Lightweight Encryption Transformation for IoT Devices

Anita Kumari
Department of IT
IIIT Allahabad
Prayagraj, India
mcl2023005@iiita.ac.in

Junaid Alam
Department of IT
IIIT Allahabad
Prayagraj, India
rsi2022512@iiita.ac.in

Soumyadev Maity
Department of IT
IIIT Allahabad
Prayagraj, India
soumyadev@iiita.ac.in

***Abstract*—Symmetric and public key cryptography are both important in the implementation of network security. Symmetric key cryptography can achieve the same level of security as public key cryptography with a significantly smaller key, making it far more efficient for the encryption or authentication of massive data sets while public key cryptography is essential for securing communications in dynamic environments. However, traditional PKI-based protocols can be inefficient, especially in resource-limited systems like the IoT. This paper highlights how modern cryptographic techniques, such as bilinear pairing operations and encryption transformation, help overcome these challenges by improving security and efficiency.**

***Index Terms*—IoT, Cryptographic encryption, Encryption Transformation., Data Sharing.**

I. Introduction

A. Context of the Problem

With the rapid increase of interconnected devices in the IoT, secure data transmission has become a fundamental challenge. Network security relies on both symmetric and public key cryptography to ensure confidentiality, integrity, and authenticity. While symmetric cryptography is computationally efficient and can achieve the same level of security as public-key cryptography, it requires an initial key exchange, which is impractical in open and dynamic environments. Public key cryptography ensures secure key establishment but with computational overhead and certificate management issues, particularly in traditional Public Key Infrastructure (PKI)-based security mechanisms. PKI requires resource-intensive processes such as certificate verification, certification revocation list (CRL) maintenance, and continuous online connectivity with a trusted Certificate Authority (CA), making it less suitable for resource-constrained environments like IoT.

B. Research Problem

Traditional PKI-based security solutions are not suitable for IoT contexts since they rely heavily on certificate management operations such as certificate verification and revocation, as well as the requirement for constant internet connectivity with a trustworthy Certificate Authority (CA). This situation underscores the importance of lightweight, efficient, and secure cryptographic solutions that can reduce certificate administration requirements while providing solid security guarantees. Furthermore, present transformation schemes are frequently inefficient and costly, emphasizing the critical need for solutions that enable safe and efficient data transfer among resource-constrained IoT devices.

TABLE I
Characteristics of IoT

Resource Constraints	**Physical Security**	**Dynamic Nature**
Battery Storage Computing Capacity Channel Bandwidth	Easy to capyure	Due to movements Due to instability of wireless links

All these characteristics of IoT make traditional data sharing methods impractical.

C. Challenges Faced

Creating a secure data-sharing solution for resource-constrained IoT devices requires overcoming numerous challenges, as follows:

- Minimizing computational costs due to IoT devices limited processing capability.
- Reducing the reliance on continual online connections for crucial management.
- Enabling flexible delegation and sharing of encrypted data while maintaining confidentiality.
- Ensuring safe data rerouting when nodes are down or in sleep mode.

D. Contributions

To address these challenges, we proposed the following solution

- A novel cross-domain encryption transformation protocol that converts Identity-Based Encryption (IBE) ciphertexts into Least Common Multiple-Based Broadcast Encryption (LBBE) ciphertexts.
- Reducing the number of bilinear pairings used with symmetric cryptographic primitives in LBBE to significantly reduce computational overhead.
- Dynamic Fog-IoT integration, allowing rerouting and secure data access through authorized neighboring nodes when a Fog node enters sleep mode.

- Secure translation key generation to enable routers to convert ciphertexts without breaching confidentiality.
- Enhancement of system reliability and resource optimization in a Fog-to-Thing (Fog-T) environment.

As you progress in this, Section 2 presents a comprehensive related work, analyzing existing solutions and identifying their limitations. Section 3 introduces the foundational concepts required to understand the proposed protocol. Section 4 provides an in-depth description of the proposed protocol, including its system architecture and operational algorithms. Sections 5 and 6 are dedicated to the security and performance analyses, respectively. Section 7 concludes the report by summarizing the key contributions and proposing potential directions for future research.

II. Related Work

In this section, we have analyzed the benefits and limitation of existing work. Like Identity-based cryptography and encryption transformation schemes have become pivotal to addressing secure and flexible data sharing in dynamic cloud, IoT, and vehicular communication scenarios. The research literature shows diverse approaches, primarily focusing on proxy re-encryption, hybrid cryptographic techniques, and Identity-based Broadcast Encryption (IBBE).

Blaze et al. [1] introduced one of the foundational works in proxy re-encryption (PRE), establishing protocols enabling ciphertexts encrypted under one public key infrastructure (PKI) to be securely transformed into ciphertexts decryptable by users within an IBE system. Their method significantly simplified the delegation of decryption rights but inherently involved PKI, thus retaining certificate management complexities.

Green and Ateniese [2] proposed a hybrid proxy re-encryption scheme explicitly bridging attribute-based encryption (ABE) and IBE. This scheme provided a robust method to incorporate expressive encryption policies into the identity-based cryptographic model. Although this method enhanced flexibility and policy expressiveness, the computational complexity involved with ABE components limited its practical efficiency, especially in constrained environments.

In a related yet distinct effort, Ateniese et al. [3] presented a fully secure unidirectional Identity-based Proxy Re-Encryption (IB-PRE) scheme. This scheme offered robust master secret security, providing the original encryptor complete control over ciphertext re-encryption. Although secure, its application scope was restricted to single-hop transformations, thus limiting flexibility and broader applicability.

Deng et al. [4] advanced Identity-based Encryption Transformation (IBET) models by integrating IBE with IBBE. This novel integration enabled ciphertext transformations from a format initially accessible to a single recipient into one decryptable by multiple recipients, facilitating flexible and efficient data sharing in cloud environments. However, the intensive computational overhead due to pairing-based cryptographic operations was noted as a significant constraint.

Further exploring revocation management, Ge et al. [5] introduced a revocable IB-PRE mechanism explicitly addressing dynamic revocation scenarios. Their work provided practical solutions to effectively handle revocation in broadcast re-encryption contexts, significantly enhancing flexibility in user management within cloud environments.

Each of these works contributes uniquely to the broader landscape of identity-based cryptographic solutions, offering varied strategies and trade-offs. Blaze et al. [1] and Green and Ateniese [2] foundationally defined the scope of proxy re-encryption schemes but introduced complexities related to PKI and ABE integrations. Ateniese et al. [3] focused on achieving unidirectional and secure single-hop re-encryption transformations. In contrast, Deng et al. [4] generalized this concept to support multi-recipient transformations, though at increased computational cost.

The comparative analysis of these related works is presented in Table II.

TABLE II
Some of the Identity-Based Encryption Transformation Techniques (IBET)

Topic	Focus	Encryption Transformation Domain
Proxy Re-encryption Systems for IBE [1]	Cross-domain ciphertext transformation between PKI and IBE	PKI → IBE
Hybrid Proxy Re-encryption for ABE [2]	Bridging ABE schemes with IBE schemes via novel proxy re-encryption	ABE → IBE
Fully Secure Unidirectional IBE Transformation [3]	Secure single-hop IB-PRE	IBE → IBE
IBET for Flexible Data Sharing [4]	Transforming IBE ciphertexts to IBBE ciphertexts for large-scale data sharing	IBE → IBBE
Revocable IB-PRE [5]	Addressing key revocation issues in broadcast re-encryption for cloud sharing	IBE → IBBE

In summary, earlier research primarily focused on developing secure identity-based ciphertext transformation mechanisms, subsequently evolving towards performance optimization specifically targeting resource-constrained environments and dynamic revocation scenarios. While foundational techniques such as IBET [4] offered strong security guarantees, their computational overhead significantly motivated the exploration of more efficient approaches, such as hybrid and symmetric encryption techniques, notably transforming ciphertexts from IBE into Least Common Multiple (LCM)-based Broadcast Encryption (LBBE). Our research builds upon these foundational insights, proposing a comprehensive and optimized IBET framework that effectively addresses challenges related to computational efficiency and dynamic adaptability, with a particular emphasis on scalable key revocation and ciphertext transformation capabilities in dynamic operational contexts

III. System Model Architecture

The proposed system architecture for IBE to LCM-based broadcast encryption (LBBE) consists of five key entities: the Key Generation Center (KGC), Sender, Receiver, Proxy Server, and Router. This architecture is designed to support secure, efficient, and flexible ciphertext transformation and delivery in Fog-to-Thing (Fog-T) environments.

Initially, the KGC executes the *Setup* algorithm to generate public parameters and a master secret key. It is also responsible for issuing identity-based private keys to all users in the network. The *Sender* encrypts the data using IBE, targeting the identity of a specific fog node (Receiver). This ciphertext is directly decryptable only by the intended Receiver.

Before entering sleep or low-power mode, the *Receiver* generates a *Translation Key* using its private key and identity information of a set of delegate nodes. This translation key is securely transferred to a trusted *Proxy Server* or *Router*.

Upon receiving this key, the Proxy Server performs an *Encryption Transformation* to convert the original IBE ciphertext into a Least Common Multiple (LCM)-based Broadcast Encryption (LBBE) ciphertext. This transformed ciphertext can now be decrypted by any pre-authorized neighboring fog nodes that share a common broadcast key structure.

This model ensures secure and dynamic rerouting of data without requiring re-encryption by the sender. It supports scalability, minimizes computational overhead, and enhances communication reliability in decentralized and resource-constrained IoT scenarios.

IV. Preliminaries

This section, contains the brief description of Bilinear Pairing, IBE, IBBE, IBBET, Computational assumption used in this reseach paper. These are the fundamental things we should know before moving to our proposed solution.

A. Bilinear Pairing

Assume $(G_1, +)$ and $(G_T, *)$ are cyclic groups with prime order n, where $P \in G_1$ is the generator of Group G_1 [6]. It can be achieved either through the use of Weil pairing [7] or Tate pairing [8]. Let $e : G_1 \times G_1 \rightarrow G_T$ constitute a pairing that must fulfill the condition of the bilinear property below:

- Bilinearity $e(P + Q, a) = e(P, a)e(Q, a)$
- Non-Degeneracy: $P \in G_1$ and $Q \in G_1$ then pairing $e(P, Q) \neq 1$
- Function $e(P, Q)$ should be computational in polynomial time. it should be easy function.

Pairing $e : G_1 \times G_1 \rightarrow G_T$ is called Symmetric and $e : G_1 \times G_2 \rightarrow G_T$ is called asymmetric.
We now discuss several encryption techniques that leverage bilinear pairing as their foundational cryptographic primitive.

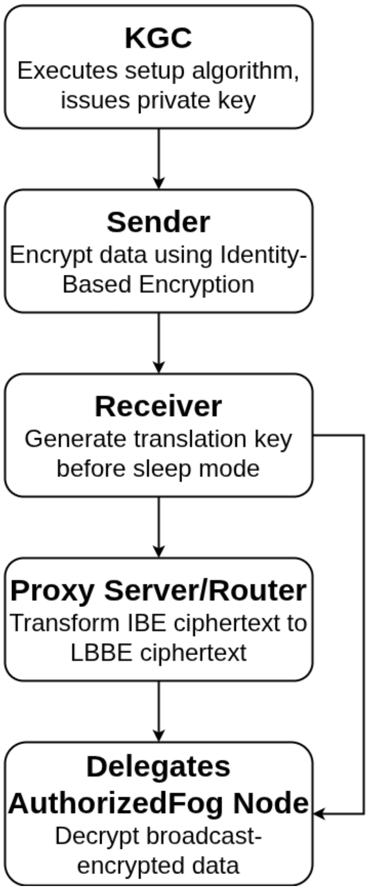
KGC
Executes setup algorithm, issues private key
Sender
Encrypt data using Identity-Based Encryption
Receiver
Generate translation key before sleep mode
Proxy Server/Router
Transform IBE ciphertext to LBBE ciphertext
Delegates AuthorizedFog Node
Decrypt broadcast-encrypted data

Fig. 1. Interaction of the LBBE Protocol

B. Identity-Based Encryption (IBE)

It is a fundamental primitive within the broader domain of identity-based cryptography. It is a form of public-key encryption wherein the public key of a user is derived directly from a unique identifier associated with that user, such as an email address, username, or other distinctive textual information. This characteristic enables a sender, equipped only with the system's public parameters, to encrypt a message using the recipient's identity string as the public key. Consequently, there is no need for pre-distributed certificates or a traditional public key infrastructure (PKI).

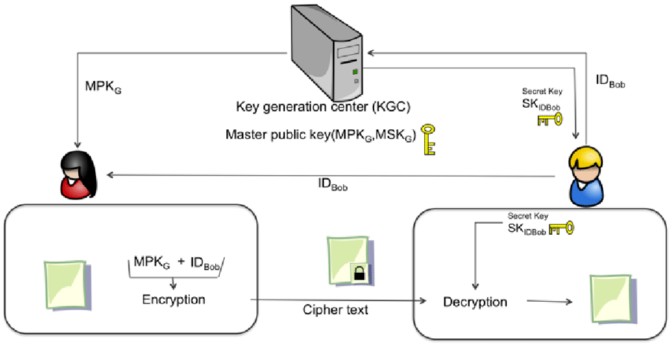

Fig. 2. IBE Environment

An IBE scheme involves three core entities: the Private Key Generator (PKG), which initializes the system and issues private keys; Senders, who encrypt data using receiver identities; and Receivers, who decrypt messages with keys issued by the PKG. This removes the need for certificates and enables identity-driven encryption.

IBE operates through four fundamental algorithms: *Setup*, where the PKG generates public parameters and the master secret; *Key Generation*, in which user-specific private keys are derived; *Encryption*, where senders encrypt messages using the receiver's identity; and *Decryption*, allowing receivers to recover the plaintext using their private keys.

The ability of IBE to eliminate the need for digital certificates and pre-shared public keys makes it particularly attractive for dynamic and decentralized environments, such as cloud computing, IoT networks, and mobile communication systems.

C. Identity Based Broadcast Encryption (IBBE)

Broadcast encryption schemes enable senders to efficiently broadcast ciphertexts to a large set of receivers in a way that only non-revoked receivers can decrypt them. IBE schemes are public key encryption schemes that can use arbitrary strings as public keys.

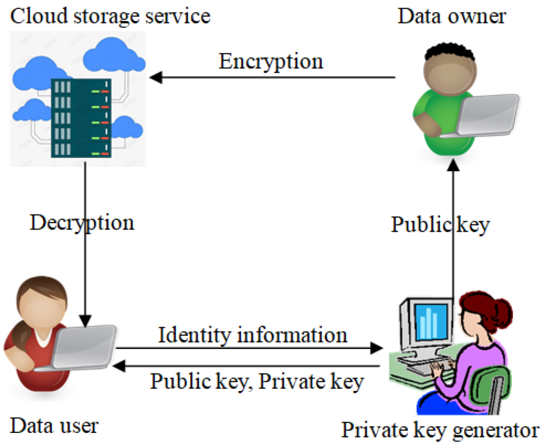

Fig. 3. IBBE

An IBBE system involves three main roles: the Key Generator Centre (KGC), which initializes the system and issues private keys; Senders, who encrypt data for a set of users based on their identities; and Receivers, who decrypt the ciphertext using keys from the KGC. This structure enables secure communication without relying on traditional certificate-based infrastructures.

IBBE operates through four core algorithms: Setup, where the PKG generates system parameters and a master secret; Key Generation, where identity-specific private keys are issued; Encryption, where the sender encrypts messages for a designated group; and Decryption, where authorized users recover the plaintext. These steps support efficient and scalable broadcast encryption based on user identities.

D. Identity Based Broadcast Encryption Transformation (IBBET)

A significant challenge arises when encrypted data must be accessed by users beyond those originally specified by the data owner. To overcome this limitation, we propose an IBET [4] framework that effectively combines two established cryptographic techniques: IBE and IBBE. In this framework, users are authenticated and granted access based on their identity information, thereby eliminating the need for complex certificate-based infrastructure typically required in conventional distributed systems. A key advantage of IBET lies in its ability to transform an IBE-encrypted ciphertext into an IBBE-encrypted one, enabling newly authorized users — not known at the time of the original encryption — to decrypt the data. We construct a concrete IBET scheme using bilinear pairing groups and rigorously prove its security against various attack models. Both theoretical evaluations and experimental results demonstrate that the proposed scheme achieves strong efficiency and practical viability.

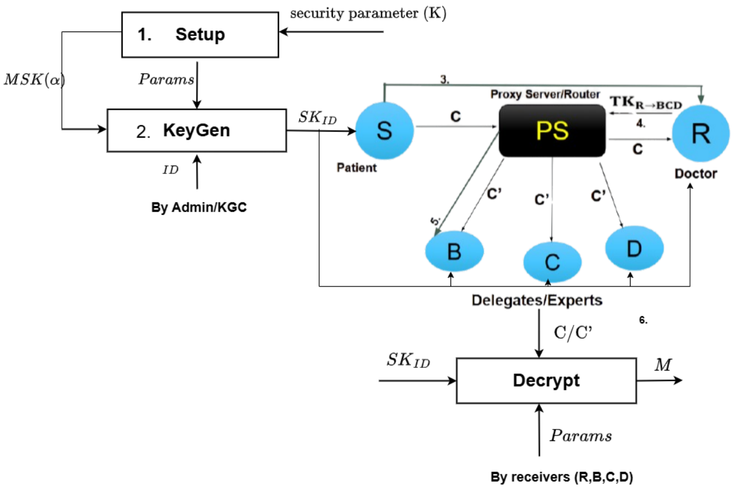

Fig. 4. IBET

Notation	Description
MSK (α)	Master Secret Key
ID_i	Public Key (Identity), where $i \in m$ m is the set of users for whom the data is to be encrypted.
M	Data to be encrypted in G_1
C	Ciphertext, i.e., data encrypted by the sender$S$.
C'	Ciphertext, i.e., data transformed by the proxy server.
SK$_{ID_i}$	Private key of the respective user (identity).
Params/PP	Public Parameters

TABLE III
NOTATION TABLE FOR IBET

The IBET framework involves five main roles: the *Key Generation Centre (KGC)*, a trusted authority that initializes the system and issues private keys; the *Sender (S)*, who encrypts data using the identity of the intended recipient; the *Receiver (R)*, who decrypts the ciphertext; the *Delegate (B, C, D)*, secondary users authorized to access transformed ciphertexts; and the *Proxy Server (PS)*, which securely transforms ciphertexts without learning the underlying plaintext.

The protocol comprises six key algorithms. *Setup* is executed by the KGC to generate public parameters and the master secret. *Key Generation* allows the KGC to issue a private key tied to a user's identity. *Encryption* enables the sender to produce ciphertext using the recipient's identity. *Translation Key Generation* is performed by the receiver to create transformation material. *Encryption Transformation*, carried out by the proxy server, converts ciphertexts for authorized delegates. Finally, *Decryption* is executed by either the receiver or delegate to retrieve the original message.

V. Proposed Work

The proposed protocol introduces a transformation mechanism tailored for secure and efficient data dissemination in Fog-to-Thing (Fog-T) architectures. Key features of the protocol are outlined below:

- The system enables the transformation of ciphertexts from IBE to Least Common Multiple (LCM)-based Broadcast Encryption (LBBE), supporting secure multi-recipient communication.
- It mitigates the computational overhead associated with bilinear pairing operations inherent in IBE/IBBE by leveraging the efficiency of symmetric-key cryptography in LBBE.
- The protocol is optimized for Fog-T environments, ensuring system reliability through dynamic task delegation among neighboring fog nodes.
- When a fog node enters a dormant state (e.g., sleep mode), a router within the system is responsible for redirecting incoming communications to other nearby fog nodes.
- Data is initially encrypted under the identity of the intended fog node using IBE, ensuring identity-based confidentiality.
- Prior to entering sleep mode, the fog node pre-generates and shares translation key material with the router to enable secure ciphertext transformation.
- The router applies the received translation key to convert IBE-encrypted ciphertexts into LBBE format, facilitating broadcast decryption.
- Only the authorized neighboring fog nodes—pre-designated by the original fog node—can decrypt the transformed LBBE ciphertexts.

TABLE IV
NOTATION AND DESCRIPTION

Symbol	Description
$MSK = \alpha$	Master Secret Key generated during setup
PP	Public Parameters including system generators and hash functions
ID	Identity of a user (used as public key)
SK_{ID}	Private key of user with identity ID
M	Original plaintext message to be encrypted
k	Random symmetric session key
s	Random secret value used during encryption
$C = \langle C_0, C_1, C_2 \rangle$	Original IBE Ciphertext
F'	AES-encrypted version of file M using key k
$NS = \{ID_1, \ldots, ID_m\}$	Delegate set authorized to access transformed data
C_i	MAC tag generated using shared key K_{ui} and $Info_i$
$l = LCM(C_i)$	Least Common Multiple of MAC tags C_i for all delegates
r	Random value used in transformation key generation
K_B	Random broadcast key used in transformation
$D_1 = l \cdot r + K_B$	First component of translation key
$D_2 = H_1(K_B \| \| ID_i \| \| F_{id}) \cdot h^r$	Second component of translation key
$D_3 = SK_{ID} \cdot u^{-r}$	Third component of translation key
$C' = \langle C_2, D_1, D_2, D_4 \rangle$	Transformed LBBE ciphertext
D_4	Encrypted symmetric key k computed during transformation
$Flag$	Indicator for whether ciphertext is original (0) or transformed (1)

The protocol is structured around six core algorithms, each performing a distinct cryptographic operation:

Algorithm 1 Setup Phase (Executed by Admin/KGC) Initializes system parameters and generates the master secret.

```
Input: security parameter (K)
Generate bilinear groups $G, G_T$ and pairing $\hat{e} : G \times G \to G_T$
Choose random $\alpha \in \mathbb{Z}_p^*(MSK), g, h, u \in G$
Compute $g_1 = g^\alpha,\ v = \hat{e}(g, h)$
Derive: $u^\alpha,\ h^\alpha,\ h^{\alpha^2}, ..., h^{\alpha^m}$
Set $PP = \langle g_1, u, u^\alpha, h, h^\alpha, ..., h^{\alpha^m}, v, H_0, H_1 \rangle$
Output: Public Parameters $PP$, Master Secret Key $MSK = \alpha$
```

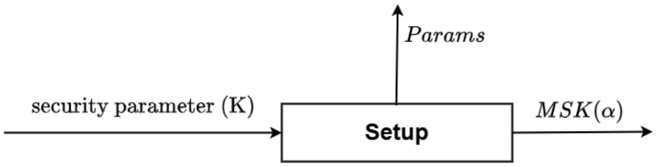

Fig. 5. Setup

Algorithm 2 Key Generation Phase (Executed by KGC) Associates a user's identity with a private key issued by the trusted authority.

```
Input: MSK , ID , PP
Compute Private Key $SK_{ID} \leftarrow g^{1/[\alpha+H_0(ID)]}$
Output: Private Key $SK_{ID}$
```

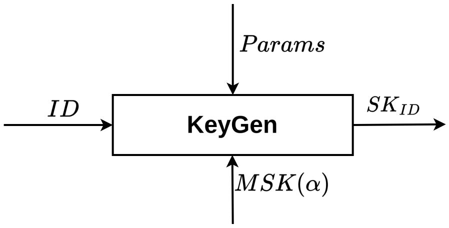

Fig. 6. Key Generation Phase

Algorithm 3 Encryption Phase (Executed by Sender)

```
Input: M , ID , PP
Randomly choose $k \in G_T$, $s \in \mathbb{Z}_p^*$
Compute:
    $C_0 = k \cdot v^s$
    $C_1 = (h^\alpha \cdot h^{H_0(ID)})^s$
    $C_2 = (u^\alpha \cdot u^{H_0(ID)})^s$
Encrypt $M$ using AES with key $k$: $F' = AES(M, k)$
Set flag = 0
Output: Ciphertext $C = \langle C_0, C_1, C_2 \rangle$, Encrypted File $F'$, Flag = 0
```

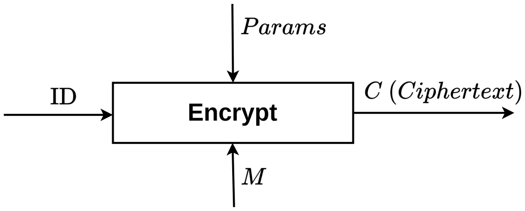

Fig. 7. Encryption Phase

Algorithm 4 Authorization Phase (Executed by Data Owner/Original receiver) Generates translation key to permit secure transformation of ciphertexts.

```
Input: Receiver's identity ID, Delegate set $NS = \{ID_1, \ldots, ID_m\}$, Private key $SK_{ID}$, Public Parameters $PP$
for each delegate $ID_i \in NS$ do
    Compute authentication tag $C_i = MAC(Info_i, K_{ui})$
end for
Compute $l = LCM(C_i)$
Randomly choose $r \in \mathbb{Z}_p^*$, $K_B \leq C_i \ \forall i$
Compute:
    $D_1 = l \cdot r + K_B$
    $D_2 = H_1(K_B || ID_i || F_{id}) \cdot h^r$
    $D_3 = SK_{ID} \cdot u^{-r}$
Output: Translation Key = $TK_{ID \to \mathrm{NS}} = \langle D_1,\ D_2,\ D_3 \rangle$
```

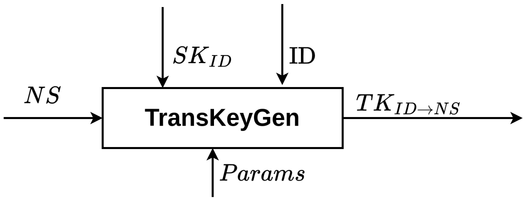

Fig. 8. Translation Key Generation Phase

Algorithm 5 Translation Phase (Executed by Proxy Server)

```
Input: Ciphertext $C = \langle C_0, C_1, C_2 \rangle$, Encrypted file F', Translation Key $TK$
Compute transformed key $D_4 \leftarrow C_0 \ / \ \hat{e}(C_1,\ D_3)$
Set Flag = 1 (means File get transformed i.e. C' )
Output: Transformed Ciphertext $C' = \langle C_2, D_1, D_2, D_4 \rangle$, Flag = 1
```

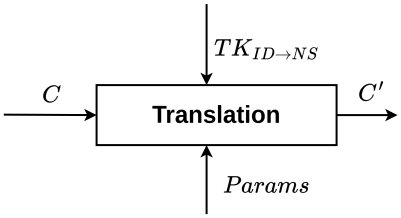

Fig. 9. Translation Phase

Algorithm 6 Decryption Phase (Executed by Delegates/Original receiver)

Input: Private key SK_{ID}, Ciphertext C or C', Encrypted File F', Flag

```
if Flag = 0  (Original Receiver) then
    Compute $k = C_0 \ / \ \hat{e}(SK_{ID},\ C_1)$
    Decrypt message: $M = AES^{-1}(F', k)$
else if Flag = 1  (Delegate) then
    Extract $K_B = D_1 \bmod C_i$
    Compute $h^r = D_2 \ / \ H_1(K_B||ID_i||F_{id})$
    Recover $k = D_4 \ / \ \hat{e}(h^r,\ C_2)$
    Decrypt message: $M = AES^{-1}(F', k)$
end if
```

Output: Original Message M/F

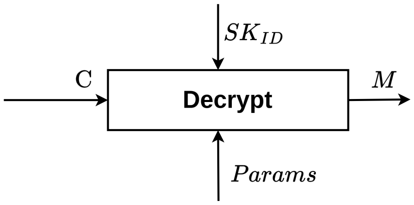

Fig. 10. Decryption by Original Receiver

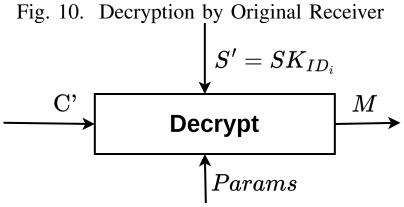

Fig. 11. Decryption by Delegates

VI. Security Analysis

In this section, we analyze the correctness and informal security guarantees provided by the proposed Identity-Based LCM-Based Broadcast Encryption (LBBE) protocol. The protocol relies on standard hardness assumptions from bilinear pairing-based cryptography and ensures confidentiality during transformation and broadcast phases.

A. Correctness Proof

To ensure successful decryption, the correct private key must be used by either the original receiver or a valid delegate. We prove that in both cases, the symmetric key k used for AES encryption can be correctly recovered.

Case A: Decryption of Original Ciphertext

Given ciphertext components C_0, C_1 and private key SK_{ID}, the receiver computes:

$$k = \frac{C_0}{\hat{e}(SK_{ID}, C_1)} = \frac{k \cdot \hat{e}(g,h)^s}{\hat{e}(g^{1/(\alpha+H_0(ID))}, h^{s(\alpha+H_0(ID))})}$$

$$= \frac{k \cdot \hat{e}(g,h)^s}{\hat{e}(g,h)^s} = k$$

Thus, the original message is recovered as $M = AES^{-1}(F', k)$.

Case B: Decryption of Transformed Ciphertext

Let $D_1 = l \cdot r + K_B$, and $C_i = MAC(Info_i, K_{ui})$. Since $K_B < C_i$ and $l \mod C_i = 0$, we get:

$$K_B = D_1 \mod C_i$$

To derive h^r, the delegate computes:

$$h^r = \frac{D_2}{H_1(K_B||ID_i||F_{ID})}$$

Using $D_4 = \frac{C_0}{\hat{e}(C_1, D_3)}$, and knowing:

$$D_3 = SK_{ID} \cdot u^{-r}, \quad C_1 = (h^{\alpha} \cdot h^{H_0(ID)})^s$$

$$\Rightarrow D_4 = k \cdot \hat{e}(g,h)^s / \hat{e}(h^{s(\alpha+H_0(ID))}, SK_{ID} \cdot u^{-r})$$

Simplifying:

$$D_4 = k \cdot \hat{e}(h^{s(\alpha+H_0(ID))}, u^r) = k \cdot \hat{e}(h,u)^{rs(\alpha+H_0(ID))}$$

The delegate computes:

$$k = \frac{D_4}{\hat{e}(h^r, C_2)} = \frac{k \cdot \hat{e}(h,u)^{rs(\alpha+H_0(ID))}}{\hat{e}(h^r, u^{s(\alpha+H_0(ID))})} = k$$

Hence, decryption of transformed ciphertext also yields the original message.

B. Informal Soundness Proof

We now analyze potential attack vectors and how the protocol defends against them.

TABLE V
Informal Soundness Analysis

Adversary	**Attack Type**	**Defense in Proposed Protocol**
Proxy Server	Illegal Transformation	Cannot generate translation keys; only authorized receiver issues them
Proxy Server	Confidentiality Breach	Cannot decrypt ciphertext; only performs ciphertext conversion using transformation key
Delegate (unauthorized)	Confidentiality Breach	Only delegates with valid MAC-authenticated broadcast keys can derive symmetric key k

The security of the protocol builds on the difficulty of the Bilinear Diffie-Hellman Problem (BDHP) and ensures that no unauthorized entity can decrypt or misuse transformed ciphertext without possessing a valid secret key. The inclusion of file identifiers and LCM-based MAC construction further prevents man-in-the-middle and replay attacks.

VII. Performance Analysis

In this section, we briefly discuss the performance benefits offered by our proposed Identity-Based LCM-Based Broadcast Encryption (LBBE) protocol, emphasizing both computational efficiency and security features.

Firstly, our proposed scheme significantly reduces computational overhead by minimizing the use of expensive bilinear pairing operations, which directly results in lower computational cost and improved efficiency. Furthermore, the integration of symmetric key cryptography within the LCM-based broadcast mechanism contributes to performance enhancement, leveraging efficient symmetric operations.

Additionally, our protocol employs unique identifiers such as file IDs, effectively preventing unauthorized file decryption and enhancing data confidentiality. Finally, the inclusion of Least Common Multiple (LCM)-based authentication ensures robust neighbor verification, thereby mitigating common vulnerabilities such as Man-in-the-Middle attacks. Collectively, these attributes demonstrate the protocol's ability to balance strong security guarantees with efficient performance suitable for deployment in resource-constrained environments such as IoT.

VIII. Conclusion

In this paper, we proposed an Identity-Based LCM-Based Broadcast Encryption (LBBE) protocol, effectively addressing critical issues of ciphertext transformation, computational efficiency, and dynamic revocation within decentralized and resource-constrained IoT environments. Our approach integrates the strengths of identity-based encryption with symmetric cryptography and Least Common Multiple (LCM)-based broadcast encryption, significantly reducing the reliance on costly bilinear pairing operations.

We rigorously analyzed the security of the proposed protocol, illustrating its resilience against common threats, including unauthorized access and Man-in-the-Middle attacks. Performance evaluations indicate our scheme offers substantial computational savings and robust security assurances, making it particularly suitable for practical deployments in IoT and Fog-to-Thing architectures.

References

[1] Matt Blaze, Gerrit Bleumer, and Martin Strauss. "Divertible Protocols and Atomic Proxy Cryptography". In: *Advances in Cryptology — EUROCRYPT'98*. Springer, 1998, pp. 127–144. DOI: 10.1007/BFb0054122.

[2] Matthew Green and Giuseppe Ateniese. "Identity-based Proxy Re-encryption". In: *Applied Cryptography and Network Security*. Springer, 2007, pp. 288–306. DOI: 10.1007/978-3-540-72738-5_19.

[3] Giuseppe Ateniese et al. "Improved proxy re-encryption schemes with applications to secure distributed storage". In: *ACM Transactions on Information and System Security (TISSEC)* 9.1 (2006), pp. 1–30. DOI: 10.1145/1127345.1127346.

[4] Hua Deng et al. "Identity-based Encryption Transformation for Flexible Sharing of Encrypted Data in Public Cloud". In: *IEEE Transactions on Information Forensics and Security* 15 (2020), pp. 3168–3180. DOI: 10.1109/TIFS.2020.2985532.

[5] Chunpeng Ge et al. "Revocable Identity-Based Broadcast Proxy Re-Encryption for Data Sharing in Clouds". In: *IEEE Transactions on Dependable and Secure Computing* 18.3 (2021), pp. 1214–1226. DOI: 10.1109/TDSC.2019.2923226.

[6] Dan Boneh, Ben Lynn, and Hovav Shacham. "Short signatures from the Weil pairing". In: *International conference on the theory and application of cryptology and information security*. Springer. 2001, pp. 514–532.

[7] Dan Boneh and Matt Franklin. "Identity-based encryption from the Weil pairing". In: *Annual international cryptology conference*. Springer. 2001, pp. 213–229.

[8] Steven D Galbraith, Keith Harrison, and David Soldera. "Implementing the Tate pairing". In: *International Algorithmic Number Theory Symposium*. Springer. 2002, pp. 324–337.