security by integrating quantum cryptographic approach and classical AES - About 10 to 21 Qubits are needed for encryption and decryption of messages	Building the quantum circuits is challenging due to noise

algorithms are Boneh–Franklin IBE, Cocks IBE, Gentry IBE and Waters IBE. Conceptually, the different types of IBE include - Basic IBE, Certificateless IBE, Hierarchical IBE, Revocable IBE and Dual-Receiver IBE.

(v) Attribute Based Encryption: The secret keys required to decrypt the cipher text are related to users' attributes such as roles, age, location, etc. The encrypted data can only be decrypted by users who have the specific set of attributes required to access the data. The main types of ABE schemes are - Key-policy ABE (KP-ABE) and Ciphertext-policy ABE (CP-ABE).

(vi) Proxy Re-encryption: Proxy Re-encryption (PRE) is a mechanism to convert the ciphertexts for one key into ciphertexts for another by using a Pproxy. An encrypted data item is stored in the Cloud and the Proxy decides to re-encrypt it and send to legitimate users. This delegation scheme is useful for resource constrained IoT devices as the computationally intensive encryption are outsourced to the proxy server. It is useful for applications such as e-mail forwarding, content distribution and law-enforcement monitoring.

(vii) Homomorphic Encryption: Homomorphic Encryption (HE) encodes data into ciphertext which can be analyzed and worked with as though it were still in its original form. HE will enable users to perform computations on the encrypted data without first decrypting it [50]. HE is expected to play an important role in Cloud based applications (e-voting, health, finance), allowing companies to encrypt and store sensitive data in a Public Clouds and take advantage of the Cloud Provider's analytic services. In 2009, Gentry proposed with proven safety, a fully HE cryptogram technique; this algorithm relies primarily on learning with error (LWE).

(viii) Verifiable Computation: Verifiable computation is a method where a device having limited computing capabilities makes a request to outsource computation services which it is unable to undertake on its own, and receives assistance from another source. The result of using outsourcing is that the results can be delivered in a time- and cost-effective manner [51]. The verifiable computation does not involve interaction between users who have had their identities validated and protects users' privacy by the comparison and verification of the values that are input and output.

(ix) Searchable Encryption: Searchable encryption is designed to boost the efficacy of searching by adding an index for searching specific information whilst still considering the safety of encrypted data. This method consists of - key generation, encryption, generation of trapdoors, and the search test process. During the stage of key generation, the user is generating as well as saving a private key, while also providing other users with a public key. The data are encrypted, and an index is too generated in order to search for keyword information within the data. By using the private key, the user creates a trapdoor for the term. The data supplied by the sender can be located by the receiver by using the trapdoor [50].

(x) Multiparty Computation: Present day collaborative applications in cloud computing desire the ability to exchange encrypted data with other users whilst also protecting their privacy and retaining their confidentiality. Multiparty computing (MPC), is a cryptographic technique that enables multiple parties to jointly perform a computation on their private data without revealing their inputs to each other. The idea is to divide the computation into smaller sub-computations that can be performed separately by each party using their private inputs. These intermediate results generated by the sub-computations are then combined in such a way that the final result can be obtained without any party having access to the private inputs of the other parties.

(xi) Quantum Encryption: This is an extremely secure encryption based on quantum mechanics, but is not yet widely available due to the technical requirements and high cost of quantum computers. Various types of Quantum Cryotography [52] are evolving such as Quantum Random Number Generation (QRNG), Quantum Key Distribution (QKD), Quantum Cryptography for Authentication, Quantum Digital Signatures and Quantum Secret Sharing.

C. Lattice-based Cryptography

In mathematics, Lattice is a multi-dimensional grid of points that are spaced at regular intervals. Since, lattices are flexible, and posses features required for obfuscation, functional encryption and homomorphism, they are being used for encryption and decryption in the lattice based cryptographic algorithms. Two key concepts about lattice are - Vector and Basis. A vector is a tuple of numbers called the coordinates of the vector, indicating the starting and ending points. Basis is a pair of vectors which can produce other points in the lattice, when added with linear integer combinations of these basis vectors.

Mathematical problems related to the lattice are used to obtain the secret from the public key. The Shortest Vector Problem, is to determine a non-zero vector in the given lattice space whose length is minimal over all non-zero lattice vectors. The Closest Vector Problem is a reduction problem to determine a vector in the lattice which is closest to the target vector. Lattice based mathematical techniques and its variations lend themselves to quantum resistant cryptography. This section explains the lattice based methods.

D. Learning With Errors (LWE)

In LWE-based encryption, the data or message is encrypted by adding a random noise (or error) value to it; the resulting ciphertext is sent over an insecure communication channel. The idea is that even if the attacker tries to learn the secret key by solving a set of equations using the ciphertext and the function, the error component makes it difficult to solve the equations and recover the key. LWE can be mathematically stated using the equation:

$$(B[\,] = A[\,] * s + e)$$

where (A) is a random vector $(a = (a_1, \ldots, a_n))$, sampled uniformly over a vector $(Z/Z_q)^n$, and B is the public key based on $b = (a, s) + m + e$, and s is the secret, the error e is drawn from Gaussian Error distribution and all arithmetic is done $modulo\ q$. Breaking the scheme in LWE scheme is based on the hardness of lattice problems.

Without the error term, an attacker could determine the secret key from a polynomial-sized collection of LWE ciphertexts by some method like Gaussian elimination. Hence the

plain text is encoded along with the errors. For encoding, we use smaller cleartext space (actual messages) and encode cleartexts by putting the messages in the higher-order bits of the plaintext space. E.g., a 10-bit message can be encoded in the top 10 bits of a 32-bit integer, and leave the remaining 22 bits of the plaintext for the error distribution. Hence size of key is large in LWE scheme. However, the key size can be compressed by changing different parameters such as size of the noise polynomial, size of error, dropping lower-order bits, changing security level etc.

E. Ring Learning With Errors (RLWE)

The RLWE is a specialized form of LWE for polynomial rings over finite fields in which the set of polynomials have coefficients in another ring [53]. In algebra, rings are structures that come with the basic operations of addition and multiplication and a multiplicative identity; addition operation is commutative and the multiplication is associative. The elements in the Ring may be integers, polynomials and matrices etc.

A ring R, denoted by $(R, +, *)$ is a set of elements (integers, polynomials, matrices), with binary operations of addition and multiplication.

For a positive integer q, we define, $Z_q = Z/qZ = \{0, 1, ...(q-1)\}$ as the integer quotient ring, with q as its modulus. The set of polynomials is denoted by $Z_q[x]$ where the coefficients of $Z[x]$ are chosen from Z_q [53]. Any polynomial is uniquely identified by its set of coefficients or the coefficient vector.

A polynomial ring R_q is any polynomial in $Z_q[x]$, modulus divided by a modular polynomial $f(x) = x^n + 1$. That is $R_q = Z_q[x]/f(x)$ so that no polynomial has higher rank than n.

The RLWE [53], [54], uses ring polynomial and error vectors over Z, sampled with discrete Gaussian Distribution (χ) (σ) with a small standard deviation σ.

$$(B[\,]_q^{m*1} = A[\,]_q^{m*n} * s[\,]_q^{n*1} + e[\,]_q^{m*1})$$

where $a \in Z_q^n$ are random coefficients, $s \in Z_q^n$ is the secret and $e \in \chi$ and χ is the error distribution with small standard deviation. A typical illustration of RLWE [55] is shown Fig. 2.

$$(B[\,]_{13}^{7*1} = A[\,]_{13}^{7*4} * s[\,]_{13}^{4*1} + e[\,]_{13}^{7*1})$$

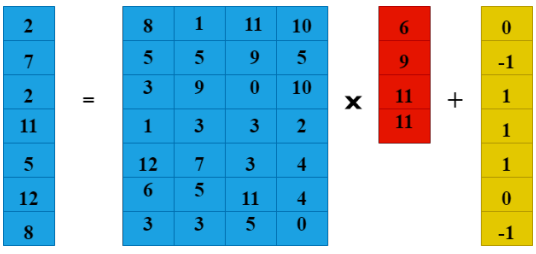

Fig. 2. A typical example of ring polynomial in RLWE.

The RLWE procedure is shown in Fig. 3. The algorithm takes the following inputs:
1. A positive integer n, which is the dimension of the polynomial used in the RLWE problem.
2. A secret key s, which is a vector of n random elements chosen from the ring.
3. A public parameter A, which is a matrix of n random elements also chosen from the ring.
4. A small error term e, which is a vector of n elements chosen from a Gaussian distribution.
The algorithm outputs the public key matrix B that can be used for encryption and the secret key that can be used for decryption. The security of the RLWE scheme is determined by the degree of polynomial n, ciphertext size C $modulus$ q and Gaussian Noise Distribution ($\chi\sigma$).

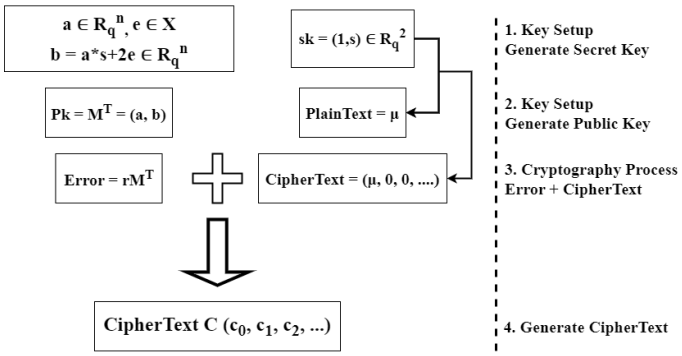

Fig. 3. General RLWE encryption procedure.

In RLWE the keys sizes are much lesser than LWE scheme; RLWE key sizes are almost the square root of LWE key size [56]. If we compare the key sizes for the current security level of 128 bits, the RLWE algorithm would use public keys of 7000 bits in length, whereas the equivalent LWE algorithm requires public keys of 49 million bits. Yet, when compared to the present public key schemes like RSA and ECC which offer public key sizes of only 3072 bits and 256 bits respectively (for a 128-bit security level), the RLWE key sizes are larger.

F. Modulus Switching

In a fully homomporphic encryption, one should be able to add and multiply the ciphertext as if it were in the original plaintext form. The encryption procedure for RLWE-based fully homomorphic encryption is carried out by making use of the elements that are contained within a vector space, and has an Error value in the resulting cryptogram. While performing a homomorphic operation $Eval_{pk}(., c, c)$ of two cryptograms for the multiplication, each having an n-dimension, the encryption expands to n^2.

Adding two homomorphic RLWE ciphertexts produces a resultant ciphertext as the sum of the plaintexts. Homomorphic addition does not increase the size of the ciphertexts. But, homomorphic RLWE multiplication increases the number of components in ciphertexts; that is,

$$(b_1 - a_1.s)(b_2 - a_2.s) = b_1b_2 - (b_1a_2 + b_2a_1)s + a_1a_2s^2$$

Now the ciphertext became larger and has 3 terms. The size of the cryptogram needs to be lessened. The concept of

Modulus Switching [57] can used to contain the growth of ciphertext size.

Modulus switching is based on changing the modulus q to a different prime number q', where q' is smaller than q. By this technique, the size of the polynomial ring can be reduced temporarily during intermediate computations.

The concept of modulus switching is explained with an example. For encoding, smaller cleartext (actual messages) are encode as plaintext by putting the messages in the higher-order bits of the plaintext space. Say, RLWE encodes 10-bit messages in the top 10 bits of a 32-bit integer, and leaves the remaining 22 bits of the plaintext for the error distribution.

In modulus switching, the modulus is changed from q to q', where $q' < q$, and we would like to produce a vector $(a'_1, ..., a'_n, b') \in (Z/q'Z)^{n+1}$, with the new modulus q', that also encrypts message m. In other words, we encrypt $m' = mq'/q$. The operation of $m \rightarrow mq'/q$ shifts up by $log_2(q')$ many bits and then shifting down by $log_2(q)$ many bits.

For example, say the number (x=6) requires only top 3 bits of a 32-bit unsigned integer $((q = 2^{32}))$, i.e., (m = $6 \cdot 2^{29}$). Let (q'= 2^{10}). Then (mq' / q = $6 \cdot 2^{29} \cdot 2^{10} / 2^{32}) = 6 \cdot 2^{29+10-32}$) (= $6 \cdot 2^7$), which stores the same underlying number (x=6), but in the top three bits of a 10-bit message. By modulus switching, the data (x) is in the same position in the plaintext space, but the plaintext space around it has shrunk, thereby making extra room for error bits in the plaintext space.

The RLWE ciphertext contain a certain amount of noise or error which increases with every multiplication; finally the decryption becomes infeasible if error size passes a certain bound. When modulus switching is incorporated and homomorphic multiplication, is performed, there is space in the plaintext to allow the error term to grow towards the most significant bits of the plaintext.

Modulus switching technique involves changing the modulus of the polynomial ring during intermediate computations to reduce the computational complexity of the algorithm. This technique could significantly reduce the execution time of the algorithm while maintaining the same level of security. The idea behind modulus switching is to reduce the size of the polynomial ring temporarily during intermediate computations to make the computation faster. Then, the size of the polynomial ring shall be increased again to its original size at the end of the computation.

IV. IoT-Cloud Application Ecosystem

The smart application ecosystem shows how data is being used for intelligent automation of various real-life tasks. There is a natural convergence of IoT, Cloud Computing, Big Data and AI to provide smart solutions for many real-world problems [1]. Fig. 4 shows the creation, capture and processing of large volume and variety of data getting ingested into the Cloud at high velocity from the network of heterogeneous devices, computers and users forming Big Data. The IoT data are stored and processed in Cloud Storage and the control decision is sent back to the actuators [1], [58], [59].

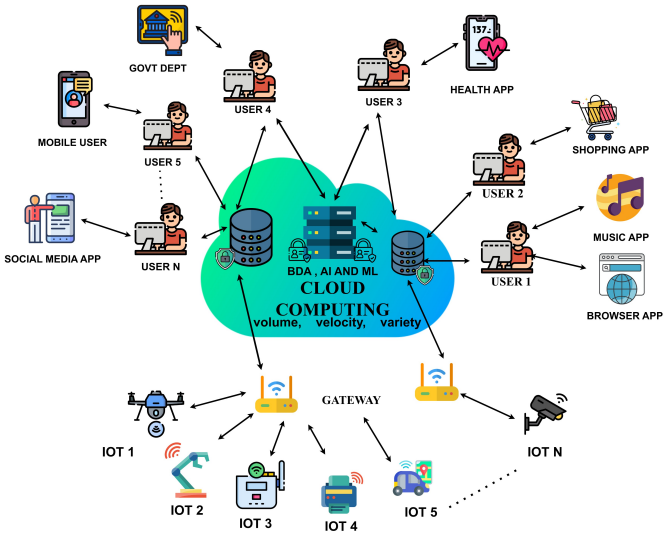

Fig. 4. Architecture of smart applications.

A. Architecture of IoT-Cloud Ecosystem

To meet the needs of real-time processing and control for time critical applications, a multi-layered IoT-Cloud architecture has evolved with intermediate mist, edge and fog layers [46] as seen in Fig. 5. The main difference between the layers is in terms of the compute-storage-communication capacities and where the data is processed and stored.

(i) IoT Device Layer: The bottom layer is the IoT devices with embedded sensors and actuators. This layer contains billions of devices across various IoT networks.

(ii) Mist Layer: On top of the IoT devices is a thin Mist layer which aids to build slightly bigger IoT systems at the local level through virtual communication mechanisms. The Mist allows to execute less computationally intensive tasks, closer to the devices thereby reducing the communication latency (while cloud handles the more computationally intensive tasks).

(iii) Gateway Layer: This is a communication layer having routers for IoT to reach the edge layer.

(iv) Edge Computing: This is composed of small nodes at the edge of the network which are capable of offering routing as well as computing services.

(v) Fog computing: This is an intermediary between the Edge and the Cloud which can carry out functions such as data filtering.

(vi) Cloud Computing: It is the topmost layer. It has large compute-storage capabilities and runs many software applications for multiple users.

B. Applications of IoT-Cloud Ecosystem

The IoT-Cloud system are easily usable and suitable for almost any type of application domain [60]. There are practically hundreds of applications; only a few common ones are listed in this subsection, to provide an understanding to the reader.

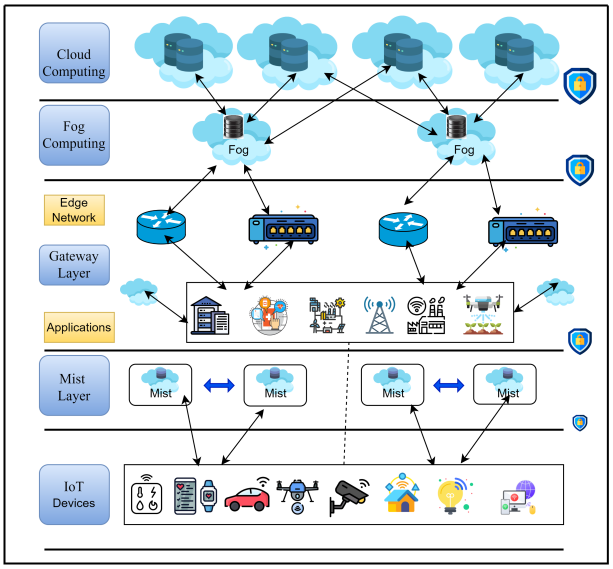

Fig. 5. IoT-Cloud layered architecture.

(i) Healthcare: Simplify healthcare processes by remote patient monitoring and emergency care, ambient assisted living, creation and management of Electronic Health Records, multimedia-based health services.

(ii) Smart Cities: Urban area planning and management, public lighting automation, electricity management, traffic control, water management, law and order, geo-tagging services, device crowdsourcing, etc.

(iii) Agriculture: Automated irrigation, water distribution, improved yield, farm surveillance, crop patterns and agri-analytics.

(iv) Manufacturing Industries: IoT devices on the factory floor monitor critical parameters like temperature and pressure, detect defective components and remove from assembly chain, quality checks for finished products.

(v) Wearables: Body worn smart IoT connected to the internet, such as fitness trackers, cardio monitors, smart glasses, smart watches, virtual reality headsets, etc. having a wide range of day-to-day applications in real-life are becoming the forefront of IoT.

C. Challenges in IoT-Cloud Ecosystem

The main challenges encountered in IoT-Cloud ecosystem are [61]:

(i) Heterogeneity: Wide variety of devices from different vendors and for different application domains, varied operating systems, platforms and services from IoT and Cloud create a disparate environment.

(ii) Resource Limitation: The IoT devices are low capacity, low power units, which cannot afford to process large codes or store huge data.

(iii) Performance: Having fast response time is important for real-world applications for which this ecosystem is utilized.

(iv) Reliability: Reliability is the property of consistently performing well. In the multi-layer IoT-Cloud ecosystem reliability is required at end point devices, data communication network, cloud infrastructure and services, and at the mist, edge, fog and so on. Reliability can be described in terms of number of users affected in unit time or Mean Time to Failure (MTTF).

(v) Security: Non standards hardware and software such as outdated OS, weak factory settings, lack of user awareness etc., create several security loopholes in IoT devices. In the multi-layer IoT to Cloud architecture security is critical for each layer and have diverse security requirements.

(vi) Standards: Unplanned surge of IoT by different vendors has lead to a lack of uniformity and standards.

(vii) Scale: IoT-Cloud systems are becoming very complex with millions of IoT devices getting interconnected and generating huge amounts of data. Monitoring and managing the hardware-software of the complex system is a challenging task.

V. Security Challenges in IoT-Cloud System

Analysis of the evolving IoT-Cloud architectures and recent publications [46], [62] reveal that security of the system is challenged by - technological progress, lack of standardization, IoT resource constraint etc. Broadly, the categories of security problems are - (i) Post Quantum Attacks and (ii) Vulnerabilities due to IoT-Cloud Characteristics.

A. Post Quantum Attacks

(i) Emergence of Quantum Computers: Gartner report indicates that Quantum Computers are rapidly evolving and a reasonable sized quantum computer would be available by 2025. The exceptionally fast computation of prime factors on quantum computers suggest that the popular public key cryptography system like RSA can be cracked down.

(ii) Requirement of Larger Key Size: Evolution of Quantum Computers and High Performance Computers makes it easy to crack even reasonably sized cryptographic schemes such as 1024 bit RSA. A simple remedy is to increase the key size; but it is impractical, especially in the context of resource constrained IoT.

(iii) Time-Space-Energy Complexity: Increasing the key size increases the computational complexity. The encryption and decryption algorithms should not be too complex if they have to run on low power and low capacity IoT and provide near real time response for real world applications. It is desirable to have minimal time-space-energy complexity for the algorithms.

For IoT applications with limited transmission bandwidth, the size of cipherext and the encapsulated key should be minimum for and ensure low communication overheads.

(iv) Key Generation - Key Exchange Time: The unprecedented growth of IoT networks combined with quantum cyber hackers is creating a highly active and complex system. The time for key generation and key exchange should be very fast, for the success of the security scheme.

(v) New Cryptanalysis Techniques: Cryptanalysts are exposing vulnerabilities in several existing crypto systems by using quantum computing algorithms, supercomputers, side channel attacks, guessing attacks, differential power analysis, differential cryptanalysis attacks etc. Rapidly evolving ecosystem, poses new challenges and requirements to develop novel Crypto schemes with higher security which are suitable for IoT and post quantum scenario, optimized algorithms with low footprint, and catering to heterogeneous devices. The concept of crypto-agility (i.e. replacing or adapting cryptographic schemes in software or hardware dynamically without interrupting the flow of a running system) is becoming a requirement for future operational application environments.

B. Vulnerabilities from IoT-Cloud Characteristics

(i) Heterogeneous Ecosystem: Independent manufacturers produce different varieties of IoT devices with varied functionality, that get integrated with different computing systems or Clouds over different networks. The heterogeneity and complexity of the ecosystem makes it very difficult to find a uniform security solution that fits all.

(ii) Lack of Standards: While there are diverse IoT manufacturers and enormous research and academic activities in IoT, there are only few standardization efforts such as from NIST, European Telecommunications Standards Institute (ETSI) and Internet Engineering Task Force (IETF).

(iii) IoT-Cloud Layered Architecture: A hierarchically increasing order of compute-storage capacities from IoT to Cloud to enable real-life applications brings along diverse security requirements at each layer

(iv) Scalability: The scheme should be able to cater for tens-of-thousands of devices and users and for huge volume of data within a reasonable turnaround time.

VI. Proposed Scheme

A. Problem Definition, Objectives and Constraints

The problem is to provide security for data in motion and at rest, for the IoT-Cloud ecosystem, taking cognizance of the emerging quantum computing.

The main constraints for this problem are:

(i) very low capacity of IoT devices and edge computers;

(ii) growing key sizes of the cryptographic schemes;

(iii) possibility of break down of the existing cryotographic schemes by quantum computing algorithms;

(iv) need to support fully homomorphic operations.

Hence, in addition to the basic authentication, confidentiality, integrity and privacy of data, the new cryptographic scheme must provide - low compute footprint and a higher security level. It is desirable to have features of homomorphic encyrption, searchable encryption and verifiable computation which are very useful for big data analytics. Crypto systems should be secure against the imminent quantum computers, which have the ability to solve certain mathematical problems exponentially faster compared to today's supercomputers.

By increasing the size of the key and cipher, we can make it more difficult for attackers to guess the key. This is because the larger the key, the more possible combinations there are, making it harder for an attacker to guess the right one. Similarly, increasing the cipher size also makes it harder for an attacker to break the encryption since the cipher has more bits that need to be decrypted. However, increasing the key and cipher size also increases the computational complexity of the algorithm, which can slow down the encryption and decryption process.

The lattice based, homomorphic and quantum secure, RLWE encryption scheme lends itself to the above requirements, and is also the most popular among the NIST PQC Competition [63]. In order to make it suitable for resource constrained IoT-Edge-Cloud environment, the RLWE should be optimized with respect to the amount of compute, memory and communication bandwidth.

Minor tweaks to the RWLE, can help to increase the security level and reduce the computation. A new algorithm named Circular Error Learning Algorithm (CELA) is proposed which uses modulus switching to improve the execution time and security level of the cryptographic scheme.

B. Circular Error Learning Algorithm (CELA)

In the proposed research, the execution time of the RLWE is reduced and security is improved, by introducing the Circular Error Learning Algorithm (CELA).

To improve security strength, the polynomial ring used in this algorithm shall have a larger dimension to accommodate the larger key size. The size of key is given by:

$$KeySize = (RingSize * Log_2(RingSize) + NoiseSize * Log_2(NoiseSize) + SecurityLevel) * 2$$

For example, if the security desired is 256 bits, and the ring size is 1024, the noise size is 128, the key size would be 22784 bits. Hence, note that the key size can be adjusted by changing the parameters used in the encryption process.

The execution time is determined by the complex calculations on the polynomial ring. The modulus switching reduces the execution time by reducing the intermediate computations without compromising its security. In modulus switching, the modulus is changed from q to a lower value q', to reduce computational complexity and improve its execution time.

Typically, the cryptographic scheme consists of Key Generation, Encryption and Decryption and the usage method is similar to the Diffie Hellman Key Exchange.

In the setup phase the public key is generated and published and corresponding secret-key generated is securely distributed among the parties. The public key is configured to be in the Hermite normal form of the lattice being examined. The keyholder creates a good basis by selecting P_k as a basis that is composed of short vectors that are 'roughly orthogonal'. The key to unlocking CELA's plan is nothing more than a simple vector. In the lattice based method, a ciphertext is a vector that is located close to the lattice called L. The message that is encrypted in this ciphertext is incorporated in the distance from the nearest lattice vector. In order to encrypt a message denoted by m, the sender must first select

a brief 'error vector' denoted by e that encodes m. Next, the sender must compute the ciphertext. We present a lattice-based key agreement mechanism for IoT devices based on CELA problem in order to perform secure data transfer.

The steps for key setup, encryption and decryption in CELA are as follows:

Key Setup:
1: Select public and private keys with a key size of at least $2n$, where n is the desired security level in bits.
2: Generate Private key and distribute among the parties.
3: Generate Public key and publish.
Note: The key size for this algorithm with an alteration is that it is at least twice the desired security level in bits; while the key size for the base RLWE algorithm is typically n.

Encryption:
1: Choose a uniformly random polynomial r with coefficients in Z_q and degree less than n. Choose a uniformly random polynomial r' with coefficients in Z_q and degree less than n.
2: Compute $e = a * r + m + e'$, where m is the message to be encrypted and e' is a small error term.
Compute $e' = b * r' + e''$, where e'' is a very small error term.
3: Return the ciphertext $(e, e' * f)$.
Note: In the encryption step of CELA, modulus switching is applied to reduce the amount of computation and error, while an additional random polynomial r' is used to add randomness to strengthen the ciphertext.

Decryption:
1: Compute $c' = e * b + e' * s \ mod \ q$. Round each coefficient of c' to the nearest integer multiple of $q/2n$.
2: Compute $m' = c' * f^{-1} \ mod \ q$.
3: Return m'.
Note: - In the decryption step of the CELA, the combined term $e' * s$ in the ciphertext makes it more difficult for an attacker to obtain information about the secret key from the ciphertext.

C. CELA Secure Communication Protocol in IoT-Cloud

This scheme uses a variety of available methods in order to conceal from parties the primary data that are not authorised for access. The protocol methodology uses the Registry Service Selection (RSS) security algorithm which has been designed to offer close assistance to the data all the way through the process of distributed computing [64].

The suggested system is divided into four distinct stages. To begin, the stage called - Client Registration to Cloud Specialist Cooperative, is responsible for managing the process through which the client is enrolled with the Cloud Service Provider (CSPs). The second stage (Distributed Storage of Information) will scramble and transfer the data and securely store it on the cloud. This includes data scrambling on the end of the customer and encryption with CELA on the end of the service provider. The third stage is - Client Authentication for Information Recovery Request. The final stage (Information Retrieval) takes the registry information by the confirmed client from the Cloud and gives to the approved client the data back after it has passed across all the security systems. The IoT devices, gateway and CSP are simulated on virtual machines (VMs) for experimentation. The flowchart of secure data communication protocol in IoT-Cloud environment is explained in Fig. 6.

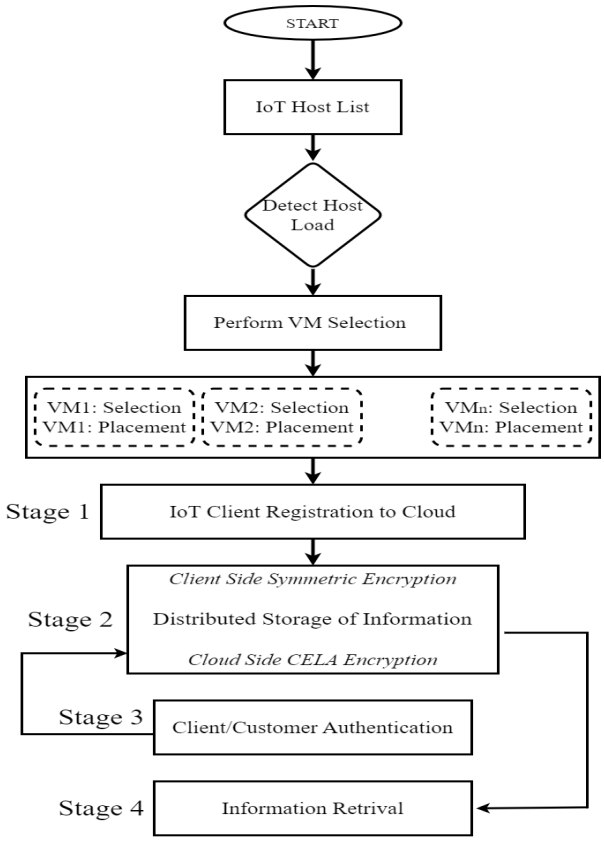

Fig. 6. Flowchart of CELA communication protocol.

D. Results

As seen in Fig. 7, the average time required for the CELA algorithm is 1628 microseconds while the average time required for the RLWE algorithm is 1811 microseconds. The ciphertext size for 100 keys in CELA algorithm is 158MB while that for the RLWE algorithm is 108MB.

Fig. 8 represents the comparison of minimum, maximum and average computation time of both RLWE and CELA algorithms. Multiple iterations have been considered for calculating the encryption and decryption time. The packet size transferred during these iterations are in the range of 1KB to 10KB. So for a 1KB packet size considered, the minimum time required is lower for CELA when compared to RLWE. For the highest packet size considered, the maximum time is also lower for CELA.

Typically, the size of the cipher text generated by CELA is slightly higher than that of RLWE as seen in Fig. 9.

E. Discussion

The following are some of the advantages offered by CELA:

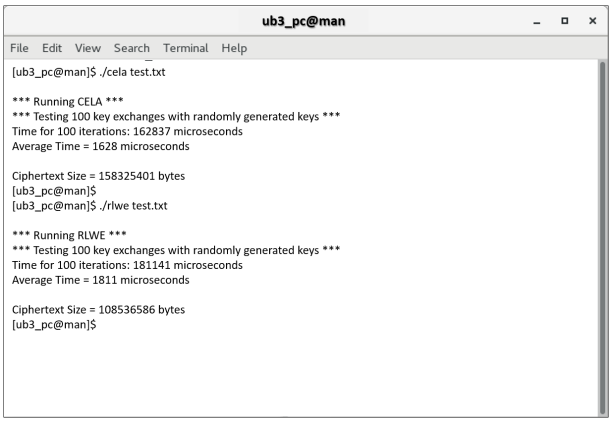

Fig. 7. Execution of CELA and RLWE.

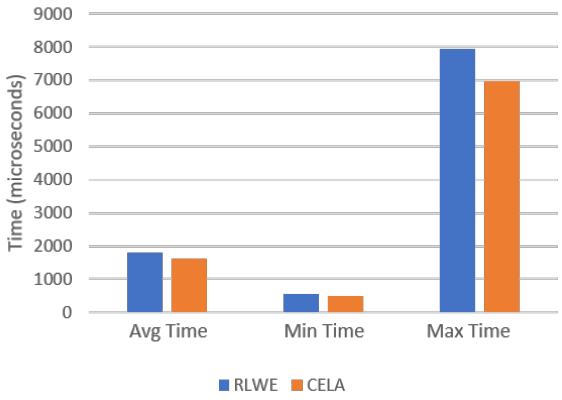

Fig. 8. Performance of CELA and RLWE.

- The proposed scheme offers an optimized lattice RLWE-based key agreement protocol for IoT devices [41], [40].

- The comparative examination of performance demonstrates that the suggested system is efficient in terms of compute and communication overhead.

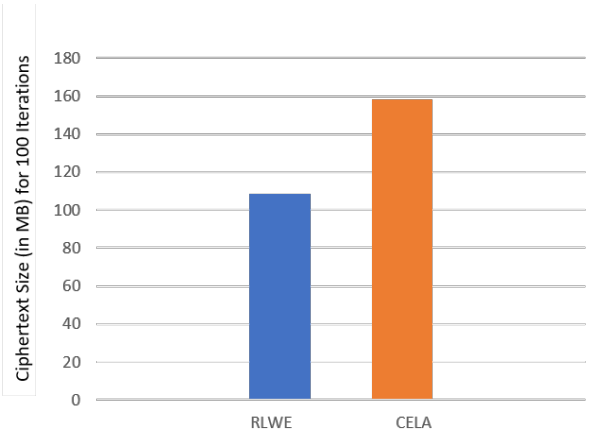

Fig. 9. Ciphertext size in CELA and RLWE.

- The system is provably secure for the longest period of time under the hardness assumption of RLWE, based on the fundamental difficulties of lattice algebraic structures [65]. Security analysis demonstrates that the system is secure against known security threats.

- The technique has been thoroughly researched for homomorphic encryption [66].

- Leveraging the RLWE scheme, CELA is capable of flexibility in dimension choice, and reduced computational requirements.

- The execution speed of CELA is superior in most cases, by virtue of modulus swicthing [57], very small size errors and Hermite Normal Form [67] matrix polynomials.

- The cipher text is more robust due to more randomness and slightly bigger size.

- The execution time of CELA is improved compared to RLWE by about 10%.As the number of IoT devices in the network increases, this can give a significant improvement in the system performance. Overall the CELA execution time being in microseconds is conducive for real-time applications.

- Increase in ciphertext size of CELA makes it computationally challenging for the adversary to break the encryption.

This research designed and demonstrated CELA as an optimized lattice based fully homomorphic cryptographic algorithm and a secure data communication protocol, for the IoT-Cloud converged system.

VII. Plausible Solutions

(i) Resistance to Post Quantum Cryptography: Subsequent to the RSA-768 getting compromised in 2010 due to availability of higher computer power, the key sizes have been increased to RSA-2048, 3072 and 7680 which are said to secure even if the supercomputing power is used to crack them. But with the emergence of Quantum Computers, it is predicted that the Shor's algorithm can calculate prime factors exceptionally fast; again posing a threat to RSA and other public key encryption schemes. While it is anticipated that factorizing the 1024-bit RSA would require about 2000 qubits [46] and the RSA-7680 would require 15362 qubits [68] on a Quantum Computer, it is currently unknown how many qubits would be needed to break 128-bit RLWE encryption as it is based polynomial equations, which is a different type of problem than factoring or discrete logarithms, and it is believed to be resistant to attacks by quantum computers. Presently the best operational quantum computer from IonQ is said to have only 79 qubits. Hence, for guaranteeing security till quantum computers become a reality, it will be sufficient to increase the key sizes.

(ii) Light Weight Cryptography: Increasing key size will improve the security strength, but disproportionately increase the complexity of the crypto system. There will be huge surge in the amount of compute-storage-power complexity, which are unaffordable for IoT-Edge-Cloud type of applications. Light weight cryptography having less computational complexity, faster execution time and low power consumption, less storage

space for key and ciphertext, is required to be developed. Complementing with a light weight key exchange mechanism will be highly useful for IoT-Cloud systems.

(iii) Novel Algorithms: Recognizing the need to transition traditional crypto systems as a preparatory for the quantum systems in the next decade, the NIST PQC Standardization Program has researched several Code-Based Techniques, Hash-based and Lattice based techniques. Novel hybrid techniques combining quantum key generation, key distribution with classical cryptography should be developed such as Quantum-AES [45], Quantum-RSA and other Quantum-PKE. Future real-world applications, require crypto-agility (i.e. replacing or adapting cryptographic schemes dynamically without interrupting the running system) to ensure dynamic adaptation in the heterogeneous hyper-connected operational environment.

VIII. Conclusion

The objective of this research was to design a light weight quantum secure communication protocol for the IoT-Cloud ecosystem. Detailed study of the state-of-the-art cyrptographic schemes and the challenges of IoT-Cloud Security in the emergence of Quantum Computing, led to identification of RLWE as a suitable base. RLWE is a homomorphic and quantum-safe cryptographic scheme and has emerged as the most popular in the NIST PQC Standardization Program.

A light weight Circular Learning Error Algorithm (CELA) has been proposed by optimizing RLWE with the modulus switching technique to make it suitable for IoT-Cloud ecosystem. A complete IoT-Cloud communication, including client registration, key generation and exchange, with CELA and RLWE encryption were experimentally compared. The CELA based protocol took 1628 microseconds compared to RLWE based protocol taking 1811 microseconds. This improvement will be significantly rewarding as the number of devices and users in the system increases.

While it may take over a decade for operational quantum computers to be established, the CELA can effectively safeguard the present IoT-Cloud application ecosystem. Meanwhile, agencies such as NIST, are taking cognizance of the threats to data security due to quantum computers. Research is rife for designing novel quantum based public key encryption schemes for the future quantum era; plausible solutions for future quantum secure cyrpto protocols have been presented in this work.

References

[1] J. Singh, T. Pasquier, J. Bacon, H. Ko, and D. Eyers, "Twenty security considerations for cloud-supported internet of things," *IEEE Internet of Things Journal*, vol. 3, p. 269–284, June 2016.

[2] M. Larouia, B. Nourd, H. Mounglaa, M. A. Cherifb, H. Afific, and M. Guizanie, "Edge and fog computing for iot: A survey on current research activities & future directions," *Elsevier Science Direct*, 2021.

[3] P. Taylors, "Amount of data created, consumed, and stored 2010-2020, with forecasts to 2025," 2022, last accessed 02 April 2023. [Online]. Available: https://www.statista.com/statistics/871513/worldwide-data-created/#:~:text=The%20total%20amount%20of%20data,replicated%20reached%20a%20new%20high

[4] M. N, V. K R, and B. E. Reddy, "Current challenges in iot cloud smart applications," *In Proc. 2021 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)*, 2021.

[5] M. Shabbir, A. Shabbir, C. Iwendi, A. Javed, M. Rizwan, N. Herencsar, and J. Lin, "Enhancing security of health information using modular encryption standard in mobile cloud computing," *IEEE Access*, vol. 9, p. 8820–8834, 2021.

[6] T. Baker, M. Mackay, M. Randles, and A. Taleb-Bendiab, "Intention-oriented programming support for runtime adaptive autonomic cloud-based applications," *Computers and Electrical Engg.*, vol. 39, p. 2400–2412, 2013.

[7] W. L. Chang, A. Roy, and M. Underwood, "Nist big data interoperability framework: Volume 4, big data security and privacy," 2018, nBD-PWG NIST Big Data Public Working Group. [Online]. Available: https://www.nist.gov/publications/nist-big-data-interoperability-framework-volume-4-big-data-security-and-privacy-v

[8] IBM, "What is data security data security definition and overview," last accessed March 28, 2023. [Online]. Available: https://www.ibm.com/topics/data-security

[9] T. K. et al., "Factorization of a 768-bit rsa modulus," *In Proc. 30th Annu. Conf. Adv. Cryptol., Santa Barbara, CA, USA*, p. 333–350, Aug 2010.

[10] S. Balogh, O. Gallo, R. Ploszek, P. Špaček, and P. Zajac, "Iot security challenges: Cloud and blockchain, postquantum cryptography, and evolutionary techniques," *MDPI Electronics*, 2021.

[11] S. Halder, M. Conti, S. Member, and IEEE, "Crypsh: A novel iot data protection scheme based on bgn cryptosystem," *IEEE Transactions on Cloud Computing*, vol. 10, pp. 2437–2450, 8 2022.

[12] A. et al., "Newhope: Algorithm specifications and supporting documentation," *Version 1.1, Updated April 10, 2020*, p. 169–180, 2020. [Online]. Available: https://newhopecrypto.org/data/NewHope_2020_04_10.pdf

[13] F. Chen, D. Luo, T. Xiang, P. Chen, J. Fan, and H. Truong, "Iot cloud security review: A case study approach using emerging consumer-oriented applications," *ACM Computer Survey*, vol. 54, pp. 1–36, 2021.

[14] T. Mather, S. Kumaraswamy, and S. Latif, "Cloud security and privacy," *OReilly Media Inc.: Sebastopol, CA, USA*, 2009.

[15] Ometov, Aleksandr, O. L. Molua, M. Komarov, and J. Nurmi, "A survey of security in cloud, edge, and fog computing," *Sensors*, vol. 22, p. 927, 2022.

[16] R. Jain, S. Madan, and B. Garg, "Privacy sustainability scheme in cloud environment," *CSI Transactions in ICT*, vol. 4, pp. 123–128, 2016.

[17] P. Rao and P. Saraswathi, "Evolving cloud security technologies for social networks," *Elsevier: Security in IoT Social Networks*, vol. 4, p. 179–203, 2021.

[18] R. Montasari, A. Daneshkhah, H. Jahankhani, and A. Hosseinian-Far, "Cloud computing security: Hardware-based attacks and countermeasures," *Springer: Digital Forensic Investigation of Internet of Things (IoT) Devices*, p. 155–167, 2021.

[19] J. Li, Y. Zhang, J. Ning, X. Huang, G. S. Poh, and D. Wang, "Attribute based encryption with privacy protection and accountability for cloudiot," *IEEE Transactions on Cloud Computing*, vol. 10, pp. 762–773, 6 2022.

[20] C. Zhao, L. Xu, J. Li, H. Fang, and Y. Zhang, "Toward secure and privacy-preserving cloud data sharing: Online/offline multiauthority cp-abe with hidden policy," *IEEE Systems Journal*, vol. 16, pp. 4804–4815, 9 2023.

[21] D. Han, N. Pan, and K.-C. Li, "A traceable and revocable ciphertext-policy attribute-base encryption scheme based on privacy protection," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, pp. 316–327, 2 2022.

[22] S. Das and S. Namasudra, "Multiauthority cp-abe-based access control model for iot-enabled healthcare infrastructure," *IEEE Transactions on Industrial Informatics*, vol. 19, pp. 821–829, 1 2023. [Online]. Available: https://ieeexplore.ieee.org/document/9760125/

[23] K. Zhang, J. Long, X. Wang, H.-N. Dai, K. Liang, and M. Imran, "Lightweight searchable encryption protocol for industrial internet of things," *IEEE Transactions on Industrial Informatics*, vol. 17, pp. 1–11, 6 2021.

[24] L. Wang, Y. Lin, T. Yao, H. Xiong, and K. Liang, "Fabric: Fast and secure unbounded cross-system encrypted data sharing in cloud computing," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–13, 2023.

[25] H. Deng, Z. Qin, Q. Wu, Z. Guan, R. H. Deng, Y. Wang, and Y. Zhou, "Identity-based encryption transformation for flexible sharing of encrypted data in public cloud," *Institutional Knowledge at Singapore Management University - Research Collection School of Computing and Information Systems*, pp. 1–13, 2020.

[26] R. F. Abdel-Kader, S. H. El-sherif, and R. Y. Rizk, "Efficient two-stage cryptography scheme for secure distributed data storage in cloud computing," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, pp. 3295–3306, 6 2020.

[27] M. H. Saracevic, S. Z. Adamovic, V. A. Miskovic, M. Elhoseny, N. D. Macek, M. M. Selim, and K. Shankar, "Data encryption for internet of things applications based on catalan objects and two combinational structures," *IEEE Transactions on Reliability*, vol. 70, pp. 819–830, 6 2021.

[28] S. Atiewi, A. Al-Rahayfeh, M. Almiani, S. Yussof, O. Alfandi, A. Abugabah, and Y. Jararweh, "Scalable and secure big data iot system based on multifactor authentication and lightweight cryptography," *IEEE Access*, vol. 8, pp. 113 498–113 511, 2020. [Online]. Available: https://ieeexplore.ieee.org/document/9118946/

[29] A. K. Singh and D. Saxena, "A cryptography and machine learning based authentication for secure data-sharing in federated cloud services environment," *Journal of Applied Security Research*, pp. 1–28, 2 2021.

[30] A. K. Sahu, S. Sharma, and D. Puthal, "Lightweight multi-party authentication and key agreement protocol in iot-based e-healthcare service," *ACM Transactions of Multimedia Compute and Communication*, vol. 17, pp. 1–20, 6 2021.

[31] H. Nejatollahi, F. Valencia, S. Banik, F. Regazzoni, R. Cammarota, and N. Dutt, "Synthesis of flexible accelerators for early adoption of ring-lwe post-quantum cryptography," *ACM Transactions on Embedded Computing Systems*, vol. 19, pp. 1–17, 3 2020.

[32] W.-K. Lee, H. Seo, Z. Zhang, and S. O. Hwang, "Tensorcrypto: High throughput acceleration of lattice-based cryptography using tensor core on gpu," *IEEE Access*, vol. 10, pp. 20 616–20 632, 2022.

[33] J. Xie, P. He, X. Wang, and J. L. Imana, "Efficient hardware implementation of finite field arithmetic ab+c for binary ring-lwe based post-quantum cryptography," *IEEE Transactions on Emerging Topics in Computing*, pp. 1–6, 4 2020.

[34] D.-E.-S. Kundi, A. Khalid, S. Bian, C. Wang, M. O'Neill, and W. Liu, "Axrlwe: A multilevel approximate ring-lwe co-processor for lightweight iot applications," *IEEE Internet of Things Journal*, vol. 9, pp. 10 492–10 501, 7 2022.

[35] J. L. Imana, P. He, T. Bao, Y. Tu, and J. Xie, "Efficient hardware arithmetic for inverted binary ring-lwe based post-quantum cryptography," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 69, pp. 3297–3307, 8 2022.

[36] X. Xi, G. Li, Y. Wang, and M. Orshansky, "A provably secure strong puf based on lwe: Construction and implementation," *IEEE Transactions on Computers*, vol. 72, pp. 346–359, 2 2023.

[37] P. He, T. Bao, J. Xie, and M. Amin, "Fpga implementation of compact hardware accelerators for ring-binary-lwe based post-quantum cryptography," *ACM Transactions on Reconfigurable Technology and Systems*, 10 2022.

[38] S. Hadayeghparast, S. Bayat-Sarmadi, and S. Ebrahimi, "High-speed post-quantum cryptoprocessor based on risc-v architecture for iot," *IEEE Internet of Things Journal*, vol. 9, pp. 15 839–15 846, 9 2022.

[39] D. Xu, X. Wang, Y. Hao, Z. Zhang, Q. Hao, and Z. Zhou, "A more accurate and robust binary ring-lwe decryption scheme and its hardware implementation for iot devices," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 30, pp. 1007–1019, 8 2022.

[40] S. Ebrahimi and S. Bayat-Sarmadi, "Lightweight and fault-resilient implementations of binary ring-lwe for iot devices," *IEEE Internet of Things Journal*, vol. 7, pp. 6970–6978, 8 2020.

[41] Z. Liu, R. Azarderakhsh, H. Kim, and H. Seo, "Efficient software implementation of ring-lwe encryption on iot processors," *IEEE Transactions on Computers*, vol. 69, pp. 1424–1433, 10 2020.

[42] F. Zhang, B. Yang, X. Dong, S. Guilley, Z. Liu, W. He, F. Zhang, and K. Ren, "Side-channel analysis and countermeasure design on arm-based quantum-resistant sike," *IEEE Transactions on Computers*, vol. 69, no. 11, pp. 1681–1693, 2020.

[43] W.-K. Lee, H. Seo, S. O. Hwang, R. Achar, A. Karmakar, and J. M. B. Mera, "Dpcrypto: Acceleration of post-quantum cryptography using dot-product instructions on gpus," *IEEE Transactions on Circuits and Systems - I*, pp. 1–14, 2022.

[44] S. Ghosh, R. Misoczki, and M. R. Sastry, "Lightweight post-quantum-secure digital signature approach for iot motes," *Security and Privacy Research, Intel Labs*, pp. 1–24, 2019.

[45] R. Golchha, J. Lachure, and R. Doriya, "Fog enabled cyber physical system authentication and data security using lattice and quantum aes cryptography," *International Journal of Computing and Digital Systems*, vol. 13, pp. 267–275, 1 2023.

[46] T. M. Fernández-Caramés, "From pre-quantum to post-quantum iot security: A survey on quantum-resistant cryptosystems for the internet of things," *IEEE Internet Of Things Journal*, vol. 7, 2020.

[47] Y. Chen, S. Dong, T. Li, Y. Wang, and H. Zhou, "Dynamic multi-key fhe is asymmetric key setting from lwe," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 5239–5249, 2021.

[48] Y. Su, B. Yang, C. Yang, and L. Tian, "Fpga-based hardware accelerator for leveled ring-lwe fully homomorphic encryption," *IEEE Access*, vol. 8, pp. 168 008–168 025, 2020.

[49] D. B. Rawat, R. Doku, and M. Garuba, "Cybersecurity in big data era: From securing big data to data-driven security," *IEEE Transactions on Services*, pp. 1–18, 2019.

[50] J. Tang, Y. Cui, Q. Li, K. Ren, J. Liu, and R. Buyya, "Ensuring security and privacy preservation for cloud data services," *ACM Computing Surveys*, vol. 49, pp. 1–39, 2017.

[51] X. Yu, Z. Yan, and A. V. Vasilakos, "A survey of verifiable computation," *Mobile Networks and Applications*, vol. 22, pp. 438–453, 2017.

[52] S. P. et al., "Advances in quantum cryptography," *Advances in optics and photonics*, vol. 12, pp. 1012–1236, 2020.

[53] O. Regev, "The learning with errors problem," 2010, last accessed 02 April 2023. [Online]. Available: http://slideplayer.com/slide/14163933/

[54] V. Vaikuntanathan, "Advanced topics in cryptography: Lattices," 2015.

[55] B. B. OBE, "Learning with errors and ring learning with errors," 2018, last Accessed 18 April 2023. [Online]. Available: https://medium.com/asecuritysite-when-bob-met-alice/learning-with-errors-and-ring-learning-with-errors-23516a502406

[56] "Ring learning with errors," last Accessed 18 April 2023. [Online]. Available: https://en.wikipedia.org/wiki/Ring_learning_with_errors

[57] J. Kun, "Modulus switching in lwe," last Accessed 01 May 2023. [Online]. Available: https://jeremykun.com/2022/07/16/modulus-switching-in-lwe/

[58] R. B. et al., "A manifesto for future generation cloud computing: Research directions for the next decade," *ACM Comput. Surv*, vol. 51, pp. 1–38, Sept 2019.

[59] Subashini and Kavitha, "A survey on security issues in service delivery models of cloud computing," *Journal of Network and Computer Applications*, vol. 34, pp. 1–11, Jan 2011.

[60] A. Botta, W. de Donato, V. Persico, and A. Pescapé, "Integration of cloud computing and internet of things: A survey," *Future Generation Computer Systems*, vol. 56, p. 684–700, 2016.

[61] C. Stergiou, K. E. Psannis, B. B. Gupta, and Y. Ishibashi, "Security, privacy & efficiency of sustainable cloud computing for big data & iot," *Sustainable Computing: Informatics and Systems*, vol. 19, pp. 174–184, 9 2018.

[62] L. Juyan, J. Peng, and Z. Qiao, "A ring learning with errors-based ciphertext-policy attribute-based proxy re-encryption scheme for secure big data sharing in cloud environment," *Big Data*, 2022.

[63] H. Q. Le, P. K. Mishra, S. Nakamura, K. Kinjo, D. H. Duong, and M. Yasuda, "Impact of the modulus switching technique on some attacks against learning problems," *IET Information Security*, vol. 14, no. 3, pp. 286–303, 2020.

[64] M. Annamalai and X. M. Jesintha, "Smart iot system based patient monitoring and medicine reminder based on registry service selection scheme," *European Journal of Molecular & Clinical Medicine*, vol. 7, 2020.

[65] G. Maringer, S. Puchinger, and A. Wachter-Zeh, "Higher rates and information-theoretic analysis for the rlwe channel," pp. 1–5, 2021.

[66] H. Bandara, Y. Herath, T. Weerasundara, and J. Alawatugoda, "On advances of lattice-based cryptographic schemes and their implementations," *Cryptography*, vol. 6, November 2022.

[67] D. Micciancio, "Improving lattice based cryptosystems using the hermite normal form," *LNCS*, vol. 2146, 11 2001.

[68] T. Gagliardoni, "Quantum attack resource estimate: Using shor's algorithm to break rsa vs dh/dsa vs ecc," 2021, last accessed 02 April 2023. [Online]. Available: https://research.kudelskisecurity.com/2021/08/24/quantum-attack-resource-estimate-using-shors-algorithm-to-\break-rsa-vs-dh-dsa-vs-ecc/