Math-for-industry
Education & Research Hub

Identity based ring signcryption schemes revisited

S. Sharmila Deva Selvi, S. Sree Vivek and C. Pandu Rangan

Received on January 24, 2011

Abstract. Ring Signcryption is used to provide a graceful way to leak trustworthy secrets in an anonymous, authenticated and confidential way. To the best of our knowledge, seven identity based ring signcryption schemes were reported in the literature. In this paper, we show that five of them are insecure in one or the other way. We then propose a new scheme and formally prove its security properties. A comparison of our scheme with the only existing secure scheme by Huang et al. shows that our scheme is more efficient than the scheme by Huang et al.

Keywords. Ring Signcryption, Cryptanalysis, Provable Security, Confidentiality, Chosen Plaintext Attack, Adaptive Chosen Ciphertext Attack, Bilinear Pairing, Random Oracle Model.

1. Introduction

Public key certification and management is often considered to be an inherent overhead in Public Key Infrastructure (PKI). Identity Based Cryptography (IBC) was introduced by Shamir [9] in the year 1984 to overcome this issue. The public key of a user in IBC is not a random string any more, instead it is an unique string such as email id, IP address, social security number etc which identifies the user in the system. The user of an IBC is not required to obtain a certificate for his public key, since his identity is well known in public or available in a public directory. IBC employs a trusted third party, namely the private key generator (PKG) who generates the private key corresponding to the identity of a user, at the time of registration with the PKG. Thus, the private keys of all the users registered with a PKG are known to the PKG.

Signcryption was proposed by Zheng [13] as a cryptographic primitive to provide both authenticity and confidentiality with a lower computational cost than linearly combining the signature and encryption of the message independently. Signcryption is an excellent invention by itself, which provides both authenticity and confidentiality for a message that is transmitted between two users in a public key system. An extension of this primitive is *Ring Signcryption*, which offers anonymity along with authenticity and confidentiality. The concept of ring signature, was proposed by Rivest et al. [7] to provide authenticity for a message in an anonymous way, i.e. the verifier does not know who has signed the message but he/she can verify that one of the person from the ring (group), formed by the signer during the signing process has done it. Ring signcryption enables an user to send an authentic message confidentially and anonymously to a specified receiver. Ring signcryption schemes do not have group managers and special setup procedure to form a group: any user can choose a list of potential senders that includes himself/herself and signcrypt a message by using his private key and the public keys of other users, without getting approval or assistance from others.

Motivation. Ring signcryption as a primitive can be motivated from the following scenario: Let us consider the same example given by Rivest et al. [7], where a member of the cabinet wants to leak a very important and juicy information regarding the president of the nation, to the press. He/she has to leak the secret in an anonymous way, else the sender will be a spotted person in the cabinet. The press will not accept the information unless it is authenticated by one of the members of the cabinet. Here, if the information is so sensitive and should not be leaked until the corresponding authority in the press receives it, we should have confidential transmission of information. Thus, we require anonymity to safeguard the cabinet member who sends the information, the information should be authenticated for the authorities in the press to consider it and it should be confidential until it reaches the hands of the right person in the press. All the three properties are together achieved by the single primitive - "Ring Signcryption".

Related Work. Careful combination of IBC and ring signcryption yields a scheme which confidentially transmits an authenticated message anonymously to a specific receiver with the advantages of IBC. The first identity based ring signcryption scheme was proposed by Huang et al. [4]. Subsequently identity based ring signcryption schemes are reported in [10, 14, 12, 6, 5, 15].

Huang et al.'s scheme [4] was considered to be inefficient because the sender has to compute $n + 2$ bilinear pairing for signcrypting a message and the receiver has to compute 3 pairings for unsigncrypting a ciphertext. In an attempt to improve [4], Yu et al. [10] proposed a scheme entitled as identity based anonymous signcryption scheme which is essentially a ring signcryption scheme. The authors have claimed that their scheme [10] is adaptive chosen ciphertext (CCA2) secure but we show in this paper that their scheme is not at all secure even with respect to chosen plaintext attack (CPA). Fagen Li et al. proposed yet another scheme in [6], where they reduce the total number of pairing operations to 4 (one for signcryption and three for unsigncryption) but in this paper we show that their scheme is faulty with respect to adaptive chosen ciphertext attack [8]. Following that, Zhang et al. [12] proposed an authenticatable identity based anonymous signcryption scheme, which is also a ring signcryption scheme where the actual sender can prove to a valid verifier that the signcrypt was indeed produced by him. In [5], Fagen Li et al. have shown that Zhang et al.'s scheme does not resist adaptive chosen ciphertext attack and have proposed an improved authenticatable identity based anonymous signcryption scheme. We show that the improvement proposed by Fagen Li et al. [5] is also not adaptive chosen ciphertext secure. Lijun et al. proposed an identity based ring signature and ring signcryption scheme in [15], whose work was followed by Zhu et al. [14] who also proposed an identity based ring signcryption scheme. In this paper we show that the former one [15] is not secure against chosen plaintext attack and the latter one [14] is not adaptive chosen ciphertext secure.

Our Contribution. Anonymous signcryption is another nomenclature for ring signcryption, so both these primitives have the same functionalities. An authenticatable anonymous signcryption is a ring signcryption scheme, which has to satisfy the security properties of ring signcryption with an additional property that an actual sender can expose himself/herself with a brief interaction with the verifier at a later point of time. We show, attacks on confidentiality and anonymity of the scheme in [6], the schemes reported in [10] and [15] does not withstand chosen plaintext attack, and the schemes reported in [5] and [14] does not resist chosen ciphertext attack, by demonstrating the attack on each scheme independently. We also provide a new scheme which is IND-CCA2 secure (indistinguishable against adaptive chosen ciphertext attack) and EUF-CMA secure (existentially unforgeable against adaptive chosen message attack). Note that these are the strongest security requirements for any signcryption scheme. The formal proof of our new scheme is given in the random oracle model. Finally, a comparison with the only existing correct scheme by Huang et al. [4] shows that our scheme is the most efficient identity based ring signcryption scheme available till date.

2. Preliminaries

2.1. Bilinear Pairing

Let $\mathbb{G}_1$ be an additive cyclic group generated by P, with prime order q, and $\mathbb{G}_2$ be a multiplicative cyclic group of the same order q. A bilinear pairing is a map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ with the following properties.

- ***Bilinearity.*** For all $P, Q, R \in \mathbb{G}_1$,
  - $\hat{e}(P+Q,R) = \hat{e}(P,R)\hat{e}(Q,R)$
  - $\hat{e}(P,Q+R) = \hat{e}(P,Q)\hat{e}(P,R)$
  - $\hat{e}(aP,bQ) = \hat{e}(P,Q)^{ab}$
- ***Non-Degeneracy.*** There exist $P, Q \in \mathbb{G}_1$ such that $\hat{e}(P,Q) \neq I_{\mathbb{G}_2}$, where $I_{\mathbb{G}_2}$ is the identity in $\mathbb{G}_2$.
- ***Computability.*** There exists an efficient algorithm to compute $\hat{e}(P,Q)$ for all $P, Q \in \mathbb{G}_1$.

2.2. Computational Bilinear Diffie-Hellman Problem (CBDHP)

Definition 1. Given $(P, aP, bP, cP) \in \mathbb{G}_1^4$ for unknown $a, b, c \in \mathbb{Z}_q^*$, the CBDH problem in $(\mathbb{G}_1, \mathbb{G}_2, \hat{e})$ is to compute $\hat{e}(P,P)^{abc} \in \mathbb{G}_2$. The advantage of any probabilistic polynomial time algorithm $\mathcal{A}$ in solving the CBDH problem in $\mathbb{G}_1$ is defined as:

$$Adv_{\mathcal{A}}^{CBDH} = Pr\left[\mathcal{A}(P,aP,bP,cP) = \hat{e}(P,P)^{abc} | a,b,c \in \mathbb{Z}_q^*\right]$$

The *CBDH Assumption* is that, for any probabilistic polynomial time algorithm $\mathcal{A}$, the advantage $Adv_{\mathcal{A}}^{CBDH}$ is negligibly small.

2.3. Computation Diffie-Hellman Problem (CDHP)

Definition 2. Given $(P, aP, bP) \in \mathbb{G}_1^3$ for unknown $a, b \in \mathbb{Z}_q^*$, the CDH problem in $\mathbb{G}_1$ is to compute abP. The advantage of any probabilistic polynomial time algorithm $\mathcal{A}$ in solving the CDH problem in $\mathbb{G}_1$ is defined as:

$$Adv_{\mathcal{A}}^{CDH} = Pr\left[\mathcal{A}(P,aP,bP) = abP \mid a,b \in \mathbb{Z}_q^*\right]$$

The *CDH Assumption* is that, for any probabilistic polynomial time algorithm $\mathcal{A}$, the advantage $Adv_{\mathcal{A}}^{CDH}$ is negligibly small.

2.4. Notations used in this paper

To have a better understanding and to enhance the readability and clarity, we use the following notations throughout the paper.

$\mathcal{U}_i$ - User with identity ID_i.

$\mathcal{U} = \{\mathcal{U}_i\}_{(i=1\,to\,n)}$ - Group of users in the ring (including the actual sender).

$\mathcal{M}$ - Message space.

m - Message.

l - Number of bits used to represent m.

Q_i - Public key corresponding to ID_i.

D_i - Private key corresponding to ID_i.

$ID_{\mathbb{S}}$ - Identity of the sender.

$ID_{\mathbb{R}}$ - Identity of the receiver.

$Q_{\mathbb{S}}$ - Public key of the sender.

$Q_{\mathbb{R}}$ - Public key of the receiver.

$D_{\mathbb{S}}$ - Private key of the sender.

$D_{\mathbb{R}}$ - Private key of the receiver.

3. Formal Security Model for Identity Based Ring Signcryption

3.1. Generic Scheme

A generic identity based ring signcryption scheme consists of the following four algorithms.

Setup(κ): Given a security parameter κ, the private key generator (PKG) generates the systems public parameters *params* and the corresponding master private key *msk* that is kept secret.

Extract(ID_i): Given a user identity ID_i, the PKG computes the corresponding private key D_i and sends D_i to ID_i via a secure channel.

Signcrypt(m,$\mathcal{U}$, $D_{\mathbb{S}}$, $ID_{\mathbb{R}}$): This algorithm takes a message $m \in \mathcal{M}$, a receiver with identity $ID_{\mathbb{R}}$, the senders private key $D_{\mathbb{S}}$ and an ad-hoc group of ring members $\mathcal{U}$ with identities $\{ID_1, \ldots, ID_n\}$ as input and outputs a ring signcryption C. This algorithm is executed by a sender with identity $ID_{\mathbb{S}} \in \mathcal{U}$. $ID_{\mathbb{R}}$ may or may not be in $\mathcal{U}$.

Unsigncrypt(C,$\mathcal{U}$, $D_{\mathbb{R}}$): This algorithm takes the ring signcryption C, the ring members(say $\mathcal{U} = \{\mathcal{U}_i\}_{(i=1\,to\,n)}$) and the private key $D_{\mathbb{R}}$ of the receiver $ID_{\mathbb{R}}$ as input and produces the plaintext m, if C is a valid ring signcryption of m from a member of the ring $\mathcal{U}$ to $ID_{\mathbb{R}}$ or "*Invalid*", if C is an invalid ring signcryption. This algorithm is executed by a receiver $ID_{\mathbb{R}}$.

3.2. Security Notion

The formal security definition of signcryption was given by Baek et al. in [1]. The security requirements for identity based ring signcryption were defined by Huang et al. [4]. We extend the security model given in [4] by incorporating security against insider attacks. The security model is defined as follows.

Definition 3. An identity based ring signcryption (IRSC) is indistinguishable against adaptive chosen ciphertext attacks (IND-IRSC-CCA2) if there exists no polynomially bounded adversary having non-negligible advantage in the following game:

Setup Phase: The challenger $\mathcal{C}$ runs the *Setup* algorithm with a security parameter κ and sends the system parameters *params* to the adversary $\mathcal{A}$ and keeps the master private key *msk* secret.

First Phase: $\mathcal{A}$ asks polynomially bounded number of queries to the oracles provided to $\mathcal{A}$ by $\mathcal{C}$. The description of the queries in the first phase are listed below:

- ***Key Extraction query***: $\mathcal{A}$ sends to $\mathcal{C}$ an identity ID_i corresponding to $\mathcal{U}_i$ and receives the private key D_i corresponding to ID_i.
- ***Signcryption query:*** $\mathcal{A}$ sends to $\mathcal{C}$ a set of users $\mathcal{U}$, a receiver identity $ID_{\mathbb{R}}$ and a plaintext $m \in_R \mathcal{M}$. $\mathcal{A}$ also specifies the sender $\mathcal{U}_{\mathbb{S}} \in \mathcal{U}$ whose identity is $ID_{\mathbb{S}}$. Then $\mathcal{C}$ signcrypts m from $ID_{\mathbb{S}}$ to $ID_{\mathbb{R}}$ with $D_{\mathbb{S}}$ and sends the result to $\mathcal{A}$.
- ***Unsigncryption query:*** $\mathcal{A}$ produces a set of users $\mathcal{U}$, a receiver identity $ID_{\mathbb{R}}$, and a ring signcryption C. $\mathcal{C}$ generates the private key $D_{\mathbb{R}}$ by querying the *Key Extraction* oracle. $\mathcal{C}$ unsigncrypts C using $D_{\mathbb{R}}$ and returns m if C is a valid ring signcryption from $\mathcal{U}$ to $ID_{\mathbb{R}}$ else outputs "*Invalid*".

$\mathcal{A}$ queries the various oracles adaptively, i.e. the current oracle requests may depend on the response to the previous oracle queries.

Challenge: $\mathcal{A}$ chooses two plaintexts $\{m_0, m_1\} \in \mathcal{M}$ of equal length, a set of n users $\mathcal{U}$ and a receiver identity $ID_{\mathbb{R}}$ and sends them to $\mathcal{C}$. $\mathcal{A}$ should not have queried the private key corresponding to $ID_{\mathbb{R}}$ in the first phase. $\mathcal{C}$ now chooses a bit $\delta \in_R \{0, 1\}$ and computes the challenge ring signcryption C^* of m_δ, and sends C^* to $\mathcal{A}$.

Second Phase: $\mathcal{A}$ performs polynomially bounded number of requests just like the first phase, with the restrictions that $\mathcal{A}$ cannot make *Key Extraction* query on $ID_{\mathbb{R}}$ and should not query for unsigncryption query on C^*. It should be noted that $ID_{\mathbb{R}}$ can be included as a ring member in $\mathcal{U}$, but $\mathcal{A}$ cannot query the private key of $ID_{\mathbb{R}}$.

Guess: Finally, $\mathcal{A}$ produces a bit δ' and wins the game if $\delta' = \delta$. The success probability is defined by:

$$Succ_{\mathcal{A}}^{IND-IRSC-CCA2}(\kappa) = \frac{1}{2} + \epsilon$$

Here, ϵ is called the advantage for the adversary in the attack.

Note: The difference between the security model for confidentiality in [4] and our model is, we allow the adversary to access the private key of the ring members (selected by the adversary during the challenge phase) and restrict access to the private key of the receiver of the challenge ring signcryption. But in [4], the adversary is not allowed to access the private keys of the ring members and the receiver (of the challenge ring signcryption).

Definition 4. An identity based ring signcryption scheme (IRSC) is said to be existentially unforgeable against adaptive chosen messages attacks (EUF-IRSC-CMA) if no polynomially bounded adversary has a non-negligible advantage in the following game:

Setup Phase: The challenger runs the *Setup* algorithm with a security parameter κ and gives the system parameters to the adversary $\mathcal{A}$.

Training Phase: $\mathcal{A}$ performs polynomially bounded number of queries as described in First Phase of definition 3. The queries may be adaptive, i.e. the current query may depend on the previous query responses.

Existential Forgery: Finally, $\mathcal{A}$ produces a new triple $(\mathcal{U}, ID_{\mathbb{R}}, C)$ (i.e. a triple that was not produced by the signcryption oracle), where the private keys of the users in the group $\mathcal{U}$ were not queried in the training phase. $\mathcal{A}$ wins the game if the result of the $Unsigncryption$ $(\mathcal{U}, ID_{\mathbb{R}}, C)$ is not "*Invalid*", in other words C is a valid signcrypt of some message $m \in \mathcal{M}$. It should be noted that $ID_{\mathbb{R}}$ can also be member of the ring $\mathcal{U}$ and in that case the private key of $ID_{\mathbb{R}}$ should not be queried by $\mathcal{A}$.

Note: The difference between the security model for unforgeability in [4] and our model is, we do not allow the adversary to access the private key of the ring members (selected by the adversary during the generation of the forgery) but the adversary is given access to the private key of the receiver of the forged ring signcryption. However, in [4], the adversary is not allowed to access the private keys of the ring members as well as the receiver (of the forged ring signcryption). A signcryption scheme is referred to be insider secure, if the scheme is secure even if the adversary is given access to the private key of the sender during the confidentiality game (sender chosen for the challenge signcryption) and the private key of the receiver during the unforgeability game (receiver chosen for the forgery).

4. Attacks on Various Ring Signcryption Schemes

This section gives an overview of several schemes and the attacks corresponding to them. We consider Li et al.'s IRSC scheme [6] and show the weakness in the anonymity and confidentiality of the scheme. Following which we review Yu et al.'s [10] anonymous signcryption scheme and show an attack on the confidentiality of the scheme. Next we consider Fagen Li et al.'s [5] authenticatable anonymous signcryption scheme and show that the scheme is not CCA2 secure. The next scheme we consider is Lijun et al.'s [15] identity based ring signcryption scheme, on which we demonstrate a CPA attack. Finally we review Zhu et al.'s [14] scheme and show that the scheme is not CCA2 secure.

4.1. Overview of Li et al. IRSC Scheme [6]

Li et al. presented an efficient identity-based ring signcryption scheme in [6]. This scheme does not use any pairing computation in ring signcryption generation and uses only two pairing for ring unsigncryption. This scheme is identity-based and it comprises of four algorithms namely: Setup, Extract, Signcrypt and Unsigncrypt, which we describe below.

Setup: The setup algorithm is run by the PKG. Given a security parameter κ as input, this algorithm performs the following:

- Choose an additive cyclic group $\mathbb{G}_1$, a multiplicative cyclic group $\mathbb{G}_2$, both of the same prime order q, $\hat{e}$ an admissible bilinear pairing given by $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$. Define three hash functions $H_1 : \{0,1\}^* \rightarrow \mathbb{G}_1$, $H_2 : \mathbb{G}_2 \rightarrow \{0,1\}^{n_1}$ and $H_3 : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$. Choose master private key $s \in_R \mathbb{Z}_q^* (Msk = s)$ and set master public key $P_{pub} = sP$, where P is a generator of $\mathbb{G}_1$. Also, choose a secure symmetric cipher (E, D). The system parameters $params$ are $(\mathbb{G}_1, \mathbb{G}_2, n_1, \hat{e}, q, P, P_{pub}, E, D, H_1, H_2, H_3)$.

Extract: Given the identity of an user ID_A, compute the private/public key pair $\langle Q_A, S_A \rangle$ as follows:

- Public key $Q_A = H_1(ID_A) \in \mathbb{G}_1$.
- private key $S_A = sQ_A$.
- PKG sends S_A to the user through secure channel.

Signcrypt: User $ID_{\mathbb{S}}$ for generating a ring signcryption provides the message m, the set of ring members $\mathcal{U}$=$\{U_1$,U_2, $\ldots$,$U_n\}$, the identity of the actual sender $ID_{\mathbb{S}} \in \mathcal{U}$, the private key S_i of $ID_{\mathbb{S}}$ and the receiver identity $ID_{\mathbb{R}}$ as input to the Signcrypt algorithm. This algorithms generates a valid ring signcryption on m with ring members $\mathcal{U}$ as senders and $ID_{\mathbb{R}}$ as receiver. This is done by performing the following:

- Choose $r_{\mathbb{S}} \in_R \mathbb{Z}_q^*$ and compute $X = r_{\mathbb{S}} Q_{\mathbb{S}}$.
- Compute $k = H_2(\hat{e}(r_{\mathbb{S}} S_{\mathbb{S}}, Q_{\mathbb{R}}))$.
- Compute $c = E_k(m)$.
- For all $i \in \{1, 2, \ldots, n\}$, $i \neq \mathbb{S}$, choose $a_i \in_R \mathbb{Z}_q^*$, compute $R_i = a_i P$ and h_i=$H_3(c \| \mathcal{U} \| R_i)$.
- Compute $R_{\mathbb{S}} = X - \sum_{i=1, i \neq \mathbb{S}}^{n} \{R_i + h_i Q_i\}$.
- Compute $h_{\mathbb{S}} = H_3(c \| \mathcal{U} \| R_{\mathbb{S}})$ and $V = (h_{\mathbb{S}}$+$r_{\mathbb{S}}) S_{\mathbb{S}}$.
- Output the ring signcryption $\sigma = \{\mathcal{U}, X, c, \bigcup_{i=1}^{n} \{R_i\}, V\}$ to $ID_{\mathbb{S}}$.

Unsigncrypt: For unsigncrypting any ring signcryption $\sigma = \{\mathcal{U}, X, c, \bigcup_{i=1}^{n} \{R_i\}, V\}$ from $ID_{\mathbb{S}}$ to $ID_{\mathbb{R}}$, the receiver $ID_{\mathbb{R}}$ provides the ring signcryption σ, the receiver identity $ID_{\mathbb{R}}$, private key $\S_{\mathbb{R}}$ of receiver $ID_{\mathbb{R}}$ as input to Unsigncrypt algorithm. Unsigncryption is carried out by doing the computations given below:

- Compute $k' = H_2(\hat{e}(X, S_{\mathbb{R}}))$.
- Recover the message $m = D'_k(c)$.
- Compute h_i=$H_0(c \| \mathcal{U} \| R_i)$ for all $i \in \{1, 2, ..n\}$.
- Check whether $\hat{e}(P_{pub}, \sum_{i=1}^{n} (R_i + h_i Q_i)) \overset{?}{=} \hat{e}(P, V)$.
- Return the message m if σ is a valid signcryption on message m from $ID_{\mathbb{S}}$ to $ID_{\mathbb{R}}$. Else, return "*Invalid*".

4.1.1. Attacks on the IRSC Scheme

We demonstrate two different attacks on [6]. The first attack is on the anonymity and the second attack is on the confidentiality the scheme.

Attack on Anonymity: We show that the ring signcryption scheme in [6] does not provide anonymity. Any passive observer including the receiver, who is in possession of a ring signcryption can identify the sender in this scheme. Let m be any message and $\sigma = \{\mathcal{U} = \{ID_1, ID_2, \ldots, ID_n\}, X, c, \bigcup_{i=1}^{n}\{R_i\}, V\}$ be the ring signcryption on m from the ring $\mathcal{U}$ to $ID_{\mathbb{R}}$ and $ID_{\mathbb{S}} \in \mathcal{U}$ be the actual sender. On seeing the ring signcryption σ anyone can do the following operations to identify the actual sender $ID_{\mathbb{S}} \in \mathcal{U}$. It is to be noted that the private key of any $ID_i \in \mathcal{U}$ or $ID_{\mathbb{R}}$ is not required during this computation.

Anyone can do the following to identify the actual signer in the ring. For all values of i ($i = 1$ to n) perform the following:

- Compute $h_i = H_3(c\|\mathcal{U}\|R_i)$, ($c, \mathcal{U}, R_i$ are taken from the cipher-text).
- Check whether $\hat{e}(V, P) \overset{?}{=} \hat{e}(h_i Q_i + X, sP)$. — (1)
- If the check holds for some value of i then ID_i is the actual sender.

The following lemmas, *Lemma* 1 and *Lemma* 2 will prove that the test given above (equation (1)) is valid.

Lemma 1: Let $\mathcal{H}_{\mathbb{S}} = X + h_{\mathbb{S}} Q_{\mathbb{S}}$ where $U_{\mathbb{S}}$ is the actual signer. Let $R' = \hat{e}(V, P)$, then $R' = \hat{e}(\mathcal{H}_{\mathbb{S}}, P_{pub})$.

Proof:

$$\begin{aligned} \mathcal{H}_{\mathbb{S}} &= X + h_{\mathbb{S}} Q_{\mathbb{S}} = (r_{\mathbb{S}} + h_{\mathbb{S}}) Q_{\mathbb{S}} \text{ and} \\ R' &= \hat{e}(V, P) = \hat{e}((r_{\mathbb{S}} + h_{\mathbb{S}}) S_{\mathbb{S}}, P) \\ &= \hat{e}((r_{\mathbb{S}} + h_{\mathbb{S}}) Q_{\mathbb{S}}, P_{pub}) = \hat{e}(\mathcal{H}_{\mathbb{S}}, P_{pub}) \end{aligned}$$

Lemma 2: Let $\mathcal{H}_i = X + h_i Q_i$ where $U_i \in U$ is the not the actual signer. Let $R' = \hat{e}(V, P)$, then $R' \neq \hat{e}(\mathcal{H}_i, P_{pub})$.

Proof:

$$\begin{aligned} H_i &= X + h_i Q_i = r_{\mathbb{S}} Q_{\mathbb{S}} + h_i Q_i \text{ and} \\ R' &= \hat{e}(V, P) = \hat{e}((r_{\mathbb{S}} + h_{\mathbb{S}}) S_{\mathbb{S}}, P) \\ &= \hat{e}((r_{\mathbb{S}} + h_{\mathbb{S}}) Q_{\mathbb{S}}, P_{pub}) \\ &\neq \hat{e}(r_{\mathbb{S}} Q_{\mathbb{S}} + h_i Q_i, P_{pub}) \\ &= \hat{e}(\mathcal{H}_i, P_{pub}) \end{aligned}$$

From *Lemma* 1 and *Lemma* 2 it is clear that $R' = \mathcal{H}_i$ if and only if $i = \mathbb{S}$.

Attack on Confidentiality: The scheme is not CCA2 secure. As per the security model of [6], during the *Challenge Phase* of confidentiality game, the adversary $\mathcal{A}$ provides two messages m_0 and m_1 and a set of ring members $\mathcal{U} = \{ID_1, ID_2, \ldots, ID_n\}$ including the actual sender $ID_{\mathbb{S}}$ to $\mathcal{C}$ (Note that $\mathcal{A}$ does not know the actual sender $ID_{\mathbb{S}}$). $\mathcal{C}$ selects randomly a bit b and builds the challenge ring signcryption $\sigma = \{\mathcal{U}, X, c, \bigcup_{i=1}^{n}\{R_i\}, V\}$ on message m_δ from the ring $\mathcal{U}$ to ID_T. $\mathcal{A}$ is given access to the secret key of all users, except the target receiver ID_T and members of the ring $\mathcal{U}$. Now, $\mathcal{A}$ can perform the following,

- Set $X^* = X$ and $c^* = c$.
- Form a new ring $\mathcal{U}^* = \{U_1, U_2 \ldots, U_t\}$ with the property that $\mathcal{U}^* \nsubseteq \mathcal{U}$ and also $\mathcal{A}$ knows the secret key of at least one user say U_j, $j \in \{1, 2, \ldots, t\}$. Let $U_{\mathbb{S}*}$ be a user from ring $\mathcal{U}^*$, for which $\mathcal{A}$ knows the private key.
- For all $j \in \{1, 2, \ldots, t\}$, $j \neq \mathbb{S}^*$, choose $a_j \in_R \mathbb{Z}_q^*$, compute $R_j = a_j P$ and $h_j{=}H_3(c\|\mathcal{U}\|R_j)$.
- Choose a random $r_{\mathbb{S}^*} \in \mathbb{Z}_q^*$ and compute $R_{\mathbb{S}^*} = r_{\mathbb{S}^*} Q_{\mathbb{S}^*} - \sum_{j=1,2,j\neq\mathbb{S}^*}^{n}\{R_j + h_j Q_j\}$.
- Compute $h_{\mathbb{S}^*} = H_3(c\|\mathcal{U}\|R_{\mathbb{S}})$ and $V* = (h_{\mathbb{S}^*} {+} r_{\mathbb{S}^*}) S_{\mathbb{S}^*}$.
- Set $\sigma^* = \{\mathcal{U}^*, X^*, c^*, \bigcup_{j=1}^{t}\{R_j\}, V^*\}$.
- σ^* is entirely different from the challenge signcryption σ and hence $\mathcal{A}$ can request the *Unsigncrypt* oracle for the unsigncryption of σ^* from ring $\mathcal{U}^*$ to receiver ID_T.

The challenger will correctly respond with m_δ.

Hence, $\mathcal{A}$ can exactly find whether σ is a signcryption of m_0 or m_1 without solving any hard problem. Thus, we break the confidentiality of the Li et al.'s identity-based ring signcryption scheme. To conclude the proof, we first show that σ^* passes the verification test and then show that σ^* is indeed encryption of m_δ.

Validity of σ^*:

$$\begin{aligned} \hat{e}(P_{pub}, \textstyle\sum_{j=1}^{t}(R_j + h_j Q_j)) &= \hat{e}((r_{\mathbb{S}^*} + h_{\mathbb{S}^*}) Q_{\mathbb{S}*}, P_{pub}) \\ &= \hat{e}((r_{\mathbb{S}^*} + h_{\mathbb{S}^*}) S_{\mathbb{S}*}, P) \\ &= \hat{e}(V^*, P) \end{aligned}$$

4.2. Overview of Anonymous Signcryption (ASC) Scheme of Yu et al.

Yu et al.'s ASC scheme [10] consists of four algorithms namely: *Setup*, *KeyGen*, *Signcryption* and *Unsigncryption*, which we describe below:

Setup(κ, l)***:*** Here, κ and l are the security parameters.

- Choose $\mathbb{G}_1$, $\mathbb{G}_2$ of same order q and a random generator P of $\mathbb{G}_1$.
- Select the master private key $s \in_R \mathbb{Z}_q^*$.
- The master public key is computed as $P_{pub} = sP$.
- Select three strong public one-way hash functions: $H_1 : \{0,1\}^* \to \mathbb{G}_1^*$, $H_2 : \mathbb{G}_2 \to \{0,1\}^l$, $H_3 : \{0,1\}^* \to \mathbb{Z}_q^*$.
- Select an admissible pairing $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$.
- The public parameters of the scheme are given by *params*=($\mathbb{G}_1$, $\mathbb{G}_2$, $\hat{e}$, P, P_{pub}, H_1, H_2, H_3,q).

KeyGen(ID_i)***:*** Here, ID_i is the identity of the user $\mathcal{U}_i$. The PKG performs the following.

- The user public key is computed as $Q_i = H_1(ID_i)$
- The corresponding private key is $D_i = sQ_i$.
- The PKG sends D_i to the user $\mathcal{U}_i$ via a secure channel.

Signcryption$(\mathcal{U}, m, ID_{\mathbb{R}}, ID_{\mathbb{S}}, D_{\mathbb{S}})$***:*** In order to signcrypt a message m, the sender has to perform the following:

- Choose $r \in_R \mathbb{Z}_q^*$, compute $R = rP$, $R' = \hat{e}(P_{pub}, Q_{\mathbb{R}})^r$, $t = H_2(R')$ and $c = m \oplus t$.
- For all $i = 1$ to n and $i \neq \mathbb{S}$, choose $U_i \in_R \mathbb{G}_1$ and compute $h_i = H_3(m, t, \mathcal{U}, U_i)$.
- For $i = \mathbb{S}$ choose $r'_{\mathbb{S}} \in_R \mathbb{Z}_q^*$ and, compute $U_{\mathbb{S}} = r'_{\mathbb{S}} Q_{\mathbb{S}} - \Sigma_{i=1, i\neq\mathbb{S}}^n (U_i + h_i Q_i)$, $h_{\mathbb{S}} = H_3(m, t, \mathcal{U}, U_{\mathbb{S}})$ and $V = (h_{\mathbb{S}} + r'_{\mathbb{S}}) D_{\mathbb{S}}$.

Finally, the sender outputs the ring signcryption $C = (\mathcal{U}$, c, R, $h_1, \ldots, h_n$, $U_1, \ldots, U_n$, $V)$.

Unsigncrypt$(C = (\mathcal{U}$***,*** c***,*** R***,*** h_1***,*** $\ldots$***,*** h_n***,***U_1***,*** $\ldots$***,*** U_n***,*** $V), D_{\mathbb{R}})$***:*** In order to unsigncrypt a ring signcryption C, the receiver has to perform the following:

- Compute $t' = H_2(\hat{e}(R, D_{\mathbb{R}}))$ and $m' = c \oplus t'$.
- For $i = 1$ to n, check whether $h'_i \stackrel{?}{=} H_3(m', t', \mathcal{U}, U_i)$.
- Check whether $\hat{e}(P_{pub}, \Sigma_{i=1}^n (U_i + h'_i Q_i)) \stackrel{?}{=} \hat{e}(P, V)$.

If all the n checks in the second step and the check in the third step are true, then output m' as the message, else output "*Invalid*".

4.2.1. Attack on ASC Scheme of Yu et al.

During the challenge phase of the confidentiality game, the challenger $\mathcal{C}$ receives two messages m_0 and m_1 from the adversary $\mathcal{A}$. The challenger chooses $\delta \in_R \{0,1\}$ and produces the challenge ring signcryption C^* using the message m_δ and delivers C^* to $\mathcal{A}$. Upon receipt of $C^* = (\mathcal{U}, c^*, R^*, h_1^*, \ldots, h_n^*, U_1^*, \ldots, U_n^*, V^*)$, $\mathcal{A}$ performs the following to check whether C^* is a signcryption of m_0 or m_1. (Since $\mathcal{A}$ knows both messages m_0 and m_1, $\mathcal{A}$ can perform the following computations.)

- Compute $t^* = c^* \oplus m_0$ and check whether $h_i \stackrel{?}{=} H_3(m_0, t^*, \mathcal{U}, U_i^*)$, for $i = 1$ to n. If all the n checks hold, then C^* is the ring signcryption corresponding to m_0.
- If any of the above checks does not hold, compute $t^* = c^* \oplus m_1$, check whether $h_i \stackrel{?}{=} H_3(m_1, t^*, \mathcal{U}, U_i^*)$, for $i = 1$ to n. If all the n checks hold then C^* is a valid ring signcryption for message m_1.
- At least one of the above checks should hold *true*, else C^* is an invalid ring signcryption.

Thus, $\mathcal{A}$ distinguishes the ring signcryption with out solving any hard problem. Here $\mathcal{A}$ does not interact with the challenger $\mathcal{C}$ after receiving the challenge ring signcryption C^*. Thus, our attack is indeed against the CPA security of the ASC scheme by Yu et al. reported in [10].

Remark: Informally, $\mathcal{A}$ is able to distinguish the ring signcryption because, the key component required to evaluate the hash value h_i is t' and it is available in $c = m_\delta \oplus t'$. $\mathcal{A}$ knows that m_δ is either m_0 or m_1 because m_0 and m_1 were chosen by $\mathcal{A}$ and submitted to $\mathcal{C}$ during the challenge phase by $\mathcal{A}$. Hence, $\mathcal{A}$ can find t' without having access to the private key of the receiver and this led to the break in confidentiality (CPA).

4.3. Overview of Authenticatable Anonymous Signcryption Scheme (AASC) of Fagen Li et al.

The AASC scheme of Fagen Li et al. [5] consists of the five algorithms. A secure symmetric key encryption scheme (E, D) is employed in this scheme where, E and D are the secure symmetric key encryption and decryption algorithms respectively.

Setup(κ)***:*** Here, κ is the security parameter.

- Choose $\mathbb{G}_1$, $\mathbb{G}_2$ of same order q and a random generator P of $\mathbb{G}_1$.
- Choose the master private key $s \in_R \mathbb{Z}_q^*$.
- Compute the master public key $P_{pub} = sP$.
- Select three strong public one-way hash functions $H_1 : \{0,1\}^* \rightarrow \mathbb{G}_1$, $H_2 : \mathbb{G}_2 \rightarrow \{0,1\}^l$, $H_3 : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$.
- Select an admissible pairing $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ and a secure symmetric key encryption system (E, D).
- The public parameters of the scheme are set to be *params*=($\mathbb{G}_1$, $\mathbb{G}_2$, $\hat{e}$, P, P_{pub}, H_1, H_2, H_3, E, D).

Extract(ID_i)***:*** Similar to the ***Extract***(ID_i) algorithm in 4.2.

Signcrypt$(\mathcal{U}, m, ID_{\mathbb{R}}, ID_{\mathbb{S}}, D_{\mathbb{S}})$***:*** In order to signcrypt a message m, the sender has to perform the following:

- Choose $r \in_R \mathbb{Z}_q^*$, compute $R = rP$, $k = H_2(\hat{e}\,(P_{pub}, Q_{\mathbb{R}})^r)$ and $c = E_k(m)$.
- For $i = 1$ to n, $i \neq \mathbb{S}$, choose $a_i \in_R \mathbb{Z}_q^*$, compute $U_i = a_i P$ and $h_i = H_3(c, \mathcal{U}, U_i)$.
- For $i = \mathbb{S}$, choose $a_{\mathbb{S}} \in_R \mathbb{Z}_q^*$ and compute $U_{\mathbb{S}} = a_{\mathbb{S}} Q_{\mathbb{S}} - \Sigma_{i=1 i\neq\mathbb{S}}^n (U_i + h_i Q_i)$.
- Compute $h_{\mathbb{S}} = H_3(c, \mathcal{U}, U_{\mathbb{S}})$ and $\sigma = (h_{\mathbb{S}} + a_{\mathbb{S}}) D_{\mathbb{S}}$.

Finally, the sender outputs the ring signcryption as $C = (\mathcal{U}, c, R, U_1, \ldots, U_n, \sigma)$.

Unsigncrypt$(C = (\mathcal{U}$***,*** c***,*** R***,*** U_1***,*** $\ldots$***,*** U_n***,*** $\sigma), D_{\mathbb{R}})$***:*** To unsigncrypt C, the receiver has to perform the following.

- Compute $k' = H_2(\hat{e}(R, D_{\mathbb{R}}))$ and recover $m' = D_{k'}(c)$.
- For $i = 1$ to n, compute $h'_i = H_3(c,\mathcal{U}, U_i)$.
- If $\hat{e}(P_{pub}, \Sigma_{i=1}^{n}(U_i + h'_iQ_i)) \overset{?}{=} \hat{e}(P, \sigma)$, accept C and the message m', else output "*Invalid*".

Authenticate(C): The actual sender $ID_{\mathbb{S}}$ can prove that the message m was indeed signcrypted by him by running the following interactive protocol.

- Sender - Choose $x \in_R \mathbb{Z}_q^*$, compute $\mu = \hat{e}(P, \sigma)^x$ and send μ to the verifier.
- Verifier - Choose $y \in_R \mathbb{Z}_q^*$ and send it to the sender.
- Sender - Compute $v = (x + y)(h_{\mathbb{S}} + a_{\mathbb{S}})$ and return v to the verifier.
- Verifier - Check whether $\hat{e}(P_{pub}, Q_{\mathbb{S}})^v \overset{?}{=} \mu.\hat{e}(P, \sigma)^y$ and accept if the check holds.

4.3.1. Attack on AASC Scheme of Fagen Li et al.

The attack on AASC scheme is quite tricky one and it shows that the model considered by the authors did not address explicitly the scenario of the attack we propose. On receiving the challenge ring signcryption $C^* = (\mathcal{U}^*, c^*, R^*,$ $U_1^*, \ldots, U_n^*, \sigma^*)$, in the challenge phase of the confidentiality game, $\mathcal{A}$ can find the message used for generating C^*. $\mathcal{A}$ knows the private keys of all the users except the receiver $ID_{\mathbb{R}}$ and the members of $\mathcal{U}^*$ (here, $\mathcal{U}^*$ is the group of adhoc members in the challenge ring signcryption C^*). Now, $\mathcal{A}$ chooses $\mathcal{U}_E^{'} \notin \mathcal{U}^*$ with identity string ID_E for which $\mathcal{A}$ knows the private key D_E. $\mathcal{A}$ distinguishes C^* as, whether it is a signcryption of m_0 or m_1, during the second phase of oracle queries by performing the following.

- Form a new group with η users who are totally different from $\mathcal{U}^*$. Let the new group be $\mathcal{U}^{'} = \{\mathcal{U}_1^{'}, \ldots, \mathcal{U}_\eta^{'}\}$, where $\mathcal{U}_E^{'} \in \mathcal{U}^{'}$ and $\mathcal{U}^{'} \neq \mathcal{U}^*$.
- For $i = 1$ to η, $i \neq E$, choose $a_i \in_R \mathbb{Z}_q^*$, compute $U_i^{'} = a_iP$ and $h_i^{'} = H_3(c^*,\mathcal{U}^{'}, U_i^{'})$.
- For $i = E$, choose $a_E \in_R \mathbb{Z}_q^*$, compute $U_E^{'} = a_EQ_E - \Sigma_{i=1,i\neq E}^{\eta}(U_i^{'} + h_i^{'}Q_i)$.
- Compute $h_E^{'} = H_3(c^*,\mathcal{U}^{'}, U_E^{'})$ and $\sigma^{'} = (h_E^{'}+a_E)D_E$.
- Now, $C^{'} = (\mathcal{U}^{'}, c^*, R^*, U_1^{'}, \ldots, U_\eta^{'}, \sigma^{'})$ is also a valid ring signcryption on the same message m_δ, which was used by $\mathcal{C}$ to generate C^* and $C^{'}$ is entirely different from C^*, since $\mathcal{U}^{'} \neq \mathcal{U}^*$. Thus, $\mathcal{A}$ can legally query the unsigncryption of $C^{'}$ during the second phase of the confidentiality game.
- Get the unsigncryption to $C^{'}$ from $\mathcal{C}$ (which results in the challenge message m_δ) and thus $\mathcal{A}$ concludes correctly whether C^* is the signcryption of m_0 or m_1.

Distinguishing the ring signcryption after the start of the second phase of interaction and a decryption query leads to a break in CCA2 security of the system. Thus, we claim that the AASC scheme by Fagen Li et al. [5] is not adaptive chosen ciphertext secure.

Remark: In this scheme, ring signcryption is achieved by using the *Encrypt-then-Sign* paradigm, where the signature part is a ring signature algorithm. This scheme lacks the binding between the encryption and signature; any adversary can alter the signature component of any ring signcryption and with the same receiver, i.e., the output of the encryption is alone used as input to for signature generation. This facilitates the adversary to generate a new valid signature and use it with the remaining components of the challenge ring signcryption, which forms a totally different valid ring signcryption. Now, the adversary can make use of the unsigncryption oracle to unsigncrypt the newly formed ring signcryption. Note that since the encryption part is same as the challenge ring signcryption and the signature part is varied, the newly formed ring signcryption yields the same message as in the challenge ring signcryption and this query is legal with respect to the security model..

4.4. Overview of Identity Based Ring Signcryption (IRSC) Scheme of Lijun et al.

The IRSC scheme of Lijun et al. [15] consists of the following four algorithms.

Setup(κ)***:*** Here, κ is the security parameters.

- Choose $\mathbb{G}_1$, $\mathbb{G}_2$ of same order q and a random generator P of $\mathbb{G}_1$.
- Choose the master private key $s \in_R \mathbb{Z}_q^*$.
- Compute $P_{pub} = sP$ as the master public key.
- Select three cryptographic hash functions $H_1 : \{0,1\}^* \rightarrow \mathbb{G}_1$, $H_2 : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$, $H_3 : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$.
- Select an admissible pairing $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$.
- The public parameters of the scheme are set to be *params*=($\mathbb{G}_1$, $\mathbb{G}_2$, $\hat{e}$, P, P_{pub}, H_1, H_2, H_3, q).

KeyGen (ID_i)***:*** Similar to the ***Extract***(ID_i) algorithm in 4.2.

Signcrypt$(\mathcal{U}, m, ID_{\mathbb{R}}, ID_{\mathbb{S}}, D_{\mathbb{S}})$***:*** In order to signcrypt the message m the sender performs the following:

- Choose $r_0 \in_R \mathbb{Z}_q^*$ and compute $R_0 = r_0P$, $W = r_0P_{pub}$.
- For $i = 1$ to n, $i \neq \mathbb{S}$, choose $r_i \in_R \mathbb{Z}_q^*$, compute $R_i = r_iP$ $h_i = H_2(m\|\mathcal{U}\|R_i\|R_0)$.
- For $i = \mathbb{S}$, choose $r_{\mathbb{S}} \in_R \mathbb{Z}_q^*$, compute $R_{\mathbb{S}} = r_{\mathbb{S}}P - \Sigma_{i=1,i\neq\mathbb{S}}^{n}(h_iQ_i)$, $h_{\mathbb{S}} = H_2(m\|\mathcal{U}\|R_{\mathbb{S}}\|R_0)$ and $V = h_{\mathbb{S}}D_{\mathbb{S}}+$ $\Sigma_{i=1}^{n}r_iP_{pub}$.

- Compute $y = \hat{e}(W, Q_{\mathbb{R}})$, $t = H_3(y)$, $c = m \oplus t$.

Finally the sender outputs the ciphertext as $C = (\mathcal{U}, c, V, R_0, R_1, \ldots, R_n)$.

Unsigncrypt$(C = (\mathcal{U}, c, V, R_0, R_1, \ldots, R_n), D_{\mathbb{R}})$**:** In-order to unsigncrypt C, the receiver performs the following.

- Compute $t' = H_3(\hat{e}(D_{\mathbb{R}}, R_0))$ and recover $m' = c \oplus t'$.
- For $i = 1$ to n, compute $h'_i = H_2(m \| \mathcal{U} \| R_i \| R_0)$.
- Check whether $\hat{e}(P_{pub}, \Sigma_{i=1}^{n}(R_i + h'_i Q_i)) \overset{?}{=} \hat{e}(P, V)$.

If all the n checks in the second step and the check in the third step are true, then output m' as the message, else output "*Invalid*".

4.4.1. Attack on IRSC Scheme of Lijun et al.

During the challenge phase of the confidentiality game, the challenger $\mathcal{C}$ receives two messages m_0 and m_1 from the adversary $\mathcal{A}$. The challenger chooses $\delta \in_R \{0, 1\}$ and generates the challenge ring signcryption C^* using the message m_δ and delivers C^* to $\mathcal{A}$. Upon receipt of $C^* = (\mathcal{U}, c^*, V^*, R_0^*, R_1^*, \ldots, R_n^*)$, $\mathcal{A}$ does the following to distinguish C^* as, whether C^* is the signcryption of m_0 or m_1. Since $\mathcal{A}$ knows both messages m_0 and m_1, $\mathcal{A}$ performs the following.

- Compute $h_i = H_2(m_0 \| \mathcal{U} \| R_i^* \| R_0^*)$ for $i = 1$ to n. (since R_i^*, R_0^* are known from the ring signcryption C^*).
- Check whether $\hat{e}(P_{pub}, \Sigma_{i=1}^{n}(R_i^* + h_i Q_i)) \overset{?}{=} \hat{e}(P, V^*)$. If this check holds, then C^* is a valid ring signcryption of m_0.
- If the above check does not hold, perform the previous two steps with m_0 replaced by m_1. If the ring signcryption was formed with one of the two messages m_0 or m_1, any one of the above checks will hold, else the ring signcryption C^* is an invalid one.

Thus, $\mathcal{A}$ can distinguish the challenge signcryption without knowing the key of the receiver in the challenge ring signcryption C^*.

Remark: The intuition behind the attack is, in the ring signcryption proposed by Lijun et al. [15] the ring signcryption can be verified if the message and the corresponding ring signcryption is known. During the confidentiality game the adversary $\mathcal{A}$ knows the message, which is either m_0 or m_1, and hence $\mathcal{A}$ can conclude whether C^* is a ring signcryption of m_0 or m_1.

4.5. Overview of IRSC Scheme of Zhu et al.

The IRSC scheme of Zhu et al. [14] consists of the following four algorithms.

Setup(κ, l)**:** Here, κ and l are the security parameters.

- Choose $\mathbb{G}_1$, $\mathbb{G}_2$ of same order q and a random generator P of $\mathbb{G}_1$.
- Choose the master private key $s \in_R \mathbb{Z}_q^*$ and compute the master public key to be $P_{pub} = sP$.
- Select four cryptographic hash functions $H_1 : \{0, 1\}^* \rightarrow \mathbb{G}_1^*$, $H_2 : \mathbb{G}_1^* \rightarrow \{0, 1\}^l$, $H_3 : \{0, 1\}^l \times \mathbb{G}_1 \rightarrow \{0, 1\}^l$, $H_4 : \{0, 1\}^* \rightarrow \mathbb{Z}_q^*$.
- Select an admissible pairing $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$.
- The public parameters of the scheme are set to be *params*=$(\mathbb{G}_1, \mathbb{G}_2, \hat{e}, P, P_{pub}, H_1, H_2, H_3, H_4, q)$.

KeyGen (ID_i)**:** Similar to the ***Extract***(ID_i) algorithm in 4.2.

Signcrypt$(\mathcal{U}, m, ID_{\mathbb{R}}, ID_{\mathbb{S}}, D_{\mathbb{S}})$**:** In order to signcrypt the message m, the sender performs the following:

- Choose $r \in_R \mathbb{Z}_q^*$, $\hat{m} \in_R \mathcal{M}$ and, compute $R_0 = rP$, $R^{'} = \hat{e}(rP_{pub}, Q_{\mathbb{R}})$, $k = H_2(R^{'})$, $c_1 = \hat{m} \oplus k$ and $c_2 = m \oplus H_3(\hat{m} || R_0)$.
- For $i = 1$ to n, $i \neq \mathbb{S}$, choose $U_i \in_R \mathbb{G}_1^*$ and compute $h_i = H_4(c_2 || U_i)$.
- For $i = \mathbb{S}$, choose $r' \in_R \mathbb{Z}_q^*$, compute $U_{\mathbb{S}} = r'Q_{\mathbb{S}} - \Sigma_{i=1, i \neq \mathbb{S}}^{n}(U_i + h_i Q_i)$, $h_{\mathbb{S}} = H_4(c_2 || U_{\mathbb{S}})$ and $V = (h_{\mathbb{S}} + r')D_{\mathbb{S}}$.

Finally, output the ring signcryption as $C = (\mathcal{U}, R_0, c_1, c_2, U_1, \ldots, U_n, V)$.

Unsigncrypt$(C = (\mathcal{U}, R_0, c_1, c_2, U_1, \ldots, U_n, V) , D_{\mathbb{R}})$**:** To unsigncrypt a ring signcryption C, the receiver performs the following.

- For $i = 1$ to n, compute $h'_i = H_4(c_2 || U_i)$.
- Check whether $\hat{e}(P_{pub}, \Sigma_{i=1}^{n}(U_i + h'_i Q_i)) \overset{?}{=} \hat{e}(P, V)$, if so, compute $k' = H_2(\hat{e}(R_0, D_{\mathbb{R}}))$, and recover $\hat{m}' = c_1 \oplus k'$ and $m' = c_2 \oplus H_3(\hat{m}' || R_0)$. Accept m' as the valid message.

Note: The actual scheme in [11] had typos in setup, keygen as well as signcryption algorithms. The definition of the hash function H_3 was inconsistent. Instead, of H_2, it was written as H_1 and instead of H_1, it was written H_0. Moreover, and instead of writing $U_{\mathbb{S}} = r'Q_{\mathbb{S}} - \Sigma_{i=1, i \neq \mathbb{S}}^{n}(U_i + h_i Q_i)$, it was written as $U_{\mathbb{S}} = r'Q_{\mathbb{S}} - \Sigma_{i=1, i \neq \mathbb{S}}^{n}(U_i + h_i Q_{\mathbb{S}})$. We have corrected all of them in our review, in order to maintain the consistency of the scheme.

4.5.1. Attack on IRSC Scheme of Zhu et al.

On receiving the challenge ring signcryption $C^* = (\mathcal{U}^*, R_0^*, c_1^*, c_2^*, U_1^*, \ldots, U_n^*, V^*)$, in the challenge phase of the confidentiality game, $\mathcal{A}$ can find the message used for generating C^*. $\mathcal{A}$ knows the private keys of all the users except the receiver $ID_{\mathbb{R}}$ and the members of $\mathcal{U}^*$ (here, $\mathcal{U}^*$ is the group of ad-hoc members in the challenge ring signcryption

C^*). Now, $\mathcal{A}$ chooses $\mathcal{U}_E^{'} \notin \mathcal{U}^*$ with identity string ID_E for which $\mathcal{A}$ knows the private key D_E. $\mathcal{A}$ performs the following steps to distinguish C^* as, whether it is a signcryption of m_0 or m_1, during the second phase of oracle queries by performing the following.

- Form a new group $\mathcal{U}^{'}$ with η members who are totally different from the users in $\mathcal{U}^*$ present in the challenge ring signcryption. Consider $\mathcal{U}^{'} = \{\mathcal{U}_1^{'}, \ldots, \mathcal{U}_\eta^{'}\}$ and $\mathcal{U}_E^{'} \in \mathcal{U}^{'}$ (The private key of $\mathcal{U}_E^{'}$ is known to $\mathcal{A}$).
- Choose a message m' and compute $c_2^{'} = c_2^* \oplus m'$.
- For all $i = 1$ to η and $i \neq E$, choose $U_i^{'} \in_R \mathbb{G}_1^*$ and compute $h_i^{'} = H_4(c_2^{'}||U_i^{\prime})$.
- For $i = E$, choose $r^{'} \in_R \mathbb{Z}_q^*$ and compute $U_E^{'} = r^{'}Q_A - \Sigma_{i=1}^{\eta}(U_i^{'} + h_i^{'}Q_i)$.
- Compute $h_E^{'} = H_4(c_2^{'}||U_E^{'})$ and $V^{'} = (r^{'} + h_E^{'})D_E$
- Now, $C^{'} = (\mathcal{U}^{'}, R_0^*, c_1^*, c_2^{'}, U_1^{'}, \ldots, U_n^{'}, V^{'})$ is a valid ring signcryption on message $m_\delta \oplus m'$.

Now, during the second phase of training, $\mathcal{A}$ requests the unsigncryption of C' to $\mathcal{C}$. Note that it is legal for $\mathcal{A}$ to ask for unsigncryption of C' because it is derived from C^* and not exactly the challenge ring signcryption C^*. $\mathcal{C}$ responds with $M = m_\delta \oplus m'$ as the output for the query. $\mathcal{A}$ now obtains $m_\delta = M \oplus m'$ and thus identifies the message in the challenge ring signcryption C^*.

Remark: This attack is possible due to the same reason as described in the remark for the attack stated in section 4.3.

5. New Ring Signcryption Scheme (New-IBRSC)

In this section, we present a new improved identity based ring signcryption scheme (New-IBRSC), taking into account the attacks carried out in the previous section. New-IBRSC consists of the following four algorithms:

Setup(κ)***:*** This algorithm is executed by the PKG to initialize the system by taking a security parameter κ as input.

- Choose an additive cyclic group $\mathbb{G}_1$, a multiplicative cyclic group $\mathbb{G}_2$, both cyclic with prime order q and a random generator P of the group $\mathbb{G}_1$.
- Select $s \in_R \mathbb{Z}_q^*$ as the master private key and compute the master public key $P_{pub} = sP$.
- Select four cryptographic hash functions $H_1 : \{0,1\}^* \rightarrow \mathbb{G}_1$, $H_2 : \mathbb{G}_2 \rightarrow \{0,1\}^{|\mathcal{M}|} \times \mathbb{Z}_q^* \times \mathbb{G}_1$, $H_3 : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$ and $H_4 : \{0,1\}^{|\mathcal{M}|} \times \mathbb{Z}_q^* \rightarrow \mathbb{Z}_q^*$.
- Selects a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$.
- The public parameter of the scheme is $params$=($\mathbb{G}_1$, $\mathbb{G}_2$, $\hat{e}$, P, P_{pub}, H_1, H_2, H_3, H_4, q).

Keygen(ID_i)***:*** This algorithm takes ID_i, the identity of a user $\mathcal{U}_i$ as input. The PKG who executes this algorithm computes the private key and public key for the user with identity ID_i as follows:

- The public key is computed as $Q_i = H_1(ID_i)$
- The corresponding private key is $D_i = sQ_i$.
- PKG sends D_i to user $\mathcal{U}_i$ via a secure channel.

Signcrypt($\mathcal{U}, m, ID_\mathbb{R}, Q_\mathbb{R}, ID_\mathbb{S}, D_\mathbb{S}$)***:*** For signcrypting a message m to the receiver $\mathcal{U}_\mathbb{R}$ with public key $Q_\mathbb{R}$ the sender with private key $D_\mathbb{S}$ and public key $Q_\mathbb{S}$ performs the following:

- Select n potential senders and forms an ad-hoc group $\mathcal{U}$, including its own identity $ID_\mathbb{S}$.
- Choose $w \in_R \mathbb{Z}_q^*$, compute $r = H_4(m, w)$, $U = rP$ and $\alpha = \hat{e}(P_{pub}, Q_\mathbb{R})^r$.
- For $i = 1$ to n, $i \neq \mathbb{S}$, choose $U_i \in_R \mathbb{G}_1$ and compute $h_i = H_3(m, U_i, \alpha, \mathcal{U}, Q_\mathbb{R})$.
- For $i = \mathbb{S}$, choose $r_\mathbb{S} \in_R \mathbb{Z}_q^*$ and, compute $U_\mathbb{S} = r_\mathbb{S}Q_\mathbb{S} - \Sigma_{i=1, i\neq\mathbb{S}}^{n}(U_i + h_iQ_i)$, $h_\mathbb{S} = H_3(m, U_\mathbb{S}, \alpha, \mathcal{U}, Q_\mathbb{R})$ and $V = (h_\mathbb{S} + r_\mathbb{S})D_\mathbb{S}$.
- Compute $y = (m\|w\|V) \oplus H_2(\alpha)$.

Finally, the sender outputs the ring signcryption $C = (y, \mathcal{U}, U, U_1, \ldots, U_n)$.

Unsigncrypt($C = (y, \mathcal{U}, U, U_1, \ldots, U_n), D_\mathbb{R}$***):*** The receiver $\mathcal{U}_\mathbb{R}$ with identity $ID_\mathbb{R}$ performs the following to unsigncrypt the ring signcryption C:

- Compute $\alpha' = \hat{e}(U, D_\mathbb{R})$, retrieve m', w' and V' as $(m'\|w'\|V') = y \oplus H_2(\alpha')$.
- Check whether $U \stackrel{?}{=} H_4(m', w')P$.
- For $i = 1$ to n, compute $h_i' = H_3(m', U_i, \alpha', \mathcal{U}, Q_\mathbb{R})$ and check whether $\hat{e}(P_{pub}, \Sigma_{i=1}^{n}(U_i + h_i'Q_i)) \stackrel{?}{=} \hat{e}(P, V')$.

If all the above checks hold, then accept C as a valid ring signcryption and the message m' as a valid message. If any one of them fail, return "*Invalid*".

Correctness: We show the correctness of the unsigncryption algorithm here. From the definition of $U_\mathbb{S}$ given in the signcryption algorithm, $\Sigma_{i=1}^{n}(U_i + h_i'Q_i) = r_\mathbb{S}Q_\mathbb{S} + h_\mathbb{S}'Q_\mathbb{S}$. Thus,

$$\begin{aligned}\text{LHS}=\ & \hat{e}(P_{pub}, \Sigma_{i=1}^{n}(U_i + h_i'Q_i)) = \hat{e}(sP, r_\mathbb{S}Q_\mathbb{S} + h_\mathbb{S}'Q_\mathbb{S}) \\ &= \hat{e}(P, (r_\mathbb{S} + h_\mathbb{S}')sQ_\mathbb{S}) = \hat{e}(P, V')\text{=RHS}\end{aligned}$$

Note that the above correctness holds only if $h_i = h_i'$ for ($i = 1$ to n).

6. Security Results for New-IBRSC:

New-IBRSC can be viewed as a signcryption scheme with the signature part replaced by the ring signature given in [2]. This composition does not induce any weakness in the anonymity property of the ring signature. The difference between the ring signature in [2] and New-IBRSC is the definition of the hash function H_3, which is used to compute h_i, for $i = 1$ to n. In New-IBRSC, the two additional components are α and $Q_{\mathbb{R}}$, where α is the session key established and $Q_{\mathbb{R}}$ is the public key of the receiver. The value α is computed as $\hat{e}(P_{pub}, Q_{\mathbb{R}})^r$, which does not provide any clue regarding the sender. Addition of α and $Q_{\mathbb{R}}$ to the hash function H_3 does not reveal any information regarding the identity of the sender. Hence the anonymity proof of New-IBRSC follows from the underlying identity based ring signature [2]. Therefore, we concentrate only on the security against adaptive chosen ciphertext attack (CCA2) and security against chosen message attack (CMA). We formally prove the security of the new identity based ring signcryption scheme (New-IBRSC), indistinguishable under chosen ciphertext attack (IND-New-IBRSC-CCA2) and existentially unforgeable under chosen message and identity attack (EUF-New-IBRSC-CMA) in the random oracle model. We consider the security model given in section 3 to prove the security of the New-IBRSC.

6.1. Confidentiality Proof of New-IBRSC (IND-IBRSC-CCA2):

Theorem 1. *If an IND-IBRSC-CCA2 adversary $\mathcal{A}$ has an advantage ϵ against New-IBRSC scheme, asking q_{H_i} (i = 1, 2, 3,4) hash queries to random oracles $\mathcal{O}_{H_i}$ (i = 1, 2, 3,4), q_e extract queries ($q_e = q_{e_1} + q_{e_2}$, where q_{e_1} and q_{e_2} are the number of extract queries in the first phase and second phase respectively), q_{sc} signcryption queries and q_{us} unsigncryption queries, then there exist an algorithm $\mathcal{C}$ that solves the CBDH problem with advantage $\epsilon\left(\frac{1}{q_{H_1} q_{H_2}}\right)$.*

Proof: The challenger $\mathcal{C}$ is challenged to solve an instance (P, aP, bP, cP) of the CBDHP. Assume that there is an adversary $\mathcal{A}$ who is capable of breaking the IND-IBRSC-CCA2 security of New-IBRSC with non-negligible advantage. $\mathcal{C}$ makes use of $\mathcal{A}$ to solve the CBDHP instance. $\mathcal{C}$ simulates the system with the various oracles $\mathcal{O}_{H_1}$, $\mathcal{O}_{H_2}$, $\mathcal{O}_{H_3}$, $\mathcal{O}_{Signcryption}$, $\mathcal{O}_{Unsigncryption}$ and allows $\mathcal{A}$ to make polynomially bounded number of queries, adaptively to these oracles. The game between $\mathcal{C}$ and $\mathcal{A}$ is demonstrated below:

Setup Phase: $\mathcal{C}$ simulates the system by setting up the system parameters in the following way.

- $\mathcal{C}$ chooses the groups $\mathbb{G}_1$ and $\mathbb{G}_2$ and the generator $P \in \mathbb{G}_1$ as given in CBDHP instance.
- Sets the master public key $P_{pub} = aP$, here $\mathcal{C}$ does not know a. $\mathcal{C}$ is using the aP value given in the instance of the CBDHP.
- Models the four hash functions as random oracles $\mathcal{O}_{H_1}$, $\mathcal{O}_{H_2}$, $\mathcal{O}_{H_3}$ and $\mathcal{O}_{H_4}$.
- Selects a bilinear pairing $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$.
- Delivers $\langle \mathbb{G}_1, \mathbb{G}_2, \hat{e}, P, P_{pub} \rangle$ to $\mathcal{A}$.

First Phase: To handle the oracle queries, $\mathcal{C}$ maintains four lists L_i, $(i = 1, 2, 3, 4)$ which keeps track of the responses given by $\mathcal{C}$ to the corresponding oracle ($\mathcal{O}_{H_1}$, $\mathcal{O}_{H_2}$, $\mathcal{O}_{H_3}$, $\mathcal{O}_{H_4}$) queries. $\mathcal{A}$ adaptively (means that, the input to the current query may depend on the outputs obtained for the previous queries) queries the various oracles in the first phase, which are handled by $\mathcal{C}$ as given below:

$\mathcal{O}_{H_1}$ **oracle Query:** We will make a simplifying assumption that $\mathcal{A}$ queries the $\mathcal{O}_{H_1}$ oracle with distinct identities in each query. There is no loss of generality due to this assumption, because, if the same identity is repeated, by definition the oracle consults the list L_1 and gives the same response. Thus, we assume that $\mathcal{A}$ asks q_{H_1} distinct queries for q_{H_1} distinct identities. Among this q_{H_1} identities, a random identity has to be selected as target identity and it is done as follows.

$\mathcal{C}$ selects a random index γ, where $1 \leq \gamma \leq q_{H_1}$. $\mathcal{C}$ does not reveal γ to $\mathcal{A}$. When $\mathcal{A}$ generates the γ^{th} query on ID_γ, $\mathcal{C}$ decides to fix ID_γ as target identity for the challenge phase. Moreover, $\mathcal{C}$ responds to $\mathcal{A}$ as follows:

- If it is the γ^{th} query, then $\mathcal{C}$ sets $Q_\gamma = bP$, returns Q_γ as the response to the query and stores $\langle ID_\gamma, Q_\gamma, * \rangle$ in the list L_1. Here, $\mathcal{C}$ does not know b. $\mathcal{C}$ is simply using the bP value given in the instance of the CBDHP.
- For all other queries, $\mathcal{C}$ chooses $x_i \in_R Z_q^*$ and sets $Q_i = x_i P$ and stores $\langle ID_i, Q_i, x_i \rangle$ in the list L_1.

$\mathcal{C}$ returns Q_i to $\mathcal{A}$. (Note that as the identities are assumed to be distinct, for each query, we create distinct entry and add in the list L_1).

$\mathcal{O}_{H_2}$ **oracle Query:** When $\mathcal{A}$ makes a query to this oracle with α as input, $\mathcal{C}$ retrieves h_2 from list L_2 and returns h_2 to $\mathcal{A}$; else, chooses a new h_2 randomly, stores $\langle \alpha, h_2 \rangle$ in L_2 and returns h_2 to $\mathcal{A}$.

$\mathcal{O}_{H_3}$ **oracle Query:** When $\mathcal{A}$ makes a query to this oracle with $(m, U_i, \alpha, \mathcal{U}, Q_{\mathbb{R}})$ as input, $\mathcal{C}$ retrieves $h_i^{(3)}$ from list L_3 and returns $h_i^{(3)}$ to $\mathcal{A}$; else, chooses a new $h_i^{(3)} \in_R \mathbb{Z}_q^*$ randomly, stores $\langle m, U_i, \alpha, \mathcal{U}, Q_{\mathbb{R}}, h_i^{(3)} \rangle$ in the list L_3 and returns $h_i^{(3)}$ to $\mathcal{A}$.

$\mathcal{O}_{H_4}$ **oracle Query:** When $\mathcal{A}$ makes a query to this oracle with (m, w) as input, $\mathcal{C}$ retrieves r from list L_4 and returns r to $\mathcal{A}$; else, chooses $r \in_R \mathbb{Z}_q^*$, stores $\langle m, w, r \rangle$ in L_4 and returns r to $\mathcal{A}$.

Extract Query: On getting a request for the private key of user $\mathcal{U}_i$ with identity ID_i, $\mathcal{C}$ aborts if $ID_i = ID_\gamma$. Else, $\mathcal{C}$ retrieves Q_i, x_i from list L_1 and returns $D_i = x_i aP = aQ_i$.

(**Note:** It is assumed throughout the confidentiality game, $\mathcal{A}$ queries $\mathcal{O}_{H_1}$ oracle with ID_i before querying other oracles with ID_i as input.)

$\mathcal{O}_{Signcryption}$**Query:** $\mathcal{A}$ chooses a message m, a set of n potential senders and forms an ad-hoc group $\mathcal{U}$ by fixing a sender $ID_{\mathbb{S}}$ and a receiver $ID_{\mathbb{R}}$ and sends them to $\mathcal{C}$. To respond correctly to the signcryption query on the plaintext m chosen by $\mathcal{A}$, $\mathcal{C}$ does the following:

$\mathcal{C}$ proceeds according to the signcryption algorithm when $ID_{\mathbb{S}} \neq ID_{\gamma}$. This is possible for $\mathcal{C}$ because $\mathcal{C}$ knows the private key $D_{\mathbb{S}}$ of the sender $ID_{\mathbb{S}}$.

If the sender's identity $ID_{\mathbb{S}} = ID_{\gamma}$ (i.e. when $\mathcal{C}$ does not know the private key corresponding to $ID_{\mathbb{S}}$), $\mathcal{C}$ cooks up a response as explained below:

- Choose $w \in_R \mathbb{Z}_q^*$, compute $r = H_4(m, w)$, $U = rP$ and $\alpha = \hat{e}(P_{pub}, Q_{\mathbb{R}})^r$.
- For $i = 1$ to n, $i \neq \mathbb{S}$, choose $U_i \in_R \mathbb{G}_1$ and query the oracle $\mathcal{O}_{H_3}$ and obtains the value $h_i^{(3)} = \mathcal{O}_{H_3}(m, U_i, \alpha, \mathcal{U}, Q_{\mathbb{R}})$.
- For $i = \mathbb{S}$,
  - Choose $r_{\mathbb{S}}, h_{\mathbb{S}}^{(3)} \in_R \mathbb{Z}_q^*$
  - Compute $U_{\mathbb{S}} = r_{\mathbb{S}}P - h_{\mathbb{S}}^{(3)}Q_{\mathbb{S}} - \Sigma_{i=1, i\neq\mathbb{S}}^{n}(U_i + h_i^{(3)}Q_i)$.
  - Add the tuple $\langle m, U_{\mathbb{S}}, \alpha, \mathcal{U}, Q_{\mathbb{R}}, h_{\mathbb{S}}^{(3)}\rangle$ to the list L_3.
- Compute $V = r_{\mathbb{S}}P_{pub}$
(***Note:*** Here $h_{\mathbb{S}}^{(3)}$ is not computed by $\mathcal{C}$, instead it is chosen at random and set as the output for the random oracle query $h_{\mathbb{S}}^{(3)} = \mathcal{O}_{H_3}(m, U_{\mathbb{S}}, \alpha, \mathcal{U}, Q_{\mathbb{R}})$. This is possible because the random oracles are manipulated by $\mathcal{C}$).
- Queries $h^{(2)} = \mathcal{O}_{H_2}(\alpha)$ and computes $y = (m\|w\|V) \oplus h^{(2)}$

Finally, $\mathcal{C}$ outputs the ring signcryption $C = (y, \mathcal{U}, U, U_1, \ldots, U_n)$ to $\mathcal{A}$ as the signcryption of m. The signcryption $C = (y, \mathcal{U}, U, U_1, \ldots, U_n)$ is considered as valid by $\mathcal{A}$ because C passes the verification tests as shown below:

From the definition of $U_{\mathbb{S}}$, $\Sigma_{i=1}^{n}(U_i + h_i' Q_i) = r_{\mathbb{S}}P$. Thus,

$$\begin{aligned}\hat{e}(P_{pub}, \Sigma_{i=1}^{n}(U_i + h_i'Q_i)) &= \hat{e}(aP, r_{\mathbb{S}}P)\\ &= \hat{e}(P, r_{\mathbb{S}}aP)\\ &= \hat{e}(P, r_{\mathbb{S}}P_{pub})\\ &= \hat{e}(P, V')\end{aligned}$$

$\mathcal{O}_{Unsigncryption}$ **Query:** Upon receiving an unsigncryption query on a ring signcryption $C = (y, \mathcal{U}, U, U_1, \ldots, U_n)$ with $ID_{\mathbb{R}}$ as receiver, $\mathcal{C}$ proceeds as follows:

$\mathcal{C}$ proceeds as per the unsigncryption algorithm, when $ID_{\mathbb{R}} \neq ID_{\gamma}$. Here, $\mathcal{C}$ can directly use the unsigncryption algorithm because, $\mathcal{C}$ knows the private key $D_{\mathbb{R}}$ of the receiver $ID_{\mathbb{R}}$.

If the receiver identity $ID_{\mathbb{R}} = ID_{\gamma}$ (i.e. When $\mathcal{C}$ does not know the private key corresponding to $ID_{\mathbb{R}}$), $\mathcal{C}$ generates the response as explained below:

- For a given signcryption $C = (y, \mathcal{U}, U, U_1, \ldots, U_n)$, a pair (m, α) is said to be a potential pair if $\langle m, U_i, \alpha, \mathcal{U}, Q_{\mathbb{R}}\rangle \in L_3$ for all $i = 1$ to n. Let $\overline{M}$ denote the set of all potential pairs for C.
- For each pair $(m, \alpha) \in \overline{M}$, the challenger $\mathcal{C}$ performs the following:
  - Retrieves m', w' and V' as $m'\|w'\|V' = y \oplus \mathcal{O}_{H_2}(\alpha)$.
  - Checks whether $m' \overset{?}{=} m$ and checks $\mathcal{O}_{H_4}(m', w') P \overset{?}{=} U$. If true, then $\mathcal{C}$ obtains the value $h_i^{(3)'} = \mathcal{O}_{H_3}(m, U_i, \alpha, \mathcal{U}, Q_{\mathbb{R}})$, for $i = 1$ to n from the list L_3 and checks whether $\hat{e}(P_{pub}, \Sigma_{i=1}^{n}(U_i + h_i^{(3)'}Q_i)) \overset{?}{=} \hat{e}(P, V)$.
- The first time when all checks in (2) passes, $\mathcal{C}$ outputs the corresponding m' and halts.
- If every $(m, \alpha) \in \overline{M}$ obtained in step (1) fails the checks in step (2), then $\mathcal{C}$ outputs "*Invalid*" and halts.

Challenge Phase: Finally, $\mathcal{A}$ chooses two equal length plaintexts m_0, $m_1 \in \mathcal{M}$, the set of ring members $\mathcal{U}^* = \{ID_i\}_{(i=1\,to\,n^*)}$, a sender identity $ID_{\mathbb{S}} \in \mathcal{U}^*$ and a receiver identity $ID_{\mathbb{R}}$ on which $\mathcal{A}$ wants to be challenged and sends them to $\mathcal{C}$. $\mathcal{A}$ should not have queried the private key corresponding to $ID_{\mathbb{R}}$ in the first phase. $\mathcal{C}$ aborts, if $ID_{\mathbb{R}} \neq ID_{\gamma}$; else, $\mathcal{C}$ chooses a bit $\delta \in_R \{0, 1\}$ and computes the challenge ring signcryption C^* of m_δ as follows:

- Sets $U^* = cP$. (Note that the challenger does not know c but uses the cP value available in the instance of CBDHP.)
- Chooses $\{U_i^*\}_{(i=1\,to\,n^*)}$ randomly from $\mathbb{G}_1$ and $y^* \in_R \{0,1\}^{|\mathcal{M}|} \times \mathbb{Z}_Q^* \times \mathbb{G}_1$ and outputs $C^* = (y^*, \mathcal{U}^*, U^*, U_1^*, \ldots, U_n^*)$.

(Note that C^* is not constructed according to the signcryption algorithm of the scheme but made up of random values.)

Second Phase: On getting the challenge ring signcryption C^*, $\mathcal{A}$ is allowed to interact with $\mathcal{C}$ as in the first phase. This time, $\mathcal{A}$ is not given access to the private key of $ID_{\mathbb{R}}$ and is also restricted from querying the decryption oracle for the ring unsigncryption of C^*.

Guess: At the end of the second phase, $\mathcal{A}$ returns its guess. $\mathcal{C}$ ignores the answer from $\mathcal{A}$, picks a random tuple $\langle\alpha, h_2\rangle$ from list L_2 and returns the corresponding α as the solution to the CBDHP instance. Since the challenge ciphertext C^* given to $\mathcal{A}$ is randomly distributed in the ciphertext space, $\mathcal{A}$ cannot gain any advantage in this simulation. Thus, any adversary that has advantage ϵ in the real IND-IBRSC-CCA2 game must necessarily recognize with probability at least ϵ that the challenge ciphertext provided by $\mathcal{C}$ is incorrect. For $\mathcal{A}$ to find that C^* is not a valid ciphertext, $\mathcal{A}$ should have queried the $\mathcal{O}_{H_2}$ oracle with $\alpha = \hat{e}(U^*, D_\gamma)$. Here D_γ is the private key of the target identity and it is $a(Q_\gamma) = abP$. Also, $\mathcal{C}$ has set $U^* = cP$. Hence $\alpha = \hat{e}(U^*, D_\gamma) = \hat{e}(cP, abP) = \hat{e}(P,P)^{abc}$. Therefore, one of the entries in list L_2 should be the value $\hat{e}(P,P)^{abc}$. With probability $\frac{1}{q_{H_2}}$, the value of α chosen by $\mathcal{C}$ from list L_2 will be the solution to CBDHP instance.

Now, we assess the probability of success of $\mathcal{C}$. The events in which $\mathcal{C}$ aborts the IND-IBRSC-CCA2 game are,

- E_1 - when $\mathcal{A}$ queries the private key of the target identity ID_γ and $Pr[E_1] = \dfrac{q_{e_1}}{q_{H_1}}$.
- E_2 - when $\mathcal{A}$ does not choose the target identity ID_γ as the receiver during the challenge and $Pr[E_2] = \left(1 - \dfrac{1}{q_{H_1} - q_{e_1}}\right)$.

The probability that, $\mathcal{C}$ does not abort the IND-IBRSC-CCA2 game is given by

$$(Pr[\neg E_1 \wedge \neg E_2]) = \left(1 - \frac{q_{e_1}}{q_{H_1}}\right)\left(\frac{1}{q_{H_1} - q_{e_1}}\right) = \frac{1}{q_{H_1}}$$

The probability that, the α chosen randomly from L_2 by $\mathcal{C}$, being the solution to CBDHP is $\left(\frac{1}{q_{H_2}}\right)$. Therefore, the probability of $\mathcal{C}$ solving CBDHP is given by,

$$Pr[\mathcal{C}(P, aP, bP, cP | a, b, c \in_R \mathbb{Z}_q^*) = \hat{e}(P,P)^{abc}] = \epsilon\left(\frac{1}{q_{H_1} q_{H_2}}\right)$$

Since ϵ is non-negligible, the probability of $\mathcal{C}$ solving CBDHP is also non-negligible. □

6.2. Unforgeability Proof of New-IBRSC (EUF-IBRSC-CMA):

Theorem 2. *If an EUF-IBRSC-CMA forger $\mathcal{A}$ exists against New-IBRSC scheme, asking q_{H_i} $(i = 1, 2, 3, 4)$ hash queries to random oracles H_i $(i = 1, 2, 3, 4)$, q_e extract secret key queries, q_{sc} signcryption queries and q_{us} unsigncryption queries, then there exist an algorithm $\mathcal{C}$ that solves the CDHP.*

Proof: The challenger $\mathcal{C}$ is challenged to solve an instance of the CDHP. $\mathcal{C}$ interacts with an adversary $\mathcal{A}$ who is capable of breaking the EUF-IBRSC-CMA security of New-IBRSC, to solve the CDHP instance. On receiving the instance $\langle P, aP, bP\rangle \in \mathbb{G}_1^3$ of the CDHP as input, $\mathcal{C}$ begins the interaction with $\mathcal{A}$ to compute the value $abP \in \mathbb{G}_1$. $\mathcal{C}$ simulates the system with the various oracles $\mathcal{O}_{H_1}$, $\mathcal{O}_{H_2}$, $\mathcal{O}_{H_3}$, $\mathcal{O}_{H_4}$, $\mathcal{O}_{Signcryption}$, $\mathcal{O}_{Unsigncryption}$ and allows $\mathcal{A}$ to adaptively ask polynomially bounded number of queries to these oracles.

Setup Phase: $\mathcal{C}$ simulates the system by setting up the system parameters in the following way.

- It takes $\mathbb{G}_1$ and $\mathbb{G}_2$, the two groups as well as the generator $P \in \mathbb{G}_1$ as given in the CDHP instance.
- Sets the master public key $P_{pub} = aP$.
- Models the four hash functions as random oracles $\mathcal{O}_{H_1}$, $\mathcal{O}_{H_2}$, $\mathcal{O}_{H_3}$ and $\mathcal{O}_{H_4}$.
- Selects a bilinear pairing $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$.

Training Phase: $\mathcal{A}$ adaptively performs polynomially bounded number of queries to the various oracles in this phase. The queries may be *Hash Queries*, *Extract Queries*, $\mathcal{O}_{Signcryption}$ *Queries* and $\mathcal{O}_{Unsigncryption}$ *Queries* with no restrictions, which are handled by $\mathcal{C}$ as in the confidentiality game for New-IBRSC.

Forgery: Finally, $\mathcal{A}$ produces a forged signcryption $C^* = (y^*, \mathcal{U}^*, U^*, U_1^*, \ldots, U_n^*)$ on the message m^* (i.e. C^* was not produced by the *Signcryption Oracle* as an output for the ring signcryption query on the message m^* with an ad-hoc set of users $\mathcal{U}^*$ and the receiver $ID_{\mathbb{R}}$), where the private keys of the users who are in the group $\mathcal{U}^*$ were not queried in the training phase. $\mathcal{C}$ aborts if $\mathcal{U}^*$ do not contain the target identity. Else, $\mathcal{C}$ can very well unsigncrypt and verify the validity of the forged ring signcryption C^* (as done in unsigncrypt oracle).

If the ring signature of the forged ring signcryption passes the verification then $\mathcal{C}$ will be able to generate one more valid ring signcryption from $C^* = (y^*, \mathcal{U}^*, U^*, U_1^*, \ldots, U_n^*)$ which is named as $C' = (y', \mathcal{U}^*, U^*, U_1^*, \ldots, U_n^*)$, using the oracle replay technique and applying extended version of forking lemma [3] applicable for ring signatures. This is achieved by running the turing machine again with the same random tape but with the different hash value. Obviously, $\mathcal{A}$, who is capable of generating a valid ring signcryption will be able to generate new valid ring signcryption again with the same randomness again. On getting two valid ring signcryptions on m^*, $\mathcal{C}$ will be able to retrieve $D_{\mathbb{S}} = abP$ as explained below:

- Computes $\alpha = \hat{e}(U, D_{\mathbb{R}})$
- Consecutively, V^* and V' are retrieved as $(m^* \| V^*) = y^* \oplus H_2(\alpha)$ and $(m^* \| V') = y' \oplus H_2(\alpha)$.
- Here, $V^* = (h_{\mathbb{S}}^* + r_{\mathbb{S}})D_{\mathbb{S}}$ and $V' = (h'_{\mathbb{S}} + r_{\mathbb{S}})D_{\mathbb{S}}$ (Since they have the same randomness).
- Thus, $V^* - V' = h_{\mathbb{S}}^* D_{\mathbb{S}} - h'_{\mathbb{S}} D_{\mathbb{S}} = (h_{\mathbb{S}}^* - h'_{\mathbb{S}})D_{\mathbb{S}}$.

Since $\mathcal{C}$ knows the hash values $h_{\mathbb{S}}^*$ and $h'_{\mathbb{S}}$, $\mathcal{C}$ can compute $D_{\mathbb{S}}$ as $D_{\mathbb{S}} = (h_{\mathbb{S}}^* - h'_{\mathbb{S}})^{-1}(V^* - V')$. This means, $\mathcal{C}$ can compute abP because $D_{\mathbb{S}} = abP$. In other words, $\mathcal{C}$ is capable of solving CDHP in polynomial time and this is a contradiction. Hence, New-IBRSC is secure against EUF-CMA. □

7. Conclusion

As a concluding remark we summarize the work in this paper. To the best of our knowledge there were seven ring signcryption schemes in the identity based setting. It was shown in [5] by Fagen Li et al. that, [12] was not CCA2 secure. So, six out of seven identity based ring signcryption schemes were believed to be secure till date. We have shown that the scheme in [6] does not provide anonymity and is not CCA2 secure; the schemes in [10] and [15] does not even provide security against chosen plaintext attack (CPA); and the schemes reported in [5] and [14] does not provide security against CCA2 attack, by demonstrating concrete attacks on these schemes. This leaves Huang et al.'s [4] scheme as the only secure identity based ring signcryption scheme. We have proposed a new identity based ring signcryption scheme for which we proved the security against chosen ciphertext attack and existential unforgeability in the random oracle model. Also we have compared our scheme with Huang et al.'s scheme below. In the comparison table, n represents the number of members in the ring.

Scheme	No of BP		CT Size
	SC	*USC*	
New-IBRSC	1	3	$n+2$
Huang et al. [4]	$n+2$	3	$2n+3$

Table 1: Comparison with [4]
BP - Bilinear Pairing, CT - Ciphertext, SC - Signcryption, USC - Unsigncryption.

Thus, our new identity based ring signcryption scheme (New-IBRSC) is a significant improvement over the scheme proposed by Huang et al. [4]

References

[1] Joonsang Baek, Ron Steinfeld, and Yuliang Zheng. Formal proofs for the security of signcryption. In *PKC 2002: Proceedings of the 5th International Workshop on Practice and Theory in Public Key Cryptography*, volume 2274 of *Lecture Notes in Computer Science*, pages 80–98. Springer-Verlag, 2002.

[2] Sherman S. M. Chow, Siu-Ming Yiu, and Lucas Chi Kwong Hui. Efficient identity based ring signature. In *Applied Cryptography and Network Security, Third International Conference, ACNS - 2005*, volume 3531 of *Lecture Notes in Computer Science*, pages 499–512, 2005.

[3] Javier Herranz and Germán Sáez. Forking lemmas for ring signature schemes. In *INDOCRYPT*, volume 2904 of *Lecture Notes in Computer Science*, pages 266–279. Springer, 2003.

[4] Xinyi Huang, Willy Susilo, Yi Mu, and Futai Zhang. Identity-based ring signcryption schemes: Cryptographic primitives for preserving privacy and authenticity in the ubiquitous world. In *AINA '05: Proceedings of the 19th International Conference on Advanced Information Networking and Applications*, pages 649–654. IEEE Computer Society, 2005.

[5] Fagen Li, Masaaki Shirase, and Tsuyoshi Takagi. Analysis and improvement of authenticatable ring signcryption scheme. In *International Conference ProvSec-08, Paper appears in Journal of Shanghai Jiaotong University (Science)*, volume 13-6, pages 679–683, December 2008.

[6] Fagen Li, Hu Xiong, and Yong Yu. An efficient id-based ring signcryption scheme. In *International Conference on Communications, Circuits and Systems, 2008. ICCCAS 2008.*, pages 483–487, May 2008.

[7] Ronald L. Rivest, Adi Shamir, and Yael Tauman. How to leak a secret. In *ASIACRYPT*, volume 2248 of *Lecture Notes in Computer Science*, pages 552–565. Springer, 2001.

[8] Sree Vivek S, Sharmila Deva Selvi S, and Pandu Rangan C. On the security of two ring signcryption schemes. Cryptology ePrint Archive, Report 2009/052; (To appear in the proceedings of SECRYPT - 09), 2009.

[9] Adi Shamir. Identity-based cryptosystems and signature schemes. In *Advances in Cryptology - CRYPTO '84: Proceedings of the 4th Annual International Cryptology Conference*, volume 196 of *Lecture Notes in Computer Science*, pages 47–53. Springer-Verlag, 1984.

[10] Yong Yu, Fagen Li, Chunxiang Xu, and Ying Sun. An efficient identity-based anonymous signcryption scheme. *Wuhan University Journal of Natural Sciences*, Volume: 13, Number: 6, December, 2008:670–674, 2008.

[11] Tzer Shyong Chen Yu Fang Chung, Zhen Yu Wu. Ring signature scheme for ecc-based anonymous signcryption. In *Computer Standards & Interfaces Journal*, 2008.

[12] Mingwu Zhang, Bo Yang, Shenglin Zhu, and Wenzheng Zhang. Efficient secret authenticatable anonymous signcryption scheme with identity privacy. In *PAISI, PACCF and SOCO '08: Proceedings of the IEEE ISI 2008 PAISI, PACCF, and SOCO international workshops on Intelligence and Security Informatics*, pages 126–137. Springer-Verlag, 2008.

[13] Yuliang Zheng. Digital signcryption or how to achieve cost(signature & encryption) << cost(signature) + cost(encryption). In *CRYPTO-97*, pages 165–179, 1997.

[14] ZhenChao Zhu, Yuqing Zhang, and Fengjiao Wang. An efficient and provable secure identity based ring signcryption scheme. In *Computer Standards & Interfaces*, Pages 649-654, http://dx.doi.org/10.1016/j.csi.2008.09.023, 2008.

[15] Lijun Zhun and Futai Zhang. Efficient identity based ring signature and ring signcryption schemes. In *International Conference on Computational Intelligence and Security, 2008. CIS '08.*, volume 2, pages 303–307, December 2008.

S. Sharmila Deva Selvi, S. Sree Vivek and C. Pandu Rangan
Indian Institute of Technology Madras,
Theoretical Computer Science Laboratory,
Department of Computer Science and Engineering,
Chennai, India - 600036
E-mail: sharmila(at)cse.iitm.ac.in
svivek(at)cse.iitm.ac.in
prangan(at)cse.iitm.ac.in