An Anonymous Multi-receiver Certificateless Hybrid Signcryption (AMCLHS) using mKEM-DEM for Broadcast Communication

No Institute Given

Abstract. Confidentiality, authentication, and anonymity are the fundamental security requirements in broadcast communication that can be achieved by Digital Signature (DS), encryption, and pseudo-anonymous identity techniques. Signcryption offer both DS and encryption in a single logical step with high efficiency. Similarly, anonymous multireceiver signcryption ensure receiver privacy by generating identical ciphertext for multiple receivers while keeping their identities private. While signcryption is a significant improvement over "sign then encrypt", it still incurs higher computational and communication cost and does not provide the required level of security.

In this paper, we propose a multiple-recipient Key Encapsulation Mechanism (mKEM) - Data Encapsulation Mechanism (DEM) based Anonymous Multireceiver Certificateless Hybrid Signcryption (AMCLHS). The AMCLHS uses a combination of symmetric key and asymmetric key cryptography to signcrypt an arbitrary length message in broadcast communication and has two unique settings as follows:

1. Pseudo-Identity (PID) Settings: We introduce a new algorithmic step in AMCLHS construction where each user (sender and receiver) is assigned a PID to enable the sender to signcrypt identical messages for multiple receivers while keeping the identities of other receivers anonymous. The receiver anonymity is achieved by choosing random Real-Identity (ID_R) to generate PID of the users in key generation algorithm of AMCLHS scheme. Our approach relies on the Elliptic Curve Discrete Logarithm (ECDL) hardness assumptions, the hash function, and verification-based secret key of the Register Authority (RA), using time ΔT.
2. mKEM-DEM Settings: We introduce the first construction that achieves optimal ciphertext from the Diffie-Hellman (DH) assumption using mKEM-DEM for Signcryption. Our scheme uses mKEM to generate a symmetric key for multiple-receivers and DEM to signcrypt message using the previously generated symmetric key and the sender's private key. Our scheme relies on DH and Bilinear Pairing (BP) assumption and uses a single key for all messages, which minimizes ciphertext length and ultimately reduces complexity overhead.

The scheme operates in a multireceiver certificateless environment, preventing the key escrow problem, and demonstrates cryptographic notions for Indistinguishability under Chosen-Ciphertext Attack (IND-CCA2) and Existential Unforgeability against Chosen Message Attack (EUF-CMA) for Type-I and Type-II adversaries under q-Decisional Bilinear Diffie-Hellman Inversion (q-DBDHI) and ECDL hard assumptions. We compare the proposed scheme with existing multireceiver hybrid signcryption schemes in terms of computation cost, communication cost, and security requirements. We show that, compared to existing

multireceiver schemes which has overall cost of $\mathcal{O}(n^2)$, our scheme is computationally more efficient and has optimal communication cost, with signcryption cost linear $\mathcal{O}(n)$ to the number of designated receivers while the unsigncryption cost remains constant $\mathcal{O}(1)$. Our scheme achieves confidentiality, authentication, anonymity, and simultaneously achieves unlinkability, non-repudiation, and forward security.

Keywords: mKEM-DEM · Hybrid Signcryption · Certificateless · Multireceiver · Pseudo-Identity · Confidentiality · Authentication · Anonymity.

1 Introduction

Confidentiality, authentication, and anonymity are the basic security requirements in a broadcast communication scenario [17,8]. The current solution to provide these security requirements is by encryption and Digital Siganture (DS). However, the traditional "sign-then-encrypt" approach leads to high computational costs. Signcryption, on the other hand, allows the encryption and signature operations to be performed simultaneously to provide both the confidentiality and authentication more efficiently. Signcryption was first proposed by Zhang et al. [27] as a novel cryptographic primitive and has been widely used in real-world applications such as e-commerce, smart cards, and mobile ad-hoc communication [28].

Since then, several traditional Public Key Cryptosystem (PKC) and Identity (ID)-based signcryption methods have been proposed. Malone-Lee [14] proposed the first ID-based signcryption scheme that provides both forward security and public verifiability. However, PKC relying on Public Key Infrastructure (PKI) requires a trusted Certification Authority (CA) to distribute public key certificates increasing the cost of certificate management, storage, and revocation. Additionally, in ID-based schemes, the Public Key Generator generates the user's private key, leading to the issue of private key escrow. To solve the key escrow problem, Al-Riyami et al. [1] proposed a Certificateless Public Key Cryptography (CLPKC) that do not require the use of certificates and does not have a key escrow problem. In CLPKC, the Key Generation Center (KGC) generates a partial private key of the user taking user's ID as input. The user then combines partial private key and a secret value to generate the actual private and public key pair. More specifically, the key escrow problem is prevented as the KGC does not have knowledge of the complete private key of the user. Following that, Barbosa and Farshim [2] proposed the first certificateless signcryption scheme that provides confidentiality and unforgeability and is secure under the Random Oracle Model (ROM).

The signcryption methods mentioned above are designed for single receiver use, which is not suitable for broadcast communication. When sending the same message to multiple recipients, the user has to encrypt a message for each individual recipient, causing an increase in computation time and communication lag. To address this, Yu et al. [25] proposed an ID-based multireceiver signcryption scheme that can encrypt a message for n designated recipients. The security of this scheme has been proven in a ROM. Later on, several ID-based signcryption schemes were proposed however, since ID-based PKC has an inherent key escrow problem, Selvi et al. [20] proposed the first multireceiver certificateless signcryption scheme and proven secure in ROM. Generally,

the construction of signcryption can be achieved through two methods: (i) Public key signcryption: With public key signcryption, both message encryption and signing take place in a public key setting [20]. However, for arbitrary length messages, public key signcryption alone can be computationally intensive and will not be feasible for large and resource-constrained environment. (ii) Hybrid signcryption: Hybrid signcryption enables a message to be signed in a public key setting and then encrypted using symmetric key, with the symmetric key being encrypted using a public key [19]. Hybrid signcryption is generally more efficient than public key signcryption alone because hybrid signcryption uses a combination of symmetric key and public key where a message is encrypted using a symmetric key, which is faster and more efficient. Typically, hybrid signcryption involves two phases: First, a KEM generates a symmetric key using the public key, second a DEM encrypts an arbitrary length message using the symmetric key generated by the KEM [6] followed by a signature with the private key of the sender.

In this paper, we propose an anonymous certificateless hybrid signcryption based on multiple-recipient Key Encapsulation Mechanism- Data Encapsulation Mechanism mKEM-DEM for broadcast communication. For confidentiality, we prove Indistinguishability under Chosen-Ciphertext Attack (IND-CCA2-I) for Type-I adversary and (IND-CCA2-II) for Type-II adversary based on q-Decisional Bilinear Diffie-Hellman Inversion (q-DBDHI) hard assumption. For unforgeability, we prove Existential Unforgeability against Chosen Message Attack (EUF-CMA-I) for Type-I adversary and (EUF-CMA-II) for Type-II adversary, respectively, based on Elliptic Curve Discrete Logarithm (ECDL) hard assumption. Moreover, to ensure anonymity, each user is assigned a Pseudo-Identity (PID). We additionally demonstrate the security for unlinkability, non-repudiation, and forward security. Finally, we compare our scheme with existing multireceiver certificateless hybrid signcryption schemes in terms of computation cost, communication cost, and security requirements. As compared to existing schemes listed at the end of the paper, our scheme is more efficient, with the signcryption cost linear with the number of designated receivers while the unsigncryption cost remains constant. Our scheme simultaneously fulfills all the security requirements in terms of confidentiality, unforgeability, anonymity, unlinkability, non-repudiation, and forward security. The main contributions are as follows:

1. We propose a mKEM-DEM based Anonymous Multireceiver Certificateless Hybrid Signcryption (AMCLHS) scheme for broadcast communication.
2. We achieve confidentiality by demonstrating security against IND-CCA2 Type-I and Type-II adversaries and unforgeability by demonstrating security against EUF-CMA Type-I and Type-II adversaries, respectively. The security is demonstrated using q-DBDHI and ECDL hard assumptions under the ROM.
3. The AMCLHS scheme achieves anonymity for each receiver by assigning a PID. Each receiver can unsigncrypt a message sent by a sender while the identities of each receiver remains anonymous from each other.
4. We evaluate our scheme and provide a comparison with other existing schemes and show that our scheme simultaneously achieves, unlinkability non-repudiation, and forward security with lower computation and communication cost.

The remainder of the paper is organized as follows: The related work is provided in Section 1.1. Section 2 describes the preliminaries and mathematical assumptions. In

Section 3, we introduce the framework and security model of the scheme. Section 4 introduces the proposed AMCLHS scheme and in Section 5, we provide the security analysis under the hardness assumption. Section 6 provide the efficiency analysis and comparison of the proposed scheme. Lastly, in Section 7, we conclude the work.

1.1 Related Work

Signcryption was first introduced by Zheng et al. [27] in 1997 that combines the signature and encryption to provide authentication and confidentiality more efficiently than sign-then-encrypt. Several ID-based signcryption schemes have been proposed, however the key issue with ID-based signcryption is the presence of a key escrow problem. To address this, Barbosa and Farshim [2] proposed the first certificateless signcryption scheme that provides both confidentiality and authentication and is secure under the ROM. The scheme relies on the BP assumption and is designed to be secure against UF-CMA and strong-UF-CMA attacks under the Gap-Bilinear Diffie Hellman (G-BDH), Decisional Bilinear Diffie-Hellman (DBDH), and Computational Bilinear Diffie Hellman (CBDH) assumption. Gong et al. [9] presented a lightweight and secure certificateless hybrid signcryption scheme for the Internet of Things (IoT). The scheme demonstrates security against IND-CCA and EUF-CMA under the hardness assumptions of Computational Diffie- Hellman (CDH) and DBDH, to provide confidentiality unforgeability. However, the scheme has a high computational cost for a single receiver and does not provide anonymity.

Similarly, Zhang et al. [26] introduced a certificateless hybrid signcryption scheme suitable for the IoT. The scheme is constructed to achieve both confidentiality and unforgeability under the hardness assumptions of Discrete Logrithmic (DL), CDH, BDH, and DBDH. A certificateless signcryption scheme without ROM was proposed by ZHOU et al. [28] that achieves confidentiality and unforgeability however, does not provide anonymity. Kasyoka et al. [11] and Yin et al. [23] proposed a certificateless signcryption and a certificateless hybrid signcryption scheme, respectively, for wireless sensor networks. Additionally, Hongzhen et al. [10] and Cui et al. [5] presented a pairing-free certificateless signcryption scheme for Vehicular Ad Hoc Networks and a certificateless signcryption scheme for the Internet of Vehicles, respectively. Li et al. [13] proposed a signcryption scheme for resource-constrained smart terminals in cyber-physical power systems. However, all the aforementioned schemes are designed for single receivers, which are not be suitable for broadcast communication. For example, to send an identical message to multiple receivers, the sender must encrypt a message for each recipient, resulting in poor performance.

Yu et al. [25] introduced the first multireceiver signcryption scheme based on ID-based PKC, where a message is encrypted for n designated receivers. The security of the scheme is based on CDH assumption under the ROM. Later on, several multireceiver certificateless signcryption schemes were proposed. In 2017, Niu et al. [15] proposed a heterogeneous hybrid signcryption for multi-message and multi-receiver. The scheme proves security against IND-CCA and EUF-CMA attacks under the ROM based on the hardness assumptions of DBDH and variants of DBDH and CBDH. In 2022, Niu et al. [16] proposed a privacy-preserving mutual heterogeneous signcryption scheme based on 5G network slicing that operates in a hybrid environment, where the sender is

in a PKI environment and the receiver is in a certificateless environment. The proposed scheme is secure against IND-CCA2 and EUF-CMA under the hardness assumptions of CDH and DL. In addition, numerous multireceiver certificateless signcryption schemes have been introduced in edge computing, smart mobile IoT, and IoT-enabled maritime transportation systems [17,18,22,24]. However, public key signcryption can be computationally inefficient and may not be feasible for large and resource-constrained environments when dealing with arbitrary length messages. On the other hand, hybrid signcryption is generally more efficient than public key signcryption alone because it uses the combination of symmetric key and PKC. A message is encrypted using a symmetric key algorithm, which is faster and more efficient. Dent et al. [6,7] proposed the first hybrid signcryption scheme with insider and outsider security. Following that, Li et al. [12] proposed the first certificateless hybrid signcryption scheme.

Our paper presents a mKEM-DEM based anonymous multireceiver certificateless hybrid signcryption scheme for broadcast communication. For confidentiality and unforgeability, the scheme demonstrates security against IND-CCA2 and EUF-CMA Type-I and Type-II adversaries under the ROM using ECDL and q-DBDHI hard assumptions. Furthermore, our scheme achieves anonymity, unlinkability, non-repudiation, and forward security.

2 Preliminaries and Assumptions

1. **Bilinear Pairing (BP)**: Let $\mathbb{G}_1$ be a cyclic additive group of prime order q and $\mathbb{G}_2$ be cyclic multiplicative group of same order. A BP be mapping $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ which satisfies the following properties:
   - *Bilinearity*. For any generator $P, Q \in \mathbb{G}_1, \mathbb{G}_2$, $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$ where $a, b \in \mathbb{Z}_q^*$.
   - *Computability*. For $P, Q \in \mathbb{G}_1$, $\hat{e}(P, Q)$ can be efficiently computed.
   - *Non-degeneracy*. There exists $\hat{e}(P, Q) \neq 1$, for some $P, Q \in \mathbb{G}_1$.

   We adopted BP definition from [4] and readers should refer to the same paper for details about the construction of such group and BPs. Next, we are recalling some hard mathematical assumptions based on BP that will be used in construction of our scheme.
2. **q-Decisional Bilinear Diffie-Hellman Inversion (q-DBDHI) Assumption** : The q-DBDHI first introduced by Boneh and Boyen [3]. The q-BDHI assumption for $(q, \mathbb{G}_1, \mathbb{G}_2, \hat{e})$ is, given the tuple $(P, \alpha P, \alpha^2 P, ..., \alpha^q P) \in \mathbb{G}_1$ where $\alpha \in \mathbb{Z}_q^*$, it is hard to compute $\hat{e}(P, P)^{1/\alpha}$. The q-DBDHI assumption also requires it to be hard to distinguish $\hat{e}(P, P)^{1/\alpha}$ from a random element in $\mathbb{G}_2$.
3. **Elliptic Curve Discrete Logarithm (ECDL) Assumption**: Given P and $Q \in \mathbb{G}_1$, it is hard to find an $x \in \mathbb{Z}_q^*$ for any Probabilistic Polynomial-Time (PPT) algorithm with non-negligible probability such that $Q = xP$.

3 Framework and Security Model

3.1 Framework

The framework of the AMCLHS scheme consists of four entities; KGC, a Registration Authority (RA), and n users such as $n = \{\mathsf{PID_s}, \{\mathsf{PID}_1, ..., \mathsf{PID}_{r_i}, ..., \mathsf{PID}_{r_t}\}\}$. Suppose, a sender with $\mathsf{PID_s}$ sends an arbitrary length message m to t designated receivers denoted with PID_{r_i} where $1 \leq i \leq t$ and $t < n$. The role of each entity is defined below:

- **KGC**: The KGC is responsible for generating public parameters (PP), master secret key (msk) of KGC, master public key (mpk) of KGC, and partial private key (ppk) for each user taking part in communication.
- **RA**: The RA is a semi-trusted authority that first generates its private key $\mathsf{sk_{RA}}$ and public key $\mathsf{pk_{RA}}$. RA is also responsible for user registration, identity verification, and PID generation.
- **Sender**: The sender with identity $\mathsf{PID_s}$ encrypts a m using the set of designated receiver's public key pk_{r_i} signs with its private key $\mathsf{sk_s}$ and sends the signcrypted ciphertext CT to t designated receivers.
- **Receiver**: The designated receiver with PID_{r_i} and sk_{r_i} decrypt the CT, and verify the signature using sender's public key $\mathsf{pk_s}$.

Definition 1. *The multiple-recipient Key Encapsulation Mechanism* (mKEM) *and Data Encapsulation Mechanism* (DEM) *algorithms are described as follows:*

The notion of mKEM was first proposed by N.P Smart [21] and has a KEM like construction which takes multiple public keys as input and generates a symmetric session key. The mKEM construction below is according to [21]:

1. mKEM: It consists of four algorithms (Setup, KeyGen, Encap, Decap) defined as follows:
   - Setup: This algorithm takes the security parameter 1^λ as input and outputs PP.
   - KeyGen: This algorithm takes PP as input and generates $(\mathsf{pk}, \mathsf{sk})$ for each user.
   - mKEM.Encaps: In this algorithm, the sender takes a set of pk_{r_i} as input and outputs a symmetric session key K and an encapsulated C.
   - mKEM.Decaps: The designated receiver takes $(\mathsf{sk}_{r_i}, \mathsf{C})$ as input and outputs a symmetric session key K'. The correctness of mKEM holds if $\mathsf{K}' = \mathsf{K}$.
2. DEM: It consists of two algorithms $(\mathsf{Enc_K}, \mathsf{Dec}_{\mathsf{K}'})$ [15] defined as follows:
   - $\mathsf{Enc_K}$: In this algorithm, the sender takes (K, m) as input and generates a ciphertext c.
   - $\mathsf{Dec}_{\mathsf{K}'}$: The designated receiver takes the $(\mathsf{K}', \mathsf{c})$ as input and outputs m'. The correctness of DEM holds if $m' = m$.

Definition 2. *The sender with* $\mathsf{PID_s}$ *sends an arbitrary length* m *to* t *designated receivers denoted with* PID_{r_i} *where* $1 \leq i \leq t$ *and* $t < n$*. The proposed scheme consists of eight polynomial time algorithms.*

1. Setup: On input security parameter 1^λ as input, the KGC runs this algorithm to generate PP, msk, and mpk. RA generates $\mathsf{sk_{RA}}$ and $\mathsf{pk_{RA}}$.

2. Pseudo-Identity: By taking input as the Real-Identity $\mathsf{ID_R}$ of each user and $\mathsf{pk_{RA}}$, this algorithm generates a PID for each user.
3. Partial private key: KGC takes an input $(\mathsf{msk}, \mathsf{mpk}, \mathsf{PID})$ and runs this algorithm to generate the ppk for each user.
4. Set secret value: On input the PID, each user runs this algorithm to generate a secret value sv.
5. Set private key: Each user takes an input ppk and sv, runs this algorithm to generate the sk.
6. Set public key: On input the sv, each user runs this algorithm to generate the pk.
7. Signcryption: On input $(m, \mathsf{PP}, \mathsf{PID_{r_i}}, \mathsf{pk_{r_i}})$, the sender runs the mKEM.Encaps algorithm to generate a K. Finally, using the K and $\mathsf{sk_s}$, the sender runs $\mathsf{Enc_K}$ to generate CT.
8. Unsigncryption: On input $(\mathsf{PP}, \mathsf{CT}, \mathsf{PID_s}, \mathsf{sk_{r_i}})$, the receiver runs mKEM.Decaps algorithm to compute K'. If $\mathsf{K}' = \mathsf{K}$, the receiver then uses the K', and $\mathsf{pk_s}$ and runs $\mathsf{Dec_{K'}}$ to retrieve m. If unsigncryption holds, the receiver accepts m, else returns $\perp$.

3.2 Security Model

For confidentiality, we define the Indistinguishability of Anonymous Multireceiver Certificateless Hybrid Signcryption against a Chosen Ciphertext Attack (IND-AMCLHS-CCA2). For unforgeability, we define Existential Unforgeability of Anonymous Multireceiver Certificateless Hybrid Signcryption against a Chosen Message Attack (EUF-AMCLHS-CMA). We consider two types of adversaries: (i) Type-I ($\mathcal{A}_{\mathrm{I}}$): $\mathcal{A}_{\mathrm{I}}$ is considered a common user who has no knowledge of msk but can replace the pk of any ID with a value of his/her own choice. (ii) Type-II ($\mathcal{A}_{\mathrm{II}}$): $\mathcal{A}_{\mathrm{II}}$ also known as malicious KGC is considered an insider adversary who has access to the msk but cannot replace the pk of a legitimate user. In Definition 3 (Game-I), we define the IND-AMCLHS-CCA2-I for $\mathcal{A}_{\mathrm{I}}$ and the IND-AMCLHS-CCA2-II for $\mathcal{A}_{\mathrm{II}}$, and in Definition 4 (Game-II), we define the EUF-AMCLHS-CMA-I for $\mathcal{A}_{\mathrm{I}}$ and the EUF-AMCLHS-CMA-II for $\mathcal{A}_{\mathrm{II}}$.
$\mathcal{A}_{\mathrm{I}}$ has following constraints:

1. $\mathcal{A}_{\mathrm{I}}$ cannot access msk.
2. $\mathcal{A}_{\mathrm{I}}$ is not allowed to ask a partial private key query q_{ppk} for any of the target identities.

$\mathcal{A}_{\mathrm{II}}$ has the following constraints:

1. $\mathcal{A}_{\mathrm{II}}$ cannot make public key replace query q_{pr} for the target ID.
2. $\mathcal{A}_{\mathrm{II}}$ is not allowed to make sv extract queries q_{sv}.
3. If the q_{pr} has been done for the target ID, then the q_{sv} is not allowed for the same ID.

Definition 3. *The IND-AMCLHS-CCA2 requires that there exists no PPT Adversary $\mathcal{A}$ which could distinguish ciphertexts. Therefore, the security game that captures confidentiality is based on the ciphertext indistinguishability. The advantage of $\mathcal{A}$ is defined as the probability that $\mathcal{A}$ wins the game.*

Game-I (IND-AMCLHS-CCA2-I, IND-AMCLHS-CCA2-II): This Game is interaction between the Challenger $\mathcal{C}$ and $\mathcal{A}$ as follows:
Phase-1: The $\mathcal{A}$ asks polynomially bounded number of adaptive hash queries q_{H_l} where $\{l = 1, 2, 3\}$. The $\mathcal{C}$ keeps a list L_l of q_{H_l} to record the responses.
Setup: The $\mathcal{C}$ generates $(\mathsf{PP}, \mathsf{msk}, \mathsf{mpk}, \mathsf{sk_{RA}}, \mathsf{pk_{RA}})$ and gives PP to $\mathcal{A}$. Then $\mathcal{A}$ selects t target $\mathsf{PID_{r_i}}$ where $1 \leq \mathsf{i} \leq t$ and $t < n$. In IND-AMCLHS-CCA2 Game, the target is $\mathsf{PID_{r_i}}$.
Phase-2: The $\mathcal{A}$ further asks a number of adaptive queries with the restrictions defined in 3.2. The queries include public key retrieve query q_{pk}, partial private key query q_{ppk}, secret value extract query q_{sv}, public key replace query q_{pr}, signcryption query q_{sc}, and unsigncryption query q_{usc}. The $\mathcal{C}$ responds to each query as follows:

1. q_{pk}: Upon receiving the first such query for PID, the $\mathcal{C}$ searches L_{pk} for pk. If it does not exists, $\mathcal{C}$ runs the Set secret value algorithm to generate a sv for PID, and then performs the Set public key algorithm to return the pk to $\mathcal{A}$.
2. q_{ppk}: Given PID as input, the $\mathcal{C}$ checks if $\mathsf{PID} = \mathsf{PID}^*$. If it does, the $\mathcal{C}$ aborts. Otherwise, it fetches the d form L_{pk}. If it does not exists in L_{pk} then $\mathcal{C}$ runs Partial private key algorithm to return d and updates L_{pk}.
3. q_{sv}: Upon receiving q_{sv} for PID, the $\mathcal{C}$ checks L_{pk} for x_{i}. If it does not exists, $\mathcal{C}$ runs q_{pk} and returns x_{i} to $\mathcal{A}$.
4. q_{pr}: Given PID as input, the $\mathcal{C}$ replaces pk with pk' and updates L_{pk}.
5. q_{sc}: On input the message m, $\mathsf{PID_s}$, and $\mathsf{PID_{r_i}}$, the $\mathcal{C}$ checks if $\mathsf{PID_{r_i}} = \mathsf{PID}^*$. If it is not, $\mathcal{C}$ performs normal signcryption operation by taking values from L_{pk}. Otherwise, $\mathcal{C}$ performs the Signcryption algorithm to generate CT.
6. q_{usc}: Upon receiving $(\mathsf{CT}, \mathsf{PID_s}, \mathsf{PID_{r_i}})$ as input, the $\mathcal{C}$ checks if $\mathsf{PID_{r_i}} = \mathsf{PID}^*$. If it is not, $\mathcal{C}$ performs normal unsigncryption operation. Otherwise, $\mathcal{C}$ performs the Unsigncryption algorithm to answer m.

Challenge: The $\mathcal{A}$ outputs a target plaintext pair (m_0, m_1). The $\mathcal{C}$ picks $\beta \in \{0, 1\}^*$ at random, sets challenge CT^*, and sends CT^* to $\mathcal{A}$.
Phase-3: The $\mathcal{A}$ can make further queries except that the target CT^* is not allowed to appear in the q_{usc}.
Guess: Finally, $\mathcal{A}$ responds with its guess $\beta \in \{0, 1\}^*$. If $\beta = \beta'$, $\mathcal{A}$ wins the game.
The advantage of $\mathcal{A}_{\mathrm{I}}$ is defined as:

$$Adv_{\mathcal{A}_{\mathrm{I}}}^{IND-AMCLHS-CCA2} = | \Pr[\beta = \beta'] - 1/2 | \tag{1}$$

The advantage of $\mathcal{A}_{\mathrm{II}}$ is defined as:

$$Adv_{\mathcal{A}_{\mathrm{II}}}^{IND-AMCLHS-CCA2} = | \Pr[\beta = \beta'] - 1/2 | \tag{2}$$

Definition 4. *For EUF-AMCLHS-CMA, we define Game-II played between $\mathcal{C}$ and $\mathcal{A}$. An AMCLHS is Type-I and Type-II EUF-CMA secure if every PPT $\mathcal{A}$ has a negligible advantage in winning the Game-II.*

Game-II (EUF-AMCLHS-CMA-I, EUF-AMCLHS-CMA-II): This Game is an interaction between $\mathcal{C}$ and $\mathcal{A}$ as follows:
Phase-1: The $\mathcal{A}$ asks polynomially bounded number of adaptive queries hash queries

q_{H_l} $\{l = 1, 2, 3\}$. The $\mathcal{C}$ keeps a list L_l of q_{H_l} to record the responses.
Setup: The $\mathcal{C}$ generates $(\mathsf{PP}, \mathsf{msk}, \mathsf{mpk}, \mathsf{sk_{RA}}, \mathsf{pk_{RA}})$ and sends PP to $\mathcal{A}$. $\mathcal{A}$ selects a target $\mathsf{PID}_\mathsf{s}^*$. In EUF-AMCLHS-CMA Game, the target is the $\mathsf{PID}_\mathsf{s}^*$.
Phase-2: The $\mathcal{A}$ first asks number of adaptive queries with the restrictions defined in 3.2. The queries include q_{pk}, q_{ppk}, q_{pr}, q_{sv}, q_{sc}, and q_{usc} and are defined in *Phase-2* of Game-I in definition. 3.

Forgery: $\mathcal{A}$ outputs the forged ciphertext under a targeted $\mathsf{PID}_\mathsf{s}^*$. $\mathcal{A}$ wins if unsigncryption does not return $\perp$.

4 mKEM-DEM based Anonymous Multireceiver Certificateless Hybrid Signcryption Scheme (AMCLHS)

In this section, we construct the proposed mKEM-DEM based AMCLHS scheme according to the framework in definition 2. The main scheme is shown in Fig. 1 and construction is given as below:

1. Setup: KGC initializes the system by taking the security parameter λ as input. It generates two large cyclic groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of a large prime order $q > 2^\lambda$, and a BP $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$, and selects a generator P of $\mathbb{G}_1$. KGC defines four hash functions $H_0 : \{0,1\}^* \times \mathbb{G}_1 \times \mathbb{Z}_q^* \to \{0,1\}^l$ where l is a positive integer, $H_1 : \{0,1\}^l \times \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_1$, $H_2 : \mathbb{Z}_q^* \times \mathbb{Z}_q^* \times \mathbb{G}_2 \times \mathbb{G}_2 \times \mathbb{G}_1 \times \mathbb{G}_1 \to \{0,1\}^k$ where k is a plaintext box length. $H_3 : \{0,1\}^* \times \{0,1\}^k \times \{0,1\}^* \times \{0,1\}^k \times \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{Z}_q^*$. KGC generates $\mathsf{PP} = \{\mathbb{G}_1, \mathbb{G}_2, P, \hat{e}, q, H_0, H_1, H_2, H_3\}$ and chooses $x_0 \in \mathbb{Z}_q^*$ at random as msk and calculates $\mathsf{mpk} = x_0.P$. Next, RA chooses $v \in \mathbb{Z}_q^*$ randomly as $\mathsf{sk_{RA}}$ and computes $\mathsf{pk_{RA}} = v.P$. RA publishes its $\mathsf{pk_{RA}}$ while keeping $\mathsf{sk_{RA}}$ secret. KGC publishes PP, mpk and keeps the msk secret.
2. Pseudo-Identity: This algorithm is run by the each user and RA as follows:
   - **User**: Each user begins by choosing random $\mathsf{ID_R} \in \{0,1\}^*$ and computes $R = \alpha.P$ where $\alpha \in \mathbb{Z}_q^*$. Taking $\mathsf{ID_R}$ and α as input, the users compute its initial $\mathsf{PID}_1 = \mathsf{ID_R} \oplus H_0(\alpha.\mathsf{pk_{RA}})$ and sends (PID_1, R) to RA.
   - **RA**: Taking (PID_1, R) as input, the RA verifies the $\mathsf{ID_R}$ of each user as $\mathsf{ID_R} = \mathsf{PID}_1 \oplus H_0(R.v)$. If it holds, the RA accepts the registration request from users and sends $\mathsf{PID} = \mathsf{ID_R} \oplus H_0((\alpha.\mathsf{pk_{RA}}) \| (\Delta T))$ to each user where $\Delta T \geq T_{current} - T_{generated}$ (T = Time of PID calculation).
3. Partial private key: Taking $(\mathsf{PID}, \mathsf{pk_{RA}})$ as input, the KGC computes $Q_{\mathsf{PID}} = H_1(\mathsf{PID} \| x_0.P)$ and the ppk as $\mathsf{d} = x_0.Q_{\mathsf{PID}}$ for each user.
4. Set secret value: Each user with PID chooses secret value $x \in \mathbb{Z}_q^*$ randomly.
5. Set private key: On input (ppk, x), each user with PID set $\mathsf{sk} = (\mathsf{d}, x)$.
6. Set public key: Taking x as input, each user with PID computes $\mathsf{pk} = x.P$.
7. Signcryption: The sender with $\mathsf{PID_s}$ and $\mathsf{sk_s}$ signcrypts a m to generate a CT. It sends the CT to t designated receivers with $\mathsf{PID_{r_i}}$ and $\mathsf{pk_{r_i}}$. The sender runs the following steps:
   - *Key encapsulation phase*
     (a) $Z_{1_i} = \mathsf{d_s}.Q_{\mathsf{PID_{r_i}}}$ where $Q_{\mathsf{PID_{r_i}}} = H_1(\mathsf{PID_{r_i}} \| x_0.P)$

(b) $Z_{2_i} = x_s.\mathsf{pk}_{r_i}$
(c) $Z_{3_i} = \hat{e}(\mathsf{mpk}, Q_{\mathsf{PID}_{r_i}})$
(d) $\psi = H_2(Z_{1_i}, Z_{2_i}, Z_{3_i})$
(e) Randomly chooses $r \in \mathbb{Z}_q^*$ and computes $U = r.P$, $\mathsf{K} = \mathsf{mKEM.Encaps}(r)$.
(f) $\mathsf{C} = r \oplus \psi$ and outputs (C, K)

– *Message encryption and signing phase*
(a) $\mathsf{c_i} = \mathsf{Enc}_{\mathsf{K}}(m)$
(b) $f = H_3(m, \mathsf{C}, \mathsf{K}, \mathsf{c_i}, \mathsf{PID_s}, \mathsf{PID}_{r_i}, \mathsf{pk_s}, \mathsf{pk}_{r_i})$
(c) Signature $S_i = r^{-1}(f + w.\mathsf{d_s}x_s)$ where $w = \mathrm{x}_U \mathrm{mod}(n)$ which is the x-coordinate of U.
(d) $\mathsf{CT} = (f, \mathsf{c_i}, S_i, U)$

The sender sends CT to t designated receivers.

8. Unsigncryption: The designated receiver with PID_{r_i} takes the $\mathsf{CT} = (f, \mathsf{c_i}, S_i, U)$, its sk_{r_i}, and $\mathsf{pk_s}$ as input and runs the following algorithms to unsigncrypt m:
– *Key decapsulation phase*
(a) $Z_{1_i} = \mathsf{d}_{r_i}.Q_{\mathsf{PID_s}}$
(b) $Z_{2_i} = \mathsf{pk_s}.x_{r_i}$
(c) $Z_{3_i} = \hat{e}(P, \mathsf{d}_{r_i})$
(d) $\psi = H_2(Z_{1_i}, Z_{2_i}, Z_{3_i})$
(e) $r = \mathsf{C} \oplus \psi$ and $\mathsf{K}' = \mathsf{mKEM.Decaps}(r)$

If $\mathsf{K}' \neq \mathsf{K}$, the receiver aborts otherwise decrypts m as follows:
– *Message decryption and verification phase*
(a) $m' = \mathsf{Dec}_{\mathsf{K}'}(\mathsf{c_i})$
(b) $f' = H_3(m', \mathsf{C}, \mathsf{K}', \mathsf{c_i}, \mathsf{PID_s}, \mathsf{PID}_{r_i}, \mathsf{pk_s}, \mathsf{pk}_{r_i})$
(c) If $f' = f$, the receiver will verify the signature. To verify the signature S_i it will check if $U = r.P$ and $w' = x_U \mathrm{mod}(n)$

If $w' = w$, the receiver will accept the signcrypted m else returns $\perp$.

Correctness

1. $\mathsf{ID_R} = \mathsf{PID}_1 \oplus H_0(R.v) = \mathsf{ID_R} \oplus H_0(\alpha.\mathsf{pk_{RA}}) \oplus H_0(Rv) = \mathsf{ID_R} \oplus H_0(R.v) \oplus H_0(Rv) = \mathsf{ID_R}$
2. $Z_{1_i} = \mathsf{d_s}.Q_{\mathsf{PID}_{r_i}} = x_0.Q_{\mathsf{PID_s}}.Q_{\mathsf{PID}_{r_i}} = \mathsf{d}_{r_i}.Q_{\mathsf{PID_s}}$
3. $Z_{2_i} = x_s.\mathsf{pk}_{r_i} = x_s.\mathsf{pk}_{r_i} = x_s.x_{r_i}.P = \mathsf{pk_s}.x_{r_i} = \mathsf{pk_s}.x_{r_i}$
4. $Z_{3_i} = \hat{e}(\mathsf{mpk}, Q_{\mathsf{PID}_{r_i}}) = \hat{e}(x_0.P, Q_{\mathsf{PID}_{r_i}}) = \hat{e}(P, x_0.Q_{\mathsf{PID}_{r_i}}) = \hat{e}(P, x_0.Q_{\mathsf{PID}_{r_i}}) = \hat{e}(P, \mathsf{d}_{r_i})$
5. $Z_{1_i} = \mathsf{d_s}.Q_{\mathsf{PID}_{r_i}}, \mathsf{pk_s} = x_s.P$. Let $u_1 = f.P$ and $u_2 = w.\mathsf{pk_s}.Z_{1_i}.Q_{\mathsf{PID}_{r_i}}^{-1}$
$U_i = S_i^{-1}(u_1 + u_2) = S_i^{-1}(f.P + w.\mathsf{pk_s}.Z_{1_i}.Q_{\mathsf{PID}_{r_i}}^{-1}) = S_i^{-1}(f.P + w.\mathsf{pk_s}.\mathsf{d_s}.Q_{\mathsf{PID}_{r_i}} Q_{\mathsf{PID}_{r_i}}^{-1}) = S_i^{-1}(f.P + w.x_s.P.\mathsf{d_s}) = \frac{(f.P + w.x_s.P.\mathsf{d_s})}{S_i} = \frac{P(f + w.x_s.\mathsf{d_s})}{r^{-1}(f + w.x_s.\mathsf{d_s})} = \frac{P}{r^{-1}} = r.P$ and $w' = \mathrm{x}_U \mod(n)$

5 Security Analysis

The security analysis of the proposed hybrid signcryption scheme is based on the security model defined in section 3.2. The message confidentiality is based on Theorems

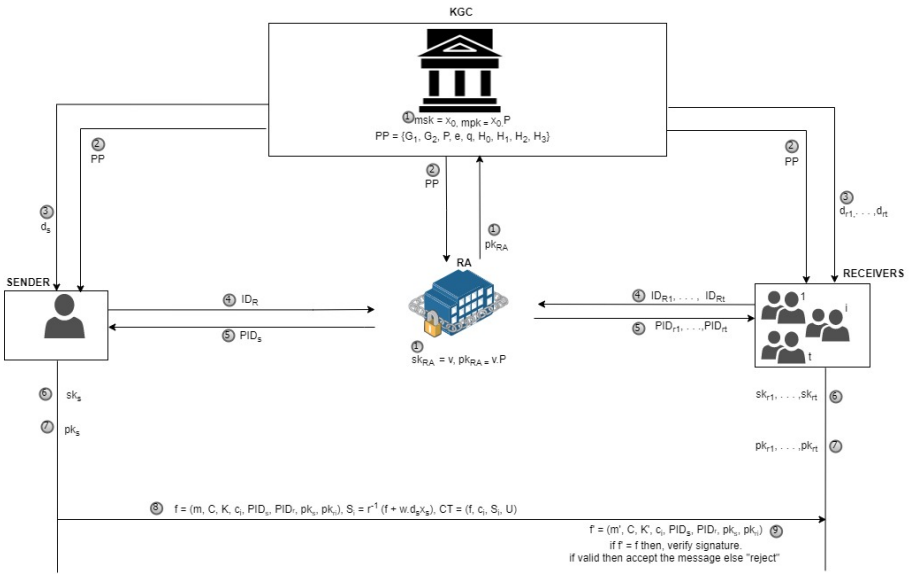

Fig. 1. The mKEM-DEM (AMCLHS) scheme

1 and 2 which demonstrates that the scheme is secure against IND-AMCLHS-CCA2 Type-I and Type-II adversaries in aforementioned Game-I in definition 3. Similarly, unforgeability is based on Theorems 3 and 4 and follows that the scheme is secure against EUF-AMCLHS-CMA Type-I and Type-II adversaries in the aforementioned Game-II in definition 4.

Confidentiality

Theorem 1. *The proposed scheme is IND-AMCLHS-CCA2-I secure under the ROM based on the hardness of the q-DBDHI assumption. Assume that the hash functions* $H_l\{l = 1, 2, 3\}$ *are secure under ROM and* $\mathcal{A}_{\mathrm{I}}$ *can make a number of* q_{H_l} *queries to the ROM including* q_{pk}*,* q_{ppk}*,* q_{pr}*,* q_{sv}*,* q_{sc}*, and* q_{usc}*. Suppose that the IND-AMCLHS-CCA2-I adversary* $\mathcal{A}_{\mathrm{I}}$ *has a non-negligible advantage* ϵ *in winning the game then, there is* $\mathcal{C}$ *that can solve the q-DBDHI with the non-negligible advantage* ϵ'*.*

Proof. Given a random instance of the q-DBDHI, the $\mathcal{C}$ has to compute $J = \hat{e}(P, P)^{1/\alpha} \in \mathbb{G}_2$ as definition given in Section 2 by interacting with the $\mathcal{A}_{\mathrm{I}}$ to solve the q-DBDHI as follows:

Phase-1: A polynomially bounded number of adaptive queries q are made by an $\mathcal{A}_{\mathrm{I}}$. The $\mathcal{C}$ keeps a list L_l of q_{H_l} to record the responses.
Setup: The $\mathcal{C}$ runs the setup algorithm to generate $\mathsf{PP} = \{\mathbb{G}_1, \mathbb{G}_2, P, \hat{e}, q, H_0, H_1, H_2, H_3\}$. The $\mathcal{C}$ sets new value for the $\mathsf{mpk} = \theta^{-1}.P$ and sends PP and mpk to the $\mathcal{A}_{\mathrm{I}}$. The $\mathcal{A}_{\mathrm{I}}$ selects t target identities denoted by $\mathsf{PID}_{\mathsf{i}}^*$ where $1 \leq \mathsf{i} \leq t$ and $t < n$.
H_1**-Query**: Upon receiving H_1 query from the $\mathcal{A}_{\mathrm{I}}$, $\mathcal{C}$ determines whether the tuple $(Q_{\mathsf{PID}_{\mathsf{i}}}, \mathsf{mpk}, \mathsf{PID}_{\mathsf{i}})$ exists in the list L_1 or not. If it already exists, $\mathcal{C}$ returns $Q_{\mathsf{PID}_{\mathsf{i}}}$ to

$\mathcal{A}_{\mathrm{I}}$. Otherwise if $\mathsf{PID_i} \neq \mathsf{PID_i^*}$, $\mathcal{C}$ sets $Q_{\mathsf{PID_i}} = H_1(\mathsf{PID_i} \| \mathsf{mpk})$. If $\mathsf{PID_i} = \mathsf{PID_i^*}$, $\mathcal{C}$ chooses $\gamma^{-1} \in \mathbb{Z}_q^*$ randomly and computes $Q_{\mathsf{PID_i}} = \gamma^{-1}.P$ and adds a new tuple $(Q_{\mathsf{PID_i}}, \mathsf{mpk}, \mathsf{PID_i})$ in L_1 and sends $Q_{\mathsf{PID_i}}$ to $\mathcal{A}_{\mathrm{I}}$.

H_2**-Query**: Upon receiving H_2 query from the $\mathcal{A}_{\mathrm{I}}$, $\mathcal{C}$ determines whether the tuple $(\psi, Z_{1_i}, Z_{2_i}, Z_{3_i})$ exists in the list L_2 or not. If it already exists, $\mathcal{C}$ returns ψ to $\mathcal{A}_{\mathrm{I}}$. Otherwise, $\mathcal{C}$ chooses $\psi \in \{0,1\}^k$ randomly, updates the tuple $(\psi, Z_{1_i}, Z_{2_i}, Z_{3_i})$ and sends ψ to $\mathcal{A}_{\mathrm{I}}$.

H_3**-Query**: Upon receiving H_3 query from the $\mathcal{A}_{\mathrm{I}}$, $\mathcal{C}$ determines whether the tuple $H_3(m, \mathsf{C}, \mathsf{K}, \mathsf{c_i}, f)$ exists in the list L_3 or not. If it already exists, $\mathcal{C}$ returns f to $\mathcal{A}_{\mathrm{I}}$. Otherwise, it chooses $f \in \mathbb{Z}_q^*$ randomly, updates the tuple $H_3(m, \mathsf{C}, \mathsf{K}, \mathsf{c_i}, f)$ and sends f to $\mathcal{A}_{\mathrm{I}}$.

Phase-2: The adversary $\mathcal{A}_{\mathrm{I}}$ asks a number queries in an adaptive manner including q_{pk}, q_{ppk}, q_{pr}, q_{sv}, and q_{usc}. An initially empty list L_{pk} is maintained by the $\mathcal{C}$. The $\mathcal{C}$ stores the public key and secret value information in L_{pk}. The $\mathcal{C}$ responds the queries as follows:

1. q_{pk}: Upon receiving the $\mathsf{pk_i}$ query for $\mathsf{PID_i}$, $\mathcal{C}$ checks if $\mathsf{pk_i}$ exists in L_{pk} If it exists, $\mathcal{C}$ returns $\mathsf{pk_i}$ to $\mathcal{A}_{\mathrm{I}}$. Otherwise, $\mathcal{C}$ chooses $x_i \in \mathbb{Z}_q^*$ and computes $\mathsf{pk_i} = x_i.P$ and adds the tuple $(\mathsf{PID_i}, -, \mathsf{pk_i}, x_i)$ in L_{pk} and returns $\mathsf{pk_i}$ to $\mathcal{A}_{\mathrm{I}}$.
2. q_{ppk}: Upon receiving the query, if $\mathsf{PID_i} = \mathsf{PID_i^*}$, the $\mathcal{C}$ aborts. Otherwise, if it exists in the list L_{pk}, $\mathcal{C}$ sends $\mathsf{d_i}$ to $\mathcal{A}_{\mathrm{I}}$, if it does not, $\mathcal{C}$ randomly chooses $Q_{\mathsf{PID_i}} = \gamma^{-1}.P$ from L_1 and return $\mathsf{d_i} = \mathsf{mpk} Q_{\mathsf{PID_i}}$ to $\mathcal{A}_{\mathrm{I}}$. The $\mathcal{C}$ then updates the tuple $(\mathsf{PID_i}, \mathsf{d_i}, \mathsf{pk_i}, x_i)$ in L_{pk}.
3. q_{sv}: Upon receiving the query, $\mathcal{C}$ checks if it exists in the list, L_{pk}, if it does, $\mathcal{C}$ sends x_i to $\mathcal{A}_{\mathrm{I}}$. If it does not, $\mathcal{C}$ performs the public key retrieve query and return x_i to $\mathcal{A}_{\mathrm{I}}$.
4. q_{pr}: Upon receiving the query, the $\mathcal{C}$ replaces the public key $\mathsf{pk_i}$ with $\mathsf{pk_i'}$ for $\mathsf{PID_i}$ and updates the tuple $(\mathsf{PID_i}, \mathsf{d_i}, \mathsf{pk_i'}, -)$ in the list L_{pk}.
5. q_{sc}: Upon receiving the query with sender's $\mathsf{PID_s}$, receiver's $\mathsf{PID_{r_i}}$ and a m, the $\mathcal{C}$ checks whether $\mathsf{PID_{r_i}} = \mathsf{PID_i^*}$. The $\mathcal{C}$ performs the normal signcryption operation if $\mathsf{PID_{r_i}} \neq \mathsf{PID_i^*}$ by taking values from L_{pk}. Otherwise, the $\mathcal{C}$ performs the signcryption as follows:
   (a) If $\mathsf{pk_i}$ is replaced, the $\mathcal{A}_{\mathrm{I}}$ will provide another value.
   (b) Gets $Q_{\mathsf{PID_{r_i}}}$ from L_1 and computes $Z_{1_i} = \mathsf{d_s}.Q_{\mathsf{PID_{r_i}}}$, $Z_{2_i} = x_{\mathsf{s}}.\mathsf{pk_{r_i}}$, $Z_{3_i} = \hat{e}(\mathsf{mpk}, Q_{\mathsf{PID_{r_i}}})$, $\psi = H_2(Z_{1_i}, Z_{2_i}, Z_{3_i})$, and updates L_2.
   (c) Chooses $r \in \mathbb{Z}_q^*$ randomly and computes $U = r.P$, $\mathsf{K} = \mathsf{mKEM.Encaps}(r)$.
   (d) $\mathsf{C} = r \oplus \psi$ $\mathsf{c_i} = \mathsf{Enc_K}(m)$.
   (e) $f = H_3(m, \mathsf{C}, \mathsf{K}, \mathsf{c_i}, \mathsf{PID_s}, \mathsf{PID_{r_i}}, \mathsf{pk_s}, \mathsf{pk_{r_i}})$ and updates L_3.
   (f) Chooses $S_i \in \mathbb{Z}_q^*$ randomly and returns $\mathsf{CT} = \{f, \mathsf{c_i}, S_i, U\}$ to adversary $\mathcal{A}_{\mathrm{I}}$.
6. q_{usc}: Upon receiving the query with sender's $\mathsf{PID_s}$, receiver's $\mathsf{PID_{r_i}}$ and a CT, the $\mathcal{C}$ checks whether $\mathsf{PID_{r_i}} = \mathsf{PID_i^*}$ or not. If $\mathsf{PID_{r_i}} \neq \mathsf{PID_i^*}$, the $\mathcal{C}$ performs the normal unsigncryption operation. Otherwise, the $\mathcal{C}$ unsigncrypts m as follows:
   (a) If $\mathsf{pk_i}$ is replaced, the $\mathcal{A}_{\mathrm{I}}$ will provide another value.
   (b) Searches the lists L_2 and L_3 for $(\psi, Z_{1_i}, Z_{2_i}, Z_{3_i})$ and $H_3(m, \mathsf{C}, \mathsf{K}, \mathsf{c_i}, f)$.
   (c) If the record does not exist, $\mathcal{C}$ returns "failure". If it exists, the $\mathcal{C}$ computes $\mathsf{K'} = \mathsf{K}$ and $m' = \mathsf{Dec_{K'}}(\mathsf{c_i})$.
   (d) Checks if $f' = f$, if it holds then checks if $U = r.P$ and $w' = \mathrm{x}_U \bmod(n)$ holds or not. If yes, the $\mathcal{C}$ answers m else, returns $\perp$.

Challenge: The $\mathcal{A}_{\text{I}}$ chooses equal length plaintext message pair (m_0, m_1) and sends the target plaintext to the $\mathcal{C}$. The $\mathcal{A}_{\text{I}}$ takes a sender $\mathsf{PID_s}$ and a target $\mathsf{PID_{r_i}}$. Moreover, the $\mathcal{A}_{\text{I}}$ can not ask for the sk of the target $\mathsf{PID_{r_i}}$. If $\mathsf{PID_{r_i}} \neq \mathsf{PID_i^*}$, the returns $\perp$. Otherwise, the $\mathcal{C}$ chooses $\beta \in \{0,1\}^*$ and performs the following steps to generate a challenge CT^*:

1. Computes $Z_{1_i} = \mathsf{d_s}.Q_{\mathsf{PID_{r_i}}}$, $Z_{2_i} = x_{\mathsf{s}}.\mathsf{pk_{r_i}}$, $Z_{3_i} = (\mathsf{mpk}, Q_{\mathsf{PID_s}})$ and $\psi^* = H_2(Z_{1_i}^*, Z_{2_i}^*, Z_{3_i}^*)$.
2. Chooses $r^* \in \mathbb{Z}_q^*$ and computes $U^* = r^*.P$, $\mathsf{K}^* = \mathsf{mKEM.Encaps}(r^*)$.
3. $\mathsf{C}^* = r^* \oplus \psi^*$, $\mathsf{c_i^*} = \mathsf{Enc}_{\mathsf{K}^*}(m)$.
4. $f^* = H_3(m, \mathsf{C}^*, \mathsf{K}^*, \mathsf{c_i^*}, \mathsf{PID_s}, \mathsf{PID_{r_i}}, \mathsf{pk_s}, \mathsf{pk_{r_i}})$.
5. Computes $S_{\mathsf{i}}^* = r^{*-1}(f^* + w.d_s x_{\mathsf{s}})$ and $\mathsf{CT}^* = (f^*, \mathsf{c_i^*}, S_{\mathsf{i}}^*, U^*)$.

Phase-3: The adversary $\mathcal{A}_{\mathcal{I}}$ may issue further polynomially bounded adaptive queries as in *Phase-1*, however, $\mathcal{A}_{\text{I}}$ cannot send the q_{ppk} of the target $\mathsf{PID_{r_i}}$, or the unsigncryption query for CT^*.
Guess: The adversary $\mathcal{A}_{\text{I}}$ will respond with the guess bit $\beta \in \{0,1\}^*$. Adversary wins the game if $\beta' = \beta$. The $\mathcal{C}$ will win the game by obtaining $X = \hat{e}(P,P)^{1/\theta\gamma}$ which is the solution to the q-DBDHI. The $\mathcal{C}$ obtains it by evaluating $Z_{3_i}^*$ from the list L_2. Since $\mathsf{mpk} = \theta^{-1}.P$, $Q_{\mathsf{PID_i}} = \gamma^{-1}.P$ from L_1, we can evaluate $X = Z_{3_i}^* = \hat{e}(P, \mathsf{d_{r_i}}) = \hat{e}(P, \mathsf{mpk}.Q_{\mathsf{PID_{r_i}}}) = \hat{e}(P, \mathsf{mpk}.\gamma^{-1}.P) = \hat{e}(P, \gamma^{-1}.\mathsf{mpk}) = \hat{e}(P, \gamma^{-1}.\theta^{-1}.P) = \hat{e}(P,P)^{1/\theta\gamma}$.

In the end, the $\mathcal{C}$ is able to find the solution to the q-DBDHI $X = \hat{e}(P,P)^{1/\theta\gamma}$. Next, we evaluate the advantage of $\mathcal{C}$ winning the Game-I (IND-AMCLHS-CCA-I) by calculating the probability of aborting the game during occurrence of the following events:

1. In partial private key query, the game aborts for $\mathsf{PID_i} = \mathsf{PID_i^*}$. The probability is $\Pr(E_{q_{\mathsf{ppk}}}) = 1/q_{\mathsf{ppk}}$.
2. In unsigncryption query, the game aborts due to invalid m. The probability is $\Pr(E_{q_{\mathsf{usc}}}) = q_{\mathsf{usc}}/2^k$.
3. In the challenge phase, $\mathcal{C}$ aborts the game if the adversary queries against the identity $\mathsf{PID_{r_i}} \neq \mathsf{PID_i^*}$. The probability is $\Pr(E_{q_{H_1}}) = (1 - 1/qH_1)$.

Moreover, the $\mathcal{C}$ fetches the list L_1 and L_2 to evaluate X with probability $(1/q_{H_1} + 1/q_{H_2})$. Therefore, the probability of the $\mathcal{C}$ winning the game with advantage ϵ' is:

$$\epsilon' \geq \epsilon \left(\frac{1}{qH_1} + \frac{1}{qH_2}\right)\left(\frac{1}{qH_1}\right)\left(1 - \frac{1}{q_{\mathsf{ppk}}}\right)\left(1 - \frac{q_{\mathsf{usc}}}{2^k}\right) \tag{3}$$

Theorem 2. *The proposed scheme is IND-AMCLHS-CCA2-II secure under the ROM based on the hardness of the q-DBDHI assumption. Assume that the hash functions $H_l\{l = 1,2,3\}$ are ROM and $\mathcal{A}_{\text{II}}$ can make a number of q_{H_l} queries to the ROM, including q_{pk}, q_{sv}, q_{sc}, and q_{usc}. Suppose that the IND-AMCLHS-CCA2-II adversary $\mathcal{A}_{\text{II}}$ has a non-negligible advantage ϵ in winning the game then, there is a $\mathcal{C}$ that can solve the q-DBDHI with the non-negligible advantage ϵ'.*

Proof. Given a random instance of the q-DBDHI and the $\mathcal{C}$ has to compute $J = \hat{e}(P,P)^{1/\alpha} \in \mathbb{G}_2$ by interacting with the $\mathcal{A}_{\text{II}}$ to solve the q-DBDHI as follows:

Phase-1: A polynomially bounded number of adaptive queries q are made by an $\mathcal{A}_{\text{II}}$.The Challenger $\mathcal{C}$ keeps a list L_l of q_{H_l} to record the responses.
Setup: The $\mathcal{C}$ runs the setup algorithm to generate $\mathsf{PP} = \{\mathbb{G}_1, \mathbb{G}_2, P, \hat{e}, q, H_0, H_1, H_2, H_3\}$. The $\mathcal{C}$ sets new $\mathsf{mpk} = \theta^{-1}.P$ and sends PP and mpk to the $\mathcal{A}_{\text{II}}$. The $\mathcal{A}_{\text{II}}$ selects the target $\mathsf{PID}_\mathsf{i}^*$ $1 \leq \mathsf{i} \leq t$ and $t < n$.
H_1**-Query**: Upon receiving H_1 query from the $\mathcal{A}_{\text{II}}$, the $\mathcal{C}$ determines whether the tuple $(Q_{\mathsf{PID}_\mathsf{i}}, \mathsf{mpk}, \mathsf{PID}_\mathsf{i})$ exists in the list L_1 or not. If it already exists, $\mathcal{C}$ returns $Q_{\mathsf{PID}_\mathsf{i}}$ to $\mathcal{A}_{\text{II}}$. Otherwise, if $\mathsf{PID}_\mathsf{i} \neq \mathsf{PID}_\mathsf{i}^*$, $\mathcal{C}$ sets $Q_{\mathsf{PID}_\mathsf{i}} = H_1(\mathsf{PID}_\mathsf{i} \| \mathsf{mpk})$. If $\mathsf{PID}_\mathsf{i} = \mathsf{PID}_\mathsf{i}^*$, $\mathcal{C}$ chooses $\gamma^{-1} \in \mathbb{Z}_q^*$ randomly and computes $Q_{\mathsf{PID}_\mathsf{i}} = \gamma^{-1}.P$ and adds a new tuple $(Q_{\mathsf{PID}_\mathsf{i}}, \mathsf{mpk}, \mathsf{PID}_\mathsf{i})$ in L_1. The $\mathcal{C}$ sends $Q_{\mathsf{PID}_\mathsf{i}}$ to $\mathcal{A}_{\text{II}}$.
H_2, H_3**-Query**: Upon receiving H_2 and H_3 queries from the $\mathcal{A}_{\text{II}}$, the $\mathcal{C}$ determines whether the tuple $(\psi, Z_{1_\mathsf{i}}, Z_{2_\mathsf{i}}, Z_{3_\mathsf{i}})$ and $H_3(m, \mathsf{C}, \mathsf{K}, \mathsf{c_i}, f)$ exists in the list L_2 and L_3 or not. If it already exists, $\mathcal{C}$ returns ψ and f to $\mathcal{A}_{\text{II}}$. Otherwise, the $\mathcal{C}$ chooses $\psi \in \{0,1\}^k$ and $f \in \mathbb{Z}_q^*$ randomly and updates the tuple $(\psi, Z_{1_\mathsf{i}}, Z_{2_\mathsf{i}}, Z_{3_\mathsf{i}})$ and $H_3(m, \mathsf{C}, \mathsf{K}, \mathsf{c_i}, f)$. The $\mathcal{C}$ sends ψ and f to $\mathcal{A}_{\text{II}}$.
Phase-2: The adversary $\mathcal{A}_{\text{II}}$ asks a number of queries in an adaptive manner, including q_{pk}, q_{sv}, and q_{usc}. An initially empty list L_{pk} is maintained by $\mathcal{C}$. The $\mathcal{C}$ stores the public key and secret value information in L_{pk}. $\mathcal{C}$ responds the queries as follows:

1. q_{pk}: Upon receiving the $\mathsf{pk_i}$ query for $\mathsf{PID_i}$, the $\mathcal{C}$ checks if $\mathsf{pk_i}$ exists in the L_{pk} as $(\mathsf{PID_i}, \mathsf{d_i}, \mathsf{pk_i}, x_\mathsf{i})$. If it exists, $\mathcal{C}$ returns $\mathsf{pk_i}$ to $\mathcal{C}$. Otherwise, $\mathcal{C}$ chooses $x_\mathsf{i} \in \mathbb{Z}_q^*$ and computes $\mathsf{pk_i} = x_\mathsf{i}.P$ and adds the tuple $(\mathsf{PID_i}, -, \mathsf{pk_i}, x_\mathsf{i})$ in L_{pk} and returns $\mathsf{pk_i}$ to $\mathcal{A}_{\text{II}}$.
2. q_{sv}: Upon receiving the query for $\mathsf{PID_i}$, the $\mathcal{C}$ checks if $\mathsf{PID_i} = \mathsf{PID_i^*}$. If it holds, the $\mathcal{C}$ aborts because in this case, the $\mathsf{PID_i}$ is a target identity. Otherwise, it checks if x_i already exists in the L_{pk} as $(\mathsf{PID_i}, \mathsf{d_i}, \mathsf{pk_i}, x_\mathsf{i})$. If it exists, the $\mathcal{C}$ returns x_i to $\mathcal{A}_{\text{II}}$. Otherwise, $\mathcal{C}$ runs q_{pk} and computes $\mathsf{pk_i} = x_\mathsf{i}.P$ and adds the tuple $(\mathsf{PID_i}, \mathsf{d_i}, \mathsf{pk_i}, x_\mathsf{i})$ in L_{pk} and returns x_i to $\mathcal{A}_{\text{II}}$.
3. q_{sc}: Upon receiving the query with sender's $\mathsf{PID_s}$, target $\mathsf{PID_{r_i}}$, and m, the $\mathcal{C}$ checks whether $\mathsf{PID_{r_i}} = \mathsf{PID_i^*}$ or not. The $\mathcal{C}$ performs the normal signcryption operation if $\mathsf{PID_{r_i}} \neq \mathsf{PID_i^*}$ by taking values from L_{pk}. Otherwise, if $\mathsf{PID_{r_i}} = \mathsf{PID_i^*}$, the $\mathcal{C}$ performs the signcryption as follows:
   (a) Gets $Q_{\mathsf{PID_{r_i}}}$ from L_1 and computes $Z_{1_\mathsf{i}} = \mathsf{d_s}.Q_{\mathsf{PID_{r_i}}}$, $Z_{2_\mathsf{i}} = x_\mathsf{s}.\mathsf{pk_{r_i}}$, $Z_{3_\mathsf{i}} = \hat{e}(\mathsf{mpk}, Q_{\mathsf{PID_{r_i}}})$, and $\psi = H_2(Z_{1_\mathsf{i}}, Z_{2_\mathsf{i}}, Z_{3_\mathsf{i}})$.
   (b) Chooses $r \in \mathbb{Z}_q^*$, computes $U = r.P$, and updates $\mathsf{K} = \mathsf{mKEM.Encaps}(r)$.
   (c) $\mathsf{C} = r \oplus \psi$, and $\mathsf{c_i} = \mathsf{Enc_K}(m)$.
   (d) Computes $f = H_3(m, \mathsf{C}, \mathsf{K}, \mathsf{c_i}, \mathsf{PID_s}, \mathsf{PID_{r_i}}, \mathsf{pk_s}, \mathsf{pk_{r_i}})$ and updates L_3.
   (e) Chooses $S_\mathsf{i} \in \mathbb{Z}_q^*$ randomly and returns $\mathsf{CT} = \{f, \mathsf{c_i}, S_\mathsf{i}, U\}$ to adversary $\mathcal{A}_{\text{II}}$.
4. q_{usc}: Upon receiving the query with sender's $\mathsf{PID_s}$, receiver's $\mathsf{PID_{r_i}}$, and a CT, the $\mathcal{C}$ checks whether $\mathsf{PID_{r_i}} = \mathsf{PID_i^*}$ or not. The $\mathcal{C}$ performs the normal unsigncryption operation if $\mathsf{PID_{r_i}} \neq \mathsf{PID_i^*}$. Otherwise, the $\mathcal{C}$ unsigncrypts m as follows:
   (a) The $\mathcal{C}$ searches the lists L_2 and L_3 for $(\psi, Z_{1_\mathsf{i}}, Z_{2_\mathsf{i}}, Z_{3_\mathsf{i}})$ and $(m, \mathsf{C}, \mathsf{K}, \mathsf{c_i}, f)$.
   (b) If the record does not exist, $\mathcal{C}$ returns "failure". If it exists, the $\mathcal{C}$ computes $\mathsf{K'} = \mathsf{K}$ and $m' = \mathsf{Dec_{K'}}(\mathsf{c_i})$.
   (c) Checks if $f' = f$, if it holds then checks if $U = r.P$ and $w' = \mathrm{x}_U \bmod(n)$ holds or not. If yes, the $\mathcal{C}$ answers m else, returns $\perp$.

Challenge: The $\mathcal{A}_{\text{II}}$ chooses target plaintext m_0, m_1 and sends the target plaintext to the $\mathcal{C}$. The $\mathcal{A}_{\text{II}}$ takes a sender $\mathsf{PID_s}$ and a target $\mathsf{PID_{r_i}}$. Moreover, the $\mathcal{A}_{\text{II}}$ can not ask for the sk of the receiver $\mathsf{PID_{r_i}}$. If $\mathsf{PID_{r_i}} \neq \mathsf{PID_i^*}$, the returns $\perp$. Otherwise, the $\mathcal{C}$ chooses $\beta \in \{0,1\}^*$ and performs the following steps to generate a challenge CT^*:

1. Computes $Z_{1_i} = \mathsf{d_s}.Q_{\mathsf{PID_{r_i}}}$, $Z_{2_i} = x_{\mathsf{s}}.\mathsf{pk_{r_i}}$, $Z_{3_i} = (\mathsf{mpk}, Q_{\mathsf{PID_s}})$, and $\psi^* = H_2(Z_{1_i}^*, Z_{2_i}^*, Z_{3_i}^*)$.
2. Chooses $r^* \in \mathbb{Z}_q^*$ and computes $U^* = r^*.P$, $\mathsf{K}^* = \mathsf{mKEM.Encaps}(r^*)$.
3. $\mathsf{C}^* = r^* \oplus \psi^*$, $\mathsf{c_i^*} = \mathsf{Enc}_{\mathsf{K}^*}(m)$.
4. $f^* = H_3(m, \mathsf{C}^*, \mathsf{K}^*, \mathsf{c_i^*}, \mathsf{PID_s}, \mathsf{PID_{r_i}}, \mathsf{pk_s}, \mathsf{pk_{r_i}})$.
5. Computes $S_{\mathsf{i}}^* = r^{*-1}(f^* + w.\mathsf{d_s}x_{\mathsf{s}})$ and $\mathsf{CT}^* = (f^*, \mathsf{c_i^*}, S_{\mathsf{i}}^*, U^*)$.

Phase-3: The adversary $\mathcal{A}_{\text{II}}$ may issue further polynomially bounded adaptive queries as in *Phase-1* however, $\mathcal{A}_{\text{II}}$ cannot send the q_{sv} for the target $\mathsf{PID_{r_i}^*}$ and the unsigncryption query for CT^*.
Guess: The adversary $\mathcal{A}_{\text{II}}$ will respond with the guess bit $\beta \in \{0,1\}^*$. Adversary wins the game if $\beta' = \beta$.

The $\mathcal{C}$ will win the game by obtaining $X = \hat{e}(P,P)^{1/\theta\gamma}$ which is the solution to the q-DBDHI. The $\mathcal{C}$ obtains it by evaluating $Z_{3_i}^*$ from the list L_2. Since $\mathsf{mpk} = \theta^{-1}.P$, $Q_{\mathsf{PID_i}} = \gamma^{-1}.P$ from L_1, we can evaluate $X = Z_{3_i}^* = \hat{e}(P, \mathsf{d_{r_i}}) = \hat{e}(P, \mathsf{mpk}.Q_{\mathsf{PID_{r_i}}}) = \hat{e}(P, \mathsf{mpk}.\gamma^{-1}.P) = \hat{e}(P, \gamma^{-1}\mathsf{mpk}) = \hat{e}(P, \gamma^{-1}\theta^{-1}.P) = \hat{e}(P,P)^{1/\theta\gamma}$.

In the end, the $\mathcal{C}$ is able to find the solution to the q-DBDHI $X = \hat{e}(P,P)^{1/\theta\gamma}$. Next, we will analyse the advantage of the $\mathcal{C}$ in winning the game. The $\mathcal{C}$ advantage is based on the occurrence of the events in which the game aborts. The $\mathcal{C}$ aborts the game under the following conditions:

1. The secret value query where the game aborts for $\mathsf{PID_i} = \mathsf{PID_i^*}$. The probability is $\Pr(E_{q_{\mathsf{sv}}}) = 1/q_{\mathsf{sv}}$.
2. An unsigncryption query where the game aborts due to invalid m. The probability is $\Pr(E_{q_{\mathsf{usc}}}) = q_{\mathsf{usc}}/2^k$.
3. In the challenge phase, the adversary queries for $\mathsf{PID_{r_i}^*} \neq \mathsf{PID_i^*}$. The probability is $\Pr(E_{q_{H_1}}) = (1 - 1/qH_1)$.

Moreover, the $\mathcal{C}$ fetches the list L_1 and L_2 to evaluate X with probability $(1/q_{H_1} + 1/q_{H_2})$. Therefore, the probability of the $\mathcal{C}$ winning the game with advantage ϵ' is:

$$\epsilon' \geq \epsilon \left(\frac{1}{qH_1} + \frac{1}{qH_2}\right)\left(\frac{1}{qH_1}\right)\left(1 - \frac{1}{q_{\mathsf{sv}}}\right)\left(1 - \frac{q_{\mathsf{usc}}}{2^k}\right) \tag{4}$$

Unforgeability

Theorem 3. *The proposed scheme is EUF-AMCLHS-CMA-I secure under the ROM based on the hardness of the ECDL assumption. Assume that the hash functions $H_l\{l = 1,2,3\}$ are ROM and $\mathcal{A}_{\text{I}}$ can make a number of q_{H_l} queries to the ROM, including q_{pk}, q_{ppk}, q_{pr}, q_{sv}, q_{sc}, and q_{usc}. Suppose that the EUF-AMCLHS-CMA-I adversary $\mathcal{A}_{\text{I}}$ has a non-negligible advantage ϵ in winning the game then, there is $\mathcal{C}$ that can solve the ECDL with the non-negligible advantage ϵ'.*

Proof. Given two random instances of the ECDL $(Q, P) \in \mathbb{G}_1$ where $Q = \phi.P$. The $\mathcal{C}$ has to find ϕ by interacting with the $\mathcal{A}_{\text{I}}$.

Phase-1: A polynomially bounded number of adaptive queries q are made by an $\mathcal{A}_{\text{I}}$. The Challenger $\mathcal{C}$ keeps a list L_l of q_{H_l} to record the responses.
Setup: The $\mathcal{C}$ runs the setup algorithm to generate $\mathsf{PP} = \{\mathbb{G}_1, \mathbb{G}_2, P, \hat{e}, q, H_0, H_1, H_2, H_3\}$. The $\mathcal{C}$ sets $\mathsf{mpk} = \theta^{-1}.P$ and sends PP and mpk to the $\mathcal{A}_{\text{I}}$. The $\mathcal{A}_{\text{I}}$ selects a target identity denoted by $\mathsf{PID}_{\mathsf{s}}^*$.
Phase-2: The $\mathcal{A}_{\text{I}}$ asks a number of queries in an adaptive manner including q_{pk}, q_{ppk}, q_{pr}, q_{sv}, and q_{sc}. An initially empty list L_{pk} is maintained by the $\mathcal{C}$. The $\mathcal{C}$ stores the public key and secret value information in L_{pk}. $\mathcal{C}$ responds to all queries as in *Phase-2* of Theorem 1, except the responds to q_{ppk} as follows:

1. q_{ppk}: Upon receiving the query, if $\mathsf{PID} = \mathsf{PID}_{\mathsf{s}}^*$, the $\mathcal{C}$ aborts. Otherwise, if it exists in the list L_{pk}, the $\mathcal{C}$ sends $\mathsf{d_i}$ to $\mathcal{A}_{\text{I}}$, if it does not, the $\mathcal{C}$ randomly chooses $\phi \in \mathbb{Z}_q^*$ and computes $\mathsf{d_i} = \phi Q_{\mathsf{PID_i}}$. The $\mathcal{C}$ return $\mathsf{d_i} = \phi Q_{\mathsf{PID_i}}$ to $\mathcal{A}_{\text{I}}$ and updates the tuple $(\mathsf{PID_i}, \mathsf{d_i}, \mathsf{pk_i}, x_{\text{i}})$ in L_{pk}.

Forgery: Taking the target sender's $\mathsf{PID}_{\mathsf{s}}^*$ and designated receiver's $\mathsf{PID}_{\mathsf{r_i}}$, the adversary outputs a forged $\mathsf{CT}^* = (f^*, \mathsf{c}_{\text{i}}^*, S_{\text{i}}^*, R_1^*)$ on a m^* which is the valid signcrypted ciphertext and is not the result of signcryption oracle. If $\mathsf{PID} \neq \mathsf{PID}_{\mathsf{s}}^*$, the $\mathcal{C}$ returns $\perp$. Otherwise, the $\mathcal{C}$ extracts the list L_{pk} for the record $(\mathsf{PID}_{\text{i}}^*, \mathsf{d}_{\text{i}}^*, \mathsf{pk}_{\text{i}}^*, x_{\text{i}}^*)$ and L_3 for the record $(m^*, \mathsf{C}^*, \mathsf{K}^*, \mathsf{c}_{\text{i}}^*, f^*)$.
According to Forking Lemma, $\mathcal{C}$ replays the $\mathcal{A}_{\text{I}}$ with the same random tape but distinct attributes from H_1 and H_3. It implies that, $h_1^* = H_1(\mathsf{mpk}, \mathsf{PID}_{\text{i}}^*)$ and $h_1'^* = H_1(\mathsf{mpk}, \mathsf{PID}_{\text{i}}^*)$ and $h_1^* \neq h_1'^*$ i.e. $Q_{\mathsf{PID}_{\mathsf{s}}}^* \neq Q_{\mathsf{PID}_{\mathsf{s}}}'^*$. Similarly, $h_3^* = H_3(m^*, \mathsf{C}^*, \mathsf{K}^*, \mathsf{c}_{\text{i}}^*, \mathsf{PID}_{\mathsf{s}}^*, \mathsf{PID}_{\mathsf{r_i}}^*, \mathsf{pk}_{\mathsf{s}}^*, \mathsf{pk}_{\mathsf{r_i}}^*)$, $h_3'^* = H_3(m^*, \mathsf{C}^*, \mathsf{K}^*, \mathsf{c}_{\text{i}}^*, \mathsf{PID}_{\mathsf{s}}^*, \mathsf{PID}_{\mathsf{r_i}}^*, \mathsf{pk}_{\mathsf{s}}^*, \mathsf{pk}_{\mathsf{r_i}}^*)$ and $h_3^* \neq h_3'^*$ i.e. $f^* \neq f'^*$. In the end, the $\mathcal{A}_{\text{I}}$ outputs another forged $\mathsf{CT}'^* = (f'^*, \mathsf{c}_{\text{i}}^*, S_{\text{i}}'^*, U^*)$ on the same m^*. Finally, $\mathcal{C}$ will have two valid signatures:

$$S_{\text{i}}^* = r^{*-1}(f^* + w.\mathsf{d}_{\mathsf{s}}^*.x_{\mathsf{s}}) \tag{5}$$

$$S_{\text{i}}'^* = r'^{*-1}(f'^* + w.\mathsf{d}_{\mathsf{s}}'^*.x_{\mathsf{s}}) \tag{6}$$

where $r^* = r'^*$ and $\mathsf{d}_{\mathsf{s}}^* = \mathsf{d}_{\mathsf{s}}'^*$. From the Equations 8 and 9 above, $\mathcal{C}$ can extract ϕ as follows:

$$\phi = r^{*-1}(f'^* - f^*) + (S_{\text{i}}^* - S_{\text{i}}'^*)(r^{*-1}(w.x_{\mathsf{s}}(Q_{\mathsf{PID}_{\mathsf{s}}}^* - Q_{\mathsf{PID}_{\mathsf{s}}}'^*))^{-1}$$

Given that, the $\mathcal{C}$ solves the ECDL $Q = \phi.P$ with the advantage ϵ':

$$\epsilon' \geq \epsilon \left(\frac{1}{qH_1} + \frac{1}{qH_2}\right)\left(\frac{1}{qH_1}\right)\left(1 - \frac{1}{q_{\mathsf{ppk}}}\right)\left(1 - \frac{q_{\mathsf{usc}}}{2^k}\right) \tag{7}$$

Theorem 4. *The proposed scheme is EUF-AMCLHS-CMA-II secure under the ROM based on the hardness of the ECDL assumption. Assume that the hash functions $H_l\{l = 1, 2, 3\}$ are ROM and $\mathcal{A}_{\text{II}}$ can make a number of q_{H_l} queries to the ROM, including q_{pk}, q_{ppk}, q_{pr}, q_{sv}, q_{sc}, and q_{usc}. Suppose that the EUF-AMCLHS-CMA-II adversary $\mathcal{A}_{\text{II}}$ has a non-negligible advantage ϵ in winning the game then, there is $\mathcal{C}$ that can solve the ECDL with the non-negligible advantage ϵ'.*

Proof. Given two random instances of the ECDL $(Q, P) \in \mathbb{G}_1$ where $Q = \pi.P$ where $\pi \in \mathbb{Z}_q^*$. The $\mathcal{C}$ has to find π by interacting with the $\mathcal{A}_{\text{II}}$ such that $Q = \pi P$.

Phase-1: Phase-1 queries are similar to Theorem 2, respectively. The $\mathcal{C}$ keeps a list L_l of q_{H_l} to record the responses.
Setup: The $\mathcal{C}$ runs the setup algorithm to generate $\mathsf{PP} = \{\mathbb{G}_1, \mathbb{G}_2, P, \hat{e}, q, H_0, H_1, H_2, H_3\}$. The $\mathcal{C}$ sets $\mathsf{mpk} = \theta^{-1}.P$ and sends PP and mpk to the$\mathcal{A}_{\text{II}}$.
Phase-2: The adversary $\mathcal{A}_{\text{II}}$ asks a number of queries in an adaptive manner, including q_{pk}, q_{ppk}, q_{pr}, q_{sv}, and q_{sc}. An initially empty list L_{pk} is maintained by the $\mathcal{C}$. The $\mathcal{C}$ stores the public key and secret value information in L_{pk}. $\mathcal{C}$ responds to all queries as in Phase-2 of Theorem 2, except the responds to secret value extract query q_{sv} as follows:

1. q_{sv}: Upon receiving the query for PID, the $\mathcal{C}$ checks if $\mathsf{PID} = \mathsf{PID}_{\mathsf{s}}^*$. If it holds, the $\mathcal{C}$ aborts because in this case, the PID is a target identity. Otherwise, it checks if x_{i} exists in the list L_{pk} $(\mathsf{PID}_{\mathsf{i}}, \mathsf{d}_{\mathsf{i}}, \mathsf{pk}_{\mathsf{i}}, x_{\mathsf{i}})$. If it exists, the $\mathcal{C}$ returns x_{i} to $\mathcal{A}_{\text{II}}$. Otherwise, $\mathcal{C}$ computes $\mathsf{pk}_{\mathsf{i}} = \pi P$ where $x_{\mathsf{i}} = \pi \in \mathbb{Z}_q^*$ and adds the tuple $(\mathsf{PID}_{\mathsf{i}}, \mathsf{d}_{\mathsf{i}}, \mathsf{pk}_{\mathsf{i}}, x_{\mathsf{i}})$ in L_{pk} and returns x_{i} to $\mathcal{A}_{\mathcal{II}}$.

Forgery: Taking the target sender $\mathsf{PID}_{\mathsf{s}}^*$ and designated receiver's $\mathsf{PID}_{\mathsf{r_i}}$, the adversary outputs a forged $\mathsf{CT}^* = (f^*, \mathsf{c}_{\mathsf{i}}^*, S_{\mathsf{i}}^*, U^*)$ on a m^* which is the valid signcrypted ciphertext and is not the result of signcryption oracle. If $\mathsf{PID} \neq \mathsf{PID}_{\mathsf{s}}^*$, the $\mathcal{C}$ returns $\perp$. Otherwise, the $\mathcal{C}$ extracts the list L_{pk} for the record $(\mathsf{PID}_{\mathsf{i}}^*, \mathsf{d}_{\mathsf{i}}^*, \mathsf{pk}_{\mathsf{i}}^*, x_{\mathsf{i}}^*)$ and L_3 for the record $(m^*, \mathsf{C}^*, \mathsf{K}^*, \mathsf{c}_{\mathsf{i}}^*, f^*)$.
According to the Forking Lemma, the $\mathcal{C}$ replays the $\mathcal{A}_{\text{II}}$ with the same random tape but distinct attributes from H_1 and H_3. It implies that, $h_1^* = H_1(\mathsf{mpk}, \mathsf{PID}_{\mathsf{i}}^*)$ i.e. , $h_1'^* = H_1(\mathsf{mpk}, \mathsf{PID}_{\mathsf{i}}^*)$ and $h_1^* \neq h_1'^*$ i.e. $Q_{\mathsf{PID_s}}^* \neq Q_{\mathsf{PID_s}}'^*$. Similarly, $h_3^* = H_3(m^*, \mathsf{C}^*, \mathsf{K}^*, \mathsf{c}_{\mathsf{i}}^*, \mathsf{PID}_{\mathsf{s}}^*, \mathsf{PID}_{\mathsf{r_i}}^*, \mathsf{pk}_{\mathsf{s}}^*, \mathsf{pk}_{\mathsf{r_i}}^*)$, $h_3'^* = H_3(m^*, \mathsf{C}^*, \mathsf{K}^*, \mathsf{c}_{\mathsf{i}}^*, \mathsf{PID}_{\mathsf{s}}^*, \mathsf{PID}_{\mathsf{r_i}}^*, \mathsf{pk}_{\mathsf{s}}^*, \mathsf{pk}_{\mathsf{r_i}}^*)$, and $h_3^* \neq h_3'^*$ i.e. $f^* \neq f'^*$. In the end, the $\mathcal{A}_{\text{II}}$ outputs another forged $\mathsf{CT}'^* = (f'^*, \mathsf{c}_{\mathsf{i}}^*, S_{\mathsf{i}}'^*, U^*)$ on the same m^*. Finally, $\mathcal{C}$ will have two valid signatures:

$$S_{\mathsf{i}}^* = r^{*-1}(f^* + w.\mathsf{d}_{\mathsf{s}}^* x_{\mathsf{s}}^*) \tag{8}$$

$$S_{\mathsf{i}}'^* = r'^{*-1}(f'^* + w.\mathsf{d}_{\mathsf{s}}^* x_{\mathsf{s}}'^*) \tag{9}$$

where $r^* = r'^*$ and $x_{\mathsf{s}}^* = x_{\mathsf{s}}'^*$. From the Equations 8 and 9 above, the $\mathcal{C}$ can extract π as follows:

$$\pi = r^{*-1}(f'^* - f^*) + (S_{\mathsf{i}}^* - S_{\mathsf{i}}'^*)(r^{*-1}(w.\mathsf{mpk}.(Q_{\mathsf{PID_s}}^* - Q_{\mathsf{PID_s}'}^*))^{-1}$$

Given that, the $\mathcal{C}$ solves the ECDL $Q = \pi P$ with the advantage:

$$\epsilon' \geq \epsilon \left(\frac{1}{qH_1} + \frac{1}{qH_2}\right)\left(\frac{1}{qH_1}\right)\left(1 - \frac{1}{q_{\mathsf{sv}}}\right)\left(1 - \frac{q_{\mathsf{usc}}}{2^k}\right) \tag{10}$$

Anonymity: In the proposed scheme, each user uses the PID to communicate with each other instead of the $\mathsf{ID_R}$. Therefore, the users will be able to validate the identity but cannot detect or modify the $\mathsf{ID_R}$. Each user create a PID from the $\mathsf{ID_R}$ as

$\mathsf{PID} = \mathsf{ID_R} \oplus (\alpha.\mathsf{pk_{RA}})$ where α is chosen randomly and $R = \alpha.P$. In order to obtain the $\mathsf{ID_R}$, the attacker need to calculate the $\mathsf{ID_R} = \mathsf{PID} \oplus (R.v)|(\Delta T)$. However, it is based on ECDL hard assumption therefore, the attacker will not be able to compute $\mathsf{ID_R}$. Moreover, to validate the $\mathsf{ID_R}$ of the user, RA will verify the $\mathsf{ID_R}$ using its private key as $\mathsf{ID_R} = H_0(R.v)$. Since, only RA knows its private key, no else could generate the $\mathsf{ID_R}$. The scheme also provide the conditional anonymity i.e. in case of a dispute, the RA will expose the $\mathsf{ID_R}$ of the user. Furthermore, ΔT shows the validity period of the PID which is defined by RA. Upon receiving message, the receiver will verify ΔT from the RA. If the ID is not valid for the defined period of time, message will not be accepted.

Unlinkability: Unlinkability is a security property that ensures that the user actions cannot be linked to the user identity, thereby preserving their privacy. Unlinkability is typically achieved using pseudo-anonymous identities. In our scheme, we ensure unlinkability by generating PID using the RA. RA generates the PID for each user as $\mathsf{PID} = \mathsf{ID_R} \oplus H_0((\alpha.\mathsf{pk_{RA}})\|(\Delta T))$ which has the validity period of (ΔT). Moreover, the PID contains a random parameter α that cannot be determined by the adversary.

Non-repudiation: Non-repudiation refers to the concept which ensures that a user cannot later deny sending a message by adding some of its unique information to the message. In communication, non-repudiation is typically achieved through the use of signature, in which the sender signs message with its $\mathsf{sk_s}$ and the message is verified using $\mathsf{pk_s}$. By signing message with their $\mathsf{sk_s}$, the sender proves that they sent the message and cannot later deny it since, only the sender knows its $\mathsf{sk_s}$. Similarly, in our scheme, message is signed by the sender with its $\mathsf{sk_s}$ as $S_\mathsf{i} = r^{-1}(f + w.\mathsf{d_s}x_\mathsf{s})$. The message is verified by the receiver using $\mathsf{pk_s}$ as $R_\mathsf{i} = S_\mathsf{i}^{-1}(f.P + w.\mathsf{pk_s}.Z_{1_\mathsf{i}}.Q_{\mathsf{PID}_{r_\mathsf{i}}}^{-1})$. Since, the sender signs message with its $\mathsf{sk_s}$ that only sender knows, it cannot deny sending a message. Hence, our scheme achieves non-repudiation.

Forward Security: Forward security is a property that ensures the security of a message even if the sk of the user is compromised. The adversary cannot extract previously exchanged messages during communication and the messages remain secure. It is typically achieved by using key agreement protocol which generate a new key for each session. Even if the key for one session is compromised, other sessions cannot be exploited by the adversary. In our scheme, the symmetric session key is generated using the sk of the users along with the random parameter $r \in \mathbb{Z}_q^*$. The key is generated as $\psi = H_2(Z_{1_\mathsf{i}}, Z_{2_\mathsf{i}}, Z_{3_\mathsf{i}})$ which is different for each session due to random selection of the parameter r. In this case, even if the sk of the user are exploited, the adversary cannot extract the plaintext message from $f = H_3(m, \mathsf{C}, \mathsf{K}, \mathsf{c_i}, \mathsf{PID_s}, \mathsf{PID}_{r_\mathsf{i}}, \mathsf{sk_s}, \mathsf{pk}_{r_\mathsf{i}})$, since r is always randomly chosen for each session. Therefore, our scheme ensure that the messages remains secure during communication, even if the $\mathsf{sk_s}$ are exploited.

6 Performance Analysis

We compare the computational cost, communication cost, security requirements of the proposed hybrid signcryption scheme with existing multireceiver signcryptions.

The computational overhead for multireceiver schemes is compared with [15,22,16] as shown in Table 1. Among the multireceiver signcryption schemes, Niu et al. [15] has highest computational overhead utilizing total $(2n+4)T_{bp} + T_{pm} + (2n+2)T_e$ oper-

ations with $(2n+4)T_{bp}$ BP operations which are considered as the most expensive and time consuming. Niu et al. [16] require total $(7n+5)T_{pm}$ operations for signcryption and unsigncryption whereas, Yang et al. [22] utilize total $5nT_{pm} + nT_e + nT_{bp}$ operations. As compare to the existing schemes, our proposed scheme require $(n+4)T_{pm} + T_e + (n+1)T_{bp}$ total operations where the signcryption cost is linear with the number of designated receivers while unsigncryption cost remains constant for each designated receiver. Overall, the proposed scheme is of high efficiency.

Furthermore, the length of ciphertext directly affects the communication overhead and system storage of ciphertext transmission, Table 2 shows the comparison results of communication cost in terms of ciphertext length and complexity of communication. Nui et al. [15] contain $2n$ ciphertext element in $\mathbb{G}_1$ with higher complexity of communication. Yang et al. [22] contains n ciphertext element in $\mathbb{G}_1$ with the same complexity as [15]. Further, [16] contain $(n+1)$ and $2n$ ciphertext elements in $\mathbb{G}_1$ for PMRCHS and CPHAS schemes, respectively which significantly increases communication overhead. Among the existing scheme listed in Table 2 our proposed scheme has less communication overhead containing $2n$ ciphertext elements in $\mathbb{G}_1$ with the lower complexity of communication for signcryption where as the complexity remains constant for unsigncryption.

Table 1. Computational Overhead Comparison with Multireceiver Schemes

Schemes	**Signcryption**	**Unsigncryption**	**Total**
Niu et al. [15]	$2nT_{bp} + T_{pm} + 2nT_e$	$4T_{bp} + 2T_e$	$(2n+4)T_{bp} + T_{pm} + (2n+2)T_e$
Yang et al. [22]	$2nT_{pm} + nT_e$	$3nT_{pm} + nT_{bp}$	$5nT_{pm} + nT_e + nT_{bp}$
Niu et al. [16]	$(4n+4)T_{pm}$	$(3n+1)T_{pm}$	$(7n+5)T_{pm}$
Our scheme	$(n+1)T_{pm} + nT_{bp}$	$3T_{pm} + T_{bp}$	$(n+4)T_{pm} + (n+1)T_{bp}$

Legend: T_{bp} : Time to execute BP operation, T_{pm} : Time to execute point multiplication operation, T_e : Time to execute an exponentiation operation in $\mathbb{Z}_q^*$

Table 2. Communication Cost

scheme	Ciphertext Length	Complexity of Communication	
		Signcryption	Unsigncryption
Niu et al. [15]	$n\|m\| + 3\|\mathbb{G}_1\| + 2n\|\mathbb{G}_1\|$	$\mathcal{O}(n^2)$	$\mathcal{O}(n)$
Yang et al. [22]	$n\|m\| + n\|\mathbb{G}_1\|$	$\mathcal{O}(n^2)$	$\mathcal{O}(n)$
Niu et al. [16]	(PMRCHS) $\|m\| + \|\mathbb{Z}_q^*\| + (n+1)\|\mathbb{G}_1\|$	$\mathcal{O}(n)$	$\mathcal{O}(n)$
Niu et al. [16]	(CPHAS) $n\|m\| + 2n\|\mathbb{Z}_q^*\| + 2n\|\mathbb{G}_1\|$	$\mathcal{O}(n)$	$\mathcal{O}(n)$
Our scheme	$n\|m\| + \|\mathbb{Z}_q^*\| + 2n\|\mathbb{G}_1\|$	$\mathcal{O}(t)$	$\mathcal{O}(1)$

Legend: n is the number of users, $|m|$ is the plaintext Length, $|\mathbb{Z}_q^*|$ is the length of an element in finite field $\mathbb{Z}_q^*$, $|\mathbb{G}_1|$ is the length of an element in $\mathbb{G}_1$.

In Table 3, we compare the security requirements of existing multireceiver hybrid signcryption schemes [15,22,16] with our scheme in terms of achieving confidentiality,

unforgeability, anonymity, unlinkability, non-repudiation, and forward security. Yang et al. [22] provides confidentiality, unforgeability, and non-repudiation that comes with signcryption however, it fails to fulfill anonymity, unlinkability, and forward security requirements. Niu et al. [16] PKI $\rightarrow$ Certificateless Public Key Cryptography (CLC) completely anonymous multireceiver signcryption (PMRCHS) and CLC $\rightarrow$ PKI heterogeneous aggregate signcryption (CPHAS) achieves confidentiality, unforgeability, and anonymity however, it does not provide unlinkability, non-repudiation, and forward security. Moreover, Niu et al. [15] fulfills each security requirement however, it achieves the security with high computational cost compared to our proposed scheme. Lastly, our scheme achieves all security requirements, including confidentiality, unforgeability, anonymity, unlinkability, non-repudiation, and forward security at a lower computational cost compared to the other schemes listed in Table 3.

Table 3. Security requirements

Schemes	Confidentiality	Unforgeability	Anonymity	Unlinkability	Non-repudiation	Forward Security
Niu et al. [15]	Yes	Yes	Yes	Yes	Yes	Yes
Yang et al. [22]	Yes	Yes	No	No	Yes	No
Niu et al. [16]	Yes	Yes	Yes	No	No	No
Our scheme	Yes	Yes	Yes	Yes	Yes	Yes

7 Conclusion

Our paper introduces a novel mKEM-DEM based AMCLHS scheme for broadcast communication. The proposed scheme is based on the multiple-recipient Key Encapsulation Mechanism (mKEM) and Data Encapsulation Mechanism (DEM), that generates a symmetric key using the public and private key pair of the users. The message is then signcrypted with the previously generated symmetric key and the private key of the sender. We provide a detailed security analysis using q-DBDHI and ECDL hard assumptions, and demonstrate that the scheme is secure against IND-AMCLHS-CCA2 and EUF-AMCLHS-CMA attacks for Type-I and Type-II adversaries. Moreover, in this scheme, each user is assigned a PID to ensure user anonymity. Lastly, we compare our scheme with existing single receiver and multireceiver certificateless hybrid signcryption schemes in terms of computation cost, communication cost, and security requirements. We show that, the proposed scheme has less communication cost and is computationally more efficient, with the signcryption cost linear with the number of designated receivers while the unsigncryption cost remains constant and simultaneously achieves confidentiality, unforgeability, anonymity, unlinkability, non-repudiation, and forward security.

References

1. Al-Riyami, S.S., Paterson, K.G.: Certificateless public key cryptography. In: Laih, C. (ed.) Advances in Cryptology - ASIACRYPT 2003, 9th International Conference on the Theory

and Application of Cryptology and Information Security, Taipei, Taiwan, November 30 - December 4, 2003, Proceedings. Lecture Notes in Computer Science, vol. 2894, pp. 452–473. Springer (2003), https://doi.org/10.1007/978-3-540-40061-5_29
2. Barbosa, M., Farshim, P.: Certificateless signcryption. In: Abe, M., Gligor, V.D. (eds.) Proceedings of the 2008 ACM Symposium on Information, Computer and Communications Security, ASIACCS 2008, Tokyo, Japan, March 18-20, 2008. pp. 369–372. ACM (2008), https://doi.org/10.1145/1368310.1368364
3. Boneh, D., Boyen, X.: Efficient selective-id secure identity-based encryption without random oracles. In: Advances in Cryptology-EUROCRYPT 2004: International Conference on the Theory and Applications of Cryptographic Techniques, Interlaken, Switzerland, May 2-6, 2004. Proceedings 23. pp. 223–238. Springer (2004)
4. Boneh, D., Franklin, M.: Identity-based encryption from the weil pairing. In: Advances in Cryptology—CRYPTO 2001: 21st Annual International Cryptology Conference, Santa Barbara, California, USA, August 19–23, 2001 Proceedings. pp. 213–229. Springer (2001)
5. Cui, B., Lu, W., He, W.: A new certificateless signcryption scheme for securing internet of vehicles in the 5g era. Security and Communication Networks **2022** (2022)
6. Dent, A.W.: Hybrid signcryption schemes with insider security. In: Boyd, C., Nieto, J.M.G. (eds.) Information Security and Privacy, 10th Australasian Conference, ACISP 2005, Brisbane, Australia, July 4-6, 2005, Proceedings. Lecture Notes in Computer Science, vol. 3574, pp. 253–266. Springer (2005). https://doi.org/10.1007/11506157_22
7. Dent, A.W.: Hybrid signcryption schemes with outsider security. In: Zhou, J., López, J., Deng, R.H., Bao, F. (eds.) Information Security, 8th International Conference, ISC 2005, Singapore, September 20-23, 2005, Proceedings. Lecture Notes in Computer Science, vol. 3650, pp. 203–217. Springer (2005). https://doi.org/10.1007/11556992_15
8. Fu, M., Gu, X., Dai, W., Lin, J., Wang, H.: Secure multi-receiver communications: Models, proofs, and implementation. In: Wen, S., Zomaya, A.Y., Yang, L.T. (eds.) Algorithms and Architectures for Parallel Processing - 19th International Conference, ICA3PP 2019, Melbourne, VIC, Australia, December 9-11, 2019, Proceedings, Part I. Lecture Notes in Computer Science, vol. 11944, pp. 689–709. Springer (2019). https://doi.org/10.1007/978-3-030-38991-8_45
9. Gong, B., Wu, Y., Wang, Q., Ren, Y., Guo, C.: A secure and lightweight certificateless hybrid signcryption scheme for internet of things. Future Gener. Comput. Syst. **127**, 23–30 (2022). https://doi.org/10.1016/j.future.2021.08.027
10. Hongzhen, D., Qiaoyan, W., Shanshan, Z., Mingchu, G.: A pairing-free certificateless signcryption scheme for vehicular ad hoc networks. Chinese Journal of Electronics **30**(5), 947–955 (2021)
11. Kasyoka, P.N., Kimwele, M.W., Mbandu, A.S.: Efficient certificateless signcryption scheme for wireless sensor networks in ubiquitous healthcare systems. Wirel. Pers. Commun. **118**(4), 3349–3366 (2021). https://doi.org/10.1007/s11277-021-08183-y
12. Li, F., Shirase, M., Takagi, T.: Certificateless hybrid signcryption. In: Bao, F., Li, H., Wang, G. (eds.) Information Security Practice and Experience, 5th International Conference, ISPEC 2009, Xi'an, China, April 13-15, 2009, Proceedings. Lecture Notes in Computer Science, vol. 5451, pp. 112–123. Springer (2009). https://doi.org/10.1007/978-3-642-00843-6_11
13. Li, X., Jiang, C., Du, D., Wang, S., Fei, M., Wu, L.: A novel efficient signcryption scheme for resource-constrained smart terminals in cyber-physical power systems. CoRR **abs/2212.04198** (2022). https://doi.org/10.48550/arXiv.2212.04198
14. Malone-Lee, J.: Identity-based signcryption. IACR Cryptol. ePrint Arch. p. 98 (2002), http://eprint.iacr.org/2002/098
15. Niu, S., Niu, L., Yang, X., Wang, C., Jia, X.: Heterogeneous hybrid signcryption for multi-message and multi-receiver. PloS one **12**(9), e0184407 (2017)

16. Niu, S., Shao, H., Hu, Y., Zhou, S., Wang, C.: Privacy-preserving mutual heterogeneous signcryption schemes based on 5g network slicing. IEEE Internet Things J. **9**(19), 19086–19100 (2022). https://doi.org/10.1109/JIOT.2022.3163607
17. Peng, C., Chen, J., Obaidat, M.S., Vijayakumar, P., He, D.: Efficient and provably secure multireceiver signcryption scheme for multicast communication in edge computing. IEEE Internet Things J. **7**(7), 6056–6068 (2020). https://doi.org/10.1109/JIOT.2019.2949708
18. Qiu, J., Fan, K., Zhang, K., Pan, Q., Li, H., Yang, Y.: An efficient multi-message and multi-receiver signcryption scheme for heterogeneous smart mobile iot. IEEE Access **7**, 180205–180217 (2019). https://doi.org/10.1109/ACCESS.2019.2958089
19. Selvi, S.S.D., Vivek, S.S., Rangan, C.P.: Certificateless KEM and hybrid signcryption schemes revisited. IACR Cryptol. ePrint Arch. p. 462 (2009), http://eprint.iacr.org/2009/462
20. Selvi, S.S.D., Vivek, S.S., Shukla, D., Rangan, C.P.: Efficient and provably secure certificateless multi-receiver signcryption. In: Baek, J., Bao, F., Chen, K., Lai, X. (eds.) Provable Security, Second International Conference, ProvSec 2008, Shanghai, China, October 30 - November 1, 2008. Proceedings. Lecture Notes in Computer Science, vol. 5324, pp. 52–67. Springer (2008). https://doi.org/10.1007/978-3-540-88733-1_4
21. Smart, N.P.: Efficient key encapsulation to multiple parties. In: Blundo, C., Cimato, S. (eds.) Security in Communication Networks, 4th International Conference, SCN 2004, Amalfi, Italy, September 8-10, 2004, Revised Selected Papers. Lecture Notes in Computer Science, vol. 3352, pp. 208–219. Springer (2004). https://doi.org/10.1007/978-3-540-30598-9_15
22. Yang, Y., He, D., Vijayakumar, P., Gupta, B.B., Xie, Q.: An efficient identity-based aggregate signcryption scheme with blockchain for iot-enabled maritime transportation system. IEEE Trans. Green Commun. Netw. **6**(3), 1520–1531 (2022). https://doi.org/10.1109/TGCN.2022.3163596
23. Yin, A., Liang, H.: Certificateless hybrid signcryption scheme for secure communication of wireless sensor networks. Wirel. Pers. Commun. **80**(3), 1049–1062 (2015). https://doi.org/10.1007/s11277-014-2070-y
24. Yu, X., Zhao, W., Tang, D.: Efficient and provably secure multi-receiver signcryption scheme using implicit certificate in edge computing. J. Syst. Archit. p. 102457 (2022). https://doi.org/10.1016/j.sysarc.2022.102457
25. Yu, Y., Yang, B., Huang, X., Zhang, M.: Efficient identity-based signcryption scheme for multiple receivers. In: Xiao, B., Yang, L.T., Ma, J., Müller-Schloer, C., Hua, Y. (eds.) Autonomic and Trusted Computing, 4th International Conference, ATC 2007, Hong Kong, China, July 11-13, 2007, Proceedings. Lecture Notes in Computer Science, vol. 4610, pp. 13–21. Springer (2007), https://doi.org/10.1007/978-3-540-73547-2_4
26. Zhang, W., Zhang, Y., Guo, C., An, Q., Guo, Y., Liu, X., Zhang, S., Huang, J.: Certificateless hybrid signcryption by a novel protocol applied to internet of things. Computational Intelligence and Neuroscience **2022** (2022)
27. Zheng, Y.: Digital signcryption or how to achieve cost(signature & encryption) $<<$ cost(signature) + cost(encryption). In: Jr., B.S.K. (ed.) Advances in Cryptology - CRYPTO '97, 17th Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 1997, Proceedings. Lecture Notes in Computer Science, vol. 1294, pp. 165–179. Springer (1997). https://doi.org/10.1007/BFb0052234
28. ZHOU, C.: Certificateless signcryption scheme without random oracles. Chinese Journal of Electronics **27**(5), 1002–1008 (2018)