A Multi-receiver Certificateless Signcryption (MCLS) Scheme

Alia Umrani*, Apurva K Vangujar and Paolo Palmieri

School of Computer Science & IT,
University College Cork, Ireland
a.umrani@cs.ucc.ie, a.vangujar@cs.ucc.ie,
p.palmieri@cs.ucc.ie

Abstract. User authentication and message confidentiality are the basic security requirements of high-end applications such as multicast communications and distributed systems. Several sign-then-encrypt schemes have been proposed to offer these security requirements however, such techniques require more computation cost and communication overhead as compared to signcryption techniques. Signcryption accomplishes both digital signature and encryption function in a single logical step and at a much lower cost than sign-then-encrypt schemes. Similarly, anonymous multi-receiver encryption has recently risen in prominence in distributed settings, where the same messages are sent to multiple receivers but the identity of each receiver remains private. Among the multi-receiver certificateless encryption schemes that have been introduced, Hung *et al.* [11] proposed an efficient Anonymous Multi-receiver Certificateless Encryption (AMCLE) scheme ensuring confidentiality and anonymity based on bilinar pairing, and the security is based on Indistinguishability against Chosen Ciphertext Attack (IND-CCA) and anonymous IND-CCA. In this paper, we substantially extend Hung *et al.*'s AMCLE scheme to a Multi-receiver Certificateless Signcryption (MCLS) scheme, which provides authentication and confidentiality and also introduces the public verifiability as an additional feature. We show that, as compared to Hung *et al.*'s encryption scheme, our signcryption scheme requires only two additional multiplication operations for the signcryption and unsigncryption phases. Additionally, the signcryption cost is linear with the number of designated receivers, while the unsigncryption cost remains constant. We compare the results with other existing single-receiver and multi-receiver signcryption schemes and show that the our proposed MCLS scheme is more efficient for single and multi-receiver signcryption schemes while providing exemption from the key escrow problem and working in certificateless public key settings.

Keywords: Signcryption · Certificateless · Multi-receiver · Public Key Cryptography

* Alia Umrani and Apurva K Vangujar are supported by PhD scholarships funded by the Science Foundation Ireland Centre under Grant No. SFI 18/CRT/6222

1 Introduction

A message in digital communication must be secure in terms of authentication, confidentiality, and integrity. Encryption schemes are generally used for confidentiality, whereas Digital Signature (DS) schemes are used for authentication, integrity, and non-repudiation. As a result, DS and Public Key Encryption (PKE) are fundamental requirements for achieving security. However, in sign-then-encrypt scheme, signing and then encrypting a message has a high computational cost. Signcryption, on the other hand, not only signs the message as the traditional approach requires but also encrypts it in a single step. This ensures that the message is meaningless to anyone but the intended recipient, who can also verify the sender's Identity (ID) and the message's integrity. Signcryption is more attractive than the sign-then-encrypt procedure because it requires less computation time and has a lower message expansion rate [26]. Furthermore, some Authenticated Encryption (AE) provides security against both Chosen Ciphertext Attack (CCA) and Chosen Plaintext Attack (CPA), and signcryption provides AE and thus CCA and CPA security.

Zheng *et al.* [25] proposed the first signcryption scheme, which combines DS and PKE to provide authentication, confidentiality, and non-repudiation at a lower cost than sign-then-encrypt operations separately. Malone-Lee [17] proposed the first ID-based signature scheme to provide public verifiability and forward security. Following that, several ID-based encryption schemes were proposed. Chen *et al.* [6] and Chow *et al.* [8] proposed ID-based signcryption schemes, respectively, to demonstrate public verifiability, forward security, unlinkability, and anonymity. However, ID-based cryptography has an inherent key escrow problem in which a malicious Key Generation Center (KGC) compromises the entity's private key. To solve the key escrow problem, Al-Riyami *et al.* proposed the concept of certificateless Public Key Cryptography (PKC) [1]. In certificateless-PKC, the KGC generates the partial private key for the user, and the full private key pair is the combination of the user's secret value and the partial private key. The above signcryption schemes are based on a single receiver, which is insufficient for broadcast communication. For example, to send an identical message to multiple receivers, a sender must encrypt the message for each designated receiver, resulting in poor performance. Yu *et al.* [24] proposed the first multi-receiver signcryption scheme based on ID-based PKC in which the message is encrypted for n designated receivers. The security is demonstrated through the Random Oracle Model (ROM) and the Computational Diffie-Hellman (CDH) assumption. Later, Hung *et al.* [11] proposed an efficient anonymous multi-receiver certificateless encryption (AMCLE) scheme based on Bilinear Pairing (BP). The scheme proves Indistinguishability against Chosen Ciphertext Attack (IND-CCA) and Anonymous Indistinguishability against Chosen Ciphertext Attack (ANON-CCA). The encryption cost in this scheme is linear with the number of designated receivers, while the decryption cost is constant for each designated receiver. Moreover, several publicly verifiable signcryption schemes have also been introduced [5,14,10]. Public verifiability enables any third party to verify the authenticity and integrity of a

signcrypted message without requiring knowledge of the private keys possessed by either the sender or the receiver. This capability significantly enhances the transparency and trustworthiness of the signcryption scheme [8].

In this paper, we extend the functionalities of Hung *et al.*'s AMCLE scheme into a Multi-receiver Certificateless Signcryption (MCLS) scheme. The MCLS scheme demonstrates security against IND-CCA for confidentiality and Existential Unforgeability against Chosen Message Attack (EUF-CMA). In addition, we incorporate a public verifiability feature into our scheme which is not provided in Hung *et al.*'s AMCLE scheme. Public verifiability enables any third party to verify the authenticity of the signcrypted message without knowing the private key. Furthermore, as signcryption focuses on authentication and confidentiality, we omit the ANON-CCA proof, which remains the same for the encryption scheme as in Hung *et al.* [11]. This signcryption scheme demonstrates efficient computation cost, with the cost of signcryption being linear with the number of designated receivers and the cost of unsigncryption remaining constant for each designated receiver. In comparison to the other existing signcryption techniques, the MCLS scheme overcomes the key escrow problem and works in a multi-receiver certificateless public key setting. Specifically, the main contributions are as follows:

1. We design a Multi-receiver Certificateless Signcryption (MCLS) scheme that significantly enhances the functionalities of the existing AMCLE scheme [11].
2. We provide a security proof in a ROM under the CDH and Decisional Bilinear Diffie-Hellman Inversion (DBDHI) assumptions which claim that the proposed scheme can achieve authentication by demonstrating EUF-CMA. We also show that, along with authentication, our scheme simultaneously achieves public verifiability.
3. We evaluate the performance of the proposed MCLS scheme and present a comparison with other existing signcryption schemes.

The reminder of this paper is described as follows. Sec. 2 reviews the research related to the scheme. Sec. 3 introduces the fundamentals of BP as well as mathematical assumptions. Sec. 4 describes the framework and security model in the MCLS scheme. Sec. 5 shows the construction of the MCLS scheme. In Sec. 6, we provide a security proof in ROM and in Sec. 7, we compare it to existing schemes. Sec. 8 concludes this paper.

2 Related Work

Barbosa and Farshim [2] proposed the first certificateless based signcryption scheme based on BP that provides authentication and confidentiality and is secure against insider attacks in a ROM. To prove the security, it employs Gap-Bilinear Diffie-Hellman (GBDH), Decisional Bilinear Diffie-Hellman (DBDH), and Computational Bilinear Diffie-Hellman (CBDH) assumptions and is shown to be IND-CPA and strong UF-CMA secure. Selvi *et al.* [21] proposed an efficient and provably secure certificateless multi-receiver signcryption scheme.

The scheme is based on the strong DH assumption, the Collusion Attack Algorithm with k-Traitors (k-CAA), Modified BDHI for k-Values (k-mBDHI), and the GBDH assumption. The scheme employs BP operations and compares the efficiency with ID-based schemes. The scheme proposed by Selvi *et al.* [21] is not secure against external adversaries and is improved as enhanced certificateless multi-receiver signcryption [20]. However, Miao *et al.* [18] proposed a cryptanalysis of a certificateless multi-receiver signcryption scheme, in which the authors demonstrated that the scheme proposed in [20] is insecure against an external adversary and presented an attack on [21]'s enhanced scheme. They demonstrate that the adversary can first replace the sender's public key and then generate ciphertext on the sender's behalf. Islam *et al.* [12] proposed an anonymous and provably secure certificateless multi-receiver encryption scheme which uses an elliptic curve cryptography based technique under the CDH assumption. In this scheme, the encryption cost is quadratic with the number of receivers, whereas the decryption cost is linear with the number of receivers. However, its security proof has the drawback that the simulator failed to successfully generate the challenge ciphertext and thus the security did not hold, and the scheme has a key escrow problem. To overcome the key escrow problem and provide more efficiency, Hung *et al.* [11] proposed an ACMLE scheme that provides confidentiality and anonymity. This scheme uses BP under the BDDH, GBDH, and CDH assumptions. To prove confidentiality, the scheme defines the IND-CLME-CCA and to achieve anonymity, the authors present the ANON-CLME-CCA. The proposed AMCLE scheme provides a constant decryption cost, which means that the required decryption cost of each receiver is independent of the number of receivers as compared to Islam *et al.*'s scheme. However, the security proof cannot cover all possible attacks due to some restrictions on attackers. Guo *et al.* [9] proposed an efficient certificateless ring signcryption scheme with conditional privacy preservation. The scheme employs a certificateless cryptographic technique and compares the results to ID-based signature schemes. Similarly, several publicly verifiable signcryption schemes have been introduced.

Chaudhry *et al.* [5] proposed an efficient signcryption scheme with authentication, forward security, and public verifiability based on hyper elliptic curve. Karati *et al.* [14] introduced provably secure and generalized signcryption scheme with public verifiability for secure data transmission between resource-constrained IoT devices. The scheme achieves authentication and confidentiality using strong DH and BDHI assumptions and is based on certificateless PKC. Similarly, Hu *et al.* [10] proposed a sanitizable signcryption scheme with public verifiability via chameleon hash function. The scheme utilizes chameleon hash function as trapdoor commitment for signcryption and proves security in the ROM. In this paper, we expand Hung *et al.*'s scheme and propose an efficient multi-receiver certificateless signcryption which provides authentication, confidentiality, and public verifiability.

3 Preliminaries

3.1 Bilinear Pairing

The BP definition is adopted from [3].

Definition 1. *Let G be cyclic additive and multiplicative group, respectively, over a prime order q. A BP is a map: $\hat{e} : G \times G \rightarrow G$ which satisfies the following properties:*

- *Bilinearity: For any $P \in G$, $\hat{e}(aP, bP) = \hat{e}(P, P)^{ab}$ where $a, b \in Z_q^*$.*
- *Computability: For some $P \in G$, $\hat{e}(P, P)$ can be efficiently computed.*
- *Non-degenerate: For some $P \in G$, $\hat{e}(P, P) \neq 1$.*

3.2 Assumptions

Decisional Bilinear Diffie-Hellman (DBDH) Assumption. The definition of DBDH assumption is according to Chow *et al.* [8].

Definition 2. *Given $P, aP, bP \in G$ and $R \in G$, the DBDH assumption holds if no Probabilistic Polynomial Time (PPT) algorithm with non-negligible advantage (ϵ) can decide whether $R = \hat{e}(P, P)^{ab}$ or not. The ϵ of an Adversary $\mathcal{A}$ is given as following:*

$$\epsilon^{DBDH} = Pr\left[\mathcal{A}(P, aP, bP, \hat{e}(P, P)^{ab}) = 1\right] - Pr\left[\mathcal{A}(P, aP, bP, R) = 1\right] \quad (1)$$

Computational Diffie-Hellman (CDH) Assumption. The definition of CDH assumption is taken from Joux *et al.* [13].

Definition 3. *Let $P, aP, bP \in G$, the CDH assumption holds if no PPT algorithm with non-negligible ϵ can compute abP. The ϵ of $\mathcal{A}$ is defined as*

$$\epsilon^{CDH} = Pr\left[\mathcal{A}(P, aP, bP) = abP\right] \quad . \quad (2)$$

GAP-Bilinear Diffie-Hellman (GBDH) Assumption. The GBDH assumption definition is according to Cha *et al.*[4].

Definition 4. *On input $P, aP, bP, cP \in G_1$ and $R \in G_2$ where $G_1 = G_2$, the GBDH assumption holds if no PPT algorithm with non-negligible ϵ can compute $\hat{e}(P, P)^{abc}$, where $DBDH(P, aP, bP, cP, R) = 1$ if $\hat{e}(P, P)^{abc} = R$ and 0 otherwise. The ϵ of $\mathcal{A}$ is defined as follows:*

$$\epsilon^{GBDH} = Pr\left[\mathcal{A}(P, aP, bP, cP) = \hat{e}(P, P)^{abc}\right] \quad (3)$$

Decisional Bilinear Diffie-Hellman Inversion (DBDHI) Assumption. DBDHI assumption is adopted from Chen *et al.* [7].

Definition 5. *Given input* $P, aP, bP, cP \in G_1$ *and* $R \in G_2$ *where* $G_1 = G_2$*, the DBDHI assumption holds if no PPT algorithm with non-negligible* ϵ *can decide whether* $R = \hat{e}(P,P)^{ab^{-1}c}$ *or not. The* ϵ *of* $\mathcal{A}$ *is given as below:*

$$\epsilon^{DBDHI} = Pr\left[\mathcal{A}(P, aP, bP, cP, \hat{e}(P,P)^{ab^{-1}c}) = 1\right] - Pr\left[\mathcal{A}(P, aP, bP, cP, R) = 1\right] \tag{4}$$

4 Framework and Security Model

4.1 Framework

This paper adopts the certificateless signcryption scheme framework from AMCLE scheme [11]. The scheme has two roles; KGC and n number of users (a sender and t receivers) where $t < n$. The AMCLE scheme allows a sender to generate a signcrypted ciphertext CT_i where $1 \leq i \leq t$ of a message m for t designated receivers. The sender encrypts a m with receiver's public key pk_{r_i} and signs it with its private key sk_s and, sends a CT_i to receivers. The designated receiver decrypts the m with its private key sk_{r_i} and verifies the signature S_i with sender's public key pk_s. Moreover, any third party can verify the authenticity of the message without having the knowledge of private keys of either sender or receiver. The scheme consists of seven polynomial-time algorithms as listed below:

1. **Setup**: On input security parameter 1^λ, the KGC runs this algorithm to generate a master secret key s, master public key P_{pub}, and public parameters PP. The PP is given as input to other algorithms.
2. **Partial Private Key**: Taking s and user $ID \in \{0,1\}^*$ as input, the KGC generates a partial private key D.
3. **Set Secret Value**: This algorithm takes user ID as input and outputs a secret value x.
4. **Set Private Key**: Each user takes (D, x) as input and outputs a full private key sk.
5. **Set Public Key**: Taking secret value x as input, each user outputs a public key pk.
6. **Signcryption**: On input a plaintext message m, sender's private key sk_s, and receivers' public key pk_{r_i} where $1 \leq i \leq t$ and $t < n$, a sender generates a CT_i.
7. **Unsigncryption**: The receiver takes CT_i, sender's public key pk_s, and designated receivers' private key sk_{r_i} as input, and retrieves m or "rejects".

4.2 Security Model

For authentication, we propose and define EUF-CMA. In Def. 6 (Game-I), we define the IND-CCA for Type-I adversary ($\mathcal{A}_{\mathrm{I}}$) and Type-II adversary ($\mathcal{A}_{\mathrm{II}}$) and for confidentiality, we define IND-CCA from Hung *et al.*'s scheme. In Def. 7 (Game-II), we define the EUF-CMA for $\mathcal{A}_{\mathrm{I}}$ and $\mathcal{A}_{\mathrm{II}}$ as follows:

1. $\mathcal{A}_{\mathrm{I}}$: $\mathcal{A}_{\mathrm{I}}$ is considered a common user who cannot access master secret key but can replace the user public key with a value of his/her own choice. It is not allowed to ask a partial private key query for any of the challenger identities.
2. $\mathcal{A}_{\mathrm{II}}$: $\mathcal{A}_{\mathrm{II}}$ is considered an insider who cannot make public key replace query for the challenge ID and is not allowed to make secret value extract queries. If the public key replace query has been done for a target identity as $ID^*_{r_i}$, then the secret value extract query for $ID^*_{r_i}$ is not allowed.

Definition 6. *The IND-CCA requires that there exists no PPT $\mathcal{A}$ which could distinguish ciphertexts. The advantage of $\mathcal{A}$ is defined as the probability that $\mathcal{A}$ wins the game.*

Proof. **Game-I**: Game-I is an interaction between the challenger $\mathcal{B}$ and $\mathcal{A}$ as follows:

1. **Setup**: In this algorithm, $\mathcal{B}$ generates the s, P_{pub}, and PP. The $\mathcal{B}$ sends PP to $\mathcal{A}$.
2. **Phase-1**: The $\mathcal{A}$ outputs i target identities denoted by $ID^*_{r_i}$ for $1 \leq i \leq t$ where $t < n$. The $\mathcal{A}$ asks $\mathsf{q}_{H_j}\{j = 0, 1, \cdots, 5\}$ hash queries, q_p public key retrieve query, q_r public key replace query, q_e partial private key query, q_s secret value extract query, q_{sc} signcryption query, and q_{usc} unsigncryption query. The $\mathcal{B}$ sets empty lists PK^{list} to record public and private key values. The response to each query is defined as follows:
   - q_p: $\mathcal{B}$ receives the query for ID and runs the **Set Secret Value** algorithm to generate a x, and then performs the **Set Public Key** algorithm to return the pk to $\mathcal{A}$.
   - q_r: Upon receiving such query, $\mathcal{B}$ replaces the pk of the user with pk^* and records the replacement.
   - q_e: Given ID, $\mathcal{B}$ checks if $ID = ID^*_{r_i}$. If it does, $\mathcal{B}$ aborts. Otherwise, it fetches the D from PK^{list} and returns. If it does not exist, $\mathcal{B}$ runs the **Partial Private Key** algorithm to return D.
   - q_s: Upon such query, $\mathcal{B}$ searches the PK^{list} for x and returns. If it does not exists, $\mathcal{B}$ runs the **Set Secret Value** algorithm to return x to $\mathcal{A}$.
   - q_{sc} On input (m, sk_s, pk_{r_i}), $\mathcal{B}$ checks if $ID = ID^*_{r_i}$, if it is not, $\mathcal{B}$ runs normal **Signcryption** by fetching values from PK^{list}. Otherwise, it performs the **Signcryption** operation to generate the CT_i and returns to $\mathcal{A}$.
   - q_{usc}: Upon receiving CT_i, $\mathcal{B}$ checks if $ID = ID^*_{r_i}$, if it does not, $\mathcal{B}$ runs normal **Unsigncryption** by taking values from PK^{list}. Otherwise, it performs the **Unsigncryption** operation to generate m.

3. **Challenge**: $\mathcal{A}$ outputs a target plaintext pair $\{m_0, m_1\}$. $\mathcal{B}$ picks $\beta \in \{0,1\}^*$ at random, sets target CT_i^*, and sends it to the $\mathcal{A}$.
4. **Phase-2**: $\mathcal{A}$ can make further queries except that the target CT_i^* is not allowed to appear in the q_{usc}.
5. **Guess**: Finally, $\mathcal{A}$ responds with its guess $\beta' \in \{0,1\}^*$. If $\beta = \beta'$, $\mathcal{A}$ wins the game.

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

$$\epsilon_{\mathcal{A}_{\mathrm{I}}}^{IND-CCA} =\mid Pr\left[\beta = \beta'\right] - 1/2 \mid \tag{5}$$

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

$$\epsilon_{\mathcal{A}_{\mathrm{II}}}^{IND-CCA} =\mid Pr\left[\beta = \beta'\right] - 1/2 \mid \tag{6}$$

Definition 7. *For EUF-CMA, we define Game-II played between a challenger $\mathcal{B}$ and an $\mathcal{A}$. The MCLS scheme is EUF-CMA secure if every PPT $\mathcal{A}$ has a negligible ϵ in winning Game-II.*

Proof. **Game-II**: This Game is interaction between the $\mathcal{B}$ and $\mathcal{A}$ as follows:

1. **Setup**: The $\mathcal{B}$ generates s, P_{pub}, PP and sends PP to $\mathcal{A}$.
2. **Phase-1**: The $\mathcal{A}$ first outputs a target ID denoted by ID_s^*. The $\mathcal{A}$ further asks $\mathsf{q}_{H_j}\{j = 0, 1, \cdots, 5\}$ hash queries, q_p, q_r, q_e, q_s, q_{sc}, and q_{usc}. The response to each query is defined in Phase-1 of Game-I in the proof of Def. 6.
3. **Forgery**: $\mathcal{A}$ outputs the forged signature under a target ID_s^*. $\mathcal{A}$ wins if unsigncryption does not return $\perp$.

5 Multi-receiver Certificateless Signcryption Scheme

In this section, we define the MCLS scheme according to the framework defined in Sec. 4.1. The main scheme is shown in Fig. 1.

1. **Setup**: Taking the security parameter λ as input, KGC initializes the system. It generates a cyclic group G of a large prime order $q \geq 2^{\lambda}$, a BP $\hat{e} : G \times G \to G$, and selects a generator P of G. KGC chooses six hash functions $H_0 : \{0,1\}^* \to G$, $H_1 : G \times G \to \{0,1\}^w$, $H_2, H_3, H_4 : \{0,1\}^w \to \{0,1\}^w$, $H_5 : \{0,1\}^* \times G \to Z_q^*$ for a positive integer w. KGC then chooses $s \in Z_q^*$ at random as master secret key and calculates master public key $P_{pub} = sP$. KGC then publishes $PP = \{G, \hat{e}, P, q, H_0, H_1, H_2, H_3, H_4, H_5\}$ and keeps s.
2. **Partial Private key**: Taking the user's ID as input, the KGC computes $Q = H_0(ID)$ and the associated $D = sQ$. The KGC sends D to the user via a secure channel.
3. **Set Secret Value**: Each user with ID selects a positive integer $x \in Z_q^*$ as a secret value.
4. **Set Public Key**: On input x, each user outputs $pk = xP$.

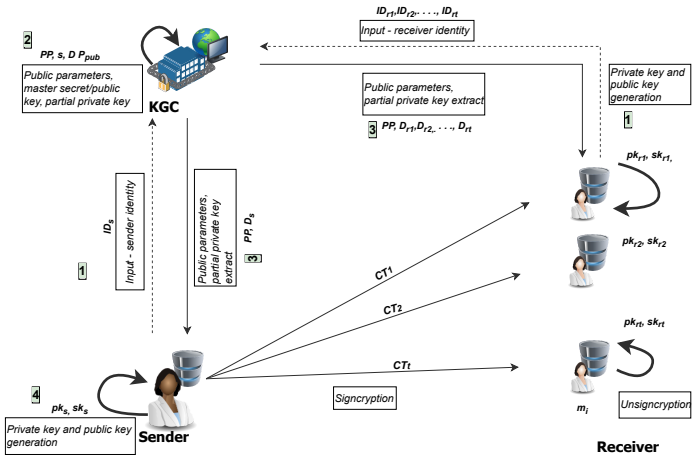

Fig. 1. The Multi-receiver Certificateless Signcryption (MCLS) Scheme

5. **Set Private Key**: Taking (D, x) as input, each user outputs a private key pair $sk = (D, x)$.
6. **Signcryption**: A sender takes its private key $sk_s = (D_s, x_s)$ and generates a CT_i to transfer a message m multi-receivers with their public key pk_{r_i} where $1 \leq i \leq t$ and $t < n$. The sender runs the following steps:
   (a) Chooses a value $r \in Z_q^*$ randomly and computes $U = rP$, $F_i = rpk_{r_i}$.
   (b) Computes $K_i = \hat{e}(P_{pub}, Q_{r_i})^r$, $Q_{r_i} = H_0(ID_{r_i})$ and $T_i = H_1(K_i, F_i)$.
   (c) Picks an ephemeral value $\sigma \in \{0,1\}^w$ randomly and computes $C_i = H_2(T_i) \parallel (H_3(T_i) \oplus \sigma)$. Use σ to compute $\alpha = H_4(\sigma)$ and generate $V_i = Enc_\alpha(m)$.
   (d) Computes $h_i = H_5(\langle C_1, C_2, \cdots, C_t \rangle, V_i, U)$ and signs the m as $S_i = (r + h_i)x_s$.
   (e) Set the $CT_i = (\langle C_1, C_2, \cdots, C_t \rangle, V_i, U, S_i, h_i)$.
7. **Unsigncryption**: Upon receiving $CT_i = (\langle C_1, C_2, \cdots, C_t \rangle, V_i, U, S_i, h_i)$, the designated receiver takes its private key $sk_{r_i} = (D_{r_i}, x_{r_i})$, sender's public key pk_s as input, and runs the following steps:
   (a) Computes $K_i = \hat{e}(U, D_{r_i})$, $F_i = x_{r_i}U$, $T_i = H_1(K_i, F_i)$ and $H_2(T_i)$.
   (b) Uses $H_2(T_i)$ to find associated C_i for $1 \leq i \leq t$ by the relation $C_i = (H_2(T_i) \parallel W_i)$ where $W_i = (H_3(T_i) \oplus \sigma)$. Computes $\sigma' = W_i \oplus H_3(T_i)$. If $\sigma' = \sigma$, the receiver runs the following steps:
   (c) Computes $\alpha' = H_4(\sigma')$ and computes $m' = Dec_{\alpha'}(V_i)$ and $h_i' = H_5(\langle C_1, C_2, \cdots, C_t \rangle, V_i, U)$. If $h_i' = h_i$, the receiver runs the following steps:
   (d) Verifies the signature and checks if $S_iP = F_i + h_i pk_s$, hold or not. If it holds, receiver gets the m, else returns 'reject'.

Correctness Proof

1. $K_i = \hat{e}(P_{pub}, Q_{r_i})^r = \hat{e}(sP, Q_{r_i})^r = \hat{e}(rP, sQ_{r_i}) = \hat{e}(U, D_{r_i})$.
2. $F_i = rpk_{r_i} = rx_{r_i}P = x_{r_i}U$.
3. $S_iP = ((r + h_i)x_s)P = rx_sP + h_ix_sP = rpk_s + h_ipk_s = F_i + h_ipk_s$.

6 Security Proof

6.1 Confidentiality

Here, we illustrate that the MCLS scheme fulfills both confidentiality and unforgeability. For confidentiality, Theorems 1 and 2 below demonstrate that the scheme is secure against IND-CCA $\mathcal{A}_{\mathrm{I}}$ and $\mathcal{A}_{\mathrm{II}}$ adversaries in the aforementioned Game-I in Def. 6. For unforgeability, Theorems 3 and 4 below demonstrate that the scheme is secure against EUF-CMA $\mathcal{A}_{\mathrm{I}}$ and $\mathcal{A}_{\mathrm{II}}$ adversaries in the aforementioned Game-II in Def. 7.

Theorem 1. *The MCLS scheme is provably secure against an IND-CCA $\mathcal{A}_{\mathrm{I}}$. Assume that an $\mathcal{A}_{\mathrm{I}}$ with a non-negligible advantage ϵ can break the MCLS scheme with running time τ, in ROM. Then, there exists an algorithm that can solve the GBDH assumption with a non-negligible advantage ϵ' with running time τ'.*

Proof. To solve the GBDH assumption, the $\mathcal{B}$ is given an instance (P, aP, bP, cP) where $P, aP, bP, cP \in G$ with unknown $a, b, c \in Z_q^*$. Let $R = \hat{e}(P, P)^{abc}$ be the solution of the GBDH assumption. The $\mathcal{B}$ would compute R by interacting with $\mathcal{A}_{\mathrm{I}}$ as follows:

1. **Setup**: $\mathcal{B}$ runs the initialized algorithm and generates $PP = \{G, \hat{e}, P, q, H_0, H_1, H_2, H_3, H_4, H_5\}$ with $P_{pub} = aP$. The $\mathcal{B}$ sends PP to $\mathcal{A}_{\mathrm{I}}$.
2. **Phase-1**: The $\mathcal{A}_{\mathrm{I}}$ first selects i target identities of receivers denoted by $ID_{r_i}^*$ for $1 \leq i \leq t$. The $\mathcal{A}_{\mathrm{I}}$ makes a number of queries including $\mathtt{q}_j\{j = 0, ..., 5\}$ hash queries, $\mathtt{q}_p$, $\mathtt{q}_r$, $\mathtt{q}_e$, $\mathtt{q}_s$, $\mathtt{q}_{sc}$, and $\mathtt{q}_{usc}$. The $\mathcal{B}$ sets empty lists PK^{list} to record public key values and maintains six empty lists $\{L_0, L_1, \cdots, L_5\}$ to record the responses of $\mathtt{q}_{H_j}$ queries. The $\mathcal{B}$ responds to $\mathcal{A}_{\mathrm{I}}$'s queries as follows:
   - *H_0-query*: If there exists (ID_i, u, Q_i) in L_0, $\mathcal{B}$ returns Q_i to $\mathcal{A}_{\mathrm{I}}$. Otherwise, $\mathcal{B}$ performs the following steps:
     (a) Picks a value $u \in Z_q^*$ at random.
     (b) If $ID_i = ID_{r_i}^*$ for some $1 \leq i \leq t$, sets $Q_i = ubP$, otherwise sets $Q_i = uP$. Stores (ID_i, u, Q_i) in L_0 and responds with Q_i.
   - *H_1-query*: Upon receiving the query, $\mathcal{B}$ checks it (K_i, F_i, T_i) exists in the L_1. If yes, $\mathcal{B}$ returns T_i to $\mathcal{A}_{\mathrm{I}}$. Else, $\mathcal{B}$ picks a string $T_i \in \{0, 1\}^w$ at random, stores (K_i, F_i, T_i) in L_1, and responds with T_i.
   - *H_2-query*: On such query, $\mathcal{B}$ checks L_2 for (T_i, x), if it exists, $\mathcal{B}$ returns x to $\mathcal{A}_{\mathrm{I}}$. Otherwise, $\mathcal{B}$ picks a string $x \in \{0, 1\}^w$ at random, stores (T_i, x) in L_2, and responds with x.
   - *H_3-query*: Upon receiving such query, if (T_i, y) exists in the L_3, $\mathcal{B}$ returns y to $\mathcal{A}_{\mathrm{I}}$. Otherwise, $\mathcal{B}$ picks a random string $y \in \{0, 1\}^w$, stores (T_i, y) in L_3, and $\mathcal{B}$ returns y to $\mathcal{A}_{\mathrm{I}}$.

- H_4-*query*: If there exists (k, w) in the L_4, $\mathcal{B}$ returns w to $\mathcal{A}_{\mathrm{I}}$. Else, $\mathcal{B}$ randomly picks a string $w \in \{0,1\}^w$, stores (k, w) in L_4, and responds with w.
- H_5-*query*: If $(\langle C_1, C_2, \cdots, C_t\rangle, V_i, U, h_i)$ exists in the L_5, $\mathcal{B}$ returns h_i to $\mathcal{A}_{\mathrm{I}}$ on receiving such query. Otherwise, $\mathcal{B}$ picks a random value $h_i \in Z_q^*$, stores the tuple $(\langle C_1, C_2, \cdots, C_t\rangle, V_i, U, h_i)$ in L_5, and then returns h_i to $\mathcal{A}_{\mathrm{I}}$.
- q_p: Upon such query, $\mathcal{B}$ checks PK^{list} for (ID_i, pk_i, x_i), if its exists, $\mathcal{B}$ returns pk_i to $\mathcal{A}_{\mathrm{I}}$. Otherwise, $\mathcal{B}$ randomly picks $x_i \in Z_q^*$, sets $pk_i = x_i P$ and stores (ID_i, pk_i, x_i) in PK^{list} and provide pk_i to $\mathcal{A}_{\mathrm{I}}$.
- q_r: $\mathcal{B}$ replaces the associated tuple (ID_i, pk_i, x_i) in PK^{list} with the new tuple $(ID_i, pk_i^*, \perp)$. Since, the x_i for pk_i^* is unknown, $\mathcal{B}$ will set $\perp$ as x_i.
- q_e: If $ID_i = ID_{r_i}^*$, for some $1 \leq i \leq t$, $\mathcal{B}$ returns $\perp$ because, $ID_{r_i}^*$ is a target ID and $\mathcal{A}_{\mathrm{I}}$ is not allowed to ask q_e for the target ID. Otherwise, if (ID_i, u, Q_i) exists in L_0, $\mathcal{B}$ computes and returns $D_i = uP_{pub}$ to $\mathcal{A}_{\mathrm{I}}$. Otherwise, $\mathcal{B}$ randomly picks a value $u \in Z_q^*$, sets $Q_i = uP$, and $D_i = uP_{pub}$, and stores (ID_i, u, Q_i) in L_0. $\mathcal{B}$ returns D_i to $\mathcal{A}_{\mathrm{I}}$.
- q_s: If (ID_i, pk_i, x_i) exists in PK^{list}, $\mathcal{B}$ returns x_i to the $\mathcal{A}_{\mathrm{I}}$. Otherwise, $\mathcal{B}$ randomly picks $x_i \in Z_q^*$, sets $pk_i = x_i P$, stores (ID_i, pk_i, x_i) in PK^{list} and returns x_i to $\mathcal{A}_{\mathrm{I}}$.
- q_{sc}: On such a query with a (m, sk_s, pk_{r_i}) $\mathcal{B}$ checks whether $ID_i = ID_{r_i}^*$ or not. If $ID_i \neq ID_{r_i}^*$, $\mathcal{B}$ performs normal signcryption as this scheme. Otherwise, $\mathcal{B}$ obtains the tuple (D_i, x_i, pk_i) via q_p, q_e, and q_s and generates a CT_i as follows:
  (a) Pick a value $r \in Z_q^*$ randomly and compute $U = rP$.
  (b) Compute $F_i = rpk_{r_i}$, $K_i = \hat{e}(P_{pub}, Q_{r_i})$, $T_i = H_1(K_i, F_i)$ and adds in L_1.
  (c) Picks an ephemeral value $\sigma \in \{0,1\}^w$ at random and computes $C_i = H_2(T_i) \parallel (H_3(T_i) \oplus \sigma)$ and adds in L_2 and L_3.
  (d) Use σ to compute $\alpha = H_4(\sigma)$ and generate $V_i = Enc_\alpha(m)$.
  (e) Compute $h_i = H_5(\langle C_1, C_2, \cdots, C_t\rangle, V_i, U)$ and update L_5.
  (f) Computes $S_i = (r+h_i)x_s$ and sets $CT_i = (\langle C_1, C_2, \cdots, C_t\rangle, V_i, U, S_i, h_i)$.
- q_{usc}: If $ID_i \neq ID_{r_i}^*$, $\mathcal{B}$ can obtain its (D_i, x_i) via the q_e and q_s, unsigncrypt CT_i and return m to $\mathcal{A}_{\mathrm{I}}$. Otherwise, $\mathcal{B}$ perform the following steps:
  (a) If $(\langle C_1, C_2, \cdots, C_t\rangle, V_i, U, h_i)$ is not in L_5, $\mathcal{B}$ terminates. Otherwise, $\mathcal{B}$ obtains σ for possible utilization further in the following:
  (b) $\mathcal{B}$ obtains Q_i from L_0 by issuing H_0 query.
  For $1 \leq i \leq t$, $\mathcal{B}$ runs the following steps:
  (a) Pick the leftmost w bits of C_i and denote it by x_i.
  (b) Pick the rightmost w bits of C_i, denote it by w_i. Compute $y_i = w_i + \sigma$.
  (c) Find a common T_i such that both the tuples (T_i, x_i) and (T_i, y_i) lie in the L_2 and L_3, respectively. If no such T_i exists, return 'abort'.
  (d) Search (K_i, F_i, T_i) associated with T_i from L_1. If not found, return 'abort'.

(e) Record the output of the query DBDH $(P, Q_i, P_{pub}, U, K_i)$ to the oracle.
(f) If DBDH $(P, pk_i, P_{pub}, U, K_i) = 1$, for some i, $\mathcal{B}$ compute $\alpha' = H_3(\sigma)$ and $m' = Dec_{\alpha'}(V_i)$.
(g) If $m' = m$, $\mathcal{B}$ returns m to the $\mathcal{A}_{\mathrm{I}}$. In all other cases, $\mathcal{B}$ terminates.

3. **Challenge**: $\mathcal{A}_{\mathrm{I}}$ gives a target plaintext pair (m_0, m_1) to $\mathcal{B}$. $\mathcal{B}$ randomly chooses $\beta \in \{0,1\}^*$ and runs the following steps:
   (a) Set $U^* = cP$. Choose $r^* \in Z_q^*$ and compute $F_i^* = r^* pk_{r_i}$
   (b) Pick a string $\sigma \in \{0,1\}^w$ at random.
   (c) For $1 \leq i \leq t$, randomly pick $x_i^* \in \{0,1\}^w$ and $y_i^* \in \{0,1\}^w$, and compute $C_i^* = x_i^* \parallel (y_i^* \oplus \sigma^*)$.
   (d) Compute $\alpha = H_3(\sigma^*)$, $V_i^* = Enc_\alpha(m_\beta)$, $h_i^* = (\langle C_1^*, C_2^*, \cdots, C_t^* \rangle, V_i^*, U^*)$, $S_i^* = (r^* + h_i^*)x_s$, and $\mathcal{B}$ returns $CT_i^* = (\langle C_1^*, C_2^*, \cdots, C_t^* \rangle, V_i^*, U^*, S_i^*, h_i^*)$.
4. **Phase-2**: The $\mathcal{A}_{\mathrm{I}}$ may ask further queries as in **Phase-2** but CT_i^* is not allowed to appear in the q_{usc}.
5. **Guess**: The $\mathcal{A}_{\mathrm{I}}$ responds with its guess $\beta' \in \{0,1\}^*$. If $\beta = \beta'$, $\mathcal{A}_{\mathrm{I}}$ wins the game. The $\mathcal{B}$ will win the Game by obtaining $R = \hat{e}(P,P)^{abc}$ which is solution to the DBDH assumption. The $\mathcal{B}$ solves the DBDH assumption by obtaining the L_1 for (K_i, F_i). Since, $Q_i = u_i bP$, $P_{pub} = aP$, and $U^* = cP$, $\mathcal{B}$ can obtain $\hat{e}(P,P)^{abc}$ by evaluating $K_i^{ui^{-1}}$.

Next, we evaluate the advantage of $\mathcal{B}$ winning by calculating the probability of occurrence of the following events:

1. In the unsigncryption query if $(\langle C_1, C_2, \cdots, C_t \rangle, V_i, U, h_i)$ cannot be found in L_5, $\mathcal{B}$ returns 'failure' and terminates. The probability is $1/\mathsf{q}_{H_5}$.
2. In the q_e, the game terminates if $ID \neq ID_{r_i}^*$, The probability is $1/\mathsf{q}_e$.
3. In the q_{usc}, the game aborts due to invalid message $m' \neq m$. The probability is q_{usc}/q.

The $\mathcal{B}$ will obtain the L_1 for the some (K_i, F_i) with the probability $1/\mathsf{q}_{H_1}$. Hence, if an IND-CCA $\mathcal{A}_{\mathrm{I}}$ can break MCLS scheme with a non-negligible ϵ, then the GBDH assumption can be solved with a non-negligible advantage ϵ' as given below:

$$\epsilon' \geq \epsilon \left(\frac{1}{\mathsf{q}_{H_5}}\right)\left(\frac{1}{\mathsf{q}_{H_1}}\right)\left(1 - \frac{1}{\mathsf{q}_e}\right)\left(1 - \frac{\mathsf{q}_{usc}}{\mathsf{q}}\right) \tag{7}$$

τ' is the required computation time while answering the queries in the aforementioned simulation Game-I. It turns out that $\tau' = \tau + \mathcal{O}(\mathsf{q}_o + \mathsf{q}_p + \mathsf{q}_e)\tau_1 + \mathcal{O}(\mathsf{q}_1 + \mathsf{q}_2 + \mathsf{q}_3 + \mathsf{q}_4 + \mathsf{q}_5 + \mathsf{q}_r + \mathsf{q}_s + \mathsf{q}_{usc})$ where τ_1 is the time to perform a scalar multiplication in G and t is the number of target identities.

Theorem 2. *The scheme is provably secure against an IND-CCA $\mathcal{A}_{\mathrm{II}}$. Assume that an $\mathcal{A}_{\mathrm{II}}$ with a non-negligible advantage ϵ can break the MCLS scheme with running time τ in ROM. Then, there is an algorithm that can solve the CDH assumption with a non-negligible advantage ϵ' with running time τ'.*

Proof. Assume that $\mathcal{B}$ is given a random instance (P, aP, bP) of the CDH assumption, where $P, aP, bP \in G$, within unknown $a, b \in Z_q^*$. Let $J = abP$ be the solution of the CDH assumption. The $\mathcal{B}$ would compute J by interacting with the $\mathcal{A}_{\text{II}}$ as follows:

1. **Setup**: $\mathcal{B}$ runs the initialized algorithm and generates $PP = \{G, \hat{e}, P, q, H_0, H_1, H_2, H_3, H_4, H_5\}$ with $P_{pub} = aP$. The $\mathcal{B}$ sends PP to $\mathcal{A}_{\text{II}}$.
2. **Phase-1**: The $\mathcal{A}_{\text{II}}$ first selects i target identities of denoted by $ID_{r_i}^*$. The $\mathcal{A}_{\text{II}}$ makes number of queries including a H_0 query, $\mathtt{q}_r$, $\mathtt{q}_s$, and $\mathtt{q}_{usc}$. The $\mathcal{B}$ sets empty PK^{list} to record public key values. $\mathcal{B}$ responds to all the queries as follows:
   - *H_0-query*: If there exists (ID_i, u, Q_i) in L_0, $\mathcal{B}$ returns Q_i to $\mathcal{A}_{\text{II}}$. Otherwise, $\mathcal{B}$ picks a value $u \in Z_q^*$ randomly and computes $Q_i = uP$. Then, $\mathcal{B}$ Stores (ID_i, u, Q_i) in L_0 and responds with Q_i.
   - $\mathtt{q}_p$: If there exists (ID_i, pk_i, x_i) in the PK^{list}, $\mathcal{B}$ returns pk_i to A. Otherwise, $\mathcal{B}$ performs the following step:
     (a) Picks a value $x_i \in Z_q^*$ at random.
     (b) If $ID_i = ID_{r_i}^*$, set $pk_i = x_i aP$ Otherwise, set $pk_i = x_i P$, store (ID_i, pk_i, x_i) in the PK^{list} and return pk_i to $\mathcal{A}_{\text{II}}$.
   - $\mathtt{q}_r$: If $ID_i = ID_{r_i}^*$, $\mathcal{B}$ reports failure and terminates because $\mathcal{A}_{\text{II}}$ is not allowed to ask $\mathtt{q}_r$ for the target identity $ID_{r_i}^*$. Otherwise, $\mathcal{B}$ replaces the associated tuple (ID_i, pk_i, x_i) in the PK^{list} with $(ID_i, pk_i', \perp)$.
   - $\mathtt{q}_s$: If $ID_i = ID_{r_i}^*$, $\mathcal{B}$ returns $\perp$. If (ID_i, pk_i, x_i) exists in the PK^{list}, $\mathcal{B}$ returns x_i to the $\mathcal{A}_{\text{II}}$. Otherwise, $\mathcal{B}$ randomly picks $x_i \in Z_q^*$ sets $pk_i = x_i P$, stores (ID_i, pk_i, x_i) in PK^{list}, and returns x_i to $\mathcal{A}_{\text{II}}$.
   - $\mathtt{q}_{usc}$: If $ID_i \neq ID_{r_i}^*$, $\mathcal{B}$ can obtain its (D_i, x_i) via the $\mathtt{q}_e$ and $\mathtt{q}_s$, unsigncrypt CT_i and return m to $\mathcal{A}_{\text{II}}$. Otherwise, $\mathcal{B}$ perform the following procedure: If $(\langle C_1, C_2, \cdots, C_t \rangle, V_i, U, h_i)$ is not in L_5, $\mathcal{B}$ terminates. Otherwise, $\mathcal{B}$ obtains σ for possible utilization further in the following:
     (a) $\mathcal{B}$ obtains Q_i associated with ID_i from the PK^{list} or by issuing $\mathtt{q}_p$.
     For $1 \leq i \leq t$, $\mathcal{B}$ runs the following steps:
     (a) Pick the leftmost w bits of C_i and denote it by x_i.
     (b) Pick the rightmost w bits of C_i, denote it by w_i. Compute $y_i = w_i \oplus \sigma$.
     (c) Find a common T_i such that both the tuples (T_i, x_i) and (T_i, y_i) lie in the L_2 and L_3, respectively. If no such T_i exists, return 'abort'.
     (d) Search the tuple (K_i, F_i, T_i) associate with T_i from L_1. If not found, return 'abort'.
     (e) If $\hat{e}(P, F_i) = \hat{e}(U, pk_i)$, record the value i.
     (f) If $\hat{e}(P, F_i) = \hat{e}(U, pk_i)$ for some $1 \leq i \leq t$, $\mathcal{B}$ computes $\alpha' = H_3(\sigma)$ and $m' = Dec_{\alpha'}(V_i)$. If $m' = m$, $\mathcal{B}$ returns m to the $\mathcal{A}_{\text{II}}$. In all the other cases, $\mathcal{B}$ terminates.
3. **Challenge**: $\mathcal{A}_{\text{II}}$ gives a target plaintext pair (m_0, m_1) to $\mathcal{B}$. Then, $\mathcal{B}$ randomly chooses $\beta \in \{0,1\}^*$ and runs the following steps:
   (a) Set $U^* = bP$. Choose $r^* \in Z_q^*$, $F_i^* = r^* pk_{r_i}$
   (b) For $1 \leq i \leq t$, randomly pick $x_i^* \in \{0,1\}^w$ and $y_i^* \in \{0,1\}^w$, and compute $C_i^* = x_i^* \parallel (y_i^* \oplus \sigma^*)$.

(c) Computes $\alpha = H_3(\sigma^*), V_i^* = Enc_\alpha(m_\beta)$, set $h_i^* = (\langle C_1^*, C_2^*, \cdots, C_t^* \rangle, V_i^*, U^*)$, $S_i^* = (r^* + h_i^*)x_s$, and $\mathcal{B}$ returns $CT_i^* = (\langle C_1^*, C_2^*, \cdots, C_t^* \rangle, V_i^*, U^*, S_i^*, h_i^*)$.

4. **Phase-2**: An $\mathcal{A}_{\text{II}}$ may issue further queries as in **Phase-2** with the restriction that CT_i^* is not allowed to appear in the q_{usc}.
5. **Guess**: An $\mathcal{A}_{\text{II}}$ responds with its guess $\beta' \in \{0,1\}^*$. If $\beta = \beta'$, $\mathcal{A}_{\text{II}}$ wins the game.

The $\mathcal{B}$ wins the game by obtaining $J = abP$, which serves as the solution to the CDH assumption. To achieve this, $\mathcal{B}$ solves the CDH assumption by obtaining the set L_1 containing pairs (K_i, F_i) such that $\hat{e}(P, F_i) = \hat{e}(U^*, pk_i)$ $\mathcal{B}$ identifies such an F_i by verifying the equality $\hat{e}(P, F_i) = \hat{e}(U^*, pk_i)$ for all instances of F_i appearing in L_1. Since, $U^* = bP$ and $pk_i = x_i(aP)$, $\mathcal{B}$ can compute $J = abP$ by evaluating $x_i^{-1}F_i$. Next, we assess the advantage of $\mathcal{B}$ winning by calculating the probability of the following events occurring:

1. In the q_{usc}, if $(\langle C_1, ..., C_t \rangle, V_i, U, h_i)$ cannot be found in L_5, $\mathcal{B}$ returns failure and terminates. The probability is $1/\mathsf{q}_{H_5}$.
2. In the q_r, the game terminates if $ID_i = ID_{r_i}^*$. The probability is $1/\mathsf{q}_r$.
3. In the q_s, the game terminates if $ID_i = ID_{r_i}^*$. The probability is $1/\mathsf{q}_s$.
4. In the q_{usc}, the game aborts due to invalid message $m' \neq m$. The probability is q_{usc}/q.

The $\mathcal{B}$ will obtain the L_1 for the some (K_i, F_i) with the probability $1/\mathsf{q}_{H_1}$. Hence, if an IND-CCA $\mathcal{A}_{\text{II}}$ can break MCLS scheme with a non-negligible ϵ, then the GBDH assumption can be solved with a non-negligible advantage ϵ'

$$\epsilon' \geq \epsilon \left(\frac{1}{\mathsf{q}_{H_5}}\right)\left(\frac{1}{\mathsf{q}_{H_1}}\right)\left(1 - \frac{1}{\mathsf{q}_r}\right)\left(1 - \frac{1}{\mathsf{q}_s}\right)\left(1 - \frac{\mathsf{q}_{usc}}{\mathsf{q}}\right) \quad . \tag{8}$$

In the following, we assess the required computation time τ' while answering queries in the aforementioned simulation game. It turns out that $\tau' = \tau + O(\mathsf{q}_0 + \mathsf{q}_p).\tau_1 + O(\mathsf{q}_{usc}).\tau_2 + O(\mathsf{q}_1 + \mathsf{q}_2 + \mathsf{q}_3 + \mathsf{q}_4 + \mathsf{q}_5 + \mathsf{q}_r + \mathsf{q}_s)$, where τ_1 is the time to perform a scalar multiplication in G, τ_2 is the time to perform a pairing operation and t is the number of target identities.

6.2 Authentication

Theorem 3. *The scheme is provably secure against EUF-CMA $\mathcal{A}_{\text{I}}$. Assume that an $\mathcal{A}_{\text{I}}$ with a non-negligible advantage ϵ can break the MCLS scheme with running time τ in ROM. Then, there exists $\mathcal{B}$, that can solve the DBDHI assumption with a non-negligible advantage ϵ' with running time τ'.*

Proof. Assume that $\mathcal{B}$ is given a random instance (P, aP, bP, cP) of the DBDHI assumption, where $P, aP,\ bP, cP \in G$ with unknown $a, b, c \in Z_q^*$. Let $R = \hat{e}(P, P)^{ab^{-1}c}$ be the solution of DBDHI assumption. The $\mathcal{B}$ would compute R by interacting with the $\mathcal{A}_{\text{I}}$ as follows:

1. **Setup**: $\mathcal{B}$ runs the initial algorithm and generates $PP = \{G, \hat{e}, P, q, H_0, H_1, H_2, H_3, H_4, H_5\}$ with $P_{pub} = aP$. The $\mathcal{B}$ sends PP to $\mathcal{A}_{\text{I}}$.

2. **Phase-1**: The $\mathcal{A}_{\mathrm{I}}$ first selects a target ID denoted by ID_s^*. The $\mathcal{A}_{\mathrm{I}}$ makes number of queries including a H_0 query, q_e, and q_{sc}. The $\mathcal{B}$ responds as follows:
   - H_0-query: If there exists (ID_i, u, Q_i) in the L_0, $\mathcal{B}$ returns Q_i to $\mathcal{A}_{\mathrm{I}}$. Otherwise, $\mathcal{B}$ performs the following steps:
     (a) Picks a value $u \in Z_q^*$ at random.
     (b) If $ID_i = ID_s^*$, sets $Q_i = ub^{-1}P$. Otherwise, sets $Q_i = uP$, stores (ID_i, u, Q_i) in L_0 and responds with Q_i to $\mathcal{A}_{\mathrm{I}}$.
   - q_e: If $ID_i = ID_s^*$, $\mathcal{B}$ returns $\perp$ because, ID_s^* is a target ID and $\mathcal{A}_{\mathrm{I}}$ is not allowed to ask for a q_e for the target ID. Otherwise, if (ID_i, u, Q_i) exists in L_0, B computes and returns $D_i = uP_{pub}$ to $\mathcal{A}_{\mathrm{I}}$. Otherwise, $\mathcal{B}$ randomly picks a value $u \in Z_q^*$, sets $Q_i = uP$, and $D_i = uP_{pub}$, and stores (ID_i, u, Q_i) in L_0. $\mathcal{B}$ returns D_i to $\mathcal{A}_{\mathrm{I}}$.
   - q_{sc}: When $\mathcal{B}$ receives a signcrypt query with a m, sender's private key sk_s, and receivers' public key pk_{r_i}, it checks if $ID_i = ID_s^*$. If $ID_i \neq ID_s^*$, the $\mathcal{B}$ runs the normal signcryption algorithm. Otherwise, $\mathcal{B}$ obtains the tuple (D_i, x_i, pk_i) via q_p, q_e, and q_s queries and generates a CT_i via following procedure:
     (a) Pick a value $r \in Z_q^*$ randomly and compute $U = rP$, and $F_i = rpk_{r_i}$
     (b) Picks an ephemeral value $\sigma \in \{0,1\}^w$ at random and compute $C_i = H_2(T_i) \parallel (H_3(T_i) \oplus \sigma)$ and adds in L_2 and L_3.
     (c) Use σ to compute $\alpha = H_5(\sigma)$ and generate $V_i = Enc_\alpha(m)$.
     (d) Set $h_i = H_5(\langle C_1, C_2, \cdots, C_t\rangle, V_i, U)$ and update L_5. Compute $S_i = (r + h_i)x_s$, and sets $CT_i = (\langle C_1, C_2, \cdots, C_t\rangle, V_i, U, S_i, h_i)$.
3. **Forgery**: After the query phase completes, $\mathcal{A}_{\mathrm{I}}$ outputs the challenge ID_s^*, a m, and a $CT_i^* = (\langle C_1^*, C_2^*, \cdots, C_t^*\rangle, V_i^*, U^*, S_i^*, h_i^*)$. However, it cannot ask for the q_{usc} for the CT_i^* with the private key of any target ID.

If the game does not abort, the $\mathcal{B}$ fetches the L_1 for (K_i, F_i) to obtain $R = \hat{e}(P,P)^{ab^{-1}c}$ which is the solution to the DBDHI assumption. Since, $Q_i = ub^{-1}P$, $P_{pub} = aP$, and $U^* = cP$, $\mathcal{B}$ can obtain $R = \hat{e}(P,P)^{ab^{-1}c}$ by evaluating $K_i^{ui^{-1}}$. Hence, if $\mathcal{A}_{\mathrm{I}}$ can break the MCLS scheme with a non-negligible advantage ϵ, the DBDHI assumption can be solved with a non-negligible advantage ϵ'

$$\epsilon' \geq \epsilon \left(\frac{1}{\mathsf{q}_{H_5}}\right)\left(\frac{1}{\mathsf{q}_{H_1}}\right)\left(1 - \frac{1}{\mathsf{q}_e}\right)\left(1 - \frac{\mathsf{q}_{usc}}{\mathsf{q}}\right) \tag{9}$$

τ' is the required computation time while answering the queries in the aforementioned simulation game. It turns out that $\tau' = \tau + O(\mathsf{q}_0 + \mathsf{q}_p + \mathsf{q}_e)\tau_1 + O(\mathsf{q}_1 + \mathsf{q}_2 + \mathsf{q}_3 + \mathsf{q}_4 + \mathsf{q}_5 + \mathsf{q}_r + \mathsf{q}_s + \mathsf{q}_{usc})$ where τ_1 is the time to perform a scalar multiplication in G and t is the number of target identities.

Theorem 4. *The scheme is provably secure against EUF-CMA $\mathcal{A}_{\mathrm{II}}$. Assume that an $\mathcal{A}_{\mathrm{II}}$ with a non-negligible advantage ϵ can break the MCLS scheme with running time τ in ROM. Then, there is $\mathcal{B}$ that can solve the CDH assumption with a non-negligible advantage ϵ' with running time τ'.*

Proof. Assume that $\mathcal{B}$ is given a random instance (P, aP, bP) of the CDH assumption, where $P, aP, bP \in G$, within unknown $a, b \in Z_q^*$. Let $R = abP$ be the solution of the CDH assumption. The $\mathcal{B}$ would compute J by interacting with the $\mathcal{A}_{\text{II}}$ as follows:

1. **Setup**: $\mathcal{B}$ runs the initialized algorithm and generates $PP = \{G, \hat{e}, P, q, H_0, H_1, H_2, H_3, H_4, H_5\}$ with $P_{pub} = aP$. The $\mathcal{B}$ sends PP to $\mathcal{A}_{\text{II}}$.
2. **Phase-1**: The $\mathcal{A}_{\text{II}}$ first selects a target ID denoted by ID_s^*. The $\mathcal{A}_{\text{II}}$ makes number of queries including a H_0 query, q_e, and q_{sc}. The $\mathcal{B}$ responds as follows:
   - q_p: If there exists (ID_i, pk_i, x_i) in the PK^{list}, $\mathcal{B}$ returns pk_i to $\mathcal{A}_{\text{II}}$. Otherwise, $\mathcal{B}$ performs the following steps:
     (a) Pick $x_i \in Z_q^*$ at random.
     (b) If $ID_i = ID_s^*$, set $pk_i = x_i aP$; otherwise, set $pk_i = x_i P$.
     (c) Store (ID_i, pk_i, x_i) in the PK^{list} and returns pk_i to $\mathcal{A}_{\text{II}}$.
   - q_{sc}: When $\mathcal{B}$ receives a signcrypt query with a m, sender's private key sk_s, it checks if $ID_i = ID_s^*$. If $ID \neq ID_s^*$, formal signcryption algorithm runs. Otherwise, $\mathcal{B}$ obtains the tuple (D_i, x_i, pk_i) via q_p, q_e, and q_s and generates a CT_i via following procedure:
     (a) Pick a value $r \in Z_q^*$ randomly and compute $U = rP$.
     (b) Compute $F_i = rpk_{r_i}$, $K_i = \hat{e}(P_{pub}, Q_{r_i})$, $T_i = H_1(K_i, F_i)$, and add to L_1.
     (c) Picks an ephemeral value $\sigma \in \{0,1\}^w$ at random and compute $C_i = H_2(T_i) \parallel (H_4(T_i) \oplus \sigma)$ and adds in L_2 and L_3.
     (d) Use σ to compute $\alpha = H_4(\sigma)$ and generate $V_i = Enc_\alpha(m)$.
     (e) Set $h_i = H_5(\langle C_1, C_2, \cdots, C_t \rangle, V_i, U)$ and update L_5. Compute $S_i = (r + h_i)x_s$, and sets $CT_i = (\langle C_1, C_2, \cdots, C_t \rangle, V_i, U, S_i, h_i)$.
3. **Forgery**: After the query phase completes, $\mathcal{A}_{\text{II}}$ outputs the $\mathcal{B}$ sender's ID_s^*, a m, and a $CT_i^* = (\langle C_1^*, C_2^*, \cdots, C_t^* \rangle, V_i^*, U^*, S_i^*, h_i^*)$. However, it cannot ask for the q_{usc} for the CT_i^* with the private key of any target ID.

If the game does not abort, the $\mathcal{B}$ fetches the L_1 for (K_i, F_i) such that $\hat{e}(P, F_i) = \hat{e}(U^*, pk_{r_i})$. The $\mathcal{B}$ can find such a F_i by verifying the equality $\hat{e}(P, F_i) = \hat{e}(U^*, pk_{r_i})$. Since, $U^* = bP$, $pk_{r_i} = x_i(aP)$, $\mathcal{B}$ can obtains $R = abP$ by evaluating $x_i^{-1} F_i$ which is the solution to the CDH assumption. Hence, if the $\mathcal{A}_{\text{II}}$ can break the proposed MCLS scheme with a non-negligible advantage ϵ, the CDH assumption can be solved with a non-negligible advantage ϵ'

$$\epsilon' \geq \epsilon \left(\frac{1}{\mathsf{q}_{H_5}}\right)\left(\frac{1}{\mathsf{q}_{H_1}}\right)\left(1 - \frac{1}{\mathsf{q}_r}\right)\left(1 - \frac{1}{\mathsf{q}_s}\right)\left(1 - \frac{\mathsf{q}_{usc}}{\mathsf{q}}\right) \quad . \tag{10}$$

In the following, we assess the required computation time τ' while answering queries in the aforementioned simulation game. It turns out that $\tau' = \tau + \mathcal{O}(\mathsf{q}_0 + \mathsf{q}_p).\tau_1 + \mathcal{O}(\mathsf{q}_{usc}).\tau_2 + \mathcal{O}(\mathsf{q}_1 + \mathsf{q}_2 + \mathsf{q}_3 + \mathsf{q}_4 + \mathsf{q}_5 + \mathsf{q}_r + \mathsf{q}_s)$, where τ_1 is the time to perform a scalar multiplication in G, τ_2 is the time to perform a pairing operation and t is the number of target identities.

6.3 Public Verifiability

In the **Signcryption** step in section 5, the sender computes $h_i = H_5(\langle C_1, C_2, \cdots, C_t\rangle, V_i, U)$ and the signs the message $S_i = (r + h_i)x_s$. The h_i value is a hash of the message and other parameters, and the S_i value is a signature of the sender that can be verified using the sender's public key pk_s. In the **Unsigncryption** step, the receiver computes $h'_i = H_5(\langle C_1, C_2, \cdots, C_t\rangle, V_i, U)$ and checks if $h'_i = h_i$. If yes, the receiver verifies if $S_iP = F_i + h_i pk_s$ holds or not. If yes, then the receiver accepts the message as valid and authentic. Otherwise, the receiver rejects the message.

The public verifiability feature allows any third party to perform these steps as well, using only the ciphertext $CT_i = (\langle C_1, C_2, \cdots, C_t\rangle, V_i, U, S_i, h_i)$ and the public keys of the sender pk_s and receiver pk_r. Thus, anyone can verify that the signcrypted message is valid and authentic without knowing the private keys of either party.

7 Performance Comparison and Discussion

Here, we compare the proposed MCLS scheme with the existing single receiver and multi-receiver encryption and signcryption schemes which are mainly based on BP operation [22,19,15,16,23,11]. The notations are defined as follows: T_p shows the time of executing a bilinear pairing operation $\hat{e} : G \times G \to G$. T_m is the time of executing a scalar multiplication operation in G, T_e shows the time of executing an exponentiation in G_2 or an exponentiation operation in Z_q^*, T_i is the time of executing modular inversion operation, T_{pm} shows the time of executing point multiplication operation, and n is the number of receivers. For single receiver schemes, Table 1 compares the computational cost of Encryption (Enc)/Decryption (Dec), signcryption/unsigncryption [16,15].

Table 1. Comparison between MCLS scheme with the existing single and multi-receiver encryption and signcryption schemes based on bilinear pairings.

Scheme	Single receiver		Multi-receiver	
	Enc/Signcrypt	Dec/Unsigncrypt	Enc/Signcrypt	Dec/Unsigncrypt
[16]	$T_e + T_p + 4T_m$	$T_p + 5T_m$	-	-
[15]	$4T_e$	$2T_e + 2T_p + T_i$	-	-
[19]	-	-	$2nT_e + 2nT_p + (n+2)T_m$	$4T_p + T_m$
[22]	-	-	$(n+1)T_e + nT_m$	$2T_p + 2T_m$
[23]	-	-	$nT_e + 2nT_{pm} + 2nT_m$	$nT_p + nT_m + 3nT_{pm}$
[11]	-	-	$nT_e + nT_p + (n+1)T_m$	$T_p + T_m$
Our scheme	$T_e + T_p + 4T_m$	$T_p + 2T_m$	$nT_e + nT_p + (2n+1)T_m$	$T_p + 2T_m$

In the encryption process, Li *et al.* [16] require $T_e + T_p + 4T_m$ to produce a ciphertext and $T_p + 5T_m$ for decryption. Karati *et al.* [15] require $4T_e$ for message encryption and $2T_e + 2T_p + T_i$ for decryption. Whereas, for single receiver,

the proposed MCLS scheme requires $T_e + T_p + 4T_m$ operations for signcryption, and it requires $T_p + 2T_m$ operations for unsigncryption. The computational cost of multi-receiver schemes is compared with [19,22,23]. Wang *et al.* [22] require $(n+1)T_e + nT_m$ to produce a signcrypted ciphertext and $2T_p + 2T_m$ for unsigncryption. Niu *et al.* [19] require $2nT_e + 2nT_p + (n+2)T_m$ in signcryption while $4T_p + T_m$ in unsigncryption for multiple recipients. Furthermore, Yang *et al.* [23] require $nT_e + 2nT_m + 2nT_{pm}$ in signcryption and $nT_p + nT_m + 3nT_{pm}$ in unsigncryption. For multi-receivers, the MCLS scheme requires $nT_e + nT_p + (2n+1)T_m$ in signcryption to produce a ciphertext while $T_p + 2T_m$ in unsigncryption. The signcryption cost in the MCLS scheme is linear with the number of designated receivers, while the unsigncryption cost is constant for each receiver. Further, Hung *et al.*'s [11] multi-receiver encryption scheme require $nT_e + nT_p + (n+1)T_m$ operations for encryption phase and $T_p + T_m$ operations for decryption. As compared to Hung *et al.*'s scheme, the MCLS require only two additional multiplication operations for signcryption and unsigncryption. In Table 2, we compare the

Table 2. Comparison of public key settings, exemption of key escrow and public verifiability requirement

Schemes	**Public key settings**	**Exemption of key escrow**	**Public verifiability**
[16]	IDPKC - CLPKC	✓	×
[15]	IDPKC	×	×
[19]	IDPKC - CLPKC	✓	×
[22]	IDPKC - PKC	×	×
[23]	IDPKC	×	×
[11]	CLPKC	✓	×
Our scheme	CLPKC	✓	✓

public key settings, exemption of key escrow problem, and public verifiability security requirements with existing signcryption schemes. Li *et al.* [16] and Niu *et al.* [19] work in ID-Public Key Cryptogrpahy (IDPKC)-CLPKC, therefore, the sending entities does not exmept the key escrow and Hung *et al.* [11] work in CLPKC, therefore, it exempts the key escrow. Further, [15,22,23] work in IDPKC environment and do not exempt the key escrow. However, none of these schemes provide public verifiability. As compared to above schemes, the proposed MCLS scheme works in CLPKC, exempts the key escrow and achieves public verifiability. Finally, while the proposed scheme significantly extends Hung *et al.*'s efficient anonymous multi-receiver certificateless encryption scheme into a multi-receiver certificateless signcryption scheme, this comes at a relatively small cost.

8 Conclusion

This paper presents a Multi-receiver Certificateless Signcryption (MCLS) scheme to fulfill both confidentiality and authentication requirements, building on Hung

et al.'s encryption [11] scheme. In the proposed scheme, the message is encrypted and signed with sender's private key, ensuring the message reliability and authenticity. In this scheme, the signcryption cost increases linearly with the number of receivers, while the required unsigncryption cost of each receiver is constant and independent of the number of receivers. We formally demonstrate the semantic security of the scheme against the IND-CCA (from Hung *et al.*'s scheme) and EUF-CMA (proposed) attacks in the ROM using the GBDH, CDH, and DBDHI assumptions, respectively. We also introduce public verifiability as an additional feature that is achieved simultaneously along with authentication. Finally, we compare the proposed MCLS scheme's performance and functionality to existing single receiver and multi-receiver signcryption approaches. In comparison to other existing single receiver and multi-receiver signcryption techniques, the MCLS scheme is more efficient, while both avoiding the key escrow problem and work in multi-receiver certificateless public key setting.

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., 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)
4. Cha, J.C., Cheon, J.H.: An identity-based signature from gap diffie-hellman groups. IACR Cryptol. ePrint Arch. p. 18 (2002), http://eprint.iacr.org/2002/018
5. Chaudhry, S.A., Nizamuddin, Sher, M., Ghani, A., Naqvi, H., Irshad, A.: An efficient signcryption scheme with forward secrecy and public verifiability based on hyper elliptic curve cryptography. Multim. Tools Appl. **74**(5), 1711–1723 (2015). https://doi.org/10.1007/s11042-014-2283-9, https://doi.org/10.1007/s11042-014-2283-9
6. Chen, L., Malone-Lee, J.: Improved identity-based signcryption. In: Vaudenay, S. (ed.) Public Key Cryptography - PKC 2005, 8th International Workshop on Theory and Practice in Public Key Cryptography, Les Diablerets, Switzerland, January 23-26, 2005, Proceedings. Lecture Notes in Computer Science, vol. 3386, pp. 362–379. Springer (2005), https://doi.org/10.1007/978-3-540-30580-4_25
7. Chen, Y., Chen, L.: The twin bilinear diffie-hellman inversion problem and applications. In: Information Security and Cryptology-ICISC 2010: 13th International Conference, Seoul, Korea, December 1-3, 2010, Revised Selected Papers 13. pp. 113–132. Springer (2011)

8. Chow, S.S.M., Yiu, S., Hui, L.C.K., Chow, K.: Efficient forward and provably secure id-based signcryption scheme with public verifiability and public ciphertext authenticity. In: Lim, J.I., Lee, D.H. (eds.) Information Security and Cryptology - ICISC 2003, 6th International Conference, Seoul, Korea, November 27-28, 2003, Revised Papers. Lecture Notes in Computer Science, vol. 2971, pp. 352–369. Springer (2003), https://doi.org/10.1007/978-3-540-24691-6_26
9. Guo, R., Xu, L., Li, X., Zhang, Y., Li, X.: An efficient certificateless ring signcryption scheme with conditional privacy-preserving in vanets. J. Syst. Archit. **129**, 102633 (2022), https://doi.org/10.1016/j.sysarc.2022.102633
10. Hu, S., Zhang, R., Wang, F., Chen, K., Lian, B., Chen, G.: A sanitizable signcryption scheme with public verifiability via chameleon hash function. J. Inf. Secur. Appl. **71**, 103371 (2022). https://doi.org/10.1016/j.jisa.2022.103371, https://doi.org/10.1016/j.jisa.2022.103371
11. Hung, Y., Huang, S., Tseng, Y., Tsai, T.: Efficient anonymous multireceiver certificateless encryption. IEEE Syst. J. **11**(4), 2602–2613 (2017), https://doi.org/10.1109/JSYST.2015.2451193
12. Islam, S.H., Khan, M.K., Al-Khouri, A.M.: Anonymous and provably secure certificateless multireceiver encryption without bilinear pairing. Secur. Commun. Networks **8**(13), 2214–2231 (2015), https://doi.org/10.1002/sec.1165
13. Joux, A., Nguyen, K.: Separating decision diffie–hellman from computational diffie–hellman in cryptographic groups. Journal of cryptology **16**, 239–247 (2003)
14. Karati, A., Fan, C., Hsu, R.: Provably secure and generalized signcryption with public verifiability for secure data transmission between resource-constrained iot devices. IEEE Internet Things J. **6**(6), 10431–10440 (2019). https://doi.org/10.1109/JIOT.2019.2939204, https://doi.org/10.1109/JIOT.2019.2939204
15. Karati, A., Islam, S.H., Biswas, G.P., Bhuiyan, M.Z.A., Vijayakumar, P., Karuppiah, M.: Provably secure identity-based signcryption scheme for crowdsourced industrial internet of things environments. IEEE Internet Things J. **5**(4), 2904–2914 (2018), https://doi.org/10.1109/JIOT.2017.2741580
16. Li, S., Tao, F., Shi, T.: Security analysis and improvement of hybrid signcryption scheme based on heterogeneous system. In: 14th International Conference on Computer Science & Education, ICCSE 2019, Toronto, ON, Canada, August 19-21, 2019. pp. 840–845. IEEE (2019), https://doi.org/10.1109/ICCSE.2019.8845053
17. Malone-Lee, J.: Identity-based signcryption. IACR Cryptol. ePrint Arch. p. 98 (2002), http://eprint.iacr.org/2002/098
18. Miao, S., Zhang, F., Zhang, L.: Cryptanalysis of a certificateless multi-receiver signcryption scheme. In: 2010 International Conference on Multimedia Information Networking and Security. pp. 593–597. IEEE (2010)
19. 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)
20. Selvi, S.S.D., Vivek, S.S., Rangan, C.P.: A note on the certificateless multi-receiver signcryption scheme. IACR Cryptol. ePrint Arch. p. 308 (2009), http://eprint.iacr.org/2009/308
21. 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

22. Wang, C., Liu, C., Li, Y., Qiao, H., Chen, L.: Multi-message and multi-receiver heterogeneous signcryption scheme for ad-hoc networks. Inf. Secur. J. A Glob. Perspect. **26**(3), 136–152 (2017), https://doi.org/10.1080/19393555.2017.1319523
23. 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, https://doi.org/10.1109/TGCN.2022.3163596
24. 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
25. Zheng, Y.: Digital signcryption or how to achieve cost(signature & encryption) textlesstextless 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, https://doi.org/10.1007/BFb0052234
26. Zheng, Y.: Signcryption or how to achieve cost (signature & encryption)¡¡ cost (signature)+ cost (encryption). In: Annu. Int. Cryptol. Conf (1999)