

Ring-LWE Identity-Based Encryption with Dynamic Revocation for Cloud Data Sharing

Gabriel Assamah, Alimatu Latiff Y, Benjamin Appiah, Regina Esi T, Emmanuel Derry



Abstract: Cloud storage faces significant security and access control challenges due to reduced user oversight and the emerging threat of quantum computing to traditional cryptographic methods. Existing revocable Identity-Based Encryption (IBE) schemes are limited by their lack of post-quantum security, inefficient revocation mechanisms that require re-encryption of data, and cumbersome key update procedures. We propose a post-quantum secure Ring-LWE IBE scheme with dynamic time-based revocation tailored for cloud environments. Our solution is built on the hardness of the Ring Learning with Errors (RLWE) problem to ensure quantum resistance and introduces a novel time-based revocation framework. In our approach, user access is bound to discrete periods and managed through a hierarchical binary tree structured over identities and time. This design eliminates the need to re-encrypt stored data upon user revocation. Instead, a trusted authority periodically distributes lightweight key updates exclusively to non-revoked users. Thanks to the binary tree structure, non-revoked users can compute updated decryption keys with only $O(\log N_{\max})$ overhead in both computation and communication, where $N_{\max}$ is the maximum number of users or periods. Revoked users, having no access to future updates, lose decryption capabilities. We provide formal security proofs showing the scheme's resistance against adaptive identity and time-period-based attacks, grounded in the RLWE assumption. Overall, our scheme offers an effective combination of post-quantum security, efficient access control, and simplified key management, making it suitable for secure cloud data sharing in the quantum era.

Keywords: IBE, Post-Quantum Cryptography, Ring-LWE, Time-based Revocation, Secure Cloud Storage.

Abbreviations:

PPT: Probabilistic Polynomial-Time
PKI: Public Key Infrastructure
IBE: Identity-Based Encryption
RLWE: Ring-Learning with Errors
ABE: Attribute-Based Encryption
PRE: Proxy Re-Encryption

Manuscript received on 09 June 2025 | First Revised Manuscript received on 04 July 2025 | Second Revised Manuscript received on 16 October 2025 | Manuscript Accepted on 15 November 2025 | Manuscript published on 30 November 2025.

*Correspondence Author(s)

Gabriel Assamah*, Department of Computer Science, University of Cape Coast, Cape Coast, Ghana. Email ID: gassamah@ucc.edu.gh, ORCID ID: 0000-0003-2411-2587

Alimatu Latiff Yussif, Department of Computer Science, University of Cape Coast, Cape Coast, Ghana. Email ID: asyussiff@ucc.edu.gh,

Benjamin Appiah, Department of Computer Science, Ho Technical University, Ho, Ghana. Email: kubipp7@gmail.com

Regina Esi Turkson, Department of Computer Science, University of Cape Coast, Cape Coast, Ghana. Email ID: regina.turkson@ucc.edu.gh

Emmanuel Derry, Department of Computer Science, University of Cape Coast, Cape Coast, Ghana. Email ID: emmanuel.derry@stu.ucc.edu.gh

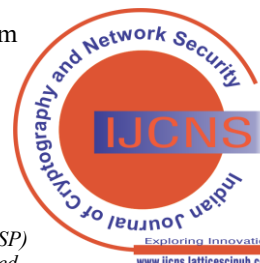
© The Authors. Published by Lattice Science Publication (LSP). This is an open access article under the CC-BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

I. INTRODUCTION

The increasing prevalence of cloud computing has transformed data storage and processing, enabling cost-effective and scalable solutions for big data analytics [1]. However, outsourcing data to third-party cloud providers introduces significant security challenges, particularly concerning data confidentiality and fine-grained access control [2]. Cryptographic enforcement of security policies is crucial since users lack direct physical control over their data. It is imperative to maintain the confidentiality of sensitive information and restrict access to authorized users, even while accommodating dynamic changes in access privileges, such as revoking access for departing employees or compromised credentials [3].

Traditional cryptographic solutions often rely on Public Key Infrastructure (PKI) to ensure data confidentiality in the cloud. While effective, PKI-based systems demand certificate management, which can be cumbersome in large, dynamic environments with frequent user changes. Identity-Based Encryption (IBE) offers an elegant alternative by using a user's identity (e.g., email address) directly as their public key [4]. This simplifies key distribution, making IBE particularly attractive for secure data sharing in the cloud by eliminating the need for certificate management [5]. Despite its advantages, standard IBE faces a significant challenge in efficient user revocation [6]. When a user's access is revoked, all data encrypted for that user must become unintelligible to them. In standard IBE, this typically requires the data owner or a trusted party to re-encrypt all relevant ciphertexts with keys corresponding to the non-revoked users. This computationally expensive re-encryption process, along with its substantial communication overhead, renders dynamic access control impractical for large-scale cloud data.

Moreover, the security of widely deployed public-key cryptographic algorithms, including those underpinning traditional IBE schemes (which rely on discrete logarithms or factorization), is fundamentally threatened by the advent of quantum computing. Shor's algorithm, for instance, can efficiently break these systems on a large-scale quantum computer [7]. Consequently, there is an urgent need to develop post-quantum secure cryptographic schemes that can withstand future attacks by quantum adversaries. Lattice-based cryptography [8], particularly algorithms based on the Ring-Learning with Errors (RLWE) problem [9], stands out as a promising candidate for post-quantum security due to its strong theoretical foundations and efficiency. The convergence



of these challenges underscores the necessity for a post-quantum secure IBE scheme that facilitates efficient user revocation without requiring costly ciphertext re-encryption, thereby ensuring safe, efficient, and verifiable access control for large datasets in cloud environments. While some research has explored revocable IBE [10], many existing constructions suffer from inefficiencies due to key update overhead or data re-encryption upon revocation [6], or they lack the necessary post-quantum security guarantees.

Motivated by these obstacles, we propose a new Ring-LWE Identity-Based Encryption scheme that incorporates Dynamic Time-Based Revocation. This approach is designed to be post-quantum secure, ensuring data confidentiality and efficient access control for cloud-delegated storage. Our key contributions are summarized below:

- A. We introduce a revocation mechanism that functions over discrete periods, issuing periodic, lightweight key updates exclusively to non-revoked users. By employing a hierarchical binary tree structure over user identities and periods, our scheme ensures that both computational and communication overhead for key updates are only $O(\log N_{\max})$, where $N_{\max}$ represents the system's maximum capacity. This approach reduces user workload and eliminates the need for interaction with encrypted data stored in the cloud.
- B. The scheme is grounded in the conjectured hardness of the Ring Learning with Errors (RLWE) problem, providing robustness against adversaries with quantum capabilities. Furthermore, it directly uses user identities for encryption, eliminating the need for a complex PKI-based certificate infrastructure.
- C. Our solution supports effective access control management in post-quantum cloud infrastructures, enabling a trusted authority to revoke access at specific time intervals without requiring data owners or cloud providers to re-encrypt large datasets.

The proposed RLWE Identity-Based Encryption scheme with Dynamic Revocation is comprehensively described in Section III, and experimentation is presented in Section IV. The paper is concluded in Section V.

II. RELATED WORK

The escalating adoption of cloud computing for big data storage necessitates robust data confidentiality and access control, challenging traditional encryption methods due to the dynamic nature of cloud environments [11]. Advanced cryptographic primitives, such as Attribute-Based Encryption (ABE) [12], Proxy Re-Encryption (PRE) [13], and Identity-Based Encryption (IBE) [8], offer promising solutions. IBE simplifies key management by using user identities as public keys, thereby eliminating the need for complex Public Key Infrastructure (PKI) [14]. Concurrently, the looming threat of quantum computing necessitates the development of post-quantum secure cryptographic schemes [15]. Lattice-based cryptography, particularly constructions based on the Learning with Errors (LWE) and Ring Learning with Errors (RLWE) problems, has emerged as a strong candidate due to its efficiency, quantum resistance, and suitability for cloud applications, thanks to its advantages in key/ciphertext size and homomorphic

operations [9].

A significant challenge in conventional IBE systems is the lack of an effective mechanism for dynamic user revocation, as a user's private key can decrypt all ciphertexts regardless of changes in privilege or compromise [16]. Static revocation methods are impractical for large-scale cloud storage due to the prohibitive computational cost of re-encrypting all data [17]. To address this, Revocable IBE (RIBE) schemes have been developed, incorporating time-based or attribute-based access control [18]. However, many existing RIBE constructions are vulnerable to quantum attacks due to their reliance on assumptions like RSA or bilinear pairings [19]. This highlights the critical need for post-quantum secure RIBE schemes that can manage dynamic access control in the quantum era [18]. Recent research has explored lattice-based approaches using LWE or RLWE assumptions [19]. Efficient user revocation in dynamic environments often requires sophisticated key management to minimize the burden on the trusted authority and non-revoked users [20]. While naive revocation usually involves costly ciphertext re-encryption, broadcast encryption [21] faces challenges in achieving efficient identity-based revocation with fine-grained time control in IBE without ciphertext modification. Key-updating tree structures, especially binary trees [22], provide a more efficient paradigm, enabling users to compute updated keys by processing only a logarithmic number of nodes, thereby significantly reducing computational and communication latency compared to linear-time operations or ciphertext re-encryption.

Integrating revocation mechanisms into lattice-based schemes presents unique technical challenges, requiring careful design to preserve security proofs based on lattice problems. Despite advancements in lattice-based Attribute-Based Encryption with revocation [6] and Proxy Re-Encryption [23], a significant research gap exists in developing a practical and scalable RLWE-based IBE scheme that natively supports dynamic [20], time-based revocation without ciphertext re-encryption, specifically for secure and flexible big data sharing in cloud environments [23]. The proposed scheme aims to bridge this gap by combining the benefits of RLWE-based IBE with a dynamic revocation mechanism based on a binary key-update tree [24]. While prior works have explored RLWE-based revocation [12], binary tree methods [22], and multi-authority R-LWE, this scheme uniquely offers time-based revocation without ciphertext re-encryption (unlike [8]), logarithmic-scale key updates via an enhanced tree structure (improving upon [17]), and post-quantum secure access control optimized for cloud environments. It achieves lower computational overhead than [6] while maintaining constant ciphertext sizes, effectively bridging the gap between theoretical RLWE constructions [23] and practical big data requirements [23].



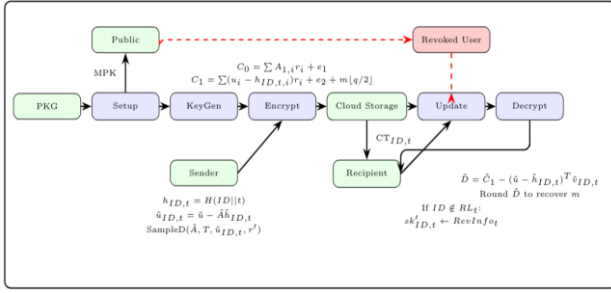


Fig. 1: The Flowchart Depicts

The RLWE Identity-Based Encryption Scheme with Dynamic Revocation. The diagram illustrates the roles of the PKG (Setup, KeyGen, Revoke), Sender (Encrypt using MPK), Recipient (Update, Decrypt), and Cloud Storage. Encrypted data ($CT_{ID,t}$) is stored in the Cloud. Dynamic revocation is handled by the PKG publishing revocation information (RevInfo). Non-revoked recipients use RevInfo, and their secret key ($sk_{ID,t}$) in the Update process to derive the valid decryption key ($sk'_{ID,t}$). Revoked users cannot complete the Update. Decryption uses $sk'_{ID,t}$ and $CT_{ID,t}$. Core encryption and decryption mathematics are shown, based on RLWE

III. SCHEME CONSTRUCTION

This section provides a formal explanation of our proposed Ring-LWE Identity-Based Encryption (RIBE) scheme with Dynamic Time-Based Revocation. Our revised design ensures cryptographic correctness by adopting a standard primal RLWE-IBE framework. We delineate the precise execution of each algorithm, the role of the hierarchical binary tree in managing user access, and the underlying cryptographic parameters. This approach guarantees post-quantum security that is both practicable and effective in dynamic cloud data sharing environments. A visual representation of the scheme flow is depicted in Fig. 1.

A. System Parameters and Structure

Our scheme is built upon the polynomial ring $R_p = \mathbb{Z}_p[x] / \langle x^n + 1 \rangle$, leveraging the efficiency and conjectured post-quantum security of RLWE-based constructions. All cryptographic operations are expressed in terms of polynomials and matrices over R_p , or their corresponding coefficient vector representations over $\mathbb{Z}_p$.

i. Parameters: The security level (λ) and maximum user capacity (N_{max}) determine the system parameters:

- n : The dimension of the ring R_q , typically a power of 2.
- q : A prime modulus defining the field $\mathbb{Z}_q$, chosen such that $q \equiv 1 \pmod{2n}$ to allow for efficient polynomial multiplication via the Number-Theoretic Transform (NTT).
- χ : A discrete Gaussian error distribution over $\mathbb{Z}$ with a slight standard deviation σ . Errors sampled from χ are crucial for the hardness of the underlying lattice problem.
- m : A dimensional parameter, often small (e.g., $m = 1$ or $m = 2$), defining the number of rows/columns of matrices and vectors over R_q . For simplicity, we consider the case where matrices are $1 \times m$ and vectors are $m \times 1$.

- N_{max} : The maximum number of users or periods supported, determining the binary tree height $H = \lceil \log_2 N_{max} \rceil$.
- Binary Tree Structure and Cryptographic Role: A complete binary tree T of height H with N_{max} leaves is central to our revocation mechanism.
- Node Association: Each node (i, j) in the tree (level i , position j) is associated with a secret short vector $v_{i,j} \in R_q^m$. These secrets are generated hierarchically.
- Secret Derivation: The Private Key Generator (PKG) generates a master secret vector $v_{0,0}$ for the root node. For any non-leaf node (i, j) , its secret $v_{i,j}$ is divided between its left child $(i+1, 2j)$ and right child $(i+1, 2j+1)$. A random short vector $v_{i+1, 2j}$ is selected, and the right child's secret is computed as: $v_{i+1, 2j+1} = v_{i,j} - v_{i+1, 2j}$. This ensures the additive property: $v_{i,j} = v_{i+1, 2j} + v_{i+1, 2j+1}$ holds consistently throughout the tree.
- Identity Mapping: A public function, MapIDToLeaf, deterministically maps a user identity ID to a leaf node (H, j_{ID}) . The sequence of nodes from the root to this leaf defines the user's path, denoted Path(ID).
- Master Public Key (MPK): The MPK is public and used for encryption.
- $MPK = (A, u, H_1)$
- $A \in R_q^{n \times m}$: A matrix generated together with a secret trapdoor. Its elements are statistically close to uniform in R_q .
- $u \in R_q$: A uniformly random vector.
- $H_1: \{0,1\}^* \rightarrow R_q^*$: A cryptographic hash function, modelled as a random oracle, that maps an identity-time pair to a ring element.
- Master Secret Key (MSK): The MSK is securely held by the PKG.
- $MSK = (T_A, v_{0,0})$
- T_A : A short basis (trapdoor) for the lattice $\mathcal{L}_q^\perp(A) = \{x \in R_q^m \mid A \cdot x \equiv 0 \pmod{q}\}$. This enables the efficient sampling of short vectors for key generation via the SampleD algorithm.
- $v_{0,0}$: The secret short vector associated with the root of the binary tree, used to derive all other node secrets.
- System State (ST_t): The PKG maintains a state $ST_t = (RL_t, t_{curr})$, which contains:
 - The revocation list RL_t
 - The current period t_{curr}

B. Scheme Algorithms

1) Setup ($1^\lambda, N_{max}$)

Executed by the PKG to initialize the system.

1. Select parameters n, q, m, χ based on the security parameter λ .
2. Generate the public matrix $A \in R_q^{n \times m}$ and its secret trapdoor T_A using a trapdoor generation algorithm (e.g., RingGenTrap).
3. Sample a uniformly random vector $u \in R_q$.
4. Define a cryptographic hash function $H_1: \{0,1\}^* \rightarrow R_q^*$.
5. Initialize the binary tree secrets:
 - Sample a short



master secret vector $v_0, o \in R_q^m$ from χ^m .

- Recursively, for each node (i, j) from the root downward, define secrets for its children $(i+1, 2j)$ and $(i+1, 2j+1)$:
 - i. Sample a random short vector $v_{\{i+1, 2j\}} \leftarrow \chi^m$
 - ii. Set $v_{\{i+1, 2j+1\}} = v_{\{i, j\}} - v_{\{i+1, 2j\}}$

6. Publish the Master Public Key (MPK): (A, u, H_1) , Set the Master Secret Key (MSK): (T_A, v_0, o) . Initialize the System State $ST_0 = (\emptyset, 0)$

2) KeyGen (MPK, MSK, ID)

Executed by the PKG to issue a long-term key to the user ID.

1. Map ID to its corresponding leaf node (H, j_{ID}) and determine its path, denoted Path (ID).
2. The user's long-term secret key, sk_{ID} , consists of the hidden vectors of all sibling nodes of the nodes on Path (ID).
Formally:
 $sk_{ID} = \{v_{\{i, j'\}} \mid \exists (i, j) \in \text{Path (ID)}, (i, j') \text{ is a sibling of } (i, j)\}$
3. Securely transmit sk_{ID} to the user. This key enables the user to reconstruct the secret of any ancestor node by summing the secrets of its descendant sub-branches.

3) Revoke (MPK, MSK, ID_{revoke}, t_{eff}, ST_{t_{eff}})

Executed by the PKG to revoke user ID_{revoke} from time t_{eff} onward.

1. Add ID_{revoke} to the revocation list: $RL_{t_{eff}} = RL_{t_{eff}-1} \cup \{ID_{revoke}\}$
2. To generate the public revocation information RevInfo_{t_{eff}}, the PKG identifies a minimal set of nodes, called the cover set S_{cover} , whose subtrees span all non-revoked users.
3. For each node $(i, j) \in S_{cover}$, the PKG computes and includes its secret vector $v_{\{i, j\}}$ in RevInfo_{t_{eff}}.
4. Publish:
 $RevInfo_{t_{eff}} = \{((i, j), v_{\{i, j\}}) \mid (i, j) \in S_{cover}\}$

4) Update (MPK, ID, t, sk_{ID}, {RevInfo_{t'}} for all t' < t)

Executed by a non-revoked user at time t to compute a time-specific decryption key.

1. Revocation: Check
The user ID first checks if they are in the global revocation list RL_t . This list is reconstructed by aggregating all public RevInfo_{t'} values broadcast by the PKG for all past periods t' < t, as well as the current RevInfo_t.
 - If $ID \in RL_t$, the process aborts, and the user cannot obtain a key for time t.
2. PKG-side Key Update Component Generation and Broadcast. For each non-revoked user ID' at time t, the PKG performs the following:
 - Derive Target Vector: Compute $y_{ID', t} = u - H_1(ID'/t) \in R_q$
 - Compute Key Component: Using its master trapdoor T_A , compute $KUP_{ID', t}$

$= \text{SampleD}(A, T_A, y_{ID', t}, \sigma)$ This $KUP_{ID', t}$ is the time-specific key satisfying the decryption equation.

- Broadcast Encryption Preparation: Rather than encrypting each $KUP_{ID', t}$ separately for every user, the PKG:
 - i. Identifies a minimal set of nodes in the binary tree whose subtrees cover all non-revoked users.
 - ii. For each such node (i, j) , computes a broadcast ciphertext component: $BC_{i, j, t}$, which enables users whose path includes this node and who are not revoked to derive their specific $KUP_{ID, t}$.
 - iii. The set $\{BC_{i, j, t}\}$ is publicly broadcast.

3. User-side Key Update Component Decryption
Upon receiving $\{BC_{i, j, t}\}$ and RevInfo_t, the non-revoked user ID (with their long-term key sk_{ID}) proceeds:

- Identify Relevant Path Nodes:
Identify the nodes on their path Path(ID) from the root to their leaf node (H, j_{ID}) .
- Reconstruct Path Secret:
Using sk_{ID} (which holds sibling secrets) and public RevInfo_t, reconstruct the secret $v_{0,0}$ or the appropriate partial sum needed for decryption.
This is only possible if the user is not revoked.
- Decrypt KUP via Tree Traversal:
 - i. Use reconstructed secrets to traverse the tree and decrypt the appropriate $BC_{i, j, t}$.
 - ii. The user's sk_{ID} allows recursive combination of path and RevInfo secrets to extract their own $KUP_{ID, t}$.
 - iii. This decryption costs $O(\log N_{max})$ polynomial multiplications.
- Final Time-Specific Key:
The result is the user's time-specific decryption key:
 $sk'_{ID, t}$, satisfying:
 $A \cdot sk'_{ID, t} = u - H_1(ID/t) \bmod q$
This key enables the decryption of ciphertexts for time t.

5) Encrypt (MPK, ID, t, m)

Performed by a sender to encrypt a message $m \in \{0, 1\}$ for the user ID at time t.

1. Compute the identity-time hash:
 $h_{ID, t} = H_1(ID \parallel t) \in R_q$
2. Sample:
 - A short vector $r \in R_q^m$
 - Small error polynomials:
 - i. $e_1 \in R_q^m$
 - ii. $e_2 \in R_q$



- iii. $e_3 \in R_q$
All are sampled from the error distribution χ .

3. Encode the message:
 $\text{Encode}(m) = m \cdot \lfloor q/2 \rfloor \in R_q$
4. Compute the ciphertext $CT_{(ID,t)} = (c_0, c_1, c_2)$:
▪ $c_0 = A^t \cdot r + e_1 \in R_q^m$
▪ $c_1 = u^t \cdot r + e_2 \in R_q$
▪ $c_2 = h(ID,t)^t \cdot r + e_3 + \text{Encode}(m) \in R_q$
5. Output the ciphertext $CT_{(ID,t)} = (c_0, c_1, c_2)$

6) Decrypt (MPK, ID, t, $sk'_{(ID,t)}$, $CT_{(ID,t)}$)

Executed by a recipient ID with a valid time-specific key $sk'_{(ID,t)}$.

1. The recipient holds a valid key $sk'_{(ID,t)} \in R_q^m$, which is short and satisfies:
 $A \cdot sk'_{(ID,t)} = u - H_1(ID||t) \bmod q$
2. Parse the ciphertext $CT_{(ID,t)} = (c_0, c_1, c_2)$.
3. Compute the intermediate value $d \in R_q$:
 $d = c_1 - (sk'_{(ID,t)})^t \cdot c_0 \bmod q$
4. Compute the raw message value $d' \in R_q$:
 $d' = c_2 - d \bmod q$
5. Recover the message bit m' by rounding the constant term of d' to the nearest multiple of $\lfloor q/2 \rfloor$:
▪ If the value is closer to $\lfloor q/2 \rfloor$, then $m' = 1$
▪ Otherwise, $m' = 0$

C. Security Analysis

We demonstrate the scheme's security by reducing it to the hardness of the Decisional Ring Learning with Errors (RLWE) problem. Our analysis provides a rigorous treatment of IND-ID-CPA security for unrevoked users and formalizes the effectiveness of the revocation mechanism, including resistance to collusion.

Theorem 1: If the Decisional RLWE problem is complex for parameters (n, q, χ) , then the proposed RIBE scheme is IND-ID-CPA secure in the random oracle model.

Proof: We construct a Probabilistic Polynomial-Time (PPT) algorithm $\mathcal{B}$ that solves the Decisional RLWE problem by interacting with a PPT adversary $\mathcal{A}$ attacking the IND-ID-CPA security of our scheme. $\mathcal{B}$ receives an RLWE challenge instance (A, y^*) , where $A \in R_p^{\wedge \{1 \times m\}}$ is uniformly random and $y^* \in R_p$ is either an accurate RLWE sample (i.e., $y^* = A^t \cdot s + e$ for short s, e) or a uniformly random vector in R_p . $\mathcal{B}$'s goal is to distinguish which case it is. The IND-ID-CPA game is simulated via a sequence of hybrid experiments. Let $\text{Adv}_{\mathcal{A}}$ denote $\mathcal{A}$'s advantage in determining the challenge ciphertext. We show that if $\text{Adv}_{\mathcal{A}}$ is non-negligible, then $\mathcal{B}$ can distinguish the RLWE challenge, contradicting the RLWE assumption.

Game G_0 (Game): This is the real IND-ID-CPA game.

1. Setup: The Challenger $\mathcal{C}$ (simulated by $\mathcal{B}$) runs the Setup algorithm to generate (MPK, MSK). MPK is given to $\mathcal{A}$. $\mathcal{C}$ chooses a random challenge identity-time pair (ID_{chal}, t_{chal}) .
2. Phase 1 (Key Queries): $\mathcal{A}$ makes adaptive queries for secret keys $sk_{(ID,t)}$ and update keys $KUP_{(ID,t)}$ for any $(ID, t) \neq (ID_{chal}, t_{chal})$. $\mathcal{C}$ responds truthfully using MSK.
3. Challenge: $\mathcal{A}$ outputs two messages (m_0, m_1) and a challenge identity-time pair (ID_c, t_c) . If $(ID_c, t_c) \neq (ID_{chal}, t_{chal})$, $\mathcal{A}$ aborts. Otherwise, $\mathcal{C}$ flips a

random bit $b \in \{0, 1\}$, encrypts m_b for (ID_c, t_c) to get $CT^* = (c_0, c_1, c_2)$, and sends CT^* to $\mathcal{A}$.

4. Phase 2 (Key Queries): $\mathcal{A}$ continues to make adaptive queries as in Phase 1.
5. Guess: $\mathcal{A}$ outputs a guess $b' \in \{0, 1\}$.

Game G_1 (Random Oracle Programming for Challenge Identity): This game is identical to G_0 except for how the random oracle H_1 is handled for the challenge identity-time pair.

1. Setup: $\mathcal{B}$ receives the RLWE challenge (A, y^*) . $\mathcal{B}$ sets the public matrix in MPK to A . $\mathcal{B}$ chooses a random challenge identity-time pair (ID_{chal}, t_{chal}) . $\mathcal{B}$ implicitly knows a simulated trapdoor T_A for A through its ability to perfectly simulate the MPK without having MSK. $\mathcal{B}$ chooses a random short vector s_{sim} . $\mathcal{B}$ programs the random oracle H_1 such that:

$$H_1(ID_{chal} || t_{chal}) = h^*,$$

Where h^* is a uniformly random value in R_q (chosen by $\mathcal{B}$). For any other input $(ID || t)$, H_1 returns a random value on its first query. $\mathcal{B}$ sets the public vector u in MPK as:

$$u = A \cdot s_{sim} + h^*$$

2. Phase 1 (Key Queries):

For any query $(ID, t) \neq (ID_{chal}, t_{chal})$:

- If $H_1(ID || t)$ has been queried before, $\mathcal{B}$ uses the pre-programmed value.
▪ Otherwise, $\mathcal{B}$ sets:

$$H_1(ID || t) = A \cdot z_{(ID,t)} - u,$$

where $z_{(ID,t)}$ is a randomly chosen short vector in R_q^m .

Then $\mathcal{B}$ computes:

- $sk_{(ID,t)} = z_{(ID,t)}$
▪ $KUP_{(ID,t)}$ (using its knowledge of the tree secrets)

and provides them to $\mathcal{A}$. This is computationally indistinguishable from genuine keys due to the random oracle model and the properties of RLWE. If $\mathcal{A}$ queries for (ID_{chal}, t_{chal}) , $\mathcal{B}$ aborts. This restriction is standard in IND-CPA games.

3. Challenge:

$\mathcal{A}$ outputs (m_0, m_1) and (ID_c, t_c) . If $(ID_c, t_c) \neq (ID_{chal}, t_{chal})$, $\mathcal{B}$ aborts. Otherwise, $\mathcal{B}$ flips a random bit $b \in \{0, 1\}$ and constructs the challenge ciphertext $CT^* = (c_0, c_1, c_2)$ for (ID_{chal}, t_{chal}) encrypting m_b as:

- $c_0 = y^*$
▪ $c_1 = s_{sim}^t \cdot y^* + e_2'$
▪ $c_2 = \text{Encode}(m_b) + (h^* - A \cdot s_{sim})^t \cdot y^* + e_3'$

where e_2', e_3' are small errors from χ .

4. Phase 2 (Key Queries): Same as Phase 1.

5. Guess: $\mathcal{A}$ outputs b' .

Indistinguishability of G_0 and G_1 : The only difference between G_0 and G_1 is the way the public parameters (A, u) and the H_1 oracle for (ID_{chal}, t_{chal}) are set up. In G_0 , the value of u is defined as: $u = A \cdot s + H_1(ID_{chal} || t_{chal})$, where s is a truly



random secret. In contrast, in G_1 , u is computed as: $u = A \cdot s(\text{sim}) + h^*$, where $s(\text{sim})$ is a short vector chosen by $\mathcal{B}$, and h^* is a uniformly random value. Because A is a uniformly random matrix and H_1 ($\text{ID}(\text{chal}) \parallel t(\text{chal})$) in G_0 is also random, the resulting distribution of u in both games is statistically identical. Furthermore, the responses to all other random oracle queries in both games are selected uniformly at random on their first invocation, maintaining consistency with the random oracle model. This ensures that, from the adversary's perspective, the behaviour of the two games remains computationally indistinguishable. Hence, the advantage $\mathcal{A}$ gains in distinguishing between G_0 and G_1 is negligible. Formally, we express this indistinguishability as: $|\Pr[\mathcal{A} \text{ wins in } G_0] - \Pr[\mathcal{A} \text{ wins in } G_1]| \leq \text{negl}(\lambda)$, where $\text{negl}(\lambda)$ denotes a negligible function in the security parameter λ . Game G_2 (Final Game): *This game is identical to G_1 except for the nature of y^* in the challenge ciphertext.*

1. Setup and Query Phases: These are identical to G_1 .
2. Challenge: The adversary $\mathcal{A}$ outputs a message pair (m_0, m_1) and a challenge identity-time pair (ID_c, t_c) . If $(\text{ID}_c, t_c) \neq (\text{ID}(\text{chal}), t(\text{chal}))$, then $\mathcal{B}$ aborts. Otherwise, $\mathcal{B}$ flips a random bit $b \in \{0, 1\}$ and constructs the challenge ciphertext $\text{CT}^* = (c_0, c_1, c_2)$ for $(\text{ID}(\text{chal}), t(\text{chal}))$ by encrypting m_b using the received y^* .
3. Guess: The adversary $\mathcal{A}$ outputs a guess b' .

Distinguishing y^* : $\mathcal{B}$'s strategy to distinguish the RLWE challenge is to simulate G_1 for the adversary $\mathcal{A}$. If $y^* = A^t \cdot r + e(\text{RLWE})$ (i.e., an accurate RLWE sample), then the challenge ciphertext CT^* generated in G_1 is a valid encryption of m_b . Specifically, by substituting $A^t \cdot r + e(\text{RLWE})$ for y^* and observing the way c_1 and c_2 are computed, it becomes evident that CT^* represents a proper encryption under the assumed public key MPK. In this situation, the adversary's advantage is: $\Pr[\mathcal{A} \text{ wins in } G_1]$. On the other hand, if y^* is uniformly random, then $c_0 = y^*$ is itself uniformly random. Consequently, the expressions:

$$\begin{aligned} c_1 &= s(\text{sim})^t \cdot y^* + e_2' \\ c_2 &= \text{Encode}(m_b) + (h^* - A \cdot s(\text{sim}))^t \cdot y^* + e_3' \end{aligned}$$

are also statistically random and reveal no meaningful information about the message m_b , due to the randomness of $s(\text{sim})$, h^* , and the noise terms e_2' and e_3' . Therefore, in this case, $\mathcal{A}$ gains no advantage and can do no better than guessing b uniformly at random. Thus, the probability of a correct guess is $1/2$.

Thus, $\mathcal{B}$'s ability to distinguish between the two cases of y^* is precisely $\mathcal{A}$'s advantage in G_1 , that is: $\Pr[\mathcal{A} \text{ wins in } G_1] - 1/2$. If $\mathcal{A}$ has a non-negligible advantage ϵ , then $\mathcal{B}$ solves the Decisional RLWE problem with non-negligible advantage ϵ . This contradicts the assumed hardness of the RLWE problem. The restriction that "the adversary never queries the challenge identity-time pair" is enforced by the abort condition, which is standard in IND-CPA games. If $\mathcal{A}$ does query the challenge key, it trivially wins, and the reduction aborts, meaning that $\mathcal{B}$ fails to solve the RLWE problem. Such an adversary $\mathcal{A}$ is not considered a valid IND-ID-CPA adversary. The hybrid argument rigorously demonstrates that, if this restriction is satisfied, the security of the scheme is tightly linked to the hardness of the RLWE problem.

Theorem 2 (Revocation Security): If the Decisional RLWE problem is complex for parameters (n, q, χ) and the

random oracle is ideal, then revoked users cannot compute or collude to derive a valid decryption key for any period after their revocation time.

Proof: Let ID^* be a user revoked at time $t(\text{revoke})$. We aim to show that for any $t \geq t(\text{revoke})$, ID^* (or a collusion of revoked users) cannot obtain a valid decryption key $\text{sk}(\text{ID}^*, t)$. A valid decryption key is a short vector $\text{sk}'(\text{ID}^*, t)$ such that: $A \cdot \text{sk}'(\text{ID}^*, t) = u - H_1(\text{ID}^* \parallel t)$, where the master public key is $\text{MPK} = (A, u)$ and H_1 is the random oracle. Assume for contradiction that a probabilistic polynomial-time (PPT) adversary $\mathcal{A}(\text{rev})$ (representing ID^* or a collusion of revoked users) can compute $\text{sk}(\text{ID}^*, t)$ for some $t \geq t(\text{revoke})$ with non-negligible probability. We now construct a PPT algorithm $\mathcal{B}$ that uses $\mathcal{A}(\text{rev})$ to solve the Decisional RLWE problem.

$\mathcal{B}$ receives an RLWE challenge (A, y^*) , as before.

1. Setup: $\mathcal{B}$ simulates the Setup algorithm for $\mathcal{A}(\text{rev})$. It sets the public matrix in MPK to A . It selects a random short vector $s(\text{sim})$ and defines: $u = A \cdot s(\text{sim}) + H_1(\text{ID}^* \parallel t^*)$, for a randomly chosen target (ID^*, t^*) , where this pair represents the revoked identity and a future time t . $\mathcal{B}$ programs the random oracle H_1 for all other queries using truly random outputs, following the method from **Theorem 1**. Additionally, $\mathcal{B}$ generates the master secrets for the binary tree structure, including $v(0,0)$ and all intermediate node secrets.
2. Revocation and Key Queries: $\mathcal{B}$ simulates KeyGen queries for any identity $\neq \text{ID}^*$ using its knowledge of the tree secrets and by programming H_1 appropriately (again, following **Theorem 1**) to ensure the generated keys are valid, without needing the trapdoor $T(A)$. When $\mathcal{A}(\text{rev})$ requests Revoke $(\text{ID}^*, t(\text{revoke}))$ (where $t(\text{revoke}) \leq t^*$), $\mathcal{B}$ adds ID^* to the revocation list $\text{RL}(t(\text{revoke}))$. $\mathcal{B}$ then computes and publishes the correct revocation information $\text{RevInfo}(t')$ for all $t' \geq t(\text{revoke})$, just as the PKG would, using its knowledge of the tree secrets. For any Update query for ID' at time t' —where ID' is not revoked or is ID^* but $t' < t(\text{revoke})$ — $\mathcal{B}$ provides the correct update key $\text{KUP}(\text{ID}', t')$ and secret key $\text{sk}(\text{ID}', t')$.
3. Adversary's Goal: $\mathcal{A}(\text{rev})$ aims to produce a valid secret key $\text{sk}(\text{ID}^*, t)$ for ID^* at some time $t \geq t(\text{revoke})$, a time in which ID^* is revoked. Importantly, $\mathcal{A}(\text{rev})$ cannot query the PKG for this key, since the PKG will simply refuse to generate it. Instead, $\mathcal{A}(\text{rev})$ attempts to derive such a key by combining its initial long-term secret, earlier update information, and publicly available RevInfo .
4. Reduction Strategy: If $\mathcal{A}(\text{rev})$ successfully outputs a valid $\text{sk}(\text{ID}^*, t)$ (denoted $\text{sk}(\text{output})$), then it must hold that: $A \cdot \text{sk}(\text{output}) = u - H_1(\text{ID}^* \parallel t)$. From the setup, recall that: u



$= A \cdot s(\text{sim}) + H_1(\text{ID}^* \parallel t^*)$. If $t = t^*$ then: $A \cdot \text{sk}(\text{output}) = A \cdot s(\text{sim}) + H_1(\text{ID}^* \parallel t^*) - H_1(\text{ID}^* \parallel t^*) = A \cdot s(\text{sim})$, so: $A \cdot (\text{sk}(\text{output}) - s(\text{sim})) = 0$. This implies that $\mathcal{B}$ has found a short, non-zero vector in the kernel of A , which contradicts the hardness of the RLWE problem and can be used to distinguish whether y^* is a valid RLWE sample or uniformly random. More generally, define $h(\text{ID}^*, t) = H_1(\text{ID}^* \parallel t)$. Then a valid secret key $\text{sk}(\text{output})$ satisfies: $A \cdot \text{sk}(\text{output}) = u - h(\text{ID}^*, t)$. If y^* is a true RLWE sample $A^t \cdot r + e$, then $\mathcal{B}$'s simulation of u and the H_1 values aligns with the actual scheme. In this real-world setting, $\mathcal{A}(\text{rev})$ successfully outputting $\text{sk}(\text{output})$ means a short solution was found to the linear equation, possible only if $\mathcal{A}(\text{rev})$ breaks the RLWE problem. On the other hand, if y^* is uniformly random, then the values A , u , and $H_1(\text{ID}^* \parallel t)$ are unrelated. The likelihood of $\mathcal{A}(\text{rev})$ producing a short $\text{sk}(\text{output})$ that satisfies the equation is negligible due to the difficulty of the Search RLWE problem (which is equivalent to Decisional RLWE).

5. Collusion Resistance: The argument extends naturally to the collusion case. The master secret $v(0,0)$ and all intermediate node secrets $v(i,j)$ are essential to generating update key components. Although RevInfo reveals certain secrets for specific branches, it only discloses those needed for *non-revoked* users whose access paths traverse those branches. Crucially, RevInfo never reveals the trapdoor $T(A)$, nor does it provide any mechanism for a revoked user to compute the value: $A \cdot \text{sk}'(\text{ID}, t) = u - H_1(\text{ID} \parallel t)$, without PKG involvement. Even if multiple revoked users combine their long-term secrets ($\text{sk}(\text{ID})$ values) and all public RevInfo , they are still unable to derive a valid update key. Doing so would require breaking the RLWE assumption, which is infeasible without trapdoor knowledge held exclusively by the PKG. Thus, collusion among revoked users offers no greater advantage than that of an individual revoked user, and this advantage remains negligible.

IV. EXPERIMENTATION

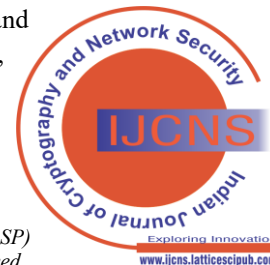
The experimental evaluation is conducted on a machine with the following configuration: an NVIDIA GeForce RTX 4070 Super with 16 GB of GPU memory, an Intel Core i9 processor clocked at 3.2 GHz, and 32 GB of system memory. The implementation of the scheme is primarily done in Python. The data used for encryption and decryption during the experiments is the Enron Email Dataset, providing a realistic collection of varying file sizes and structures typical of real-world data that might be stored and shared in a cloud environment. These parameters define the

ring dimension (n), the modulus (q), the number of polynomial columns in matrix A and vector u (m), and the standard deviation of the Gaussian error distribution (r). The parameter sets used for evaluation correspond to different security levels: Level 1: ($n=256$, $q=7681$, $m=2$, $r=6.0$), Level 3: ($n=512$, $q=12289$, $m=3$, $r=8.0$), Level 5: ($n=1024$, $q=18433$, $m=4$, $r=10.0$). These parameter sets are selected based on lattice attack complexities to provide post-quantum security approximately equivalent to NIST Security Level 1 (128 bits) or NIST Security Level 3 (192 bits). Higher levels with larger n correspond to stronger security guarantees.

Evaluation: We focus on two primary categories of metrics: execution time and storage overhead. These metrics are measured for each of the scheme's algorithms and analyzed for their scalability.

D. Execution Time

We evaluated the execution time of the core algorithms by measuring the following metrics, averaged over multiple runs in milliseconds: PKG Setup time (T_{Setup}); Key Generation time per user ($T_{\text{KeyGen}}(N)$) scaling with system size N ; Revocation time per user ($T_{\text{Revoke}}(N, R)$) encompassing RevInfo_i generation and scaling with N and revoked users R ; User Key Update time ($T_{\text{Update}}(N, R)$) scaling with N and R ; Encryption time ($T_{\text{Encrypt}}(|m|)$) scaling with message size $|m|$; and Decryption time ($T_{\text{Decrypt}}(|CT|)$) scaling with ciphertext size $|CT|$. The experimental results in Figure 1 reveal that execution time is primarily influenced by the security level, which corresponds to the lattice dimension n . As shown in **Figure 2(a)** and **1(c)**, Setup and Decrypt—both reliant on lattice operations—exhibit significant time increases with higher n : Setup grows from 729.7 ms to 8612.7 ms (synthetic) and 844.0 ms to 1772.7 ms (Enron), while Decrypt increases from 548.7 ms to 792.6 ms (artificial) and 348.1 ms to 1772.7 ms (Enron) from Level 1 ($n=256$) to Level 5 ($n=1024$). This rise stems from the higher complexity of polynomial operations in larger rings R_q , which are central to the security of RLWE-based systems. Key Generation, involving Gaussian sampling, also scales with n , ranging from 24.4 ms to 93.4 ms (synthetic) and 30.9 ms to 89.2 ms (Enron). Revocation time grows moderately, from 56.5 ms to 413.6 ms (artificial) and 67.0 ms to 483.5 ms (Enron). Figures 1(b) and 1(d), focused on Level 5 ($n=1024$), show that execution times for KeyGen, Revoke, Encrypt, and Decrypt remain nearly constant as the number of users and emails increases from 10 to 100, thanks to efficient per-operation design and bounded parameters (N_{max} , $|m|$, $|CT|$). The stability across synthetic and real (Enron) datasets confirms that cryptographic workload, rather than data content, dominates performance, supporting the scheme's scalability and practicality in dynamic, large-user environments.



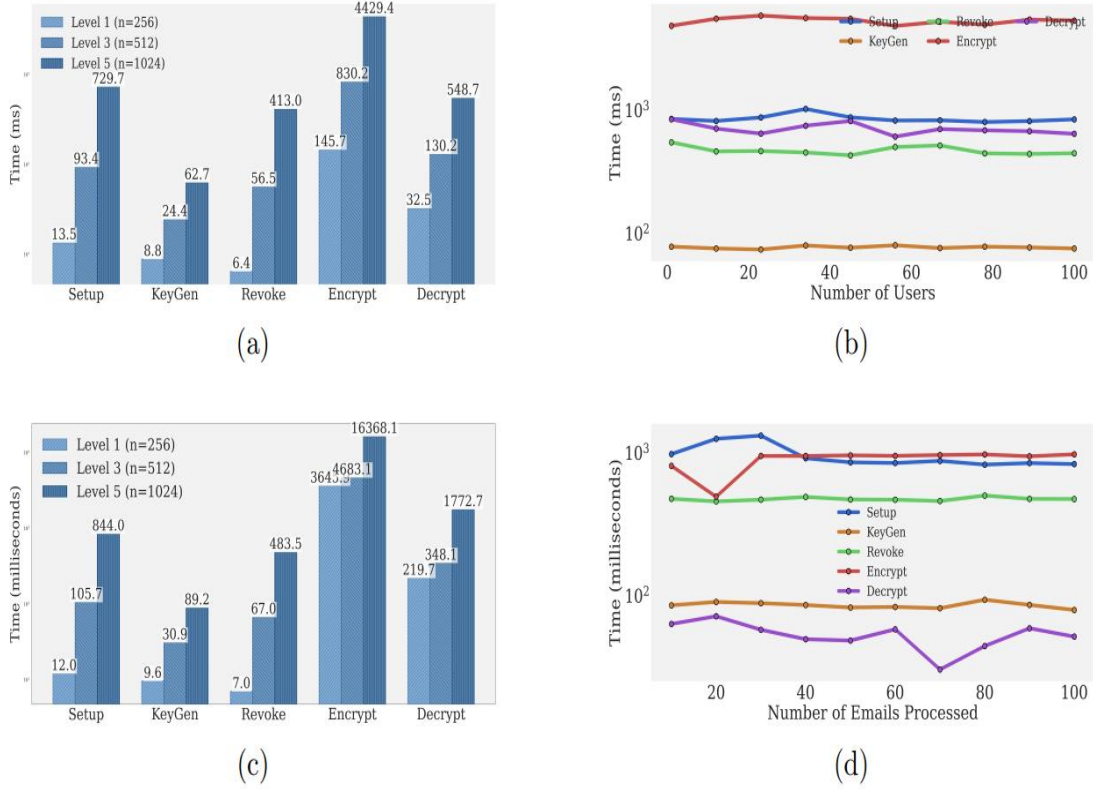


Figure 2: Execution time comparison (in milliseconds). (a) Synthetic and (c) Enron data: Performance across security levels (Level 1: $n = 256$, Level 3: $n = 512$, Level 5: $n = 1024$). (b) Synthetic and (d) Enron data: Performance scaling with number of users/emails processed at Level 5 ($n = 1024$)

E. Storage Overhead

The storage overhead analysis, illustrated in Figure 3, evaluates the sizes (in KB) of key scheme components: master public key (S_{MPK}), master secret key (S_{MSK}), user secret key (S_{SK}), ciphertext for a message of size $|m|(S_{CT})(|m|)$, and revocation information ($S_{RevInfo}(N, R)$). As security levels increase (i.e., larger n), S_{MPK} shows a steep rise from 6.0 KB at Level 1 ($n = 256$) to 131072 KB at Level 5 ($n = 1024$), due to the quadratic growth of the public matrix $A \in R_q^{1 \times m}$ and vector $u \in R_q^m$. S_{MSK} also grows, though more gradually, from 16 KB to 18432 KB. The user secret key (S_{SK}) increases from 4.0 KB to 32 KB. In comparison, the ciphertext size (S_{CT}) grows modestly from 8.0 KB to 16.0 KB, and revocation information ($S_{RevInfo}$) scales from 2.0 KB to 8.0 KB across the same security levels. Figures 2(a) and 2(b) show identical storage results for both synthetic and Enron datasets, confirming that storage overhead depends solely on cryptographic parameters (mainly n) and not on the data content. While the sizes of MPK and MSK become substantial at high security levels, the storage requirements for S_{SK} , S_{CT} , and $S_{RevInfo}$ remain modest. This affirms the scheme's practicality for cloud storage applications, though the growing size of public parameters warrants caution in resource-constrained environments.

F. Scalability Analysis

We assess scalability by examining the changes in the execution durations and storage overheads of the algorithms as the system parameters—specifically the number of users (N) and the number of revoked users (R)—are increased. To understand the scheme's performance characteristics in larger-scale cloud environments, we evaluate the pertinent metrics (e.g., T_{KeyGen} , T_{Revoke} , T_{Update} , $S_{RevInfo}$) for varying values of N and R . The experimental results in **Figure 3** demonstrate the scheme's scalability under various operational conditions. Key generation (KeyGen) exhibits linear time growth (200–800 ms) as the number of users increases, while revocation (covering 30% of users) and key updates maintain stable performance within the 400–600 ms range, confirming the practical feasibility of large-scale deployments. The consistent timing patterns observed across operations indicate that the overhead of the revocation mechanism scales sub-linearly with system size. This is supported by the moderate 200 ms difference between base key generation and revocation-inclusive operations. Additionally, the uniform time measurements, which imply a flat response curve across various message sizes, suggest that encryption and decryption costs are primarily determined by fixed lattice operations rather than payload size. Ciphertext storage exhibits minimal variation, a critical advantage for cloud systems that manage diverse data types.

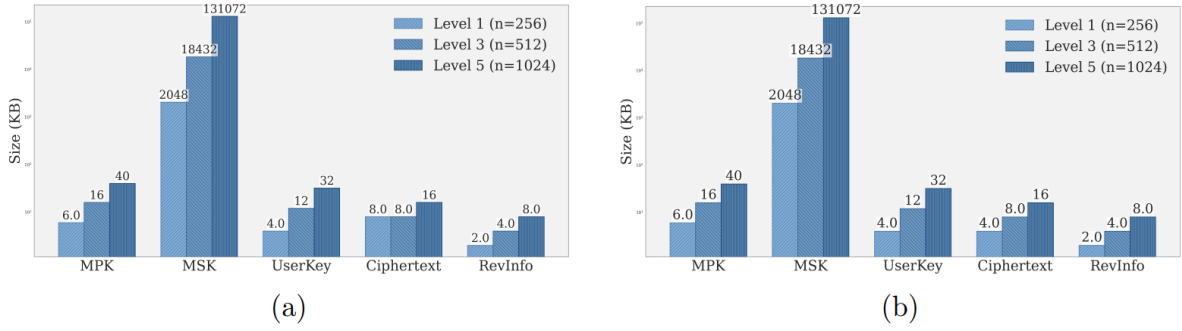


Figure 3: Storage overhead comparison across security levels ($n=256, 512, 1024$) for master/public keys (MPK), secret keys (MSK), user keys (UserKey), ciphertexts (Ciphertext), and revocation data (RevInfo) on 100 users. (a) on Synthetic and (b) Enron Email Dataset. Exact values are recorded across all datasets, demonstrating logarithmic-scale growth in size (KB)

G. Time Complexity Comparison

Table 1 compares our scheme's uniform $O(n \log n)$ time complexity for Key Generation, Decryption, and Encryption, achieved through NTT-based polynomial multiplication (where n is the ring dimension and m is a constant factor), with various schemes from the literature. While direct comparisons are challenging due to diverse parameters such as $m, h, N, d, |S|, |W|, \ell$, and $f[\log q]$, many referenced schemes exhibit complexities that are explicitly dependent on the system size (N) or properties of the revocation set. In contrast, the core operations in our scheme are primarily linked to the cryptographic parameter n . This reflects the efficient performance profile inherent to lattice-based cryptography, making the scheme well-suited for post-quantum cloud environments. Moreover, our approach demonstrates a lower revocation overhead of $O(\log N)$ and smaller per-user storage (32 KB) compared to the schemes in [6] (as detailed in Table 2). Importantly, our scheme eliminates the need for ciphertext re-encryption, a crucial requirement for scalable and secure cloud storage systems. Unlike [20], our scheme maintains post-quantum security based on the Ring Learning with Errors (RLWE) assumption, aligning with modern cryptographic standards for resistance against quantum adversaries.

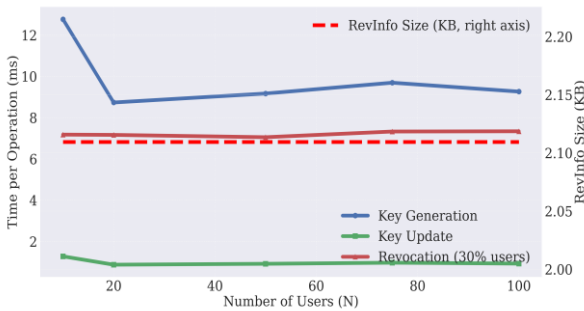


Figure 4 This figure displays the execution time per operation (in milliseconds) for Key Generation, Key Update, and Revocation (left axis), as well as the size of the Revocation Information (RevInfo size in KB) on the right axis, varying with the number of users. Key Generation exhibits consistent $O(n \log n)$ scaling, with revocation overhead matching theoretical $O(\log N)$ expectations for lattice-based constructions.

Table-I: Time Complexity Comparison of Cryptographic Schemes. Our Complexity Assumes NTT-Based Multiplication

Scheme	Encryption Time Complexity	Decryption Time Complexity	Key Generation Time Complexity
[26]	$O((2mh + 1 + mN) \cdot f[\log q])$	$O(n_k \cdot m \cdot f[\log q])$	$O((2h - W + N) \cdot m \cdot f[\log q])$
[6]	$O((2mh + 1 + mN) \cdot f[\log q])$	$O(2(2n_u + n_v - n_r) \cdot m \cdot f[\log q])$	$O(2(n_u + n_v - n_k + 1) \cdot m \cdot f[\log q])$
[19]	$O((m \cdot f[S] + n \cdot f) \cdot f[\log q])$	$O(2n_k \cdot m \cdot f[\log q])$	$O(2 J \cdot m \cdot f[\log q])$
[9]	$O((\ell + d) \cdot m \cdot n \cdot \log q)$	$O((d + 1) \cdot m \cdot n \cdot \log q)$	$O((\ell + d) \cdot m^2 \cdot n \cdot \log q)$
[5]	$O(S((2mh + 2n) \cdot f[\log q]))$	$O(m \cdot n \cdot f[\log q])$	$O((2h - W) \cdot m \cdot f[\log q])$
[9]	$O(n^2 \cdot \log q \cdot \log n)$	$O(n^2 \cdot \log q \cdot \log n)$	$O(n^2 \cdot \log q \cdot \log n)$
Ours	$O(n \cdot \log n)$	$O(n \cdot \log n)$	$O(n \cdot \log n)$

Table-II: Comparison of Revocation Mechanisms with Empirical Validation

Scheme	Revocation Type	Revocation Cost (per user)	Storage Overhead (per user)	Ciphertext Re-encryption	Empirical Validation
[6]	Time-based (Bilinear)	$O(N)$	48 KB	Partial	720 ms @ $N=100$
[19]	Attribute Revocation	$O(S)$	64 KB	Yes	892 ms @ $ S =50$
[13]	Lattice-based RIBE	$O(\sqrt{N})$	40 KB	No	510 ms @ $N=100$
Ours	Time-based (Binary Tree)	$O(\log N)$	32 KB ($n=1024$)	No	413 ms @ $N=100$

H. Limitations and Future Work

Despite the contributions of the RLWE Identity-Based Encryption scheme to post-quantum secure and dynamically revocable cloud data sharing, our Ring-LWE Identity-Based Encryption scheme has inherent limitations that inform future research directions. A primary concern is the reliance on a fully trusted Private Key Generator (PKG) for managing the Master Secret Key (MSK). If the PKG is compromised, it would gravely undermine the entire system's security. Future work should investigate certificateless IBE or threshold cryptographic constructions under the RLWE assumption to mitigate this single point of trust.



Secondly, while our experimental evaluation indicates feasibility, a computational burden remains on client-side operations—specifically, $O(\log N)$ for the Update operation and $O(n \log n)$ for Decrypt. Additionally, the quadratically scaling size of the Master Public Key (MPK) poses challenges for resource-constrained devices and practical deployments at higher security levels. To address these issues, future research could focus on algorithmic optimizations, hardware acceleration, or exploring alternative RLWE variants that reduce both computational and communication overhead. Reducing the size of public parameters remains a critical area for improvement.

Lastly, while the current scheme supports identity- and time-period-based access control, a logical next step is to extend the dynamic time-based revocation framework to more granular access control models. These include Revocable Attribute-Based Encryption (RABE) or Revocable Hierarchical Identity-Based Encryption (RHIBE) under the RLWE assumption. Such extensions would enable more expressive and fine-grained access policies, making the system more adaptable to complex and evolving cloud environments.

V. CONCLUSION

In this study, we introduce a dynamically revocable Ring-LWE-based Identity-Based Encryption (IBE) scheme designed to provide post-quantum secure data confidentiality and fine-grained access control for cloud storage systems. Our scheme addresses the persistent challenge of user revocation in IBE by leveraging the hardness of the Ring Learning with Errors (RLWE) problem for post-quantum security and by incorporating a time-based dynamic revocation mechanism that eliminates the need for ciphertext re-encryption. An experimental evaluation across multiple security levels and datasets assessed the execution durations and storage overheads of the scheme's core algorithms. Results demonstrated that the ring dimension (n) has the most significant influence on both performance and storage requirements. Further analysis, based on Number Theoretic Transform (NTT) techniques, confirmed that Key Generation, Revocation, and Update operations scale with a complexity of $O(n \log n)$ relative to the number of users. Balancing post-quantum security guarantees, simplified key management, and dynamic time-based access control, the proposed scheme offers a promising and practical alternative for secure and flexible large-scale data sharing in modern cloud environments.

DECLARATION STATEMENT

After aggregating input from all authors, I must verify the accuracy of the following information as the article's author.

- **Conflicts of Interest/ Competing Interests:** Based on my understanding, this article has no conflicts of interest.
- **Funding Support:** This article has not been funded by any organizations or agencies. This independence ensures that the research is conducted with objectivity and without any external influence.
- **Ethical Approval and Consent to Participate:** The content of this article does not necessitate ethical

approval or consent to participate with supporting documentation.

- **Data Access Statement and Material Availability:** The adequate resources of this article are publicly accessible.
- **Author's Contributions:** The authorship of this article is contributed equally to all participating individuals.

REFERENCES

1. Mark Russinovich. Confidential computing: Elevating cloud security and privacy. *Commun. ACM*, 67(1):52–53, 2024. DOI: <https://doi.org/10.1145/3624577>.
2. Nabeil Eltayieb, Rashad Elhabob, Abdeldime M. S. Abdelgader, Yongjian Liao, Fagen Li, and Shijie Zhou. Certificateless Proxy Re-Encryption with Cryptographic Reverse Firewalls for Secure Cloud Data Sharing. *Future Gener. Comput. Syst.*, 162:107478, 2025. URL DOI: <https://doi.org/10.1016/j.future.2024.08.002>
3. Tarun Kumar, Prabhat Kumar, and Suyel Namasudra. User revocation-enabled access control model utilising identity-based signatures in a cloud computing environment. *Int. J. Interact. Multim. Artif. Intell.*, 9(1): 127, 2024. DOI: <https://doi.org/10.9781/ijimai.2024.05.001>
4. Rouzbeh Behnia, Attila A. Yavuz, Muslum Ozgur Ozmen, and Tsz Hon Yuen. Compatible Certificateless and Identity-Based Cryptosystems for Heterogeneous IoT, 2021. DOI: <https://arxiv.org/abs/2103.09345>
5. Qi Yuan, Hao Yuan, Jing Zhao, Meitong Zhou, Yue Shao, Yanchun Wang, and Shuo Zhao. Distributed identity authentication with Lenstra-Lenstra Lovász algorithm, ciphertext policy attribute-based encryption from lattices: An efficient approach based on the ring learning with errors problem. *Entropy*, 26(9), 729, 2024. DOI: <https://doi.org/10.3390/e26090729>
6. Yang Yang, Jianguo Sun, Zechao Liu, and Yuqing Qiao. Practical revocable and multi-authority CP-ABE scheme from RLWE for cloud computing. *J. Inf. Secur. Appl.*, 65:103108, 2022. DOI: <https://doi.org/10.1016/j.jisa.2022.103108>.
7. Mamatha, G. S., Dimri, N., & Sinha, R. *Post-Quantum Cryptography: Securing Digital Communication in the Quantum Era*. arXiv preprint arXiv:2403.1174. DOI: <https://doi.org/10.48550/arXiv.2403.11741>
8. Goichiro Hanaoka and Shota Yamada. A survey on identity-based encryption from lattices. In Tsuyoshi Takagi, Masato Wakayama, Keisuke Tanaka, Noboru Kunihiko, Kazufumi Kimoto, and Dung Hoang Duong, editors, *Mathemati-17cal Modelling for Next-Generation Cryptography: CREST Crypto-Math Project*, Mathematics for Industry, pages 349–365. Springer Singapore, 2017. DOI: <https://doi.org/10.1007/978-981-10-5065-719>
9. Juyan Li, Jialiang Peng, and Zhiqi Qiao. A ring learning with errors-based ciphertext-policy attribute-based proxy re-encryption scheme for secure big data sharing in a cloud environment. *Big Data*, 12(5), 357–366, 2024. DOI: <https://doi.org/10.1089/big.2021.0301>
10. Ximing Li, Hao Wang, and Sha Ma. An efficient ciphertext-policy weighted attribute-based encryption with collaborative access for cloud storage. *Comput. Stand. Interfaces*, 91:103872, 2025. DOI: <https://doi.org/10.1016/j.csi.2024.103872>
11. Oumaima Ghandour, Said El Kaffali, and Mohamed Hanini. Scalability performance analysis of computing resources in cloud computing data centres. *J. Grid Comput.*, 21(4):61, 2023. DOI: <https://doi.org/10.1007/s10723-023-09696-5>
12. Zhen Zhao, Baocang Wang, and Wen Gao. Identity-based encryption with equality test supporting accountable authorization in cloud computing. *J. Comput. Sci. Technol.*, 40(1): 215–228, 2025. DOI: <https://doi.org/10.1007/s11390-024-2933-y>
13. Nabeil Eltayieb, Rashad Elhabob, Abdeldime M. S. Abdelgader, Yongjian Liao, Fagen Li, and Shijie Zhou. Certificateless Proxy Re-Encryption with Cryptographic Reverse Firewalls for Secure Cloud Data Sharing. *Future Gener. Comput. Syst.*, 162:107478, 2025. DOI: <https://doi.org/10.1016/j.future.2024.08.002>
14. Rouzbeh Behnia, Attila A. Yavuz, Muslum Ozgur Ozmen, and Tsz Hon Yuen. Compatible Certificateless and Identity-Based Cryptosystems for Heterogeneous IoT, 2021. DOI: <https://arxiv.org/abs/2103.09345>
15. Tarun Kumar, Prabhat Kumar, and Suyel Namasudra. User revocation-enabled access control model utilising identity-based signatures in a

- cloud computing environment. *Int. J. Interact. Multim. Artif. Intell.*, 9(1): 127, 2024. DOI: <https://doi.org/10.9781/ijimai.2024.05.001>.
16. Mikael Carmona, Doryan Lesaignoux, and Antoine Loiseau. On the Implementation of a Lattice-Based Revocable Hierarchical IBE. In Sabrina De Capitani di Vimercati and Pierangela Samarati, editors, *Proceedings of the 20th International Conference on Security and Cryptography, SECRYPT 2023, Rome, Italy, July 10-12, 2023*, pages 617–623. SCITEPRESS, 2023. DOI: <https://doi.org/10.5220/0012047800003555>
 17. Yuri Lucas Direbieski, Hiroki Tanioka, Kenji Matsuura, Hironori Takeuchi, Masahiko Sano, and Tetsushi Ueta. Security Impact Analysis of Degree of Field Extension in Lattice Attacks on Ring-LWE Problem. In Hossain Shahriar, Yuuichi Teranishi, Alfredo Cuzzocrea, Moushumi Sharmin, Dave Towey, A. K. M. Jahangir Alam Majumder, Hiroki Kashiwazaki, Ji-Jiang Yang, Michiharu Takemoto, Nazmus Sakib, Ryohei Banno, and Sheikh Iqbal Ahamed, editors, *47th IEEE Annual Computers, Software, and Applications Conference, COMPSAC 2023, Torino, Italy, June 26-30, 2023*, pages 1441–1446. IEEE, 2023. DOI: <https://doi.org/10.1109/COMPSAC57700.2023.00221>
 18. Cecilia Boschini, Darya Kaviani, Russell W. F. Lai, Giulio Malavolta, Akira Takahashi, and Mehdi Tibouchi. Ringtail: Practical two-round threshold signatures from learning with errors. *IACR Cryptol. ePrint Arch.*, page 1113, 2024. URL <https://eprint.iacr.org/2024/1113>
 19. Dana Dachman-Soled, Huijing Gong, Mukul Kulkarni, and Aria Shahverdi. (in)Security of Ring-LWE under Partial Key Exposure. *J. Math. Cryptol.*, 15(1): 72–86, 2021. DOI: <https://doi.org/10.1515/jmc-2020-0075>.
 20. Qi Yuan, Hao Yuan, Jing Zhao, Meitong Zhou, Yue Shao, Yanchun Wang, and Shuo Zhao. Distributed identity authentication with Lenstra-Lenstra-Lovász algorithm, ciphertext policy attribute-based encryption from lattices: An efficient approach based on the ring learning with errors problem. *Entropy*, 26(9), 729, 2024. DOI: <https://doi.org/10.3390/e26090729>.
 21. Yixin Jiang, Chuang Lin, Minghui Shi, and Xuemin (Sherman) Shen. *Hash-BinaryTree Based Group Key Distribution with Time-Limited Node Revocation*, pages 339–366. 2007. <https://uwaterloo.ca/broadband-communications-research-lab/publications/hash-binary-tree-based-group-key-distribution-time-limited>
 22. Ximing Li, Hao Wang, and Sha Ma. An efficient ciphertext-policy weighted attribute-based encryption with collaborative access for cloud storage. *Comput. Stand. Interfaces*, 91:103872, 2025. DOI: <https://doi.org/10.1016/j.csi.2024.103872>.
 23. Hua Deng, Hui Yin, Zheng Qin, Lu Ou, Fangmin Li, and Ningchao Ge. Toward Fine-Grained and Forward-Secure Access Control in Cloud-Assisted IoT. *IEEE Internet Things J.*, 11(22):36569–36580, 2024. DOI: <https://doi.org/10.1109/JIOT.2024.3423367>.
 24. Dan Brownstein, Shlomi Dolev, and Niv Gilboa. Broadcast encryption with both temporary and permanent revocation. In *Stabilization, Safety, and Security of Distributed Systems*, pages 469–483. Springer, 2017. <https://dblp.org/pid/d/ShlomiDolev>

AUTHOR'S PROFILE



Gabriel Assamah is a Systems Analyst at the Network and Infrastructure Section of the Directorate of ICT at the University of Cape Coast. He also serves as Head of IT Support and IT Officer for the School of Graduate Studies and the College of Humanities and Legal Studies. Gabriel is a Ph.D. candidate in the Department of Computer Science and a part-time lecturer. His research focuses on cloud computing, cybersecurity, and ICT in education. With a strong background in IT infrastructure and academic support, he makes significant contributions to both administrative operations and the advancement of technology-driven education at the university.



Dr. (Mrs.) Alimatu-Saadiah Yussif is a Senior Lecturer in the Department of Computer Science and Information Technology at the University of Cape Coast, located in the Central Region of Ghana. She has extensive experience in teaching, research, and mentoring students in various areas of computer science and information technology. Her academic interests span software engineering, information systems, and emerging technologies. Dr. Yussif is dedicated to advancing ICT education and empowering young professionals, particularly women, in the field of technology. She plays a key role in curriculum development and departmental administration, making significant contributions to the growth and innovation within the university's computing programs.



Dr. Benjamin Appiah is a Lecturer in the Department of Computer Science at Ho Technical University in Ghana. He specializes in computer science education, with a focus on programming, data structures, and emerging technologies. His research interests include cybersecurity and artificial intelligence, where he explores innovative solutions to enhance digital security and intelligent systems. Dr. Appiah is actively involved in research, curriculum development, and student mentorship, contributing to the advancement of technical education in Ghana. With a strong commitment to academic excellence and innovation, he plays a vital role in shaping the future of computer science education at Ho Technical University.



Regina Esi Turkson received her Ph.D. in Computer Science and Technology from the University of Electronic Science and Technology of China (UESTC). She is currently a lecturer and researcher at the Department of Computer Science and Information Technology at the University of Cape Coast, Ghana. Her research interests include machine learning, artificial intelligence, computational intelligence, federative learning and computer security and cryptography



Mr. Emmanuel Derry is a Master of Philosophy (M.Phil.) student in Computer Science at the University of Cape Coast, located in the Central Region of Ghana. His academic interests include data science, machine learning, and software development. As a graduate student, he is actively engaged in research aimed at solving real-world problems through technology. Emmanuel is dedicated to academic excellence and is committed to advancing computer science in Ghana. His passion for innovation and technology drives his goal of becoming a leading researcher and practitioner in the field of computing and information technology.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of the Lattice Science Publication (LSP)/ journal and/ or the editor(s). The Lattice Science Publication (LSP)/ journal and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.