1-1-2014

A provably secure identity-based proxy ring signature based on RSA

Maryam Rjabzadeh Asaar
Sharif University of Technology

Mahmoud Salmasizadeh
University of Wollongong

Willy Susilo
University of Wollongong, wsusilo@uow.edu.au

A provably secure identity-based proxy ring signature based on RSA

Abstract

Proxy ring (anonymous proxy) signatures allow an entity to delegate its signing capability to a group of entities (proxy group) such that only one of the members in the proxy group can generate a proxy signature on behalf of the delegator while privacy of the proxy signer is protected. Identity-based versions of proxy ring signatures employ identity strings in place of randomly generated public keys. Our contribution is two-fold. First, we formalize a security model for identity-based proxy ring signatures. We note that there exists no formal security model for identity-based proxy ring signatures prior to our work. Second, we present the rst provably secure identity-based proxy ring signature scheme using a new paradigm called sequential aggregation. The construction is proved secure, under the one-wayness assumption of RSA, in the random oracle model by presenting a new forking lemma.We should highlight that the proxy key exposure attack cannot be applied to our scheme. Further, in contrast to the existing schemes that are based on pairings, our scheme is based on RSA; therefore, it outperforms the existing schemes in terms of eciency and practicality.

Keywords

proxy, rsa, identity, secure, ring, provably, signature

Disciplines

Engineering | Science and Technology Studies

Publication Details

Asaar, M., Salmasizadeh, M. & Susilo, W. (2015). A provably secure identity-based proxy ring signature based on RSA. Security and Communication Networks, 8 (7), 1223-1236.

A Provably Secure Identity-based Proxy Ring Signature based on RSA

Maryam Rajabzadeh Asaar*[1], Mahmoud Salmasizadeh*[2], and Willy Susilo**[2]

[1] Department of Electrical Engineering,
[2] Electronics Research Institute (Center),
Sharif University of Technology, Tehran, Iran.
[3] Centre for Computer and Information Security Research,
University of Wollongong, Australia.
asaar@ee.sharif.ir, salmasi@sharif.edu, wsusilo@uow.edu.au

Abstract. Proxy ring (anonymous proxy) signatures allow an entity to delegate its signing capability to a group of entities (proxy group) such that only one of the members in the proxy group can generate a proxy signature on behalf of the delegator while privacy of the proxy signer is protected. Identity-based versions of proxy ring signatures employ identity strings in place of randomly generated public keys. Our contribution is two-fold. First, we formalize a security model for *identity-based* proxy ring signatures. We note that there exists no formal security model for *identity-based* proxy ring signatures prior to our work. Second, we present *the first* provably secure *identity-based* proxy ring signature scheme using a new paradigm called sequential aggregation. The construction is proved secure, under the one-wayness assumption of RSA, in the random oracle model by presenting a new forking lemma. We should highlight that the proxy key exposure attack cannot be applied to our scheme. Further, in contrast to the existing schemes that are based on pairings, our scheme is based on RSA; therefore, it outperforms the existing schemes in terms of efficiency and practicality.

Keywords: identity-based proxy ring signature, random oracle model, RSA assumption.

1 Introduction

Digital signatures are widely deployed around the world and have the backing of significant international legislation to support their use in electronic environment. One of these cryptographic primitives is *proxy ring signature* which supports ensuring service availability for the customers in distributed networks to avoid the dependency to a single server in addition to preserving privacy of proxy signers. In this work, we are interested in exploring proxy ring signatures in the identity-based setting, due to its practicality.

Identity-based cryptography. Public-key cryptography has many different applications, but in its basic form, it requires extensive public-key infrastructure for practical use. In order to provide more flexible management of public keys the notion of identity-based cryptography was introduced by Shamir [1]. The main feature of identity-based cryptosystems is to remove the requirement of certification of the public keys. The public key of each party is obtained from its public identity, such as the IP address or email address, which can uniquely identify the party. Since the introduction of the notion in [1], various identity based schemes ([2–4]) have been proposed.

Identity-based cryptography has attracted a lot of interest since the elliptic curve pairings are shown to provide an elegant way for implementing identity-based encryption schemes. In the past ten years, the majority of identity-based cryptosystems proposed have relied on pairings. While extensive research has led to vast improvements in implementation of pairings, their computational cost is still higher than that of traditional public key algorithms which use the exponentiation operation in various groups. Moreover, pairing-based cryptosystems rely on newer and less analyzed computational assumptions in their security analysis compared to traditional schemes that are based on classical assumptions like the widely studied RSA assumption. There has been a proliferation of pairing-based assumptions whose difficulty is not widely understood and

* This research was supported in part by the Office of Vice-President for Science and Technology, I.R. Iran.
** W. Susilo is supported by the Australian Research Council Discovery Project (ARC DP130101383).

whose connection to established assumptions, and to each other, remains unknown [5]. Therefore, when designing new identity-based cryptographic primitives it is desirable to diversify the computational assumptions and to use widely accepted assumptions where possible.

PROXY RING SIGNATURES. The notion of proxy signatures was introduced by Mambo et al. [6] in 1996. In a proxy signature scheme, an original signer, Alice, can delegate her signing right for signing messages to another signer, Bob, called the proxy signer. Since the introduction of the notion of proxy signatures, several variants of proxy signatures such as proxy signatures from RSA and integer factorization problem ([7–12]), identity-based proxy signature schemes based on bilinear pairings ([13–19]), designated-verifier proxy signatures ([20–22]), short proxy signature [23], proxy verifiably encrypted signatures [24], proxy signature schemes without random oracles [25], identity-based multi-proxy signatures [26], proxy ring signatures ([27–31]) and identity-based proxy ring signatures from bilinear pairings ([32–39]) have been proposed.

In a proxy ring signature scheme, an original signer can delegate her signing right for signing messages to a group of proxy signers with different public keys, called the proxy agent, such that only one of the proxy signers in the proxy group can generate proxy signatures on behalf of the original signer while he could be anonymous. As mentioned in ([27–29]), this primitive can be used when the requirement of proxy signer's privacy protection is necessary. For example, it is assumed that a parliament member would like to reveal an important news on behalf of the cabinet, while he wants to be anonymous. However, one still needs to verify public keys of proxy signers and the original signer in addition to verifying the validity of a proxy ring signature.

Cheng et al. proposed the first identity-based proxy ring signature [35] to facilitate public key certificate management of these kinds of signatures by merely employing signer's identities in place of the public keys and their certificates. Subsequently, there have been some follow-up works in the area of identity-based proxy ring signatures ([32–34, 36–39]), but unfortunately, *none* of them supports provable security. Hence, the formal definition and security model for *identity-based* proxy ring signature schemes do not yet exist in the literature.

In the proxy key exposure attack [40] proposed by Schuldt et al., it is assumed that temporal secret keys of proxy signers stored in a less trusted device can be leaked, while secure storage (for example in a TPM within a laptop) is available for long term secret keys of proxy signers. With this attack not only long term secret keys of proxy signers are compromised but also an adversary (with having proxy secret keys) can generate valid (identity-based) proxy signatures. Therefore, it is vital to consider the proxy key exposure attack when we present other extensions of proxy signatures, identity-based proxy ring signatures. Unfortunately, identity-based proxy ring signatures proposed in ([32–39]) are vulnerable to this attack.

1.1 Our Contribution

The main goal of this paper is to fill the gap in the literature, by proposing a formal definition and security model for identity-based proxy ring signatures. Subsequently, we present the first provably secure identity-based proxy ring signature scheme. The paradigm used in designing this primitive is sequential aggregation of an identity-based signature and an identity-based ring signature scheme. As a result, our scheme is the first identity-based proxy ring signature scheme such that the proxy key exposure attack [40] cannot be applied to it. We achieved this by employing sequential aggregation of two signatures (i.e., the original signer's signature and the proxy signer's ring signature). This is in contrast to the previous technique that employs proxy key generation algorithm, a function of delegation and proxy long term secret key, to generate proxy secret key. The latter is vulnerable to the proxy key exposure attack since proxy secret keys are assumed to be stored on a less trusted device and there is a possibility that the adversary can find proxy secret keys and then proxy long term secret keys are compromised [40].

To prove security of the scheme, we present a new forking lemma and employ it in the proof of unforgeability. The general forking lemma [41] cannot be applied directly into our scheme since this scheme is the result of sequential aggregation of two different types of signatures such that we have two different types of random oracle responses. Hence, we need to consider the probability of happening some random responses before the forking point in the proposed forking lemma which makes it different from previous forking lemma.

We should highlight that our scheme not only is the first identity-based proxy ring signature with provable security according to a formally defined security model and is resistance against proxy key exposure attack,

but also it is the first identity-based proxy ring signature from RSA, which is more efficient than the existing constructions due to not relying on pairing computations.

1.2 Paper Organization

The rest of this paper is organized as follows. Section 2 presents notations and RSA complexity assumption employed as the signature foundation. The security model of identity-based proxy ring signature including outline of the identity-based proxy signature scheme and its security properties are given in Section 2. The proposed scheme and its formal security proofs are presented in Section 3. Section 4 and 5 present the comparison and conclusion.

2 Background

In this section, first we give notations used throughout the paper and review the RSA assumption, and then we present the outline and our security definitions for the identity-based proxy ring signature schemes.

2.1 Notations.

If S is a set, then $s \xleftarrow{\$} S$ denotes the operation of assigning to s an element of S chosen uniformly at random, and if $(i_0, ..., i_z)$ is a vector, then $i \in (i_0, ..., i_z)$ means that $\exists 0 \leq u \leq z \; i = i_u$. If $x_1, x_2, ...$ are objects then $x_1||x_2||...$ denotes an encoding of them as strings from which the constituent objects are effectively recoverable. Let $\perp$ be an empty string and $\theta \leftarrow C(x_1, ...)$ stands for the operation of assigning the output of algorithm C on inputs $x_1, ...$ to θ. Let A be an algorithm which has access to H, K, KeyExtract, DelegationGen and ProxyRingSign oracles of a signature scheme, and can win a game in which a security property of the scheme is violated by A. If algorithm A is $(t, q_h, q_k, q_e, q_d, q_{prs}, \epsilon)$-bounded, we mean that the algorithm A which runs in time at most t, makes at most q_h queries to random oracle H, q_k queries to random oracle K, q_e queries to KeyExtract oracle, q_d queries to DelegationGen and q_{prs} queries to ProxyRingSign oracle can win the game with probability at least ϵ. If probability of an algorithm in doing some tasks is negligible, it means that its value is less than inverse of a polynomial of input's length for all sufficiently large values of input's length.

2.2 The RSA assumption.

An RSA key generator KG_{rsa} is an algorithm that generates triplets (N, e, d) such that N is the product of two large primes p and q and $ed = 1 \bmod \varphi(N)$, where $\varphi(N) = (p-1)(q-1)$. The advantage of an algorithm B in breaking the one-wayness of RSA related to KG_{rsa} is defined as

$$Adv^{ow-rsa}_{KG_{rsa}}(B) = \Pr\begin{bmatrix}(N, e, d) \xleftarrow{\$} KG_{rsa}; \gamma \xleftarrow{\$} \mathbb{Z}_N; \\ y = \gamma^e \bmod N : \\ \gamma \longleftarrow B(N, e, y)\end{bmatrix}. \quad (1)$$

We say that B, (t', ϵ')-breaks the one-wayness of RSA with respect to KG_{rsa} if it runs in time at most t' and has advantage $Adv^{ow-rsa}_{KG_{rsa}}(B) \geq \epsilon'$. We say that the RSA function associated to KG_{rsa} is (t', ϵ')-one-way if no algorithm B, can (t', ϵ')-break it.

2.3 Outline of identity-based proxy ring signature schemes

When describing the signature scheme, let identity of each original signer be ID_0, and identity set of proxy agent and each subset of that be $\mathbf{ID}$ and $\widetilde{ID}$, respectively. The indices used in the signature description have no global meaning outside this protocol instance which means that there is no certified relationship between indices and identities, and just serve as local pointers for original and proxy signers. An identity-based proxy ring signature scheme consists of five algorithms: ParaGen, KeyExtract, DelegationGen, ProxyRingSign and ProxyRingVer as follows.

- ParaGen: This algorithm takes as input the system security parameter l and outputs system's parameters $Para$ and the system's master key (msk, mpk), i.e. $(Para, (msk, mpk)) \leftarrow ParaGen(l)$.

- KeyExtract: This algorithm takes as input the system's parameter $Para$, master public key mpk, master secret key msk, and an identity ID_u. It outputs the corresponding secret key x_u for the identity ID_u, i.e. $x_u \leftarrow KeyExtract(Para, mpk, msk, ID_u)$.

- DelegationGen: This algorithm takes as input the system's parameter $Para$, the master public key mpk, an identity ID_0 and an identity set $\mathbf{ID}$, including at least two identities, for an original signer and a proxy agent, respectively. It also takes as input the secret key x_0 of the original signer with identity ID_0 and a message space descriptor $w \subseteq \{0,1\}^*$ for which the original signer with identity ID_0 delegates its signing right to a proxy agent with identity set $\mathbf{ID}$, then, it outputs a delegation $\sigma_0 \leftarrow DelegationGen(Para, mpk, ID_0, \mathbf{ID}, w, x_0)$.

- ProxyRingSign: This algorithm takes as input the system's parameter $Para$, the master public key mpk, the identity set $\widetilde{ID}$ of proxy signers including at least two identities, a valid delegation σ_0 for a message space descriptor w and an identity set $\mathbf{ID}$ of proxy signers such that $\widetilde{ID} \subseteq \mathbf{ID}$ and the delegation indicates that an original signer with identity ID_0 delegates its signing right on w to a proxy agent with identity set $\mathbf{ID}$, a proxy signer's secret key x_j corresponding to an identity $ID_j \stackrel{\$}{\leftarrow} \widetilde{ID} \subseteq \mathbf{ID}$ and a message $m \in w$, then, it outputs the identity-based proxy ring signature θ on behalf of the original signer with identity ID_0, i.e. $\theta \leftarrow ProxyRingSign(Para, mpk, ID_0, \mathbf{ID}, \widetilde{ID}, (m, w, \sigma_0), x_j)$.

- ProxyRingVer: This algorithm takes as input the system's parameter $Para$, an original signer's identity ID_0, the proxy signers' identity sets $\mathbf{ID}$ and $\widetilde{ID}$, a message space descriptor w, a signed message m and a proxy ring signature θ, then, it outputs 1 if θ is a valid identity-based proxy ring signature of the message m which means that it satisfies the verification equation, $m \in w$ and $\widetilde{ID} \subseteq \mathbf{ID}$ and outputs 0 otherwise, i.e. $\{0,1\} \leftarrow ProxyRingVer(Para, mpk, ID_0, \mathbf{ID}, \widetilde{ID}, w, m, \theta)$.

2.4 Security models of identity-based proxy ring signature schemes

An identity-based proxy ring signature must satisfy two independent notions of security: unforgeability and privacy of proxy signer' identity. We present the first formal definitions for unforgeability and privacy of proxy signer's identity, respectively. To achieve existential unforgeability against adaptive chosen message (chosen warrant: chosen message space descriptor and identity set of proxy signers) and chosen identity attack for identity-based proxy ring signature schemes, three types of potential adversaries as mentioned in [27] are considered as follows.

- Type I: This type adversary A_I only has identities of the original signer and proxy signers, and aims to forge a valid identity-based proxy ring signature w.r.t. identities of the original signer and proxy signers.
- Type II: This type adversary A_{II} has secret keys of some (one/all) proxy signers in a proxy group in addition to identities of the original signer and proxy signers, and aims to forge a valid identity-based proxy ring signature w.r.t. identities of the original signer and proxy signers.
- Type III: This type adversary A_{III} has the secret key of the original signer in addition to identities of the original signer and proxy signers, and aims to forge a valid identity-based proxy ring signature w.r.t. identities of the original signer and proxy signers.

Clearly, if an identity-based proxy ring signature scheme is secure against Type II (or Type III) adversaries then it is also secure against Type I adversary. Unforgeability against Type I, Type II and Type III adversaries (A_I, A_{II} and A_{III}) is formalized using the following game between a challenger C and an adversary A.

1. Setup: C runs the $ParaGen$ algorithm with a security parameter l to obtain system's parameter $para$ and the master key (mpk, msk), then it sends $(mpk, para)$ to A.

   A issues a polynomially bounded number of queries to the following oracles adaptively:

2. KeyExtract queries: A can ask for the secret key corresponding to each identity ID_u, then C returns the private key x_u to the adversary with running the KeyExtract algorithm.

3. DelegationGen queries: Adversary A can request a delegation under the identity ID_0 of an original signer on a message space descriptor w and an identity set $\mathbf{ID}$ of its choice for which the original signer with identity ID_0 delegates its signing right on w to a proxy agent with identity set $\mathbf{ID}$. In response, C runs the KeyExtract algorithm to obtain the secret key x_0 of the original signer, and returns $\sigma_0 \leftarrow DelegationGen(Para, mpk, ID_0, \mathbf{ID}, w, x_0)$ to A.

4. ProxyRingSign queries: Adversary A can request the proxy ring signature of m w.r.t. $\widetilde{ID}$ to C.
In addition, adversary A provides a delegation σ_0 of an original signer with identity ID_0 for a message space descriptor w and an identity set $\mathbf{ID}$ of proxy signers. This delegation was obtained from DelegationGen algorithm or was generated by adversary A.
Algorithm C checks that σ_0 is a valid delegation in which the original signer with identity ID_0 delegates its signing right for the message space descriptor w to the proxy agent with identity set $\mathbf{ID}$; that $\widetilde{ID} \subseteq \mathbf{ID}$; and that $m \in w$. If any of these fails to hold, returns $\perp$. Otherwise, C runs the KeyExtract algorithm to obtain the secret key x_j corresponding to one of the proxy signers with identity ID_j such that $ID_j \xleftarrow{\$} \widetilde{ID}$. Next, C runs ProxyRingSign algorithm $\theta \leftarrow ProxyRingSign(Para, mpk, ID_0, \mathbf{ID}, \widetilde{ID}, (m, w, \sigma_0), x_j)$ to generate the proxy ring signature θ and returns it to the adversary A.

5. Finally, A outputs a valid identity-based proxy ring signature (m^*, w^*, θ^*) w.r.t. original signer's identity ID_0^* and proxy signers' identity sets $\mathbf{ID}^*$ and $\widetilde{ID^*} \subseteq \mathbf{ID}^* \setminus \widehat{ID^*}$, where $\widehat{ID^*}$ is the set of corrupted proxy signers, and wins the game if the following conditions hold.

   For $A = A_I$:
   - E_0: ID_0^* and all identities in $\widetilde{ID^*}$ have not been requested to the KeyExtract oracle which means that A_I does not have secret keys corresponding to them.
   - E_1: The pair $(w^*, \mathbf{ID}^*)$ has not been requested as one of the DelegationGen queries under the identity ID_0^*.
   - E_2: m^* has not been requested as one of the ProxyRingSign queries under the identity set $\widetilde{ID^*}$.

   The formal definition of existential unforgeability against adversary A_I is expressed in Definition 1.

Definition 1. *An identity-based proxy ring signature is $(t, q_h, q_e, q_d, q_{prs}, \epsilon)$-existentially unforgeable against adaptive chosen message (warrant) and chosen identity attack if there is no $(t, q_h, q_e, q_d, q_{prs}, \epsilon)$-bounded adversary A which wins the aforementioned game.*

For $A = A_{II}$:

- E_0: ID_0^* has not been requested as one of the KeyExtract queries which means A_{II} does not have the secret key corresponding to ID_0^*.
- E_1: The pair $(w^*, \mathbf{ID}^*)$ has not been requested as one of the DelegationGen queries under the identity ID_0^*. .

The formal definition of existential unforgeability against adversary A_{II} is expressed in Definition 2.

Definition 2. *An identity-based proxy ring signature is $(t, q_h, q_e, q_d, \epsilon)$-existentially unforgeable against adaptive chosen message (warrant) and chosen identity attack if there is no $(t, q_h, q_e, q_d, \epsilon)$-bounded adversary A which wins the aforementioned game.*

For $A = A_{III}$:

- E_0: Each identity in $\widetilde{ID^*}$ has not been requested as one of the KeyExtract queries which means that A_{III} does not have the secret keys corresponding to identities in $\widetilde{ID^*}$.
- E_1:m^* has not been requested as one of the ProxyRingSign queries under identity set $\widetilde{ID^*} \subseteq \mathbf{ID}^*$.

The formal definition of existential unforgeability against adversary A_{III} is expressed in Definition 3.

Definition 3. *An identity-based proxy ring signature is* $(t, q_h, q_e, q_{prs}, \epsilon)$- *existentially unforgeable against adaptive chosen message (warrant) and chosen identity attack if there is no* $(t, q_h, q_e, q_{prs}, \epsilon)$*-bounded adversary A which wins the aforementioned game.*

Privacy of proxy signer's identity (PPSI) in an identity-based proxy ring signature means that it should be infeasible for any probabilistic polynomial time (PPT) distinguisher D to tell which proxy signer in a proxy group generates θ on a message m. To have a formal definition for this property consider the following game between a challenger C and a distinguisher D.

1. Setup: C runs the $ParaGen$ algorithm with a security parameter l to obtain system's parameter $para$ and the master key (mpk, msk), then it sends $(mpk, para)$ to D.

   The distinguisher D issues a polynomially bounded number of KeyExtract, DelegationGen and ProxyRingSign queries adaptively as explained in the forgery game.

2. the distinguisher D chooses two honest identities ID_1 and ID_2 (D never make KeyExtract query for these two identities), and makes a DelegationGen and ProxyRingSign query on $(w, \mathbf{ID})$ under an identity ID_0 and on the message $m \in w$ under the identity set $\widetilde{ID} = \{ID_1, ID_2\} \subseteq \mathbf{ID}$, respectively. In response, C chooses $j \xleftarrow{\$} \{1, 2\}$, runs KeyExtract for ID_0 and ID_j to obtain their corresponding secret keys, and runs DelegationGen on $(w, \mathbf{ID})$ under the identity ID_0 to obtain σ_0 and returns $\theta \leftarrow ProxyRingSign(Para, mpk, ID_0, \mathbf{ID}, \widetilde{ID}, (w, m, \sigma_0), x_j)$ to D.
3. Finally, the distinguisher D outputs j' and wins the game if $j' = j$.

The formal definition for privacy of proxy signer's identity is given in definition 4.

Definition 4. *(Privacy of the proxy signer's identity). An identity-based proxy ring signature scheme is* $(t, q_h, q_e, q_d, q_{prs}, \epsilon + \frac{1}{2})$*-PPSI-secure if there is no* $(t, q_h, q_e, q_d, q_{prs}, \epsilon + \frac{1}{2})$*-bounded adversary D which can win the aforementioned game.*

If the probability is equal to $\frac{1}{2}$, the scheme satisfies privacy of the proxy signer's identity perfectly.

3 Our identity-based proxy ring signature scheme

In this section, we present an identity-based proxy ring signature scheme using a new paradigm called sequential aggregation of GQ identity-based signature [42] and GQ identity-based ring signature scheme [43]. The main reason of employing this paradigm is to ensure that the proxy key exposure attack does not have any impact on the scheme. Our scheme generates an identity-based proxy ring signature scheme in a way that a delegation is original signer's GQ identity-based signature on a message space descriptor and proxy signers' identities concatenated with "11"[4] to differentiate delegations from standard signatures, and a proxy ring signature is sequential aggregation of a delegation and a ring signature generated by one of the proxy signers on a message, which belongs to the message space descriptor concatenated with "11" to differentiate them from sequential aggregation of delegations and ordinary ring signatures generated by proxy signers. We note that the trick to concatenate with "11", as suggested by Boldyreva et al. [44], prevents trivial attacks to the scheme.

3.1 Details of identity-based proxy ring signature scheme

In this section, we present the details of identity-based proxy ring signature scheme. When describing the signature scheme, let identity of each original signer be ID_0, and identity set of each proxy agent and each subset of that be $\mathbf{ID}$ and $\widetilde{ID}$, respectively. The indices used in the signature description have no global

[4] We borrow this technique from Boldyreva et al. [44] to realize this.

meaning outside this protocol instance which means that there is no certified relationship between indices and identities, and just serve as local pointers for original and proxy signers.

It is assumed that $n \geq 2$ is the number of identities for proxy signers in the proxy agent, and $z \geq 2$ is the size of each subset $\widetilde{ID}$ of **ID**. Our scheme consists of five algorithms as follows.

1. Setup: The system parameters are as follows. Let l_1 and $l_N \in \mathbb{N}$ and let $K : \{0,1\}^* \rightarrow \{0,1\}^{l_1}$ and $H : \{0,1\}^* \rightarrow \mathbb{Z}_N^*$ be random oracles. Let KG_{rsa} be a RSA key pair generator that outputs triplets (N, e, d) such that $\varphi(N) > 2^{l_N}$ and with prime encryption exponents e of length strictly greater than l_1 bits. The key distribution center runs KG_{rsa} to generate RSA parameters (N, e, d). It publishes $mpk = (N, e)$ as the master public key, and keeps the master secret key $msk = d$ secret. Therefore, public parameters are $Para = \{K, H\}$ and mpk.

2. KeyExtract: On input master secret key $msk = d$ and the user identity ID_u, the key distribution center computes $x_u = H(ID_u)^d \bmod N$, and sends the user secret key x_u over a secure and authenticated channel to the user with identity ID_u.

3. DelegationGen: Let w be a message space descriptor for which an original signer with identity ID_0 would like to delegate her signing right to a group of proxy signers with an identity set **ID**, the delegation is $\sigma_0 = (R_0, s_0) = (r_0^e \bmod N, r_0 x_0^{c_0} \bmod N)$, where $r_0 \xleftarrow{\$} \mathbb{Z}_N^*$ and $c_0 = K(R_0||w||\mathbf{ID}||11)$. Then, the original signer publishes the delegation σ_0 on $(w, \mathbf{ID})$.

4. ProxyRingSign: A proxy signer with identity $ID_j \xleftarrow{\$} \widetilde{ID} \subseteq \mathbf{ID}$ $(j \in \{1, ..., z\})$ can sign a message $m \in w$ anonymously on behalf of the original signer with identity ID_0 with his secret key x_j and the delegation σ_0 as follows.
   - For $1 \leq u \neq j \leq z$, the proxy signer ID_j chooses $r_u \xleftarrow{\$} \mathbb{Z}_N^*$, and computes $R_u = r_u^e \bmod N$ and $c_u = K(R_u||\mathbf{ID}||\widetilde{ID}||R_0||w||m||11)$.
   - The proxy signer ID_j chooses $r \xleftarrow{\$} \mathbb{Z}_N^*$, and computes $R_j = r^e[\prod_{u \neq j} H(ID_u)^{-c_u}] \bmod N$ and $c_j = K(R_j||\mathbf{ID}||\widetilde{ID}||R_0||w||m||11)$.
   - The proxy signer ID_j computes $s = s_0 x_j^{c_j} r \prod_{u \neq j} r_u \bmod N$.
   - The proxy ring signature is $\theta = (R_0, ..., R_z, s)$ on the message m and the message space descriptor w w.r.t. original signer's identity ID_0 and a subset $\widetilde{ID} \subseteq \mathbf{ID}$ of proxy signers.

5. ProxyRingVer: Given the identity ID_0 of an original signer and identity sets **ID** and $\widetilde{ID}$ of the proxy signers, a message space descriptor w, a message m, and a proxy ring signature θ, a verifier operates as follows:
   - Checks if $m \in w$, otherwise, it stops.
   - Checks if $\widetilde{ID} \subseteq \mathbf{ID}$, otherwise, it stops.
   - Accepts the proxy ring signature if and only if $s^e = R_0 H(ID_0)^{c_0}[\prod_{u=1}^{z} R_u H(ID_u)^{c_u}]$, where $c_0 = K(R_0||w||\mathbf{ID}||11)$ and $c_u = K(R_u||\mathbf{ID}||\widetilde{ID}||R_0||w||m||11)$ for $1 \leq u \leq z$.

3.2 Analysis of the scheme

In this section, we verify the correctness, and prove the privacy of the proxy signer's identity and existential unforgeability of the proposed scheme in the random oracle model (see [45] for the background).

In order to prove unforgeability of the proposed scheme, we need to show that it is unforgeable against adversaries of types II and III (as defined in Section 2.4). Since our security proofs are quite similar in both cases, we have parametrized these proofs to prevent unnecessary repetitions of arguments. Hence, just for notational settings, we refer to the adversary as $A_{(1-\zeta)II+\zeta III}$ in which the parameter $\zeta \in \{0,1\}$ makes the difference between adversaries of types II and III (i.e. notationally we assume that we have an adversary of type II, A_{II}, when $\zeta = 0$ and an adversary of type III, A_{III}, when $\zeta = 1$). Note that, the proofs for different values of ζ are independent.

To prove the security of our proposed scheme, and by contradiction, assuming an adversary $A_{(1-\zeta)II+\zeta III}$, we show that there is a solver (algorithm B) that can solve a random instance of the RSA problem with a non-negligible probability. To do this, we first show that there exists a simulator called $C_{A_{(1-\zeta)II+\zeta III}}$ that can simulate the signature scheme without knowing the secret key(s) of the honest signer(s), and runs the adversary $A_{(1-\zeta)II+\zeta III}$ as its sub-routine. In this regard, we compute the run-time and a lower-bound for the success (returning a *useful output* $(\{R_u\}_{0\leq u\leq z}, s, \{c_u\}_{0\leq u\leq z}, \{x_u\}_{0\leq u\leq z}, m, w)$ (see Definition 5)) probability of this simulator in terms of the run-time and success (returning a valid forgery $\theta = (R_0, ..., R_z, s, c_0, ..., c_z)$ on a message m under the message space descriptor w w.r.t. original signer's identity ID_0 and a subset $\widetilde{ID} \subseteq \mathbf{ID}$ of proxy signers) probability of the adversary and the number of queries to the oracles (see Lemma 1).

At the final stage, we use a forking strategy to solve an instance (N, e, y) of the RSA problem, using a *useful pair* (see Definition 6) of the simulator $C_{A_{(1-\zeta)II+\zeta III}}$ when the random string used in both simulations are the same. Hence, we concentrate on computing a lower bound for the probability of producing such a *useful pair* and solving the RSA instance as the main body of the solver algorithm B (see Lemma 3). We should highlight that the general Forking Lemma [41] cannot be applied directly into our scheme since this scheme is the result of sequential aggregation of two different types of signatures such that we have two different types of random oracle responses. Hence, we need to consider the probability of happening some random responses before the forking point in the proposed forking lemma, and this is the main difference of our Forking Lemma from previous ones.

Our main result on the security of the proposed scheme is summarized in Theorem 1, where the parameter ζ is used to code the result for both adversaries of types II and III.

To start let us verify the correctness of the proposed scheme. Note that, all computations are done modulo N, but we omit this for simplicity.

$$\begin{aligned} s^e &= s_0^e (x_j^{c_j} r \textstyle\prod_{u\neq j} r_u)^e \\ &= s_0^e (x_j^e)^{c_j} r^e (\textstyle\prod_{u\neq j} r_u^e) \\ &= R_0 H(ID_0)^{c_0} H(ID_j)^{c_j} R_j \textstyle\prod_{u\neq j} (H(ID_u)^{c_u}) (\textstyle\prod_{u\neq j} R_u) \\ &= R_0 H(ID_0)^{c_0} [\textstyle\prod_{u=1}^{z} R_u H(ID_u)^{c_u}]. \end{aligned} \tag{2}$$

The equality $r^e = R_j[\prod_{u\neq j} H(ID_u)^{c_u}] \bmod N$ is used in Eq. (2).

Definition 5. Let $\zeta \in \{0,1\}$ be a constant and the algorithm $C_{A_{(1-\zeta)II+\zeta III}}$ return $(\{R_u\}_{0\leq u\leq z}, s, \{c_u\}_{0\leq u\leq z}, \{x_u\}_{0\leq u\leq z}, m, w)$ derived from a valid forgery $((ID_0, \widetilde{ID} \subseteq \mathbf{ID}, m, w), \theta)$ produced by an adversary $A_{(1-\zeta)II+\zeta III}$ when $C_{A_{(1-\zeta)II+\zeta III}}$ simulates the signature scheme. The tuple $(\{R_u\}_{0\leq u\leq z}, s, \{c_u\}_{0\leq u\leq z}, \{x_u\}_{0\leq u\leq z}, m, w)$ is a *useful output* if $s^e = R_0(x_0^e y^{1-\zeta})^{c_0}[\prod_{u=1}^{z} R_u (x_u^e y^{\zeta})^{c_u}]$ holds.

Lemma 1. *Let $\zeta \in \{0,1\}$ and $z \geq 2$ be a constant and l_N be a security parameter. Assuming the existence of an $(t, q_h, q_k, q_e, (1-\zeta)q_d, \zeta q_{prs}, \epsilon)$-bounded adversary $A_{(1-\zeta)II+\zeta III}$ with success probability at least ϵ and run-time t, there exists a simulator $C_{A_{(1-\zeta)II+\zeta III}}$ for the signature scheme that does not use the secret key(s) of the honest signer(s), and produces a useful output $(\{R_u\}_{0\leq u\leq z}, s, \{c_u\}_{0\leq u\leq z}, \{x_u\}_{0\leq u\leq z}, m, w)$ such that,*

a) *the success probability of $C_{A_{(1-\zeta)II+\zeta III}}$ is greater than*

$$\varepsilon \stackrel{\text{def}}{=} \zeta(\frac{\epsilon}{2^{2z} q_e^z} - (2q_{prs}^2 + q_{prs}q_k)2^{-l_N}) + (1-\zeta)(\frac{\epsilon}{4q_e}$$

$$-(2q_d^2 + q_d q_k)2^{-l_N},$$

b) *the run-time of $C_{A_{(1-\zeta)II+\zeta III}}$ is less than*

$$\tau \stackrel{\text{def}}{=} t + (1q_e + 1q_h + (1-\zeta)2q_d + \zeta 2z q_{prs})t_{exp},$$

where t_{exp} is the time of one exponentiation in $\mathbb{Z}_N^*$, and q_h, q_k, q_e, q_d and q_{prs} are the number of queries to the random oracle H, the random oracle K, KeyExtract, DelegationGen and ProxyRingSign oracle, respectively.

Proof. Assume the existence of an $(t, q_h, q_k, q_e, \zeta q_d, (1-\zeta)q_{prs}, \epsilon)$-bounded adversary $A_{(1-\zeta)II+\zeta III}$ on the public data $mpk = (N, e)$ which runs in time at most t, makes q_h queries to the random oracle H, q_k queries to the random oracle K, q_e queries to the KeyExtract, $(1-\zeta)q_d$ queries to the DelegationGen and ζq_{prs} queries to ProxyRingSign oracle, and can win the unforgeability game with probability at least ϵ. The algorithm $C_{A_{(1-\zeta)II+\zeta III}}$ maintains initially empty associative arrays $T_K[.]$ and $T[.]$, and answers $A_{(1-\zeta)II+\zeta III}$'s oracle queries as described below.

- $K(Q)$ queries: If $T_K[Q]$ is defined then $C_{A_{(1-\zeta)II+\zeta III}}$ returns its value, otherwise $C_{A_{(1-\zeta)II+\zeta III}}$ chooses $T_K[Q] \xleftarrow{\$} \{0,1\}^{l_1}$, and returns $T_K[Q]$ to $A_{(1-\zeta)II+\zeta III}$. Note that, in DelegationGen $Q = (R_0||w||\mathbf{ID}||11)$ and in ProxyRingSign $Q = (R_u||\mathbf{ID}||\widetilde{ID}||R_0||w||m||11)$.

- $H(ID_u)$ queries: We employ Coron's technique [46] to obtain a tighter security bound when simulating H. If $T[ID_u] = (b, x_u, X_u)$ then $C_{A_{(1-\zeta)II+\zeta III}}$ returns X_u. If this entry is not yet defined, it chooses $x_u \xleftarrow{\$} \mathbb{Z}_N^*$ and tosses a biased coin b so that $b = 0$ with probability β and $b = 1$ with probability $1-\beta$. If $b = 0$, then $C_{A_{(1-\zeta)II+\zeta III}}$ sets $X_u = x_u^e \bmod N$; if $b = 1$, it sets $X_u = x_u^e y \bmod N$. It stores $T[ID_u] \leftarrow (b, x_u, X_u)$ and returns X_u to $A_{(1-\zeta)II+\zeta III}$.

- KeyExtract queries for ID_u: Algorithm $C_{A_{(1-\zeta)II+\zeta III}}$ looks up $T[ID_u] = (b, x_u, X_u)$, if this entry is not yet defined, it performs a query $H(ID_u)$. If $b = 0$, then $C_{A_{(1-\zeta)II+\zeta III}}$ returns x_u; otherwise, it sets $bad_{KE} \leftarrow true$ and aborts the execution of $A_{(1-\zeta)II+\zeta III}$.

- DelegationGen queries for $(w, \mathbf{ID})$ under identity ID_0: Algorithm $C_{A_{II}}$ performs a query $H(ID_0)$ and looks up $T[ID_0] = (b, x_0, X_0)$. If $b = 0$, then $C_{A_{II}}$ simulates the delegation of ID_0 with the DelegationGen algorithm $\sigma_0 \leftarrow DelegationGen(Para, mpk, x_0, w, \mathbf{ID})$ since $C_{A_{II}}$ knows x_0, the original signer's secret key. If $b = 1$, $C_{A_{II}}$ first chooses $c_0 \xleftarrow{\$} \{0,1\}^{l_1}$ and $s_0 \xleftarrow{\$} \mathbb{Z}_N^*$ and computes $R_0 \leftarrow s_0^e X_0^{-c_0} \bmod N$. If $T_K[R_0||w||\mathbf{ID}||11]$ has already been defined, then $C_{A_{II}}$ sets $bad_{DG} \leftarrow true$ and halts; otherwise, it sets $T_K[R_0||w||\mathbf{ID}||11] \leftarrow c_0$, and returns $\sigma_0 = (R_0, s_0, c_0)$ to the adversary A_{II}.

- ProxyRingSign queries for a message m w.r.t. $\widetilde{ID}$: Adversary A_{III} provides a delegation σ_0 on a message space descriptor w and an identity set $\mathbf{ID}$. Algorithm $C_{A_{III}}$ first checks if the delegation for $(w, \mathbf{ID})$ is valid under identity ID_0, if $m \in w$ and if $\widetilde{ID} \subseteq \mathbf{ID}$. If so, $C_{A_{III}}$ proceeds as follows. If $b_u = 0$ for some $1 \le u \le z$, $C_{A_{III}}$ knows some x_u and can generate a valid proxy ring signature following ProxyRingSign algorithm. If for all $1 \le u \le z$, we have $b_u = 1$, $C_{A_{III}}$ for $1 \le u \ne j \le z$ chooses $r_u \xleftarrow{\$} \mathbb{Z}_N^*$, and computes $R_u = r_u^e \bmod N$ and $c_u = K(R_u||\mathbf{ID}||\widetilde{ID}||R_0||w||m||11)$ by querying to oracle K. Next, $C_{A_{III}}$ selects $c_j \xleftarrow{\$} \{0,1\}^{l_1}$ and $\hat{s} \xleftarrow{\$} \mathbb{Z}_N^*$, and computes $R_j \leftarrow \hat{s}^e X_j^{-c_j} \prod_{u \ne j}(R_u^{-1} X_u^{-c_u}) \bmod N$ and $s = s_0 \hat{s}$. If $T_K[R_j||\mathbf{ID}||\widetilde{ID}||R_0||w||m||11]$ has already been defined, then $C_{A_{III}}$ sets $bad_{PS} \leftarrow true$ and halts; otherwise, it sets $T_K[R_j||\mathbf{ID}||\widetilde{ID}||R_0||w||m||11] \leftarrow c_j$ and returns the proxy ring signature $\theta = (R_0, ..., R_z, s, c_0, ..., c_z)$ on the message m and the message space descriptor w w.r.t. original signer's identity ID_0 and two identity sets $\mathbf{ID}$ and $\widetilde{ID}$ for proxy signers.

To lower-bound the probability that $C_{A_{(1-\zeta)II+\zeta III}}$ does not abort at answering to queries of $A_{(1-\zeta)II+\zeta III}$, we need to compute $\eta = (1-\zeta)\Pr[\neg bad_{KE}]\Pr[\neg bad_{DG}|\neg bad_{KE}] + \zeta \Pr[\neg bad_{KE}]\Pr[\neg bad_{PS}|\neg bad_{KE}]$, where events bad_{KE}, bad_{DG} and bad_{PS} indicate that $C_{A_{(1-\zeta)II+\zeta III}}$ aborts in signature simulation as a result of any of $A_{(1-\zeta)II+\zeta III}$'s KeyExtract, DelegationGen and ProxyRingSign queries, respectively. These probabilities are computed as follows.

Claim 1. $\Pr[\neg bad_{KE}] \ge \beta^{q_E}$.

Proof. $\Pr[\neg bad_{KE}]$ is the probability that $C_{A_{(1-\zeta)II+\zeta III}}$ does not abort as a result of $A_{(1-\zeta)II+\zeta III}$'s KeyExtract queries. The algorithm $C_{A_{(1-\zeta)II+\zeta III}}$ aborts at answering to a KeyExtract query when bad_{KE} is set to true which means that $b = 1$ for a given identity. The probability of this event is $1-\beta$, so the probability that $C_{A_{(1-\zeta)II+\zeta III}}$ does not abort for one KeyExtract query is β. Since $A_{(1-\zeta)II+\zeta III}$ makes

at most q_e KeyExtract queries, the probability that $C_{A_{(1-\zeta)II+\zeta III}}$ does not abort as a result of q_e KeyExtract queries is at least β^{q_e}.

Claim 2. $\Pr[\neg bad_{DG}|\neg bad_{KE}] \geq 1 - q_d(q_d+q_k)2^{-l_N} - q_d^2 2^{-l_N}$.

Proof. Events $\neg bad_{KE}$ and $\neg bad_{DG}$ are independent, so $\Pr[\neg bad_{DG}|\neg bad_{KE}] = \Pr[\neg bad_{DG}]$. The value of $\Pr[\neg bad_{DG}]$ is the probability that $C_{A_{II}}$ does not abort as a result of DelegationGen queries. The algorithm $C_{A_{II}}$ aborts at answering to a DelegationGen query if bad_{DG} is set to true which means that there is a conflict in the table $T_K[.]$ for these kinds of queries. The probability of finding a conflict in $T_K[.]$ for one DelegationGen query $(w, \mathbf{ID})$ equals the probability that $(R_0||w||\mathbf{ID}||11)$ generated in a DelegationGen simulation has been occurred by chance in a previous query to the oracle K. Since there are at most $q_k + q_d$ entries in the table $T_K[.]$ for these kinds of queries and the number of R_0, uniformly distributed in $\mathbb{Z}_N$, is 2^{l_N}, the probability of this event for one DelegationGen query is at most $(q_d+q_k)2^{-l_N}$. Hence, the probability of this event for q_d queries is at most $q_d(q_d+q_k)2^{-l_N}$. In addition, this probability includes the probability that $C_{A_{II}}$ previously used the same randomness R_0, uniformly distributed in $\mathbb{Z}_N$, in one DelegationGen simulation. Since there are at most q_d DelegationGen simulations, this probability is at most $q_d 2^{-l_N}$. Therefore, for q_d DelegationGen queries the probability of this event is at most $q_d^2 2^{-l_N}$.

Claim 3. $\Pr[\neg bad_{PS}|\neg bad_{KE}] \geq 1 - q_{prs}(q_{prs}+q_k)2^{-l_N} - q_{prs}^2 2^{-l_N}$.

Proof. Events $\neg bad_{KE}$ and $\neg bad_{PS}$ are independent, so $\Pr[\neg bad_{PS}|\neg bad_{KE}] = \Pr[\neg bad_{PS}]$. The value of $\Pr[\neg bad_{PS}]$ is the probability that $C_{A_{III}}$ does not abort as a result of ProxyRingSign queries. The algorithm $C_{A_{III}}$ aborts at answering to a ProxyRingSign query if bad_{PS} is set to true which means that there is a conflict in table $T_K[.]$ for these kinds of queries. The probability of finding a conflict in $T_K[.]$ for one ProxyRingSign query equals the probability that $(R_j||\mathbf{ID}||\widetilde{ID}||R_0||w||m||11)$ generated in ProxyRingSign simulation has been occurred by chance in a previous query to the oracle K. Since there are at most q_k+q_{prs} entries in the table $T_K[.]$ for these kinds of queries and the number of R_j, uniformly distributed in $\mathbb{Z}_N$, is 2^{l_N}, the probability of this event for one ProxyRingSign is at most $(q_{prs}+q_k)2^{-l_N}$. Hence, the probability of this event for q_{prs} queries is at most $q_{prs}(q_{prs}+q_k)2^{-l_N}$. In addition, this probability includes the probability that $C_{A_{III}}$ previously used the same randomness R_j, uniformly distributed in $\mathbb{Z}_N$, in one ProxyRingSign simulation. Since there are at most q_{prs} ProxyRingSign simulations, this probability is at most $q_{prs}2^{-l_N}$. Therefore, for q_{prs} ProxyRingSign queries the probability of this event is at most $q_{prs}^2 2^{-l_N}$.

Finally, it is assumed that $A_{(1-\zeta)II+\zeta III}$ outputs a valid forgery $\theta = (R_0, ..., R_z, s, c_0, ..., c_z)$ on a message m under message space descriptor w w.r.t. original signer's identity ID_0 and proxy signers' identity sets $\mathbf{ID}$ and $\widetilde{ID}$ with probability at least ϵ in time bound t. Since the forgery is valid, we have

$$s^e = R_0 H(ID_0)^{c_0}[\prod_{u=1}^{z} R_u H(ID_u)^{c_u}],$$

and A_{II} has not asked $(w, \mathbf{ID})$ from DelegationGen algorithm under original signer's identity ID_0 and A_{III} has not asked the message m from ProxyRingSign algorithm under proxy signer's identity set $\widetilde{ID} \subseteq \mathbf{ID}$. In addition, a valid forgery has to contain one uncorrupted identity or z uncorrupted identities for A_{II} and A_{III}, respectively. These probabilities are computed as follows.

Claim 4. The probability that A_{II} outputs a valid forgery including one uncorrupted identity is at least $\epsilon(1-\beta)$.

Proof. It is assumed that A_{II} outputs a valid forgery with probability at least ϵ. The probability that a valid forgery contains one uncorrupted identity is $1-\beta$. The probability of existence of one honest identity with $b=1$ is $1-\beta$. Therefore, the probability that A_{II} outputs a valid forgery containing one uncorrupted identity is at least $\epsilon(1-\beta)$.

Claim 5. The probability that A_{III} outputs a valid forgery including z uncorrupted identities is at least $\epsilon(1-\beta)^z$.

Proof. The proof is similar to the proof of Claim 4.

Therefore, the probability that $C_{A_{(1-\zeta)II+\zeta III}}$ returns a useful output is at least $\epsilon((1-\zeta)(1-\beta)+\zeta(1-\beta)^z)\eta \geq \epsilon((1-\zeta)(1-\beta)+\zeta(1-\beta)^z)\beta^{q_e} - ((1-\zeta)q_d((2q_d+q_k)2^{-l_N}) - \zeta q_{prs}(2q_{prs}+q_k)2^{-l_N})$. The value of $\beta^{q_e}(1-\beta)$ is maximized for $\beta = \frac{q_e}{q_e+1}$. With substituting the value of β, we obtain $\beta^{q_e}(1-\beta) = (\frac{q_e}{q_e+1})^{q_e}\frac{1}{q_e+1} = \frac{1}{q_e}(1-\frac{1}{q_e+1})^{1+q_e}$. If $q_e = 0$, this value is 1 and $(1-\frac{1}{q_e+1})^{1+q_e}$ is a monotonically increasing sequence for $q_e \geq 1$. Therefore, the lower bound of $\beta^{q_e}(1-\beta)$ is $\frac{1}{4q_e}$. Similarly, the value of $\beta^{q_e}(1-\beta)^z \geq \frac{1}{2^{2z}q_e^z}$.

To estimate the required time of $C_{A_{(1-\zeta)II+\zeta III}}$ in returning a useful output, the required time t_C in which $C_{A_{(1-\zeta)II+\zeta III}}$ answers $A_{(1-\zeta)II+\zeta III}$'s queries is computed as follows. Since it is assumed that a (multi-) exponentiation in $\mathbb{Z}_N$ takes time t_{exp} while all other operations take zero time, each random oracle or KeyExtract query takes at most one exponentiation, a delegation simulation takes 2 exponentiations, and a proxy ring signature simulation takes $2z$ exponentiations, we therefore have that $t_C \leq (1q_e + 1q_h + (1-\zeta)2q_d + \zeta 2zq_{prs})t_{exp}$.

Finally, $C_{A_{(1-\zeta)II+\zeta III}}$ performs additional random oracle queries $H(ID_u)$ for identities in the forgery to find $T[ID_u] = (b, x_u, X_u)$ for them, and returns $(\{R_u\}_{0\leq u\leq z}, s, \{c_u\}_{0\leq u\leq z}, \{x_u\}_{0\leq u\leq z}, m, w)$ with probability at least $\zeta(\frac{\epsilon}{2^{2z}q_e^z} - (2q_{prs}^2 + q_{prs}q_k)2^{-l_N}) + (1-\zeta)(\frac{\epsilon}{4q_e} - (2q_d^2 + q_dq_k)2^{-l_N}$ in time bound $t + (1q_e + 1q_h + (1-\zeta)2q_d + \zeta 2zq_{prs})t_{exp}$.

Also, in what follows we will be needing the following Splitting lemma.

Lemma 2. [47]. *Let $A \subset X \times Y$ such that $\Pr[(x,y) \in A] \geq \delta$. For any $\alpha < \delta$, define $B = \{(x,y) \in X \times Y | \Pr_{y' \in Y}[(x,y') \in A] \geq \delta - \alpha\}$ and $\bar{B} = (X \times Y) \setminus B$, then the following statements hold:*

- $\Pr[B] \geq \alpha$
- $\forall (x,y) \in B, \Pr_{y' \in Y}[(x,y') \in A] \geq \delta - \alpha$
- $\Pr[B|A] \geq \frac{\alpha}{\delta}$.

Definition 6. Let $\zeta \in \{0,1\}$ be a constant. A pair of useful outputs $(\{R_u\}_{0\leq u\leq z}, s, \{c_u\}_{0\leq u\leq z}, \{x_u\}_{0\leq u\leq z}, m, w)$ and $(\{R'_u\}_{0\leq u\leq z}, s', \{c'_u\}_{0\leq u\leq z}, \{x'_u\}_{0\leq u\leq z}, m', w')$ is said to be a *useful pair* if $R_u = R'_u$, $0 \leq u \leq z$, $s \neq s'$, $c_u \neq c'_u$ for one $0 \leq u \leq z$ and $c_u = c'_u$ other u, $x_u = x'_u$, $0 \leq u \leq z$, $m = m'$ and $w = w'$ hold.

Definition 7. The probabilistic polynomial time algorithm $C_{A_{(1-\zeta)II+\zeta III}}$ at each run proceeds based on a random string ω and answers $\rho \stackrel{\text{def}}{=} (\rho_1, ..., \rho_{q_t})$ to the queries $\mathcal{Q} \stackrel{\text{def}}{=} (Q_1, ..., Q_{q_t})$ made to the random oracle K. A pair of (ω, ρ) is said to be a *successful pair* if $C_{A_{(1-\zeta)II+\zeta III}}$ produces a useful output $(\{R_u\}_{0\leq u\leq z}, s, \{c_u\}_{0\leq u\leq z}, \{x_u\}_{0\leq u\leq z}, m, w)$ based on them.

Lemma 3. *(*A Forking Lemma). Let $\zeta \in \{0,1\}$ be a constant, z and l_1 be the number of proxy signers in a proxy ring and a security parameter, respectively. Also let K be a random oracle, and q_t be the total number of queries to K. It is assumed that $C_{A_{(1-\zeta)II+\zeta III}}$ returns a useful output $(\{R_u\}_{0\leq u\leq z}, s, \{c_u\}_{0\leq u\leq z}, \{x_u\}_{0\leq u\leq z}, m, w)$ with probability at least ε in time bound τ. Then, a replay of $C_{A_{(1-\zeta)II+\zeta III}}$ with the same random string and a different random oracle gives a useful pair in time $t' \leq 2\tau$ with probability $\epsilon' \geq \frac{\epsilon_1^2(1-2^{-l_1})}{8\pi}$, where $\epsilon_1 \geq \varepsilon - (z+1)2^{-l_1}$ and $\pi = \sum_{j=1}^{q_t-z-1}[\prod_{i=0}^{z-1}(q_t - i - j)]$.

Proof. Consider the probabilistic polynomial time Turing machine $C_{A_{(1-\zeta)II+\zeta III}}$ with a random string ω, that answers to the queries $\mathcal{Q} \stackrel{\text{def}}{=} (Q_1, ..., Q_{q_t})$ made to the random oracle K, and stores these queries and the corresponding answers $\rho \stackrel{\text{def}}{=} (\rho_1, ..., \rho_{q_t})$ in the table $T_K[.]$. Let $i_u \in \{1, ..., q_t\}$ be the index of c_u for $0 \leq u \leq z$, such that $\rho_{i_u} = c_u$. For a given value c_u, if there is no $\rho_{i_u} = c_u$ in $T_K[.]$, we define $i_u \stackrel{\text{def}}{=} \infty$. By hypothesis, for a random choice of (ω, ρ), $C_{A_{(1-\zeta)II+\zeta III}}$ produces a useful output $(\{R_u\}_{0\leq u\leq z}, s, \{c_u\}_{0\leq u\leq z}, \{x_u\}_{0\leq u\leq z}, m, w)$ with probability at least ε in time bound τ.

Since K is a random oracle, the probability of the event

$$c_0 = K(R_0||w||\mathbf{ID}||11)$$

and

$$c_u = K(R_u||\mathbf{ID}||\widetilde{ID}||R_0||w||m||11)$$

for $1 \leq u \leq z$ is less than $(z+1)2^{-l_1}$, unless they are asked during the attack. Hence, it is likely that the questions $(R_u||\mathbf{ID}||\widetilde{ID}||R_0||w||m||11)$ for $1 \leq u \leq z$ and $(R_0||w||\mathbf{ID}||11)$ are asked during a successful attack.

We define set Υ as the set of successful pairs (ω, ρ), $\Upsilon = \{(\omega,\rho)|\ C_{A_{(1-\zeta)II+\zeta III}(\omega)}$ produces a useful output & $\infty \notin (i_0, ..., i_z)\}$. The lower bound of probability of producing a useful output is $\epsilon_1 = \Pr[\Upsilon] \geq \varepsilon - (z+1)2^{-l_1}$, and let $\kappa = \max\{i \in (i_0, ..., i_z)\}$.

We also define $I_{z+1} = \{(i_0, ..., i_z) \mid 1 \leq i_u \leq q_t$ & $\forall u \neq v\ i_u \neq i_v$ & $i_0 < \max\{i_u$ for $1 \leq u \leq z\}\}$ when $\zeta = 1$, and $I_{z+1} = \{(i_0, ..., i_z) \mid 1 \leq i_u \leq q_t$ & $\forall u \neq v\ i_u \neq i_v$ & $i_0 > \max\{i_u$ for $1 \leq u \leq z\}\}$ when $\zeta = 0$.

Since $C_{A_{(1-\zeta)II+\zeta III}}$ makes query to the random oracle K for a successful pair $(\omega, \rho) \in \Upsilon$, then we define set $\Upsilon_{\boldsymbol{i}} = \{(\omega,\rho)|\ C_{A_{(1-\zeta)II+\zeta III}(\omega)}$ produces a useful output & $(i_0, ..., i_z) = \boldsymbol{i}\}$ for all vectors $\boldsymbol{i} \in I_{z+1}$ as a subset of Υ in which query $Q_{i_0} = (R_0||w||\mathbf{ID}||11)$ was made to random oracle K before one of queries $Q_{i_u} = (R_u||\mathbf{ID}||\widetilde{ID}||R_0||w||m||11)$ for $1 \leq u \leq z$ when $\zeta = 1$, and when $\zeta = 0$, queries $Q_{i_u} = (R_u||\mathbf{ID}||\widetilde{ID}||R_0||w||m||11)$ for $1 \leq u \leq z$ were made to random oracle K before query $Q_{i_0} = (R_0||w||\mathbf{ID}||11)$.

The cardinality of set I_{z+1} in both cases is $\pi = \sum_{j=1}^{q_t - z - 1}[\prod_{i=0}^{z-1}(q_t - i - j)]$. This gives us a partition of Υ in exactly π classes. Let I be the set consisting of most likely vectors $\boldsymbol{i} \in I_{z+1}$, $I = \{\boldsymbol{i} \in I_{z+1}|\ \Pr[\Upsilon'_{\boldsymbol{i}}|\Upsilon] \geq \frac{1}{2}\frac{1}{\pi}\}$. Hence, for each $\boldsymbol{i} \in I$, $\Upsilon_{\boldsymbol{i}}$ is denoted as $\Upsilon'_{\boldsymbol{i}}$, we have $\Pr[\Upsilon'_{\boldsymbol{i}}] = \Pr[\Upsilon'_{\boldsymbol{i}}|\Upsilon]\Pr[\Upsilon] \geq \frac{\epsilon_1}{2\pi}$.

With splitting the randomness ρ related to the oracle K as $\rho' = (\rho_1, ..., \rho_{\kappa-1})$, where ρ' denotes a vector of answers to all queries to the random oracle K before index κ. We employ Splitting Lemma, taking $X = (\omega, \rho')$, $Y = (\rho_\kappa, ..., \rho_{q_t})$, $A = \Upsilon'_{\boldsymbol{i}}$, $\delta = \frac{\epsilon_1}{2\pi}$ and $\alpha = \frac{\epsilon_1}{4\pi}$. This lemma ensures the existence of a subset $\Omega_{\boldsymbol{i}}$ of executions (ω, ρ) such that $\Pr[\Omega_{\boldsymbol{i}}|\Upsilon'_{\boldsymbol{i}}] \geq \frac{\alpha}{\delta} = \frac{1}{2}$ and for each $(\omega, \rho) \in \Omega_{\boldsymbol{i}}$, $\Pr_{(\rho'_\kappa, ..., \rho'_{q_t})}[(\omega, (\rho', \rho'_\kappa, ..., \rho'_{q_t})) \in \Upsilon'_{\boldsymbol{i}}] \geq \delta - \alpha = \frac{\epsilon_1}{2\pi}$.

Since $\Upsilon'_{\boldsymbol{i}}$ are disjoint, and we have $\Pr_{(\omega,\rho)}[\exists \boldsymbol{i} \in I$ s.t. $\Omega_{\boldsymbol{i}} \cap \Upsilon'_{\boldsymbol{i}}|\Upsilon] = \sum_{\boldsymbol{i}\in I}\Pr[\Omega_{\boldsymbol{i}} \cap \Upsilon'_{\boldsymbol{i}}|\Upsilon] = \sum_{\boldsymbol{i}\in I}\Pr[\Omega_{\boldsymbol{i}}|\Upsilon'_{\boldsymbol{i}}]$ $\Pr[\Upsilon'_{\boldsymbol{i}}|\Upsilon] \geq \frac{\sum_{\boldsymbol{i}\in I}\Pr[\Upsilon'_{\boldsymbol{i}}|\Upsilon]}{2} \geq \frac{1}{4}$.

Let $\boldsymbol{i}$ denote a vector of a successful pair with probability at least $\frac{1}{4}$, $\boldsymbol{i} \in I$ and $(\omega, \rho) \in \Omega_{\boldsymbol{i}} \cap \Upsilon'_{\boldsymbol{i}}$. If we replay the attack with fixed (ω, ρ') and randomly chosen $(\rho'_\kappa, ..., \rho'_{q_t})$, we get another successful pair $(\omega, (\rho', \rho'_\kappa, ..., \rho'_{q_t}))$ such that $\rho_\kappa \neq \rho'_\kappa$ with probability $\frac{\epsilon_1(1-2^{-l_1})}{2\pi}$.

After two successful executions of $C_{A_{(1-\zeta)II+\zeta III}}$, the algorithm B obtains $(\omega, (\rho', \rho_\kappa, ..., \rho_{q_t}))$ and $(\omega, (\rho', \rho'_\kappa,$ $..., \rho'_{q_t}))$, $\rho_{i_u} \neq \rho'_{i_u}$ for $\kappa \leq i_u \leq q_t$ which means that B obtains a useful pair $(\{R_u\}_{0\leq u\leq z}, s, \{c_u\}_{0\leq u\leq z}, \{x_u\}_{0\leq u\leq z}, m, w)$ and $(\{R'_u\}_{0\leq u\leq z}, s', \{c'_u\}_{0\leq u\leq z}, \{x'_u\}_{0\leq u\leq z}, m', w')$ with probability $\epsilon' \geq \frac{\epsilon_1^2(1-2^{-l_1})}{8\pi}$, where $\epsilon_1 \geq \varepsilon - (z+1)2^{-l_1}$, $\pi = \sum_{j=1}^{q_t-z-1}[\prod_{i=0}^{z-1}(q_t - i - j)]$.

The running time t' of B is twice that of $C_{A_{(1-\zeta)II+\zeta III}}$, $t' \geq 2\tau$.

Theorem 1. *If the RSA function associated to Kg_{rsa} is (t', ϵ')-one-way, then the proposed signature scheme is $(t, q_h, q_k, q_e, (1-\zeta)q_d, \zeta q_{prs}, \epsilon)$-secure against $(t, q_h, q_k, q_e, (1-\zeta)q_d, \zeta q_{prs}, \epsilon)$-bounded adversary $A_{(1-\zeta)II+\zeta III}$ for a constant $\zeta \in \{0, 1\}$ such that*

$$\begin{aligned} \epsilon' &\geq \frac{\epsilon_1^2(1-2^{-l_1})}{8\sum_{j=1}^{q_k+(1-\zeta)q_d+\zeta q_{prs}-z-1}[\prod_{i=0}^{z-1}(q_k+(1-\zeta)q_d+\zeta q_{prs}-i-j)]} \\ t' &\leq 2t + 2(1q_e + 1q_h + (1-\zeta)2q_d + \zeta 2z q_{prs})t_{exp}, \end{aligned} \tag{3}$$

where $\epsilon_1 \geq \zeta(\frac{\epsilon}{2^{2z}q_e^z} - (2q_{prs}^2 + q_{prs}q_k)2^{-l_N}) + (1-\zeta)(\frac{\epsilon}{4q_e} - (2q_d^2 + q_d q_k)2^{-l_N} - (z+1)2^{-l_1}$, t_{exp} is the time of an exponentiation in $\mathbb{Z}_N^*$, and z and l_1 are the number of proxy signers in a proxy ring and a security parameter, respectively.

Proof. In the proof, we consider two cases for the forgery depending on type of adversaries. In the first case, we type II adversary, while in the second one type III adversary. Then, we show that the algorithm B can solve a random instance of the RSA problem (N, e, y) such that $\gamma = y^{\frac{1}{e}} \bmod N$.

Case 1. In this case, we consider adversaries of type II (i.e., $\zeta = 0$). According to Lemma 1, $C_{A_{II}}$ returns a useful output $(\{R_u\}_{0\leq u\leq z}, s, \{c_u\}_{0\leq u\leq z}, \{x_u\}_{0\leq u\leq z}, m, w)$ in time $\tau = t + (1q_e + 1q_h + 2q_d)t_{exp}$ with probability at least $\varepsilon = \frac{\epsilon}{4q_e} - (2q_d^2 + q_d q_k)2^{-l_N}$. Then, the algorithm B, the RSA solver, will produce a useful pair of $(\{R_u\}_{0\leq u\leq z}, s, \{c_u\}_{0\leq u\leq z}, \{x_u\}_{0\leq u\leq z}, m, w)$ and $(\{R_u\}_{0\leq u\leq z}, s', c'_0, \{c_u\}_{1\leq u\leq z}, \{x_u\}_{0\leq u\leq z}, m, w)$

with probability at least $\epsilon' \geq \frac{(\frac{\epsilon}{4q_e} - (2q_d^2 + q_d q_k)2^{-l_N} - (z+1)2^{-l_1})^2(1-2^{-l_1})}{8\sum_{j=1}^{q_k+q_d-z-1}[\prod_{i=0}^{z-1}(q_k+q_d-i-j)]}$ in time $t' \geq 2t + 2(1q_e + 1q_h + 2q_d)t_{exp}$ (see Lemma 3). Since a useful pair contains two useful outputs, we have

$$s^e = R_0(x_0^e y)^{c_0}[\prod_{u=1}^{z} R_u x_u^{ec_u}]$$

and

$$s'^e = R_0(x_0^e y)^{c_0'}[\prod_{u=1}^{z} R_u x_u^{ec_u}].$$

By dividing the two aforementioned equations, we obtain $(x_0^{(c_0'-c_0)}\frac{s}{s'})^e = y^{(c_0-c_0')} \bmod N$.
Since $c_0 \neq c_0' \in \{0,1\}^{l_1}$ and e is a prime of length strictly greater than l_1, we have $e > (c_0 - c_0')$ and therefore $gcd(e, (c_0 - c_0')) = 1$. Using the extended Euclidean algorithm, one can find $a, b \in \mathbb{Z}$ such that $ae + b(c_0 - c_0') = 1$. Hence, we have $y = y^{ae+b(c_0-c_0')} = (y^a(x_0^{(c_0'-c_0)}\frac{s}{s'})^b)^e \bmod N$. Therefore, algorithm B outputs $(y^a(x_0^{(c_0'-c_0)}\frac{s}{s'})^b)$ as the RSA inversion of y in time t' with probability ϵ'.

Case 2. In this case, we consider adversaries of type III (i.e., $\zeta = 1$). According to Lemma 1, $C_{A_{III}}$ returns a useful output $(\{R_u\}_{0\leq u\leq z}, s, \{c_u\}_{0\leq u\leq z}, \{x_u\}_{0\leq u\leq z}, m, w)$ in time bound $\tau = t + (1q_e + 1q_h + 2zq_{prs})t_{exp}$ with probability at least $\varepsilon = \frac{\epsilon}{2^{2z}q_e^z} - (2q_{prs}^2 + q_{prs}q_k)2^{-l_N}$. Then, the algorithm B, the RSA solver, will produce a useful pair of $(\{R_u\}_{0\leq u\leq z}, s, \{c_u\}_{0\leq u\leq z}, \{x_u\}_{0\leq u\leq z}, m, w)$ and $(\{R_u\}_{0\leq u\leq z}, s', c_\kappa', \{c_u\}_{0\leq u\neq\kappa\leq z}, \{x_u\}_{0\leq u\leq z}, m, w)$ with probability at least $\epsilon' \geq (\frac{\frac{\epsilon}{2^{2z}q_e^z} - (2q_{prs}^2 + q_{prs}q_k)2^{-l_N} - (z+1)2^{-l_1})^2(1-2^{-l_1})}{8\sum_{j=1}^{q_k+q_{prs}-z-1}[\prod_{i=0}^{z-1}(q_k+q_{prs}-i-j)]}$ in time $t' \geq 2t + 2(1q_e + 1q_h + 2zq_{prs})t_{exp}$(see Lemma 3). Since a useful pair contains two useful outputs, we have

$$s^e = R_0 x_0^{ec_0}[\prod_{u=1}^{z} R_u(x_u^e y)^{c_u}]$$

and

$$s'^e = R_0 x_0^{ec_0}[(x_v^e y)^{c_v'} \prod_{1\leq u\neq v\leq z} R_u(x_u^e y)^{c_u}].$$

By dividing the two aforementioned equations, we obtain $(x_v^{c_v-c_v'}\frac{s}{s'})^e = y^{(c_v'-c_v)} \bmod N$.
Since $c_v \neq c_v' \in \{0,1\}^{l_1}$ and e is a prime of length strictly greater than l_1, we have $e > (c_v' - c_v)$ and therefore $gcd(e, (c_v' - c_v)) = 1$. Using the extended Euclidean algorithm, one can find $a, b \in \mathbb{Z}$ such that $ae + b(c_v' - c_v) = 1$. Hence, we have $y = y^{ae+b(c_v'-c_v)} = (y^a(x_v^{c_v-c_v'}\frac{s}{s'})^b)^e \bmod N$. Therefore, algorithm B can output $(y^a(x_v^{c_v-c_v'}\frac{s}{s'})^b)$ as the RSA inversion of y in time t' with probability ϵ'.

Theorem 2. *The identity-based proxy ring signature scheme is $(t, q_h, q_k, q_e, q_d, q_{prs}, \frac{1}{2})$-PPSI-secure since the probability of D in guessing the identity of the proxy signer for a given signature θ, $\Pr[D(\theta) = ID_j]$ (where $ID_j \in \widetilde{ID} = \{ID_1, ID_2\}$), is $\frac{1}{2}$ against $(t, q_h, q_k, q_e, q_d, q_{prs}, \epsilon)$-bounded adversary D.*

Proof. The distinguisher D issues a polynomially bounded number of random oracle, KeyExtract, DelegationGen and ProxyRingSign queries adaptively as explained in the forgery game.

Then, D chooses two honest identities ID_1 and ID_2 for proxy ring (D never make KeyExtract query for these two identities), and makes a DelegationGen and ProxyRingSign query on $(w, \mathbf{ID})$ under an identity ID_0 and on the message $m \in w$ under the identity set $\widetilde{ID} = \{ID_1, ID_2\} \subseteq \mathbf{ID}$, respectively. In response, C chooses $j \xleftarrow{\$} \{1,2\}$, runs KeyExtract for ID_0 and ID_j to obtain their corresponding secret keys, then runs DelegationGen on $(w, \mathbf{ID})$ under an identity ID_0 to obtain σ_0 and returns $\theta \leftarrow ProxyRingSign(Para, mpk, ID_0, \mathbf{ID}, \widetilde{ID}, (w, m, \sigma_0), x_j)$ to D. Finally, the distinguisher D outputs $j' = j$ with probability $\frac{1}{2}$. To show the value of this probability, we compute the probability that ID_j generates valid values for R_1 and R_2 of θ which are pairwise different. The probability of choosing different values for

Table 1. Comparison between our proposal and provably secure schemes

Scheme	DeleGen Cost	DeleVer Cost	PRSign Cost	PRVer Cost	Sign Size	Hard problem	ID -based
Ours	$2exp$ $+1m_N$	$2exp$ $+1m_N$	$(2n+1)exp$ $+(2n+2)m_N$	$(n+2)exp$ $+(n+2)m_N$	$(n+2)\mathbb{Z}_N^*$	RSA	✓
Li et al.[28]	$1e_G$	$2P$	$(2n-1)e_G$	$2nP+(2n-2)m_{G_T}+1e_G$	nG	Bilinear Pairing	×

R_1 and R_2 is $\frac{1}{2^{l_N}-1}\frac{1}{2^{l_N}-2}$. Then, s is computed from random numbers r_u for $u \neq j$ in R_u and r employed in R_j. The probability of generation of the proxy ring signature $\theta = (R_0, R_1, R_2, s)$ is independent from the identity of the real signer ID_j, then, this probability is the same for two members in the set of proxy signers. Therefore, the probability of D in guessing the real signer is $\frac{1}{2}$.

4 Comparison

The comparison for some provably secure (identity-based) proxy ring signature schemes is summarized in Table 1. The comparison is in terms of *DeleGen-Cost*, *DeleVer-Cost* , *PRSign-Cost* and *PRVer-Cost* , dominating computational cost in delegation generation, delegation verification, proxy ring signature generation and proxy ring signature verification, respectively. In Table 1, P, e_G, m_{G_2}, exp and m_N denote the pairing evaluation, exponentiation in group G, pairing multiplication in G_2, exponentiation in $\mathbb{Z}_N^*$ and multiplication in $\mathbb{Z}_N^*$, respectively. For the sake of comparison it is assumed that other operations take zero time and $z = n$ which means that $\widetilde{ID} = \mathbf{ID}$.

Since previous identity-based proxy ring signature schemes ([32–39]) do not support provable security, they are not considered in comparison. All *Traditional* proxy ring signature schemes [27–31] are based on bilinear pairings, where some schemes [29–31] do not support provable security, and the provably secure scheme [27] presented by Yu et al. is not secure as shown by J.S. Chou [31]. Hence, we just consider the only provably secure proxy ring signature scheme [28] proposed by Li et al. in comparison. As shown in Table 1, our scheme compared to Li et al.'s provably secure proxy ring signature scheme [28] has a proper advantage in efficiency since one pairing computation costs roughly 2.3 exponentiations according to the current MIRACL implementation [48], a 512-bit Tate pairing takes 20 ms whereas a 1024-bit prime modular exponentiation takes 8.8 ms at the same security level, and with considering the most costly operation, pairing computation, the cost of PRVer algorithm of Li et al.'s scheme is as 4.6 times as that of ours and the cost of DeleVer algorithm of Li et al.'s scheme is as 2.3 times as that of ours.

However, it is obvious that the size of our signature is increased due to the size of public parameters in RSA problem, this scheme is the first identity-based proxy ring signature scheme form RSA (widely used assumption) assumption. Furthermore, all (identity-based) proxy ring signatures are not resistant against proxy key exposure attack [40], while the proxy key exposure attack can not be applied to our scheme since it is sequential aggregation of delegation and proxy group's ring signature, and there is no proxy secret key stored in a less trusted device to be leaked.

5 Conclusion

In this paper, first, we formalized a security model for identity-based proxy ring signatures, then, we present the first provably secure identity-based proxy ring signature scheme. This scheme is the first identity-based proxy ring signature scheme from RSA, and consequently it is a response to the need of some companies for having identity-based proxy ring signature from RSA (sine they are reluctant to reinvesting in new implementation). Hence, it has a proper advantage in efficiency due to the avoiding pairing computations since the cost of each pairing computation is roughly that of 2.3 exponentiations. Furthermore, the proxy key exposure attack is not applicable to our scheme since it is generated based on sequential aggregation paradigm.

References

1. Shamir, A. (1985) Identity-based cryptosystems and signature schemes. *Proc. of the 4th Annual Int. Cryptology Conf. on Advances in Cryptology-CRYPTO* 1984, Santa Barbara, CA, USA, 19-22 August, pp. 47–53. Springer Berlin Heidelberg.
2. Choon, J. and Cheon, J. (2002) An identity-based signature from gap diffie-hellman groups. *Proc. of the 6th Int. Workshop on Practice and Theory in Public Key Cryptography, Public Key Cryptography PKC 2003*, Miami, FL, USA, 6-8 January, pp. 18–30. Springer Berlin Heidelberg.
3. Hess, F. (2002) Efficient identity based signature schemes based on pairings. *Proc. of the 9th Annual Int. Workshop on Selected Areas in Cryptography (SAC* 2002*)*, Newfoundland, Canada, 15-16 August, pp. 216–231. Springer Berlin Heidelberg.
4. Barreto, P., Libert, B., McCullagh, N., and Quisquaterd, J. (2005) Efficient and provably-secure identity-based signatures and signcryption from bilinear maps. *Proc. of the 11th Int. Conf. on the Theory and Application of Cryptology and Information Security, Aavances in Cryptology-ASIACRYPT* 2005, Chennai, India, 4-8 December, pp. 515–532. Springer Berlin Heidelberg.
5. Boyen, X. (2008) The uber-assumption family. *Proc. of the 2nd Int. Conf. on Pairing-Based Cryptography (Pairing 2008)*, Egham, UK, 1-3 September, pp. 39–56. Springer Berlin Heidelberg.
6. Mambo, M., Usuda, K., and Okamoto, E. (1996) Proxy signatures: Delegation of the power to sign messages. *IEICE Transactions on Fundamentals of Electronics,Communications and Computer Sciences*, **79**, 1338–1354.
7. Shao, Z. (2009) Provably secure proxy-protected signature schemes based on RSA. *Computers & Electrical Engineering*, **35**, 497–505.
8. Shao, Z. (2003) Proxy signature schemes based on factoring. *Information Processing Letters*, **85**, 137–143.
9. Zhou, Y., Cao, Z., and Lu, R. (2005) Provably secure proxy-protected signature schemes based on factoring. *Applied Mathematics and Computation*, **164**, 83–98.
10. Park, J. H., Kang, B. G., and Han, J. W. (2005) Cryptanalysis of Zhou et al.'s proxy-protected signature schemes. *Applied Mathematics and Computation*, **169**, 192–197.
11. Liu, Y.-C., Wen, H.-A., Lin, C.-L., and Hwang, T. (2007) Proxy-protected signature secure against the undelegated proxy signature attack. *Computers & Electrical Engineering*, **33**, 177–185.
12. Hu, X., Xu, H., and Si, T. (2010) Analysis and improvement of proxy-protected signature secure against the undelegated proxy signature attack. *Journal of Computational Information Systems*, **6**, 2997–3002.
13. Gu, C. and Zhu, Y. (2005) Provable security of ID-based proxy signature schemes. *Proc. of the 3rd Int. Conf. on Networking and Mobile Computing (ICCNMC 2005)*, Zhangjiajie, China, 2-4 August, pp. 1277–1286. Springer Berlin Heidelberg.
14. Zhang, J. and Zou, W. (2007) Another ID-based proxy signature scheme and its extension. *Wuhan University Journal of Natural Sciences*, **12**, 33–36.
15. Wu, W., Mu, Y., Susilo, W., Seberry, J., and Huang, X. (2007) Identity-based proxy signature from pairings. *Proc. of the 4th Int. Conf. on Autonomic and Trusted Computing (ATC* 2007*)*, Hong Kong, China, 11-13 July, pp. 22–31. Springer Berlin Heidelberg.
16. Gu, C. and Zhu, Y. (2008) An efficient ID-based proxy signature scheme from pairings. *Proc. of the 3rd SKLOIS Conf. on Information Security and Cryptology (Inscrypt 2007)*, Xining, China, 31 August- 5 September, pp. 40–50. Springer Berlin Heidelberg.
17. Ji, H., Wang, Y., Han, W., and Zhao, L. (2009) An identity-based proxy signature from bilinear pairings. *Proc. of WASE Int. Conf. on Information Engineering (ICIE* 2009*)*, Taiyuan, Shanxi, 10-11 July, pp. 14–17. IEEE.
18. Xu, J., Zhang, Z., and Feng, D. (2005) ID-based proxy signature using bilinear pairings. *Proc. of Parallel and Distributed Processing and Applications Workshops(ISPA* 2005*)*, Nanjing, China, 2-5 November, pp. 359–367. Springer Berlin Heidelberg.
19. Shim, K. (2006) An identity-based proxy signature scheme from pairings. *Proc. of the* 8*th Int. Conf. on Information and Communications Security (ICICS* 2006*)*, Raleigh, NC, USA, 4-7 December, pp. 60–71. Springer Berlin Heidelberg.
20. Lu, R. and Cao, Z. (2005) Designated verifier proxy signature scheme with message recovery. *Applied Mathematics and Computation*, **169**, 1237–1246.
21. Yu, Y., Xu, C., Zhang, X., and Liao, Y. (2009) Designated verifier proxy signature scheme without random oracles. *Computers & Mathematics with Applications*, **57**, 1352–1364.
22. Shim, K. (2011) Short designated verifier proxy signatures. *Computers & Electrical Engineering*, **37**, 180–186.
23. Huang, X., Mu, Y., Susilo, W., Zhang, F., and Chen, X. (2005) A short proxy signature scheme: efficient authentication in the ubiquitous world. *Proc. of Embedded and Ubiquitous Computing Workshops(EUC* 2005*)*, Nagasaki, Japan, 6-9 December, pp. 480–489. Springer Berlin Heidelberg.
24. Zhang, J., Liu, C., and Yang, Y. (2010) An efficient secure proxy verifiably encrypted signature scheme. *Journal of Network and Computer Applications*, **33**, 29–34.

25. Huang, X., Susilo, W., Mu, Y., and Wu, W. (2006) Proxy signature without random oracles. *Proc. of the 2nd Int. Conf. on Mobile Ad-hoc and Sensor Networks (MSN* 2006*)*, Hong Kong, China, 13-15 December, pp. 473–484. Springer Berlin Heidelberg.
26. Cao, F. and Cao, Z. (2009) A secure identity-based multi-proxy signature scheme. *Computers & Electrical Engineering*, **35**, 86–95.
27. Yu, Y., C. Xu, X. H., and Mu, Y. (2009) An efficient anonymous proxy signature scheme with provable security. *Computer Standards & Interfaces*, **31**, 348–353.
28. Li, J., Chen, X., and Yuen, T. H. (2006) Proxy ring signature: formal definitions, efficient construction and new variant. *Proc. of Int. Conf. on Computational Intelligence and Security (CIS 2006)*, Guangzhou, China, 3-6 November, pp. 545–555. Springer Berlin Heidelberg.
29. Zhang, F., Safavi-Naini, R., and Lin, C. (2003) New proxy signature, proxy blind signature and proxy ring signature schemes from bilinear pairings. *Cryptography ePrint Archive*, pp. 1–11.
30. Zhang, J. (2009) On the security of a proxy ring signature with revocable anonymity. *Proc. of Int. Conf. on Multimedia Information Networking and Security (MINES'09)*, Hubei, China, 18-20 November, pp. 205–209. IEEE.
31. Chou, J.-S. (2012) A novel anonymous proxy signature scheme. *Advances in Multimedia*, **2012**, 1–10.
32. Awasthi, A. K. and Lal, S. (2005) ID-based ring signature and proxy ring signature schemes from bilinear pairings. *IACR Cryptology ePrint Archive cs*/0504097 , **?**
33. Awasthi, A. K. and Lal, S. (2007) ID-based ring signature and proxy ring signature schemes from bilinear pairings. *International Journal of Network Security*, **4**, 187–192.
34. Zhao, Z., X. Tang, B. L., and Zhu, L. (2006) An ID-based anonymous proxy signature from bilinear pairings. *Proc. of the 2006 Int. Conf. on Security and Management (SAM 2006)*, Las Vegas, Nevada, USA, 26-29 June, pp. 138–144. CSREA Press.
35. Cheng, W., Lang, W., Yang, Z., Liu, G., and Tan, Y. (2004) An identity-based proxy ring signature scheme from bilinear pairings. *Proc. of the 9th Int. Symposium on Computers and Communications (ISCC* 2004*)*, Nanjing, China, 28 June-1 July, pp. 424–429. IEEE.
36. Lei, W. and Daxing, L. (2009) An efficient ID-based proxy ring signature scheme. *Proc. of WRI Int. Conf. on Communications and Mobile Computing (CMC'09)*, Yunnan, China, 6-8 January, pp. 560–563. IEEE.
37. Wu, L. and Kong, F. (2009) An efficient ID-based proxy ring signature scheme. *Journal of Shandong University (Natural Science)*, **1**, 2441–2447.
38. Cui, S. and Wen, F. (2010) An identity-based multiple grade anonymous proxy signature scheme. *Journal of Computational Information Systems*, **6**, 2441–2447.
39. Ajmath, K. A., Reddy, P. V., Rao, B. U., and Varma, S. V. K. (2012) Identity-based directed proxy ring signature scheme. *Journal of Discrete Mathematical Sciences and Cryptography*, **15**, 181–192.
40. Schuldt, J., Matsuura, K., and Paterson, K. (2008) Proxy signatures secure against proxy key exposure. *Proc. of the 11th Int. Workshop on Practice and Theory in Public-Key Cryptography (PKC 2008)*, Barcelona, Spain, 9-12 March, pp. 141–161. Springer Berlin Heidelberg.
41. Bellare, M. and Neven, G. (2006) Multi-signatures in the plain public-key model and a general forking lemma. *Proc. of the 13th ACM Conf. on Computer and Communications Security (CCS 2006)*, Alexandria, Virginia, USA, 4-8 December, pp. 390–399. ACM, New York, NY.
42. Guillou, L. and Quisquater, J. (1990) A paradoxical identity-based signature scheme resulting from zero-knowledge. *Proc. of the 8th Annual Int. Cryptology Conf. on Advances in Cryptology-CRYPTO* 1988, Santa Barbara, CA, USA, 21-25 August, pp. 216–231. Springer Berlin Heidelberg.
43. Herranz, J. (2007) Identity-based ring signatures from RSA. *Theoretical Computer Science*, **389**, 100–117.
44. Boldyreva, A., Palacio, A., and Warinschi, B. (2010) Secure proxy signature schemes for delegation of signing rights. *Journal of Cryptology*, **25**, 57–115.
45. Bellare, M. and Rogaway, P. (1993) Random oracles are practical: A paradigm for designing efficient protocols. *Proc. of the 1st ACM Conf. on Computer and Communications Security (CCS 1993)*, Fairfax, VA, USA, 3-5 November, pp. 62–73. ACM, New York, NY.
46. Coron, J. (2000) On the exact security of full domain hash. *Proc. of the 20th Annual Int. Cryptology Conf. on Advances in Cryptology-CRYPTO 2000*, Santa Barbara, CA, USA, 20-24 August, pp. 229–235. Springer Berlin Heidelberg.
47. Pointcheval, D. and Stern, J. (2000) Security arguments for digital signatures and blind signatures. *Journal of Cryptology*, **13**, 361–396.
48. Miracl. *http://www.shamus.ie/*.