Identity-based Ring Signcryption Schemes: Cryptographic Primitives for Preserving Privacy and Authenticity in The Ubiquitous World

Xinyi Huang[1], Willy Susilo[2], Yi Mu[2] and Futai Zhang[1] *
[1]College of Mathematics and Computer Science
Nanjing Normal University, P.R. China
Email: xinyinjnu@126.com, zhangfutai@njnu.edu.cn
[2]School of Information Technology and Computer Science
University of Wollongong, Australia
Email: {wsusilo, ymu}@uow.edu.au

Abstract

*In this paper, we present a new concept called an identity based ring signcryption scheme (*IDRSC*). We argue that this is an important cryptographic primitive that must be used to protect* privacy *and* authenticity *of a collection of users who are connected through an ad-hoc network, such as Bluetooth. We also present an efficient* IDRSC *scheme based on bilinear pairing. As a regular signcryption scheme, our scheme combines the functionality of signature and encryption schemes. However, the idea is to have an identity based system. In our scheme, a user can anonymously signcrypts a message on behalf of the group. We show that our scheme outperforms a traditional identity based scheme, that is obtained by a standard sign-then-encrypt mechanism, in terms of the length of the ciphertext. We also provide a formal proof of our scheme with the chosen ciphertext security under the Decisional Bilinear Diffie-Hellman assumption, which is believed to be intractable.*

Keywords: Identity-based, Ring Signcryption, Bilinear Pairing, Cryptography, Privacy and Trust

1. Introduction

The emergence of more powerful computers and networks led to the distributed computing phenomenon, in which an organization's computing is distributed over networks, instead of being performed only at a central computer installation. Ubiquitous computing plays a great role in a very difficult integration of human factors, computer science, engineering, and social sciences. With the great influence of powerful computers and networks, placing computers in human life would face an essential problem, namely how to implement *trust* among the users that are connected in a network. A specific example can be derived from a collection of ad-hoc users, that are connected via a Bluetooth device. The connection allows anyone who knows the Bluetooth device address (BD_ADDR) to connect and talk to the each other. However, how can they be sure that the people that they are talking to are really the real people who claim who they are? In some cases, we would also like to protect user's privacy. For instance, a parliament member would like to reveal an important news about the cabinet, but he would like to remain anonymous. Nevertheless, the news must be authenticated, or otherwise it could be misused. An additional requirement that we would like to achieve is the way the news is spread, namely via an ad-hoc connection like Bluetooth. This way, the only available information that can be used is the Bluetooth device address. In this paper, we present a novel solution by introducing a new cryptographic primitive called Identity-based Ring Signcryption Schemes.

The idea of identity-based cryptosystem was introduced by Shamir in his seminal paper in [1]. The main essence of identity-based cryptosystem is to remove the need of certification of the public keys, that are required in the conventional public key cryptography setting. The public key of each participant is obtained from his/her public identity, such as email address, IP address combined with a user name, social security number, etc. that can uniquely identify the participant. This model requires the existence of a trusted authority called Private Key Generator (PKG), whose task is to generate user's private key from their identity information, after a successful identification. Since its introduction in [1], many identity based schemes have been proposed (eg. [2, 3, 4, 5, 6]) and the most notably one is the identity-based encryption scheme proposed by Boneh

* Partially supported by National "211" Project, Project ID:181070H901, Education of JiangSu Project,Project ID:03KJA520066.

and Franklin in [7] that takes advantage of the properties of suitable bilinear maps (the Weil or Tate pairing) over supersingular elliptic curves. Other identity based schemes using pairing were subsequently proposed following this paper [8, 9, 10, 11].

In Asiacrypt 2001, Rivest, Shamir and Tauman firstly addressed and formalized the notion of ring signatures [12]. A ring signature can be considered as a simplified group signature with no manager, no group setup procedure, and no revocation mechanism, and hence, it provides signer's ambiguity. In a ring signature scheme, the information of all possible signers, i.e. ring members, serves as a part of the ring signature for the signed message. A valid ring signature will convince a verifier that the signature is generated by one of member in the ring, *without* revealing any information about which participant is the actual signer. Hereby, the anonymity property is referred to as *signer-ambiguity*. The first ID-based ring signature scheme was proposed by Zhang and Kim in [13].

The concept of public key signcryption schemes was proposed by Zheng in [14]. The idea of this kind of primitives is to perform encryption and signature in a single logical step to obtain confidentiality, integrity, authentication and non-repudiation more efficiently than the sign-then-encrypt approach. Several efficient signcryption schemes have been proposed since then, including [15, 16, 17, 18, 19]. Long after its first introduction, a formal security proof of signcryption scheme was proposed in [20].

Our Contribution

In this paper, we propose a new concept of identity-based ring signcryption based on the identity-based ring signature. Throughout this paper, let IDRSC denote our identity-based ring signcryption. With the IDRSC, a user can anonymously signcrypts a message on behalf of a set of users including himself. Our scheme is motivated by Herranz and Sáez's ID-based ring signature scheme proposed in [21]. The underlying security model is based on the difficulty of the *Decisional Bilinear Diffie-Hellman problem*. The length of our ciphertext is shorter compared to the traditional sign-then-encrypt method. Hence, our scheme is more practical.

2. Preliminaries

In this section, we briefly review some preliminaries that will be used throughout this paper.

2.1. Basic Concepts on Bilinear Pairings

Let $\mathbb{G}_1, \mathbb{G}'_1$ be cyclic additive groups generated by P_1, P'_1, respectively, whose order are a prime q. Let $\mathbb{G}_2$ be a cyclic multiplicative group with the same order q. We assume there is an isomorphism $\psi : \mathbb{G}'_1 \rightarrow \mathbb{G}_1$ such that $\psi(P'_1) = P_1$. Let $\hat{e} : \mathbb{G}_1 \times \mathbb{G}'_1 \rightarrow \mathbb{G}_2$ be a bilinear mapping with the following properties:

1. *Bilinearity*: $\hat{e}(aP, bQ) = \hat{e}(P,Q)^{ab}$ for all $P \in \mathbb{G}_1, Q \in \mathbb{G}'_1, a, b, \in \mathbb{Z}_q$.
2. *Non-degeneracy*: There exists $P \in \mathbb{G}_1, Q \in \mathbb{G}'_1$ such that $\hat{e}(P,Q) \neq 1$.
3. *Computability*: There exists an efficient algorithm to compute $\hat{e}(P,Q)$ for all $P \in \mathbb{G}_1, Q \in \mathbb{G}'_1$.

For simplicity, hereafter, we set $\mathbb{G}_1 = \mathbb{G}'_1$ and $P_1 = P'_1$. We note that our scheme can be easily modified for a general case, when $\mathbb{G}_1 \neq \mathbb{G}'_1$.

Complexity Assumptions

Bilinear Diffie-Hellman problem and Decisional Bilinear Diffie-Hellman problem

Definition 1 *Given two groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of the same prime order q, a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ and a generator P of $\mathbb{G}_1$, the* **Bilinear Diffie-Hellman problem (BDHP)** *in $(\mathbb{G}_1, \mathbb{G}_2, \hat{e})$ is to compute $\hat{e}(P,P)^{abc}$ given (P, aP, bP, cP).*

- *The Decisional Bilinear Diffie-Hellman problem (DBDHP) is, given a tuple of points (P, aP, bP, cP) and an element $h \in \mathbb{G}_2$, to decide whether $h \overset{?}{=} \hat{e}(P,P)^{abc}$ holds.*
- *We define the advantage of a distinguisher against the DBDH problem as follows.*

$$\begin{aligned} Adv(B) &= |P_{a,b,c\in_R F_q, h\in_R G_2}[1 \leftarrow B(aP,bP,cP,h)] \\ &\quad -P_{a,b,c\in_R F_q}[1 \leftarrow B(aP,bP,cP, \\ &\quad \hat{e}(P,P)^{abc})] \end{aligned}$$

- *The decisional bilinear Diffie-Hellman problem is no harder than the computational bilinear Diffie-Hellman problem. However, there is no known polynomial algorithm that can solve the decisional bilinear Diffie-Hellman problem efficiently.*

3. Formal Model of Identity based Ring Signcryption Schemes

Definition 2 *An identity based ring signcryption scheme consists of the following algorithms.*

Setup*: given a security parameter k, the private key generator (PKG) generates the system's public parameters.*

Keygen*: given an identity ID, the PKG computes the corresponding private key D_{ID} and delivers it to the user via an authenticated channel.*

Signcryption*: To send a message m to the receiver Bob whose identity is ID_B, Alice chooses some other users to form a group $\mathcal{U}$ including herself and computes Signcrypt $(\mathcal{U}, ID_B, m)$ on the behalf of the group $\mathcal{U}$ to obtain the ciphertext C.*

Unsigncryption*: when Bob receives the ciphertext C, to get the plaintext he computes $Unsigncrypt(\mathcal{U}, D_{ID_B}, C)$ and obtains the plain text m or the symbol $\perp$ if C was an invalid cipher text between the group $\mathcal{U}$ and Bob.*

Consistency
An identity based ring signcryption scheme is said to be *consistent* iff

$$Pr \quad [C \leftarrow Signcrypt(\mathcal{U}, ID_B, m), \\ m \leftarrow Unsigncrypt(\mathcal{U}, D_{ID_B}, C)\ ] = 1$$

3.1. Security Notions

Baek et. al. gave a formal security definition of signcryption in [20]. Our definition of the security of identity based ring signcryption scheme is a natural adaptation of theirs. We define our security requirement for identity based ring signcryption as follows.

Definition 3 *We say that an identity based ring signcryption (*IDRSC*) is indistinguishable against adaptive chosen ciphertext ring attacks(*IND-IDRSC-CCA*) if there exists no polynomially bounded adversary has a non-negligible advantage in the following game:*

- *The challenger runs the Setup algorithm with a security parameter k and sends the system parameters to the adversary $\mathcal{A}$.*
- *The adversary $\mathcal{A}$ performs a polynomially bounded number of requests:*
  - Signcryption request*: $\mathcal{A}$ produces a set of users $\mathcal{U}$, an identity ID_j and a plaintext m. The challenger randomly chooses a user $u_i \in \mathcal{U}$ whose identity is ID_i and computes $D_{ID_i} = Keygen(ID_i)$. Then the challenger acts as u_i to $Signcrypt(\mathcal{U}, ID_j, m)$ on the behalf of $\mathcal{U}$ and sends the result to $\mathcal{A}$.*
  - Unsigncryption request*: $\mathcal{A}$ produces a set of users $\mathcal{U}$, an identity ID, and a ciphertext C. The challenger generates the private key $D_{ID} = Keygen(ID)$ and sends the result of $Unsigncrypt(\mathcal{U}, D_{ID}, C)$ to $\mathcal{A}$ (this result can be the $\perp$ symbol if C is an invalid ciphertext).*
  - Key extraction request*: $\mathcal{A}$ produces an identity ID and receives the extracted private key $D_{ID} = Keygen(ID)$.*

    $\mathcal{A}$ can present its requests adaptively: every request may depend on the answer to the previous ones.
- *$\mathcal{A}$ chooses two plaintexts $m_0, m_1 \in \mathcal{M}$, a user set $\mathcal{U}_A$ and an identity ID_B on which he wants to be challenged. He cannot have asked the private key corresponding to any user in the group $\mathcal{U}_A$ nor ID_B in the first stage.*
- *The challenger takes a bit $b \in_R \{0,1\}$ and computes the ciphertext C of m_b which is sent to $\mathcal{A}$.*
- *$\mathcal{A}$ asks again a polynomially bounded number of requests just like in the first stage. This time, he cannot make a key extraction request on any user in the group $\mathcal{U}_A$ nor ID_B and he cannot ask the plaintext corresponding to C.*
- *Finally, $\mathcal{A}$ produces a bit b' and wins the game if $b' = b$.*

The adversary's success probability is defined as

$$\mathsf{Succ}_{\mathcal{A}}^{IND-RSC-CCA}(k) = \frac{1}{2} + \epsilon$$

We require that ϵ to be negligible in k.

Definition 4 *An identity based ring signcryption scheme(*IDRSC*)is said to be secure against an existential forgery for adaptive chosen messages attacks(EF-IDRSC-ACMA) if no polynomially bounded adversary has a non-negligeable advantage in the following game:*

- *The challenger runs the Setup algorithm with a security parameter k and gives the system parameters to the adversary $\mathcal{A}$.*
- *The adversary $\mathcal{A}$ performs a polynomial bounded number of requests as in the previous definition.*
- *Finally, $\mathcal{A}$ produces a new triple $(\mathcal{U}, ID, C)$(i.e. a triple that was not produced by the signcryption oracle), where the private keys of the users in the group $\mathcal{U}$ and the receiver (whose identity is ID) were not asked in the second stage and wins the game if the result of the Unsigncryption$(\mathcal{U}, ID, C)$ is not the $\perp$ symbol.*

In this definition, the adversary is allowed to ask the private key corresponding to the receiver's identity ID for which the ciphertext he produces must be valid. This condition is necessary to obtain the non-repudiation property and to prevent a dishonest recipient to send a cipertext to himself on the group $\mathcal{U}$'s behalf and try to convince a third party that the group $\mathcal{U}$ was the sender.

4. Our ID-Based Ring Signcryption Scheme

In this section, we present our identity based ring signcryption scheme from bilinear pairing. We describe our scheme by providing the description of the following algorithms: Key Generation, Signcryption, Unsigncryption.

4.1. Key Generation

Given security parameters k and l, a trusted private key generator (PKG) chooses two groups $\mathbb{G}_1, \mathbb{G}_2$ of prime order $q > 2^k$, a bilinear map $\hat{e}$ from $\mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$, and a generator P of $\mathbb{G}_1$. Next, PKG picks a random number $s \in Z_q^*$ as its master key and computes its public key $P_{pub} = sP$. Then it chooses

1550-445X/05 $20.00 © 2005 **IEEE**

some cryptographic hash functions described as follows: $H_1 : \{0,1\}^* \rightarrow \mathbb{G}_1^*$; $H_2 : \mathbb{G}_2 \rightarrow \{0,1\}^l$; $H_3 : \{0,1\}^l \rightarrow \{0,1\}^l$; $H_4 : \{0,1\}^* \rightarrow Z_q^*$. The security analysis will view H_1, H_2, H_3, H_4 as random oracles. The message space is $\mathcal{M} = \{0,1\}^l$. Finally, PKG publishes$\{\mathbb{G}_1, \mathbb{G}_2, \hat{e}, P, P_{pub}, H_1, H_2, H_3, H_4, q\}$, but s is kept secret.

Extract: For a user $\mathcal{U}_i$ whose identification information is ID_i, PKG computes $Q_i = H_1(ID_i)$ and calculates the user's secret key as $D_i = sQ_i$ where s is the PKG's master key and sends D_i to $\mathcal{U}_i$ via a secure and authenticated channel.

4.2. Signcryption

Consider a set of users $\mathcal{U} = \{u_1, u_2, \cdots, u_n\}$. Let Q_i denote $\mathcal{U}_i$'s public information (where $Q_i = H_1(ID_i)$) and D_i denotes his secret key. In the following scenario, without losing generality, we assume a user $u_s \in \mathcal{U}$ would like to signcrypt a message m on behalf of the group $\mathcal{U}$ and send it to a receiver, Bob, whose identity is ID_B. Then, u_s will perform the following.

- Choose $a_0 \in_R Z_q^*$, $m_r \in_R \mathcal{M}$ and compute $R_0 = a_0P, R_0^{'} = \hat{e}(a_0P_{pub}, Q_B), k = H_2(R_0^{'}), c_1 = m_r \oplus k, c_2 = m \oplus H_3(m_r)$.
- For $i \neq s$, choose $a_i \in_R Z_q^*$. Compute $A_i = a_iP, R_i = \hat{e}(A_i, P), h_i = H_4(\mathcal{U}, m, k, R_i)$.
- For $i = s$, choose $a_s \in_R Z_q^*$. Compute $A_s = a_sP, R_s = \hat{e}(A_s, P) \cdot \hat{e}(-P_{pub}, \sum_{i \neq s} h_i \cdot Q_i)$. If $R_s = 1_{G_2}$ or $R_s = R_i$ for some $i \neq s$, then repeat step 3 until he obtains an admissible R_s.
- Compute $h_s = H_4(\mathcal{U}, m, k, R_s)$, $\sigma = h_s \cdot D_s + \sum_{i=1}^{n} A_i$.
- Define the ciphertext of message m as:

  $$C = (\mathcal{U}, c_1, c_2, \sigma, R_0, R_1, \cdots, R_n, h_1, h_2, \cdots, h_n)$$

  and sends C to Bob.

4.3. Unsigncryption

Upon receiving the ciphertext $C = (\mathcal{U}, c_1, c_2, \sigma, R_0, R_1, \cdots, R_n, h_1, h_2, \cdots, h_n)$, Bob unsigncrypts the ciphertext, C, using his secret key D_B:

- $k' = H_2(\hat{e}(R_0, D_B))$, recovers $m_r' = c_1 \oplus k'$, $m' = c_2 \oplus H_3(m_r')$.
- For $i \in \{1, 2, \cdots, n\}$, checks whether $h_i = H_4(\mathcal{U}, m', k', R_i)$
- Checks whether $\hat{e}(\sigma, P) = R_1 \cdot R_2 \cdot ... \cdot R_n \cdot \hat{e}(P_{pub}, \sum_{i=1}^{n} h_i \cdot Q_i)$.

If for all $i \in \{1, 2, \cdots, n\}$, $h_i = H_4(\mathcal{U}, m', k', R_i)$ and $\hat{e}(\sigma, P) = R_1 \cdot R_2 \cdot ... \cdot, R_n \cdot \hat{e}(P_{pub}, \sum_{i=1}^{n} h_i \cdot Q_i)$, Bob accepts m as an valid message. Otherwise, Bob rejects m.

Correctness and Consistency.
If the ciphertext C is not altered, the following equations will hold.

$$\begin{aligned} \hat{e}(R_0, D_B) &= \hat{e}(a_0P, D_B) = \hat{e}(sa_0P, Q_B) \\ &= \hat{e}(a_0P_{pub}, Q_B) = R_0' \end{aligned}$$

Hence,

$$\begin{aligned} k' &= H_2(\hat{e}(R_0, D_B)) = k \\ m_r' &= c_1 \oplus k' = m_r \\ m' &= c_2 \oplus H_3(m_r') = m \end{aligned}$$

and finally

$$\forall i \in \{1, 2, \cdots, n\}, h_i \overset{?}{=} H_4(\mathcal{U}, m', k', R_i)$$

will hold with equality.
Moreover,

$$\begin{aligned} R_1 \cdot R_2 \cdot ... \cdot R_n \cdot \hat{e}(P_{pub}, \sum_{i=1}^{n} h_i \cdot Q_i) &= \\ \hat{e}(A_1, P) \cdot \hat{e}(A_2, P) \cdot ... \cdot \hat{e}(A_n, P) \cdot & \\ \hat{e}(-P_{pub}, \sum_{i \neq s} h_i \cdot Q_i) \cdot \hat{e}(P_{pub}, \sum_{i=1}^{n} h_i \cdot Q_i) &= \\ \hat{e}(\sum_{i=1}^{n} A_i, P) \cdot \hat{e}(P_{pub}, h_s \cdot Q_s) &= \\ \hat{e}(\sum_{i=1}^{n} A_i + h_s \cdot D_s, P) &= \hat{e}(\sigma, P). \end{aligned}$$

■

Anonymity.
The unconditional anonymity of the scheme is also clear. The scheme is completely symmetric, and hence, any third party outside the group $\mathcal{U}$ has probability $1/n$ (where n denotes the size of the group) to guess which member of the ring has actually signcrypted the message. ■

5. Security Analysis

Indistinguishability
In this section, we will provide a formal proof that the IDRSC is IND-IDRSC-CCA assuming the *Decisional Bilinear Diffle-Hellman problem* is hard.

Theorem 1 *In the random oracle model, we assume an adaptive chosen ciphertext attacks adversary $\mathcal{A}$ that can distinguish ciphertexts from the users set $\mathcal{U}$ during the game of definition 1 with an advantage ε when running in a time t and asking at most q_{H_1} identity hashing requests, at most $q_{H_2} H_2$ requests, $q_{H_3} H_3$ requests, $q_{H_4} H_4$ requests, at most q_E Key extraction requests, q_k Signcryption requests and q_U Unsigncryption requests. Then there exists a distinguisher $\mathcal{B}$ that can solve the* **Decisional Bilinear Diffle-Hellman problem** *with an advantage:*

$$Adv(\mathcal{B}) \geq \frac{1}{e^{n+q_E}} \cdot \frac{|\varepsilon - q_U/2^{k-1}|}{2\, q_{H_1}}$$

Here $e \approx 2.71$ is the base of the natural logarithm, n is the number of the users set $\mathcal{U}$, k is system's security parameter. The running time of $\mathcal{B}$ is $O(t)$.

Proof: The distinguisher $\mathcal{B}$ receives a random instance (P, aP, bP, cP, h) of the *Decisional Bilinear Diffle-Hellman problem*. His goal is to decide whether $h = \hat{e}(P,P)^{abc}$ or not. $\mathcal{B}$ will run $\mathcal{A}$ as subroutine and act as $\mathcal{A}$'s challenger in the *IND-IDRSC-CCA* game. $\mathcal{B}$ needs to maintain lists L_1, L_2, L_3, L_4 that are initially empty and are used to keep track of answers to queries asked by $\mathcal{A}$ to oracles H_1, H_2, H_3, H_4 respectively. We assume that any signcryption or unsigncyption request between a group $\mathcal{U}$ and an identity ID happens after $\mathcal{A}$ asked the hashing H_1 of this ID and the identities in the group $\mathcal{U}$. Any key extraction query on the identity is also preceded by a hash query on the same identity. We also assume that $\mathcal{A}$ never makes an unsigncryption query on a ciphertext obtained from the signcryption oracle. He only makes unsigncryption queries for observed ciphertext.

At the beginning of the game , $\mathcal{B}$ runs the **Setup** program with the parameter k, and gives $\mathcal{A}$ the system parameters $\{\mathbb{G}_1, \mathbb{G}_2, \hat{e}, P, P_{pub}, H_1, H_2, H_3, H_4, q\}$ with $P_{pub} = cP$(c is unknown to $\mathcal{B}$ and plays the role of the PKG's master-key). H_1, H_2, H_3, H_4 are random oracles described as follows:

H_1 **requests:** At any time, $\mathcal{A}$ can ask a polynomially bounded number of H_1 requests on identities of his choice. To respond these queries, $\mathcal{B}$ maintains the list L_1 of tuples (ID, Q_{ID}, b, c). The list is initially empty. When $\mathcal{A}$ queries the oracle H_1, $\mathcal{B}$ responds as follows:

- At the j^{th} H_1 request, $\mathcal{B}$ answers by $H_1(ID_j) = bP$, and let $c_j = 0$(We assume that before the j^{th} H_1 requests, there is no tuple (ID_j, Q_j, b_j, c_j) in the list L_1).
- For $i \neq j$, $\mathcal{B}$ responds as follows:
  - If the ID_i already appears on the L_1 in the tuple(ID_i, Q_i, b_i, c_i), then $\mathcal{B}$ responds with $H_1(ID_i) = Q_i$.
  - Otherwise, $\mathcal{B}$ generates a random $coin \in \{0,1\}$ so that $\Pr[coin = 1] = \delta$, for some δ that will be determined later. Let $c_i = coin$.
- $\mathcal{B}$ picks a random $b_i \in Z_q^*$, computes $Q_i = b_i P$.
- $\mathcal{B}$ adds the tuple (ID_i, Q_i, b_i, c_i) to the list L_1, and responds to $\mathcal{A}$ with $H_1(ID_i) = Q_i$.

H_2 **requests:** At any time, $\mathcal{A}$ can ask a polynomially bounded number of H_2 requests of his choice. To respond these queries, $\mathcal{B}$ maintains the list L_2 of tuples (R_e, k_e). The list is initially empty. When $\mathcal{A}$ queries the oracle H_2 of the request $H_2(R_i)$, $\mathcal{B}$ first searches a pair (R_i, k_i) in list L_2. If such a pair is found, $\mathcal{B}$ answers by k_i. Otherwise he answers $\mathcal{A}$ by a random binary sequence $k_i \in_R \{0,1\}^l$ such that no entry $(\cdot, k_i)$ appears in L_2(in order to avoid collisions on H_2) and adds the pair (R_i, k_i) to L_2.

H_3 **requests:** At any time, $\mathcal{A}$ can ask a polynomially bounded number of H_3 requests of his choice. To respond these queries, $\mathcal{B}$ maintains the list L_3 of tuples (m_e, n_e). The list is initially empty. When $\mathcal{A}$ queries the oracle H_3 of the request $H_3(m_i)$, $\mathcal{B}$ first searches a pair (m_i, n_i) in list L_3. If such a pair is found, $\mathcal{B}$ answers by n_i. Otherwise, he answers $\mathcal{A}$ by a random binary sequence $n_i \in_R \{0,1\}^l$ such that no entry $(\cdot, n_i)$ appears in L_3(in order to avoid collisions on H_3) and adds the pair (m_i, n_i) in L_3.

H_4 **requests:** At any time, $\mathcal{A}$ can ask a polynomially bounded number of H_4 requests of his choice. To respond these queries, $\mathcal{B}$ maintains the list L_4 of tuples (x_e, y_e). The list is initially empty. When $\mathcal{A}$ queries the oracle H_4 of the request $H_4(U_i, m_i, k_i, R_i)$, $\mathcal{B}$ computes $x_i = U_i \parallel m_i \parallel k_i \parallel R_i$ and searches a pair (x_i, y_i) in list L_4. If such a pair is found, $\mathcal{B}$ answers by y_i. Otherwise he answers $\mathcal{A}$ by a random binary sequence $y_i \in_R F_q^*$ such that no entry $(\cdot, y_i)$ appears in L_4(in order to avoid collisions on H_4) and adds the pair (x_i, y_i) in L_4.

Key Extraction requests: At any time, $\mathcal{A}$ can ask a polynomially bounded number of key extraction requests of his choice. When $\mathcal{A}$ asks a query keygen(ID_i), $\mathcal{B}$ first finds the corresponding tuple (ID_i, Q_i, b_i, c_i) in L_1(From the assumption we know that there must be such a tuple in L_1). If $c_i = 0$, $\mathcal{B}$ fails and stops. Otherwise if $c_i = 1$, $\mathcal{B}$ computes the secret key $D_i = b_i \cdot P_{pub} = c \cdot Q_i$, then $\mathcal{B}$ returns D_i to $\mathcal{A}$.

Signcryption requests: At any time, $\mathcal{A}$ can perform a signcryption request for a plaintext m , a user group $\mathcal{U}$ and a designated receiver with identity ID. $\mathcal{B}$ randomaly chooses a user u_A in the group $\mathcal{U}$ whose identity is ID_A and not ID_j (in this case, $\mathcal{B}$ can computes $u_A's$ secret key $D_A = b_A \cdot P_{pub}$ where b_A is in the corresponding tuple(ID_A, Q_A, b_A, c_A) in the list L_1). Then B uses $u_A's$ secret key and runs *Signcryption*$(\mathcal{U}, ID, m)$ to signcrypt the message on the behalf of the group $\mathcal{U}$. At last, $\mathcal{B}$ returns the result ciphertext C to $\mathcal{A}$.

Unsigncryption requests: At any time, $\mathcal{A}$ can perform an unsigncryption request for a ciphertext $C = (\mathcal{U}, c_1, c_2, \sigma, R_0, R_1, \cdots, R_n, h_1, h_2, \cdots, h_n)$ between the group $\mathcal{U}$ and receiver whose identity is ID. If $ID = ID_j$, $\mathcal{B}$ always notifies $\mathcal{A}$ that the ciphertext is invalid(because $\mathcal{B}$ does not know the secret key of the user whose identity is ID_j). If this ciphertext C is a valid one, the probability that A will find is no more than $1/2^k$. In other case where the receiver's identity is not ID_j, $\mathcal{B}$ computes $k' = H_2(\hat{e}(R_0, D_{ID}))$, $m'_r = c_1 \oplus k'$,$m' = c_2 \oplus H_3(m'_r)$. Then $\mathcal{B}$ checks whether $h_i = H_4(\mathcal{U}, m, k, R_i)$ and $\hat{e}(\sigma, P) = R_1 \cdot R_2 \cdot \ldots \cdot R_n \cdot \hat{e}(P_{pub}, \sum_{i=1}^{n} h_i \cdot Q_i)$. If for all $i \in \{1, 2, \cdots, n\}$, $h_i = H_4(\mathcal{U}, m, k, R_i)$ and $\hat{e}(\sigma, P) = R_1 \cdot R_2 \cdot \ldots \cdot, R_n \cdot \hat{e}(P_{pub}, \sum_{i=1}^{n} h_i \cdot Q_i)$, $\mathcal{B}$ notifies $\mathcal{A}$ that the ciphertext C is valid one, otherwise $\mathcal{B}$ notifies $\mathcal{A}$ that C is not a valid ciphertext between $\mathcal{U}$ and some user whose identity is ID.

Challenge: After a polynomially bounded number of queries, $\mathcal{A}$ chooses two messages $m_0, m_1 \in \mathcal{M}$, n users whose identities are $\{ID_1, ID_2, \cdots, ID_n\}$ to form a users set $\mathcal{U}$ and another user whose identity is ID. If $ID \neq ID_j$, $\mathcal{B}$ fails and stops. For $\forall\, i \in \{1, 2, 3, \cdots, n\}$, if $c_i = 1$ in the corresponding tuple (ID_i, Q_i, b_i, c_i) in L_1, $\mathcal{B}$ also fails and stops. If such $\mathcal{U}$ and the receiver are admissible, $\mathcal{B}$ chooses $b \in_R \{0, 1\}$ and let $R_0 = aP, R_0^{'} = h$, then $\mathcal{B}$ signcrypts the message m_b as described in the signcryption request and sends the ciphertext C to $\mathcal{A}$.

$\mathcal{A}$ asks again a polynomially bounded number of requests just like in the first stage. This time, he cannot know the secret key of any user in the group U nor ID_j and he cannot ask the plaintext corresponding to the ciphertext C. At the end of the simulation, he produces a bit b' for which he believes the relation $C = Signcrypt(\mathcal{U}, ID_j, m_b)$ holds and sends b' to $\mathcal{B}$. At this moment, if $b = b'$, $\mathcal{B}$ then answers 1 as a result because his selection h allowed him to produce a ciphertext C that appeared to $\mathcal{A}$ as a valid signcrypted text of m_b. If $b \neq b'$, $\mathcal{B}$ then answers 0.

We now consider $\mathcal{B}$'s probability of success. We find that the probability that $\mathcal{B}$ does not fail during the key extraction requests is obvious δ^{q_E} where q_E is the number of key extraction requests. Then we also see that during the challenge process the probability that $\mathcal{B}$ does not fail is $(1-\delta)^n/q_{H_1}$. Therefore, the probability that $\mathcal{B}$ does not abort during the simulation is $\delta^{q_E}(1-\delta)^n/q_{H_1}$. This value is maximized at $\delta_{opt} = 1 - n/(q_E + n)$. Using δ_{opt}, the probability that $\mathcal{B}$ does not abort is at least $(1/q_{H_1})(1/e)^{n+q_E}$. The probability that $\mathcal{B}$ gave an false answer during the Unsigncryption process is no more than $q_U/2^k$. Finally, let

$$p_1 = P[b = b' | C = Signcrypt(\mathcal{U}, ID_j, m_b)] = \frac{\varepsilon + 1}{2} - \frac{q_U}{2^k},$$

$$p_0 = P[b' = i | h \in_R \mathbb{G}_2] = 1/2 \quad for\, i = 0, 1$$

and hence, we obtain

$$\begin{aligned} Adv(\mathcal{B}) &= |P_{a,b,c \in_R F_q, h \in_R \mathbb{G}_2}[1 \leftarrow B(aP, bP, cP, h)] \\ &\quad - P_{a,b,c \in_R F_q}[1 \leftarrow B(aP, bP, cP, \hat{e}(P,P)^{abc})] \\ &\geq \frac{|p_1 - p_0|}{q_{H_1} e^{n+q_E}} \\ &= \frac{|\varepsilon - q_U/2^{k-1}|}{2 q_{H_1} e^{n+q_E}} = \frac{1}{e^{n+q_E}} \cdot \frac{|\varepsilon - q_U/2^{k-1}|}{2\, q_{H_1}} \end{aligned}$$

■

Unforgeability

The unforgeability against adaptive chosen-messages attacks can be derived directly from the security of Herranz and Sáez's ID-based ring signature scheme [21] under the computational Diffle-Hellman assumption. One can find that an attacker who can forge a valid signcrypted message of IDRSC must be able to forge a valid signature for the scheme of Herranz and Sáez's ID-based ring signature. ■

References

[1] A. Shamir, Identity-based cryptosystems and signature schemes, Adv in Cryptology-Crypto '84, LNCS 196, pp.47-53, 1984.

[2] S. Tsuji and T. Itoh, An ID-based cryptosystem based on the discrete logarithm problem, IEEE Journal on Sel Areas in Comm, vol.7, no. 4, pp. 467-473, 1989.

[3] H. Tanaka, A realization scheme for the identity-based cryptosystem, in Adv in Cryptology - Crypto '87, LNCS 293, pp. 341-349, 1987.

[4] Y. Desmedt and J. Quisquater, Public-key systems based on the diffculty of tampering, in Adv in Cryptology-Crypto '86, LNCS 263, pp. 111-117, 1986.

[5] D. Hühnlein, M. Jacobson, D. Weber, Towards Practical Non-interactive Public Key Cryptosystems Using Non-maximal Imaginary Quadratic Orders, in Selected Areas in Cryptography, LNCS 2012, pp. 275-287, 2000.

[6] U. Maurer and Y. Yacobi, Non-interactive public-key cryptography, in Adv in Cryptology-Crypto'91, LNCS 547, pp. 498-507, 1991.

[7] D. Boneh and M. Franklin, Identity-based encryption from the Weil Pairing, Crypto'01, pp.213-229, 2001.

[8] R. Dupont, A. Enge, Practical Non-Interactive Key Distribution Based on Pairings, available at http://eprint.iacr.org/2002/136.

[9] F. Hess, Efficient identity based signature schemes based on pairings, SAC 2002, LNCS, 20002.

[10] N.P. Smart, An identity based authenticated key agreement protocol based on the Weil pairing, Electronic Letters, 38(13): 630-632, 2002.

[11] F. Zhang, S. Liu, K. Kim, ID-Based One Round Authenticated Tripartite Key Agreement Protocol with Pairings, available at http://eprint.iacr.org/2002/122.

[12] R.L. Rivest, A. Shamir and Y. Tauman, How to leak a secret, Adv in Cryptology-Asiacrypt 2001, LNCS 2248, pp.552-565, 2001.

[13] F. Zhang and K. Kim, ID-based blind signature and ring signature from pairings, Adv in Cryptology - Asiacrypt 2002, LNCS 2501, pp. 533-547, 2002.

[14] Y. Zheng, Digital Signcryption or How to Achieve Cost (Signature & Encryption) $\ll$ Cost (Signature) + Cost (Encryption), Adv in Cryptology - Crypto'97, LNCS 1294, pp. 165-179, 1997.

[15] R. Steinfeld, Y. Zheng, A Signcryption Scheme Based on Integer Factorization, ISW00, LNCS, pp. 308-322, 2000.

[16] B.H. Yum, P.J. Lee, New Signcryption Schemes Based on KCDSA, ICISC01, LNCS 2288, pp. 305-317, 2001.

[17] Y. Zheng, H. Imai, Efficient Signcryption Schemes On Elliptic Curves, Proc. of IFIP/SEC98, Chapman & Hall, 1998.

[18] Y. Zheng, Identification, Signature and Signcryption using High Order Residues Modulo an RSA Composite, PKC01, LNCS 1992, Springer-Verlag, pp. 48-63, 2001.

[19] Y. Zheng, Signcryption and its applications in efficient public key solutions, ISW97, LNCS, pp. 291-312, 1998.

[20] J. Baek, R. Steinfeld, Y. Zheng, Formal Proofs for the Security of Signcryption, PKC02, LNCS 2274, pp.81-98.

[21] J. Herranz and G. Sáez, A provably secure ID-based ring signature scheme,available at http://eprint.iacr.org/2003/261.