An Efficient Identity-based Ring Signcryption Scheme

ZhenChao ZHU[1, 2] Yuqing ZHANG[2, *] Fengjiao WANG[2]

1. （Key Lab of Computer Networks and Information Security of Ministry of Education, Xidian University, Xi'an, 710071,P.R.China）

2. （National Computer Network Intrusion Protection Center, GUCAS, Beijing 100049,P.R.China）

Abstract: ID-based ring signcryption schemes (IDRSC) are usually derived from bilinear parings, a powerful but computationally expensive primitive. The number of paring computations of all existing ID-based ring signcryption schemes from bilinear pairings grows linearly with the group size, which makes the efficiency of ID-based schemes over traditional schemes questionable. In this paper, we present a new identity-based ring signcryption scheme, which only takes four pairing operations for any group size and the scheme is proven to be indistinguishable against adaptive chosen ciphertext ring attacks (IND-IDRSC-CCA2) and secure against an existential forgery for adaptive chosen messages attacks (EF-IDRSC-ACMA).

Keywords: Identity-based cryptography, ring signcryption, bilinear pairing

1. Introduction

Shamir introduced the concept of identity-based cryptography in 1984[1]. The idea is that the public key of a user can be publicly computed from his identity (for example, from his / her name, an e-mail or an IP address). Then, the secret key is derived from the public key. In this way, digital certificates are not needed, because anyone can easily verify that public key. The concept of public key signcryption was proposed by Zheng[2]. 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 [3, 4, 5, 6, 7]. A formal security proof of signcryption scheme was proposed in [8]. In 2005, Xinyi Huang proposed the concept of identity-based ring signcryption and give a scheme[9], in the scheme a user can anonymously signcrypts a message on behalf of a set of users including himself. The idea of ring signcryption comes from the ring signature, so, fully comprehending of ring signature is the base of truly

This work is supported in part by The National Natural Science Foundation of China (60573048, 60773135, 90718007); The High Technology Research and Development Program of China (863 Program) (2007AA01Z427，2007AA01Z450). Corresponding author. E-mail address: zhangyq @gucas.ac.cn *

comprehending of ring signcryption, let us review the concept of ring signature firstly.

2001, Rivest proposed a new type of signature which is called ring signature in the background of how to leak a secret [10]. The idea of ring signature is the following: a user wants to compute a signature on a message on behalf of a set (or ring) of users which includes himself. He wants the verifier of the signature to be convinced that the signer of the message is in effect some of the members of this ring. But he wants to remain completely anonymous. That is, nobody will know which member of the ring is the actual author of the signature. The ring signature can be seen as a special group signature, it has no trusted center and the course of building group, the signer is fully anonymous to the verifiers. Ring signature provides an artful method to leak secrets. This unconditioned anonymity of ring signature is very useful in the special circumstance which the information needs to be protected for a long time. Since the concept of ring signature was proposed, the researchers pay much attention to it. Fangguo Zhang constructed the first identity based ring signature scheme with bilinear parings in 2002[11]; in the same year, Emmanuel Bresson proposed the concept of threshold ring signature and applied it in the ad-hoc network[12]. Javier Herranz in 2003 presented the Forking lemma which makes the security proofs of ring signature schemes become easy[13]; and in the same year , FangguoZhang brought the concept of proxy signature into the ring signature and got the concept of proxy ring signature[14]. In 2004, Amit KAwasthi proposed an efficient identity based ring signature scheme and proxy ring signature scheme[15]; Tony K. Chan proposed the concept of blind ring signature scheme[16]; Javier Herranz proposed a new identity based ring signature[17]. In 2005, Chow S S M used a new technique to construct a new identity based signature scheme [18]which only takes two pairing operations for any group size, the generation of the signature involves no pairing computations at all, and the proposed scheme is proven to be existential unforgeable against adaptive chosen message-and identity attack under the random oracle model. In 2006, Yiqun Chen presented an identity based anonymous designated ring Signature scheme which is suitable for the P2P networks[19].

The development of identity based ring signcryption does not grow quickly like that of the identity based ring signature , the reason is that ID-based ring signcryption schemes are usually derived from bilinear pairings, a powerful but computationally expensive primitive , and we know that the number of pairing computations of all existing identity-based ring signcryption schemes in the literature from bilinear pairings grows linearly with the group size, which makes the efficiency of ID-based schemes over traditional schemes questionable. It is fair to say that devising an ID-based ring signcryption using sublinear numbers of pairing computation remains an important problem. We will settle this problem in this paper. we propose an efficient ID-based ring

signcryption scheme which only takes four pairing operations for any group size, the proposed scheme is proven to be indistinguishable against adaptive chosen ciphertext ring attacks (IND-IDRSC-CCA2) and secure against an existential forgery for adaptive chosen messages attacks (EUF- IDRSC -ACMA) under the random oracle model.

Roadmap: The paper is organized as follows: In section 2 we give some mathematical background which will be used in our scheme; The framework and the security notion of ID-based ring signcryption schemes are discussed in section 3; Then, we present our ID-based ring signcryption scheme in section 4; We prove the security of this scheme in the random oracle model in section 5, the underlying security model is based on the difficulties of Decisional Bilinear Diffie-Hellman problem (DBDHP) and Computational Diffie-Hellman problem (CDHP).

2. Preliminaries

2.1 Bilinear Pairings

Bilinear pairing is an important primitive for many cryptographic schemes. In this section, we briefly review some preliminaries that will be used throughout this paper.

Let G_1 be an additive group of prime order q, generated by p, and let G_2 be a multiplicative group with the same order *q*. We assume that there is a bilinear map e from $G_1 \times G_1 \to G_2$ with the following properties:

(1) Bilinearity: Which means that given elements $A_1, A_2, A_3 \in G_1$, we have that $e(A_1 + A_2, A_3) = e(A_1, A_3) \cdot e(A_2, A_3)$ and $e(A_1, A_2 + A_3) = e(A_1, A_2) \cdot e(A_1, A_3)$. In particular, for $e(aP, bP) = e(P, P)^{ab} = e(P, abP) = e(abP, P)$;

(2) Non-degeneracy: Which means that there exists $A_1, A_2 \in G_1$ such that $e(A_1, A_2) \neq 1_{G_2}$;

(3) Computability: Which means that there exists an efficient algorithm to compute $e(A_1, A_2)$ $\forall A_1, A_2 \in G_1$.

The typical way of obtaining such pairings is by deriving from Weil or Tate pairing on an elliptic curve over finite field.

2.2 Related Complexity Assumptions

We consider the following problems in the group G_1 of prime order *q*, generated by p.

Definition 1. The Decisional Bilinear Diffie-Hellman problem (DBDHP) is, given a generator p of a group G, a tuple (aP, bP, cP) and an element $h \in G_2$, to decide whether $h = e(P, P)^{abc}$.

Definition 2. Given a generator p of a group G and a tuple (aP, bP), the Computational Diffie-Hellman problem (CDHP) is to compute abP.

3. Formal Model of Identity based Ring Signcryption Schemes

Definition 3, definition 4 and definition 5 come from the paper [9], here we use them directly. The definition 3 gives the formal model of ID-based ring signcryption schemes, the definition 4 and definition 5 are the security requirement for identity based ring signcryption schemes. The definition of the security of identity based ring signcryption schemes is a transmogrification of the first formal security definition of singcryption given by Baek et.al in [8].

Definition 3 . An identity based ring signcryption scheme consists of the following algorithms.

Setup: given a security parameter k, a trusted private key center (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 receiver Bob whose identity is ID_B , Alice chooses some other users to form a group U including herself and computes $Signcrypt(U, ID_B, m)$ on the behalf of the group U to obtain the ciphertext δ .

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

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

$$\Pr[\delta \leftarrow signcrypt(U, ID_B, m), m \leftarrow Unsigncrypt(U, D_{ID_B}, \delta)] = 1$$

Definition 4 . We say that an identity based ring signcryption (IDRSC) is indistinguishable against adaptive chosen ciphertext ring attacks (IND-IDRSC-CCA2) 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 A .

• The adversary A performs a polynomially bounded number of requests:

– Signcryption request: A produces a set of users U , an identity ID_j and a plaintext m . The challenger randomly chooses user $U_i \in U$ whose identity is ID_i and computes $D_{ID_i} = Keygen(ID_i)$. Then the challenger acts as U_i to Signcrypt (U, ID_j, m) on the behalf of U and sends the result to A .

– Unsigncryption request: A produces a set of users U , an identity ID , and a ciphertext δ . The challenger generates the private key $D_{ID} = Keygen(ID)$ and sends the result of $UnSigncrypt(U, D_{ID}, \delta)$ to A (this result can be the "$\perp$" symbol if δ is an invalid ciphertext).

– Key extraction request: A produces an identity ID and receives the extracted private key

$D_{ID} = Keygen(ID)$. A can present his requests adaptively: every request may depend on the answers to the previous ones.

• A chooses two plaintexts $m_0, m_1 \in M$, n users whose identities are $\{ID_1, ID_2, \cdots ID_n\}$ to form a users set U and an identity ID_B on which he wants to be challenged. He can not have asked the private key corresponding to any user in the group U nor ID_B in the first stage.

• The challenger takes a bit $b \in_R \{0,1\}$ and computes the ciphertext δ of m_b which is sent to A

• 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 U nor ID_B and he can not ask the plaintext corresponding to δ.

• Finally, A produces a bit b^* and wins the game if $b^* = b$. The adversary's success probability is defined as $Succ_A^{IND-RSC-CCA}(k) = \frac{1}{2} + \varepsilon$,We require that ε to be negligible in k.

Definition 5. 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 A.

• The adversary A performs a polynomial bounded number of requests as in the previous definition.

• Finally, A produces a new triple $\delta = Signcrypt(U, ID, m)$ (i.e. a triple that was not produced by the signcryption oracle), where the private keys of the users in the group U and the receiver (whose identity is ID) were not asked in the second stage and wins the game if the result of the $UnSigncrypt(U, D_{ID}, \delta)$ is not the $\perp$ symbol. The advantage of A is defined as the probability that it wins.

4. Our scheme

We present our identity based ring signcryption scheme from bilinear pairing.

Setup: Given security parameters k and L, a trusted private key generator (PKG) chooses two groups G_1, G_2 of prime order $q > 2^k$, a bilinear map $\hat{e}$ from $G_1 \times G_1 \to G_2$, and a generator p of 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 some cryptographic hash functions described as follows: $H_1 : \{0,1\} \to G_1^*; H_2 : G_2 \to \{0,1\}^l; H_3 : \{0,1\}^l \times G_2 \to \{0,1\}^l; H_4 : \{0,1\}^* \to Z_q^*$, the security analysis will view H_1, H_2, H_3, H_4 as random oracles. The message space is $M = \{0,1\}^l$. Finally,

PKG publishes $\{G_1, G_2, \hat{e}, P, P_{Pub}, H_1, H_2, H_3, H_4, q\}$, but s is kept secret.

Keygen: For a user whose identity information is ID_i, PKG computes $Q_{ID_i} = H_1(ID_i)$ and calculates the user's secret key as $D_{ID_i} = sQ_{ID_i}$ where s is the PKG 's master key and sends D_{ID_i} to ID_i via a secure and authenticated channel.

Signcryption: Let $U = \{ID_1, ID_2, ..., ID_n\}$ be the set of all identities of n users. The actual signcrypter, indexed by ID_S, carries out the following steps to give an ID-based ring signcryption ciphertext on behalf of the group U and sends it to a receiver, Bob, Whose identity is ID_B.

(1) Randomly chooses $r \in_R Z_q^*$, $m^* \in_R M$ and computes $R_0 = rP$, $R' = \hat{e}(r \cdot P_{Pub}, Q_{ID_B})$, $k = H_2(R')$, $c_1 = m^* \oplus k$, $c_2 = m \oplus H_3(m^* \| R_0)$.

(2) Randomly chooses $U_i \in_R G_1^*$, $h_i = H_4(c_2 \| U_i)$, $\forall i \in \{1,2...n\} \setminus \{S\}$, Randomly chooses $r' \in_R Z_q^*$, $U_S = r' \cdot Q_{ID_S} - \sum_{i \neq S} \{U_i + h_i \cdot Q_{ID_S}\}$, $h_S = H_4(c_2 \| U_S)$,and $V = (h_S + r') \cdot S_{ID_S}$. Define the ciphertext of message m as:

$$\delta = (R_0, c_1, c_2, \bigcup_{i=1}^{n} \{U_i\}, V\}$$

and sends δ to Bob.

Designcryption: Upon receiving the ciphertext $\delta = (R_0, c_1, c_2, \bigcup_{i=1}^{n} \{U_i\}, V\}$, Bob designcrypts the ciphertext using his secret key D_{ID_B} :

(1) For $i \in \{1,2...n\}$, computes $h_i = H_4(c_2 \| U_i)$.

(2)Checking whether $\hat{e}(P_{Pub}, \sum_{i=1}^{n} (U_i + h_i \cdot Q_{ID_S})) = \hat{e}(P, V)$, if so, Computes $k' = H_2(R') = H(\hat{e}(R_0, D_{ID_B}))$, recoveries $m^* = c_1 \oplus k'$, $m' = c_2 \oplus H_3(m^* \| R_0)$, and accepts m' as an valid message. Otherwise, Bob rejects the ciphertext.

5. Security Analysis

In this section, we will provide two formal proofs that our scheme is IND-IDRSC-CCA2 assuming the Decisional Bilinear Diffle-Hellman problem (DBDHP) is hard and EF-IDRSC -ACMA assuming the Diffie-Hellman problem (CDHP) is hard.

Theorem 1. In the random oracle model, we assume an adaptive chosen ciphertext attacks adversary A that can distinguish ciphertexts from the users set U during the game of definition 4 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_S Signcryption requests and q_U Unsigncryption requests. Then there exists a distinguisher B that can solve the Decisional Bilinear Diffle-Hellman problem（DBDHP）with an advantage:

$$\varepsilon - q_U / 2^k / q_{H_1} \cdot e^{n+q_E}$$

Proof of the Theorem 1. The distinguisher B receives a random instance (P, aP, bP, cP, h) of the Decisional Bilinear Diffie-Hellman Problem, and his goal is to decide whether $h = \hat{e}(P,P)^{abc}$ or not. B will run A as a subroutine and act as A's challenger in the IND-IDRSC-CCA2 game. 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 A to oracles H_1, H_2, H_3, H_4 respectively. We assume that any Signcryption or Designcryption request between a group U and an identity ID happens after A asked the hashing H_1 of this ID and the identities in the group U. Any key extraction query on the identity is also preceded by a hash query on the same identity. We also assume that A never makes a Designcryption query on a ciphertext obtained from the signcryption oracle, and he only makes Designcryption queries for observed ciphertext.

At the beginning of the game, B runs the Setup program with the parameter k, and gives A the system parameters $\{G_1, G_2, \hat{e}, P, P_{Pub}, H_1, H_2, H_3, H_4, q\}$ with $P_{Pub} = cP$, H_1, H_2, H_3, H_4 are random oracles described as follows:

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

- At the j^{th} H_1 request, 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 $(\mathrm{ID}_j, \mathrm{Q}_{\mathrm{ID}_j}, b_j, c_j)$ in the list L_1).
- For $i \neq j$, B responds as follows:

– If the ID_i already appears on the L_1 in the tuple $(\mathrm{ID}_i, \mathrm{Q}_{\mathrm{ID}_i}, b_i, c_i)$, then B responds with $H_1(ID_i) = Q_{ID_i}$.

– Otherwise, B generates a random $coin \in \{0,1\}$ so that $\Pr[coin = 1] = \eta$, for some η that will be determined later. Let $c_i = coin$.

- B picks a random $b_i \in Z_q^*$, computes $Q_{ID_i} = b_i P$.
- B adds the tuple $(\mathrm{ID}_i, \mathrm{Q}_{\mathrm{ID}_i}, b_i, c_i)$ to the list L_1, and responds to A with $H_1(ID_i) = Q_{ID_i}$.

H_2 requests : At any time, A can ask a polynomially bounded number of H_2 requests of his choice. To respond these queries, B maintains the list L_2 of tuples (R_i, k_i). The list is initially empty. When A queries the oracle H_2 of the request $H_2(R_i)$, B first searches a pair (R_i, k_i) in list L_2. If such a pair is found, B answers by k_i. Otherwise he answers A by a random binary Sequence $k_i \in \{0,1\}^l$ 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, A can ask a polynomially bounded number of H_3 requests of his choice. To respond these queries, B maintains the list L_3 of tuples (m_x^*, R_x, y_x). The list is initially empty. When A queries the oracle H_3 of the request $H_3(m_x^*, R_x)$, B first searches (m_x^*, R_x, y_x) in

list L_3. If such a pair is found, B answers by y_x. Otherwise, he answers A by a random binary sequence $y_x \in \{0,1\}^l$ such that no entry (m_x^*, R_x, y_x) appears in L_3 (in order to avoid collisions on H_3) and adds the pair (m_x^*, R_x, y_x) to L_3.

H_4 requests: At any time, A can ask a polynomially bounded number of H_4 requests of his choice. To respond these queries, B maintains the list L_4 of tuples (c_2, U_i, x_i). The list is initially empty. When A queries the oracle H_4 of the request $H_4(c_2, U_i)$, B searches a pair (c_2, U_i, x_i) in list L_4. If such a pair is found, B answers by x_i. Otherwise he answers A by a random binary sequence $x_i \in_R Z_q^* \in R$ such that no entry (c_2, U_i, x_i) appears in L_4 (in order to avoid collisions on H_4) and adds the pair (c_i, U_i, x_i) to L_4.

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

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

Unsigncryption requests: At any time, A can perform an unsigncryption request for a ciphertext $\delta = (R_0, c_1, c_2, \bigcup_{i=1}^{n}\{U_i\}, V\}$ between the group U and receiver whose identity is ID. In other case while the receiver's identity is not ID_j, For $i \in \{1,2...n\}$, B checks whether:

$$\hat{e}(P_{Pub}, \sum_{i=1}^{n}(U_i + h_i \cdot Q_{ID_S})) = \hat{e}(P, V), \quad h_i = H_4(c_2 \| U_i)$$

if so, Compute $k' = H_2(R') = H(\hat{e}(R_0, D_{ID}))$, $m^* = c_1 \oplus k'$, $m' = c_2 \oplus H_3(m^* \| R_0)$ and accepts m' as an valid message. Otherwise, Bob rejects the ciphertext. If $ID = ID_j$, B always notifies A that the ciphertext is invalid (because B does not know the secret key of the user whose identity is ID_j). If this ciphertext δ is a valid one, the probability that A will find is no more than $1/2^k$.

Challenge: After a polynomially bounded number of queries, A chooses two messages $m_0, m_1 \in M$, n users whose identities are $\{ID_1, ID_2, \cdots ID_n\}$ to form a users set U and another user whose identity is ID .If $ID \neq ID_j$, B fails and stops. $\forall i \in \{1, 2, \cdots, n\}$, if $c_i = 1$ in the

corresponding tuple $(ID_i, Q_{ID_i}, b_i, c_i)$ in L_1, B also fails and stops. If such U and the receiver are admissible, B chooses $b \in_R \{0,1\}$,let $R_0 = aP$, $R' = h$, $k = H_2(h)$, randomly chooses $m^* \in_R M$,computes $c_1 = m^* \oplus k$, $c_2 = m_b \oplus H_3(m^* \| aP)$; randomly chooses $U_i \in_R G_1^*$, computes $h_i = H_4(c_2 \| U_i), \forall i \in \{1,2...n\} \setminus \{S\}$; and randomly chooses $r' \in_R Z_q^*$, computes $U_S = r' \cdot Q_{ID_S} - \sum_{i \neq S} \{U_i + h_i \cdot Q_{ID_S}\}$, $h_S = H_4(c_2 \| U_S)$, $V = (h_S + r') \cdot S_{ID_S}$ .Define the ciphertext of message m_b as:

$$\delta = (aP, c_1, c_2, \bigcup_{i=1}^{n} \{U_i\}, V\}$$

and sends δ to Bob.

then B signcrypts the message m_b as described in the signcryption request and sends the ciphertext δ to A.

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 δ. At the end of the simulation, he produces a bit b^* for which he believes the relation $\delta = Signcrypt(U, ID_j, m_{b^*})$ holds and sends b^* to B . At this moment, if $b^* = b$, B then answers 1 as a result because his selection h allowed him to produce a ciphertext C that appeared to A as a valid signcrypted text of m_b. If $b^* \neq b$, B then answers 0. The analysis of B's probability of success is as follows:

The probability that B does not fail during the key extraction requests is η^{q_E} , during the challenge process , the probability that B does not fail is $(1-\eta)^n / q_{H_1}$, the probability that B does not fail is $\eta^{q_E}(1-\eta)^n / q_{H_1}$, the value of probability get its maximum at the point $\eta = 1 - \frac{n}{(q_E + n)}$, which is $(1/q_{H_1})(1/e)^{n+q_E}$, adding the false answers during the Unsigncryption process. We first let

$$\Pr = P[b = b^* \,|\, \delta = Signcrypt(U, ID_j, m_b)] = \frac{1}{2} + \varepsilon$$

$$P_0 = P[b^* = i \,|\, h \in_R G_2] = \frac{1}{2}, i = 0,1$$

$$Adv(B) = \left| P_{a,b \in_R F_q, h \in_R G_2}[1 \leftarrow B(aP, bP, cP)] - P_{a,b \in_R F_q}[1 \leftarrow B(aP, bP, cP, e(P,P)^{abc})] \right|$$

$$\geq \frac{|P_1 - P_0|}{q_{H_1} \cdot e^{n+q_E}} = \frac{\left| \varepsilon - q_U / 2^k \right|}{q_{H_1} \cdot e^{n+q_E}} = \varepsilon - q_U / 2^k \,/\, q_{H_1} \cdot e^{n+q_E}$$

Theorem 2. In the random oracle model (the hash functions are modeled as random oracles), if there is an algorithm A that can win the EF-IDRSC-CMIA game with non-negligible probability by making a valid ring signcryption of group size n, in polynomial time with probability 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_S Signcryption requests and q_U Unsigncryption requests. Then there exists a challenger C that can solve the computional Diffle-Hellman problem $(CDHP)$ with an advantage

$$\varepsilon_C \geq \frac{\varepsilon^2}{66C_n^{q_{H_1}}} \cdot (1-\frac{1}{2^k})^{q_U} \cdot (\frac{n}{e \cdot q_E})^n .$$

Proof of the Theorem 2: Suppose the challenger C receives a random instance (P, aP, bP) of the $CDHP$ and has to compute the value of abP, C will run A as a subroutine and act as A's challenger in the EUF-IDRS-CMIA game. During the game, A will consult C for answers to the random oracles H_1, H_2, H_3, H_4 respectively. Roughly speaking, these answers are randomly generated, but to maintain the consistency and to avoid collision, C keeps four lists to store the answers used. We assume that any Signcryption or Designcryption request between a group U and an identity ID happens after A asked the hashing H_1 of this ID and the identities in the group U. Any key extraction query on the identity is also preceded by a hash query on the same identity. We also assume that A never makes a Designcryption query on a ciphertext obtained from the signcryption oracle. He only makes Designcryption queries for observed ciphertext . C gives A the system parameters with $P_{Pub} = bP,$ the value b is unknown to C.

H_1 **requests:** We embed part of the challenge aP in the answer of many H_1 queries. When A asks queries on the hash value of identity ID, C picks $Y_i \in_R Z_q^*$ and repeats the process until Y_i is not in the list L_1. C then flips a coin $W \in \{0,1\}$ that yields 0 with probability η and 1 with probability $1-\eta$ (η will be determined later.). If $W = 0$ then the hash value $H_1(ID)$ is defined as $Y_i P$; else if $W = 1$ then returns $H_1(ID) = Y_i(aP)$. In either case, C stores (ID, Y_i, W) in the list L_1.

H_2 **requests** : At any time, A can ask a polynomially bounded number of H_2 requests of his choice. To respond these queries, C maintains the list L_2 of tuples (R_i, k_i). The list is initially empty. When A queries the oracle H_2 of the request $H_2(R_i)$, C first searches a pair (R_i, k_i) in list L_2. If such a pair is found, B answers by k_i. Otherwise he answers A by a random binary Sequence $k_i \in \{0,1\}^l$ 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, A can ask a polynomially bounded number of H_3 requests of his choice. To respond these queries, C maintains the list L_3 of tuples (m_x^*, R_x, y_x). The list is initially empty. When A queries the oracle H_3 of the request $H_3(m_x^*, R_x)$, C first searches a pair (m_x^*, R_x, y_x) in list L_3. If such a pair is found, C answers by y_x. Otherwise, he answers A by a random binary sequence $y_x \in \{0,1\}^l$ such that no entry (m_x^*, R_x, y_x) appears in L_3 (in order to

avoid collisions on H_3) and adds the pair (m_x^*, R_x, y_x) in L_3.

H_4 **requests**: At any time, A can ask a polynomially bounded number of H_4 requests of his choice. To respond these queries, C maintains the list L_4 of tuples (c_2, U_i, x_i). The list is initially empty. When A queries the oracle H_4 of the request $H_4(c_2, U_i)$, C searches a pair (c_2, U_i, x_i) in list L_4. If such a pair is found, B answers by x_i. Otherwise he answers A by a random binary sequence $x_i \in_R Z_q^*$ ∈R such that no entry (c_2, U_i, x_i) appears in L_4 (in order to avoid collisions on H_4) and adds the pair (c_i, U_i, x_i) in L_4.

Key Extraction requests: At any time, A can ask a polynomially bounded number of key extraction requests of his choice. When A asks a query $Keygen(ID_i)$, C first finds the corresponding tuple (ID, Y_i, W) in L_1 (From the assumption we know that there must be such a tuple in L_1). If $W = 1$, B fails and stops. Otherwise if $W = 0$, C computes the secret key $Y_i(bP)$, then C returns $Y_i(bP)$ to A.

Signcryption requests: A chooses a group of n users' identities $U = \{ID_1, ID_2, \cdots ID_n\}$ where $1 \le i \le n$, another user whose identity is ID and any message $m \in_R M$. On input of (U, ID, m), C outputs an ID-based ring signcryption δ send it to the receiver, Bob, Whose identity is ID_B as follows.

1.Randomly chooses $r \in_R Z_q^*$, $m^* \in_R M$ and computes $R_0 = rP$, $R' = \hat{e}(r \cdot P_{Pub}, Q_{ID_B})$, $k = H_2(R')$, $c_1 = m^* \oplus k$, $c_2 = m \oplus H_3(m^* \| R_0)$;

2. Chooses an index $S \in_R \{1, 2, \cdots n\}$;

3. Randomly chooses $U_i \in_R G_1^*$ and compute $h_i = H_4(c_2 \| U_i)$, $\forall i \in \{1,2...n\} \setminus \{S\}$;

4. Chooses $h_S \in_R Z_q^*$ and $r' \in_R Z_q^*$ and computes $U_S = r'P - h_S Q_{ID_S} - \sum_{i \ne S} \{U_i + h_i Q_{ID_i}\}$;

5. Stores the relationship $H_4(c_2, U_i, x_i) = h_S$ to the list L_4 and computes $V = r'(bP)$, repeats the Step 4 in case collision occurs;

6. the ciphertext of message m as:

$$\delta = (R_0, c_1, c_2, \bigcup_{i=1}^{n} \{U_i\}, V\}$$

and sends δ to A.

Unsigncryption requests: At any time, A can perform an unsigncryption request for a ciphertext $\delta = (R_0, c_1, c_2, \bigcup_{i=1}^{n} \{U_i\}, V\}$ between the group U and receiver whose identity is ID. In other case where the receiver's identity is not ID_j, For $i \in \{1,2...n\}$, B checks whether:

$$h_i = H_4(c_2 \| U_i), \hat{e}(P_{Pub}, \sum_{i=1}^{n} (U_i + h_i \cdot Q_{ID_S})) = \hat{e}(P, V)$$

if so, Computes $k' = H_2(R') = H(\hat{e}(R_0, D_{ID}))$, $m^* = c_1 \oplus k'$, $m' = c_2 \oplus H_3(m^* \| R_0)$ and accepts m' as an valid message. Otherwise, Bob rejects the ciphertext. If $ID = ID_j$, B always notifies A that the ciphertext is invalid (because B does not know the secret key of the user

whose identity is ID_j). If this ciphertext δ is a valid one, the probability that A will find is no more than $1/2^k$.

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 can not ask the plaintext corresponding to the ciphertext.

Finally, A outputs a forged ciphertext $\delta = (R_0, c_1, c_2, \bigcup_{i=1}^{n}\{U_i\}, V\}$ that is signcrypted by a certain member in the group $U = \{ID_1, ID_2, \cdots ID_n\}$ where $Q_{ID_i} = H_1(ID_i) = Y_i(aP)$, $\forall$ $i \in \{1, 2, \cdots, n\}$, i.e. A has not requested for any one of the private keys of members in the group. Solving CDHP: It follows from the forking lemma for generic ring signature schemes [12] that if $\varepsilon_C \geq 7C_n^{q_H}/2^k$, and A can give a valid forged signature within time T_A in the above interaction, then we can construct another algorithm A' that outputs within time $2T_A$ two signed messages with at least $\varepsilon_C^2/66C_n^{q_H}$ probability. For the resemble construction we can get the same result in our scheme, Suppose $h_i = H_4(c_2 \| U_i)$ and $h_i' = H_4(c_2 \| U_i)$ for all $i \in \{1,2...n\}$, we have $h_i' = h_i$ for all $i \in \{1, 2, \cdots, n\} \setminus \{S\}$. Given A' derived from A, we can solve the CDHP by computing $abP = Y_S^{-1}(h_S - h_S')^{-1}(V - V')$, where Y_S can be found by looking for ID_S in the list L_1.

Probability of success: Now we determine the value of η. The probability that C does not fail in all the q_E private key extraction queries is η^{q_E}, and the probability that A forged a signature that C does not know all the corresponding private keys involved in the signcrypt ciphertext is $(1-\eta)^{n'}$. So the combined probability is $\eta^{q_E}(1-\eta)^n$. the value of η that maximize this probability is $q_E/q_E + n$, the maximized probability is $(1 - n/q_E + n)^{q_E + n}(\frac{n}{q_E + n})^n$. For enough large q_E:

$$(1 - n/q_E + n)^{q_E + n}(\frac{n}{q_E + n})^n = (1 - 1/(1 + \frac{q_E}{n})^{n \cdot (1 + \frac{q_E}{n})}(\frac{1}{\frac{q_E}{n} + 1})^n = (\frac{n}{e \cdot q_E})^n.$$

The probability for C not to fail in all the signcryption queries is $(1 - \frac{1}{2^k})^{q_U} \cdot (\frac{n}{e \cdot q_E})^n$, if the attacker A can succeed with the probability ε, the probability for C to succeed is

$$\varepsilon_C \geq \frac{\varepsilon^2}{66C_n^{q_{H_1}}} \cdot (1 - \frac{1}{2^k})^{q_U} \cdot (\frac{n}{e \cdot q_E})^n$$

6. conclusion

We present a new identity-based ring signcryption scheme, which only takes four pairing operations for any group size while the number of paring computations of all existing ID-based ring signcryption from bilinear pairing grows linearly with the group size, the reason why our scheme can use four paring computations for any group size is that the scheme does not take the method of choosing random numbers and applying them to the paring computations to get a part of ciphertext in the signcryption procedure; we also notice that if we use the character of bilinear paring to verify the validity of the ciphertext , the number is the least. The scheme is proven to be indistinguishable against adaptive chosen ciphertext ring attacks (IND-IDRSC-CCA2) and secure against an existential forgery for adaptive chosen messages attacks (EF-IDRSC-ACMA).

References

[1] A. Shamir, Identity-based cryptosystems and signature schemes, Advin Cryptology-Crypto'84, LNCS196, pp.47-53, 1984.

[2] Y. Zheng, Digital Signcryption or How to Achieve Cost (Signature & Encryption) Cost (Signature)+Cost (Encryption), Adv in Cryptology –Crypto'97,LNCS1294, pp.165-179, 1997.

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

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

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

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

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

[8] J. Baek, R. Steinfeld, Y. Zheng, Formal Proofs for the Security of Signcryption, PKC 02, LNCS 2274, pp.81-98.

[9] 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. 19th International Conference on Advanced Information Networking and Applications (AINA'05) Volume 2 (INA, USW, WAMIS, and IPv6 papers) pp.649-654.

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

[11] Fangguo Zhang and Kwangjo Kim. ID-Based Blind Signature and Ring Signature from Pairings. In Yuliang Zheng, editor, Advances in Cryptology–ASIACRYPT 2002, 8th International Conference on the Theory and Application of Cryptology and Information Security, Queenstown, New Zealand, December 1-5, 2002, Proceedings, volume 2501 of Lecture Notes in Computer Science, pages 533–547. Springer, 2002.

[12] BRESSON E, STERN J, SZYDLO M. Threshold ring signatures and applications to ad-hoc groups [A]. Proc CRYPTO'02[C].Springer-Verlag, 2002, 465-480.

[13] HERRANZ J，S'AEZ G. Forking lemmas for ring signature schemes[A]. Proc INDOCRYPT' 03[C]. Springer-Verlag, 2003, 266-279.

[14] Fangguo Zhang, REIHANEHSN, LINC Y. New Proxy Signature, Proxy blind signature and proxy ring signature schemes from bilinear pairings [EB/OL]. http//eprint.iacr.org/2003/104,2003.

[15] AWASTHI AK, SUNDER L.ID-based ring signature and proxy ring signature schemes from bilinear pairings [EB/OL]. http：//eprint.iacr.org/2004/184.2004.

[16] CHAN T K，FUNG K，LIU J K，et a1. Blind spontaneous an onymous group signatures for ad hoc groups [A], ESAS 2004[C].Springer-Verlag, 2005, 82-94.

[17] HERRANZ J，S'AEZ G. New identity-based ring signature schemes[A]．ICICS2004[C]. Springer-Verlag, 2004.27-39.

[18] Chow S S M, Hui L C K, Yiu S M. Efficient identity based ring signature[C] // Proc of ACNS 2005, Berlin: Springer-Verlag, 2005: 499-512.

[19] Y Q CHEN, SUSILO W, Y MU. Identity based anonymous designated ring Signatures [A]. IWCMC'06[C].Vancouver, British Columbia , Canada, 2006.189-194.