1-1-2013

Realizing fully secure unrestricted ID-based ring signature in the standard model based on HIBE

Man Ho Au
University of Wollongong, aau@uow.edu.au

Joseph K. Liu
Institute for Infocomm Research, Singapore

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

Jianying Zhou
Institute for Infocomm Research Singapore, jyzhou@i2r.a-star.edu.sg

Realizing fully secure unrestricted ID-based ring signature in the standard model based on HIBE

Abstract

We describe a secure and unrestricted identity-based ring signature scheme in the standard model. Our construction is provably fully secure in the standard model under static assumptions and is motivated by an existing HIBE scheme. Our observation on the relationship between ID-based ring signatures and HIBE is of interest to researchers in other fields as well. Following the strongest anonymity definition in ring signatures, we define a new level of anonymity for ID-based ring signatures in which the attacker can specify the randomness used in the creation of the user secret key, in addition to the private key generators master key. Our scheme provides unconditional anonymity in this model.

Keywords

era2015

Disciplines

Engineering | Science and Technology Studies

Publication Details

Au, M., Liu, J. K., Susilo, W. & Zhou, J. (2013). Realizing fully secure unrestricted ID-based ring signature in the standard model based on HIBE. IEEE Transactions on Information Forensics and Security, 8 (12), 1909-1922.

Realizing Fully Secure Unrestricted ID-based Ring Signature in the Standard Model Based on HIBE

Man Ho Au, *Member, IEEE,* Joseph K. Liu*, Willy Susilo$, *Senior member, IEEE,*, Jianying Zhou

***Abstract*—We describe a secure and *unrestricted* identity-based ring signature scheme in the standard model. Our construction is provably fully secure in the standard model under static assumptions and is motivated by an existing HIBE scheme. Our observation on the relationship between ID-based ring signatures and HIBE is of interest to researchers in other fields as well. Following the strongest anonymity definition in ring signatures, we define a new level of anonymity for ID-based ring signatures in which the attacker can specify the randomness used in the creation of the user secret key, in addition to the PKG's master key. Our scheme provides unconditional anonymity in this model.**

Keywords: ring signature, HIBE, anonymity, standard model

I. Introduction: Motivation and Background

Consider a scenario where Alice, who is an innocent member of a minority political party, has had charges brought against her and is being tried in court. One of the key pieces of evidence in support of her innocence is a copy of a digital photo, but this must be authenticated to ensure that it is genuine. Further, since the owner of the digital photo is from the minority party, he wants to ensure that his identity will not be revealed after submitting the evidence, even though the forensic photography[1] team must verify the evidence. In order to enable this, a promising cryptographic approach is the identity-based (ID-based) ring signature scheme, which allows the formation of a spontaneous signature on behalf of the minority political party while still ensuring that the real identity of the source remains hidden. The protection of the real identity of the signer is unconditional, which means that nobody will be able to find out who has actually issued the signature on the photograph provided. A thorough survey of the early constructions of ID-based ring signatures as well as of the advantages of ring signatures without the need for public key infrastructure can be found in [25]. It is unfortunate that the existing constructions of ID-based ring signatures in the standard model is not yet efficient[2], either because of the size of the parameters involved or because of the security assumptions. In this work, we aim to bridge this gap by providing an ID-based ring signature which is provably secure in the standard model under static assumptions. Our ID-based ring signatures also exhibit interesting properties such as unrestricted ring size and full anonymity even when the attacker has control over the randomness used in all key generations.

$This work is supported by ARC Future Fellowship FT0991397.

Joseph K. Liu and Jianying Zhou are with Institute for Infocomm Research, Singapore; Man Ho Au and Willy Susilo are with University of Wollongong, Australia.

*Joseph K. Liu is the corresponding author.

[1]http://en.wikipedia.org/wiki/Forensic_photography

[2]We will discuss in Section I-A that the two schemes presented in [3] are either insecure or with a flawed security analysis. On the contrary, we note that efficient constructions based on static assumptions in the random oracle model exist [27]. Our goal in this paper is to consider security in the standard model, since recent results demonstrated [4] that schemes secure in the random oracle model may be insecure in practice.

A. Related Cryptographic Notions

Ring Signature: Ring signature [49] is a group-oriented signature which takes into account privacy concerns. A user can choose autonomously to sign anonymously on behalf of a group, while group members can be totally unaware of being included in the group. Any verifier can be assured that a message has been signed by one of the members in this group, but the actual identity of the signer remains hidden. Unlike group signatures [5], there is no group manager and no revocation. The formation of the group is spontaneous. Due to these properties, it can be used for many different applications, including the following:

1) Whistle Blowing [49]: Assume there is a council meeting and members of the council are not allowed to tell outsiders about anything discussed during the meeting. Because as the discussion topic is related to the public, Alice (one of the council members) would like to leak the news to some journalists. If she sends an anonymous letter or email to the journalists, they may not believe it is from a council member as anyone could do this. If she signs the email using her secret key, although the journalists will believe it is Alice (a council member) who is leaking the news, she will be punished because she has violated the regulation. In this case, she can generate a ring signature with the public keys (or identities) of all members of the council. To do so, she does not need to seek any collaboration (she only needs to know their public keys or identities, which are known to the public), and she can simply use her own secret key. In this way, any verifier can be sure that the signature is generated by one of the members of the council but no one is able to find out who the actual signer is, even if all the secret keys are revealed. Thus Alice avoids being punished.
2) Ad Hoc Networks Authentication [43]: The fast growing popularity of portable devices and mobile applications has formed new types of groups of interacting parties: *ad hoc* groups. An *ad hoc* group has a very dynamic nature. Each member can join and leave spontaneously without seeking approval from any centralized party. Due to the lack of any trusted party, security of *ad hoc* network is a

challenging problem. Ring signatures are perfectly suited to the authentication scenario of such networks, since no set-up protocol or trusted centralized party is required and each member can authenticate itself using only its own secret key and other public information. In particular, Vehicular *Ad Hoc* Networks (VANET) (a kind of *ad hoc* network specifically designed for vehicles) can also deploy ring signatures for authentication purposes [17].

3) Smart Grid Security [45]: The smart grid is a new form of electricity network. One of the distinctive features in a smart grid is its capabilities for two-way data communication. While the supplier can output electricity to consumers, on the other side consumers and their devices can also send data to grid controllers. The ability to analyze and have access to more precise and detailed data from consumers and their devices is a major benefit of the smart grid. Based on this infrastructure, some third-parties may provide a platform for consumers to upload energy usage data, based on which a statistical report is created [28]. The purpose is to encourage consumers to compare their energy consumption with others and, as a result, use electricity more efficiently. From a security point of view, data integrity is necessary since the comparison would be meaningless if the data were faked or modified by some malicious person, however privacy is also an important concern as consumers may not want to give their identification information to any third-party service provider. A ring signature can act as a promising solution in this scenario to provide both integrity and privacy. Using a ring signature means that a valid signature will convince the service provider that the data is uploaded by a consumer on a certain street, without telling who exactly the consumer is.
4) E-voting [24]: A variant of ring signature, called 'linkable ring signature' [43], allows any verifier to know whether two ring signatures are signed by the same signer. This variant can be applied for e-voting [26], [24].
5) *Ad hoc* anonymous key agreement [21]: Due to their anonymous and *ad hoc* nature, ID-based ring signatures can be used in key agreement protocol in the ID-based setting to provide privacy protection. For instance, the *ad hoc* anonymous key agreement protocol [21] is based on the ID-based ring signatures from [27].

There is only a small number of secure ring signature schemes without random oracles. Xu et al. [57] described a ring signature scheme in the standard model but the proof is not rigorous and is apparently flawed [7]. Chow et al. [23] gave a ring signature scheme with proof in the standard model, but this requires a new assumption. Bender et al. [7] presented a ring signature which is secure in the standard model, assuming trapdoor permutations exist. Their scheme uses generic ZAPs for NP as a building block, which is inefficient. Shacham and Waters [51] proposed an efficient ring signature scheme without using random oracles. The scheme requires a computational assumption for anonymity. Chandran et al. [16] gave a sub-linear size ring signature scheme in the untrusted common reference string model. The scheme provides unconditional anonymity. Schäge and Schwenk [50] provided another ring signature scheme in the standard model using basic assumptions. However, their scheme can be only proven in a model which does not allow the adversary to query any private key.

ID-BASED RING SIGNATURE: The identity-based (ID-based) cryptosystem, introduced by Shamir [52], eliminates the necessity for testing the validity of the certificates. In an ID-based cryptosystem, the public key of each user is easily computable from a string corresponding to this user's identity, such as an email address or a telephone number. A private key generator (PKG) then computes the private keys for the users from a master secret key. This property avoids the need for using certificates and associates an implicit public key (user identity) to each user within the system.

ID-based ring signature combines the properties of ring signatures and ID-based signatures. The first construction is in Zhang and Kim [61]. Since then, several constructions have been proposed [22], [37], [42], [25][3],[27], [18], [19], [20]. All the above schemes are based on pairings with signature sizes which are proportional to the number of members in the ring. A non-pairing-based approach can be found in Herranz [36] and Tsang et al. [54]. The first constant-size construction, given by Nguyen [47], used an accumulator. The public parameters determine an upper bound n on the number of identities in the ring. We regard it as a *restricted* ring signature. The scheme was found insecure, however, by Zhang and Chen [60] and they outlined a patch. All the above constructions are only secure in the random oracle model.

Au et al. [3] proposed two ID-based ring signature schemes in the standard model. Their first construction was discovered to be flawed [29], while the second construction is a restricted ring signature, and it can be only proven in a very weak model: The adversary needs to select the challenged identities and message in the beginning, before getting the public parameters from the simulator, however, we discovered that there is a flaw in the proof of the unforgeability of this scheme. The flaw is presented in Appendix A.

Han et al. [35] claimed their scheme is secure in the standard model but Tsang et al. [54] showed that their proof is incorrect. We do not regard [35] as a provably secure scheme.

Boneh and Hamburg [13] suggested a construction of an identity-based ring signature scheme from their spatial encryption primitive, but they could only prove it selectively secure. Although full security can be achieved in a way pointed out in [2], it is also a restricted ring signature. Although the size of their signature is constant, the private key of each user is of length $O(n)$, where n is the upper bound on the number of identities in the ring determined by the public parameters.

Generic transformation from public-key based signature to (1 user) ID-based signature is possible [6], however, the transformation does not work in the case of ring signatures. In the case of ring signatures, ID-based scheme can be constructed generically but based on non-interactive zero-knowledge proof-of-knowledge. The commonly-used technique without relying on random oracles is due to [33]. Schemes constructed generically using this technique would be unrestricted and possess

[3]The work in [25] did not propose any new scheme. Instead it explains the design of a number of schemes.

comparable asymptotic complexity in the same way as as our construction. However, anonymity depends on the trusted set-up assumption[4] as the proof technique requires a common reference string.

ATTRIBUTE-BASED SIGNATURE: An attribute-based signature (ABS) assures the verifiers that a single user with a determined set of attributes satisfying a predicate has attested to the message. It is a versatile variant of digital signatures. Some of them [48], [46], [59] support non-monotone access structures and offer attribute privacy. While attribute-based encryption is a generalization of identity-based encryption, it might be natural to think that an attribute-based signature scheme supporting a predicate consisting of disjunctions with attribute privacy will provide the essential feature of an identity-based ring signature. For instance, one could set each identity to be an attribute and issue the key for this attribute to the owner of this identity. Existing construction of ABS, however, does not support this naive construction efficiently. The reason is that the maximum number of attributes or the maximum complexity of the access structure must be given in the public parameter. In other words, existing ABS can only be used to realize a ring signature scheme with restricted ring size (or *restricted* ring signature). Another possibility is to set each bit of an identity as an attribute, say, $a_1, \ldots, a_\ell$, where ℓ is the identity bit-length. The identity owner is given the signing key of the attribute set corresponding to the bits of his identity string. For example, owners of $ID_1 = 1001$ and $ID_2 = 0011$ will be assigned the attributes (a_1, a_4) and (a_3, a_4), respectively. To realize a two-party ID-based ring signature on ring (ID_1, ID_2), the access polity will be set as $(a_1 \wedge \bar{a}_2 \wedge \bar{a}_3 \wedge a_4) \vee (\bar{a}_1 \wedge \bar{a}_2 \wedge a_3 \wedge a_4)$, where $\bar{a}$ represents the negation of an attribute a. Nonetheless, the complexity of the resulting access structure is $O(\ell n)$, where n represents the size of the ring. Thus, realizing the functionality of ID-based ring signatures with ABS in this way incurs an overhead of complexity $O(\ell)$, which can be quite inefficient. We call the ID-ring signature scheme constructed this way IDR-from-ABS and its properties are shown and compared with existing schemes in Table I.

MESH SIGNATURE: The mesh signature [14] is a variant of ring signatures. While a mesh signature can be used to realize the functionalities of a ring signature, it is not clear whether or not it is sufficient to realize an ID-based ring signature. One reason is that the clauses in the access policy of a mesh signature must not share the same public key. Another issue is related to the expressiveness of a mesh signature, in which a basic policy $[VK_1 : M_1]$ assures the verifier that the mesh signature creator is in possession of a valid signature on message M_1 under verification key VK_1. The access policy of a mesh signature is an arbitrary combination of the basic policy using AND, OR as well as THRESHOLD gates. While it is easy to represent a ring signature using the policy $[VK_1 : M]$ or $[VK_2 : M]$, ..., $[VK_n : M]$, it seems to be insufficient for the realization of an ID-based ring signature. Nonetheless, the anonymity model of the mesh signature motivates the definition of anonymity in our scenario. It is pointed out in [14] that the existing anonymity notion for ring signatures is constrained in the sense that the secret keys are not chosen by the adversary. This concern is particularly relevant in our case as the adversary is the public key generator itself which is responsible for the creation of the users' signing keys.

HIERARCHICAL IBE: An identity-based encryption (IBE) [10], [11], [8], [55], [31] system is a public key system where an encryptor uses only the identity of the recipient and a set of global public parameters, so a separate public key for each entity is not required. A trusted authority holds a master secret key which allows it to create secret keys for identities and distribute them to authenticated users. A hierarchical IBE system (HIBE) [32], [38], [9], [40] provides more functionality by creating levels of an organizational hierarchy. A user at level k can delegate secret keys to descendant identities at lower levels, but cannot decrypt messages intended for a recipient that is not among its descendants.

HIBE has also been used as a building block of digital signatures where signer anonymity is of concern. For instance, a generic transformation from HIBE to ID-based group signatures is presented in [53]. There are several subtle differences between a group signature and a ring signature. Specifically, no party should be able to reveal the identity of the signer when a ring signature is used. In addition, the group formation is *ad hoc*, and the signer can select members arbitrarily. Finally, there should be no central set-up in a ring signature. While the last difference is not applicable to the case of identity-based cryptography (since the public key generator can be viewed as the central party), the differences between ID-based group signature and ID-based ring signature are still substantial.

The second ID-based ring signature scheme in [3] was built based on HIBE, but due to the differences in the two primitives (more on that discussed in Section III), their construction is only proven secure in a restricted model that seems to be tightly coupled with the definition of the selective-ID security for HIBE. We observe that even in the restricted model, the proof is flawed. (This will be further explained in Appendix A)

B. Level of Anonymity

According to [7], the strongest level of anonymity in ring signature is against full key exposure. That is, even if the secret keys of all members of the ring are exposed, or, more generally, even if the randomness used to generate the secret keys of all members of the ring are exposed, anonymity is still preserved. It makes sense when erasure cannot be ensured, or when it cannot be guaranteed that all users will comply with the directive to erase their random coins.

A direct transposition of this definition to the case of ID-based ring signatures is not straightforward. In particular, the master key and the user secret keys are created by the PKG, which may be the adversary itself. Thus, it may also be desirable for anonymity to be preserved even when the randomness used to generate the keys is *chosen* by the adversary. In this case, the adversary not only gives the indexes of the challenged

[4]The common reference string model requires that the randomness used in the setup should be erased.

users to the challenger, it also instructs the challenger to use its chosen randomness to generate two secret keys, and uses either one to generate the challenged signature. We call this 'anonymity against chosen key randomness'.

In the context of our definition, we also allow the adversary to specify the randomness used in the creation of the public parameters, the master secret key and all the other keys of the users in the system. It is reasonable because we are modelling against a dishonest PKG. The ability to choose this randomness may increase the power of the adversary to guess the actual signer. In our definition, the adversary gives the randomness to the challenger and instructs it to use this randomness to generate the secret keys for the challenged identities, one of which will be used to generate the challenged signature.

We will formally define it in a later section.

C. Comparison with different ring signature schemes

Here we compare our scheme with different provably secure unrestricted ring signature schemes that do not rely on random oracles[5]. Note that since there are no concrete unrestricted ID-based ring signature schemes without random oracles, we also include public key-based schemes in the comparison. We compare the following aspects: size of secret key, size of signature, unforgeability and anonymity. We summarize it in Table I. We use n to represent the number of users in a signature.

We also note:

(1) Unforgeability is divided into the following categories:
- ID-EF-CMA: ID-based existential unforgeablility against chosen message attack. This is also known as fully secure unforgeability in the ID-based setting.
- EF-CSA: Existential unforgeablility against chosen sub-ring attack [50]. The adversary is not allowed to query private keys.
- EF-CMA: Existential unforgeablility against chosen message attack. The adversary is allowed to corrupt any users (to obtain their secret keys) except those included in the forged signature.

(2) The legends are: U for unconditional, HS for heuristically statistical and C for computational.

(3) The unconditional anonymity provided in [16] is heuristically statistical. As analyzed in their paper, assuming the key size is 1024 bits and the ring size is less than 10000, there is less than one in a million risk of the signature not being unconditionally anonymous.

(4) m represents the number of bits in the message space.

(5) For the assumptions: DsjSDH is a new and strong assumption proposed in [23]. 'generic' means the assumption depends on the underlying schemes, as the construction is generic. 'subgp' stands for subgroup decisional assumption.

(6) The generic construction of an ID-based ring signature from ABS is described in Section I-A. In this construction, the size of the secret key and the size of the signature do not depend on the underlying ABS scheme. ℓ is the bit-length of an identity and we use $\ell = 160$. Under our generic construction, the size of the secret key and the signature depends on the number of '1' in the bit string of the (hash of the) identity. $O(\ell)$ just gives the upper bound of the complexity. We state here as an open problem to construct an unrestricted ID-based ring signature scheme from any ABS that does not rely on random oracle with the upper bound of the secret key less than $O(\ell)$.

[5]We exclude the restricted ID-ring signature by [13], [2] and following schemes although they also claim to be secure without random oracles: For the scheme in [57], the proof presented is not rigorous and is apparently flawed as stated in [7]. For the scheme in [35], the proof is incorrect as indicated in [54]. For the first scheme from [3], it is proven insecure by [29]. For the second scheme from [3], it is a restricted one and there is a flaw in the proof as explained in Appendix A.

D. Our Contribution

In this work, we propose a feasible ID-based ring signature scheme based on an existing HIBE scheme. We define a strong level of anonymity for ID-based ring signatures. Our scheme provides anonymity against a computationally unbounded adversary in this strong model. Our observation and explanation of the relationship between ID-based ring signatures and HIBE is of more general interest.

II. Preliminaries

A. Notations

If a, b are integers with $a < b$, we use $[a, b]$ to represent the set $\{a, \ldots, b\}$. For simplicity, we use $[n]$ to represent the set $\{1, \ldots, n\}$ if n is a positive integer. If $\mathbb{G}$ is a cyclic group, we abuse the notation and use 1 to represent the identity element of $\mathbb{G}$ regardless of its actual structure. As usual, $||$ denotes string concatenation.

B. Composite Order Bilinear Pairing

Composite order bilinear groups were first introduced in [12]. We define them by using a group generator ($\mathcal{G}$), an algorithm which takes a security parameter (λ) as input and outputs a description of the bilinear group ($\mathbb{G}$). In our case, $\mathcal{G}$ outputs $(N = p_1p_2p_3, \mathbb{G}, \mathbb{G}_T, e)$ where p_1, p_2, p_3 are distinct primes, $\mathbb{G}$ and $\mathbb{G}_T$ are cyclic groups of order $N = p_1p_2p_3$ and $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is a map such that :

1) *Bilinearity*: For all $g, h \in \mathbb{G}$, and $a, b \in \mathbb{Z}_N$, $e(g^a, h^b) = e(g, h)^{ab}$.
2) *Non-degeneracy*: There exists $g \in \mathbb{G}$ such that $e(g, g)$ has order N in $\mathbb{G}_T$.
3) *Computability*: It is efficient to compute $e(g, h)$ for all $g, h \in \mathbb{G}$.

Also we assume the group descriptions of $\mathbb{G}$ and $\mathbb{G}_T$ include generators of the respective cyclic groups. We let $\mathbb{G}_{p_1}$, $\mathbb{G}_{p_2}$ and $\mathbb{G}_{p_3}$ denote the subgroups of order p_1, p_2 and p_3 in $\mathbb{G}$ respectively. Likewise $\mathbb{G}_{p_ip_j}$ denotes a subgroup of $\mathbb{G}$ of order p_ip_j. We note that when $h_i \in \mathbb{G}_{p_i}$ and $h_j \in \mathbb{G}_{p_j}$ for $i \neq j$, $e(h_i, h_j)$ is the identity element in $\mathbb{G}_T$. To see this, suppose $h_1 \in \mathbb{G}_{p_1}$ and $h_2 \in \mathbb{G}_{p_2}$. We let g denote a generator of $\mathbb{G}$. Then $g^{p_1p_2}$ generates $\mathbb{G}_{p_3}$, $g^{p_1p_3}$ generates $\mathbb{G}_{p_2}$ and $g^{p_2p_3}$ generates $\mathbb{G}_{p_1}$. Hence, for some α_1, α_2, $h_1 = (g^{p_2p_3})^{\alpha_1}$ and $h_2 = (g^{p_1p_3})^{\alpha_2}$. We note:

$$e(h_1, h_2) = e(g^{p_2p_3\alpha_1}, g^{p_1p_3\alpha_2}) = e(g^{\alpha_1}, g^{p_3\alpha_2})^{p_1p_2p_3} = 1$$

TABLE I
COMPARISON OF UNRESTRICTED RING SIGNATURES WITHOUT RANDOM ORACLES

Scheme	Size of secret key	Size of signature	ID / PKI-based	Unforgeability[1]	Anonymity[2]	Assumption[5]	Pairing (group order)
Chow *et al.* [23]	$O(1)$	$O(n)$	PKI	EF-CSA	U	DsjSDH	prime
Bender *et al.* [7]	$O(1)$	$O(n)$	PKI	EF-CMA	U	generic	non-pairing
Shacham *et al.* [51], Yuen *et al.* [58]	$O(1)$	$O(n)$	PKI	EF-CMA	C	SDH, subgp	composite (2 primes)
Chandran *et al.* [16]	$O(1)$	$O(\sqrt{n})$	PKI	EF-CMA	HS[3]	SDH, subgp	composite (2 primes)
Brakerski *et al.* [15]	$O(m)$[4]	$O(mn)$	PKI	EF-CMA	unspecify	generic	non-pairing
Schäge *et al.* [50]	$O(1)$	$O(n)$	PKI	EF-CSA	U	CDH	prime
IDR-from-ABS[6]	$O(\ell)$	$O(\ell n)$	ID	ID-EF-CMA	U	generic	prime
Our Scheme	$O(1)$	$O(n)$	ID	ID-EF-CMA	U	subgp	composite (3 primes)

This orthogonality property of $\mathbb{G}_{p_1}$, $\mathbb{G}_{p_2}$ and $\mathbb{G}_{p_3}$ has been used in [40].

Every element $E \in \mathbb{G}$ can be uniquely decomposed into three components $(E_1, E_2, E_3) \in (\mathbb{G}_{p_1}, \mathbb{G}_{p_2}, \mathbb{G}_{p_3})$ such that $E = E_1E_2E_3$. For notational convenience, we use $\Psi_i(E)$ to denote E_i for $i = 1$ to 3. Note that 1 is the identity element of $\mathbb{G}$ as well as $\mathbb{G}_{p_i}$ for $i = 1$ to 3 since $\mathbb{G}_{p_i}$ is a subgroup of $\mathbb{G}$.

C. Intractability Assumptions

We review some intractability assumptions in bilinear groups, which have been defined in [41].

Assumption 1 (Subgroup decision problem for 3 primes): Given a group generator ($\mathcal{G}$), we define the following distribution:

$$G = (N = p_1p_2p_3, \mathbb{G}, \mathbb{G}_T, e) \xleftarrow{R} \mathcal{G},$$
$$g \xleftarrow{R} \mathbb{G}_{p_1}, X_3 \xleftarrow{R} \mathbb{G}_{p_3},$$
$$D = (G, g, X_3),$$
$$T_1 \xleftarrow{R} \mathbb{G}_{p_1p_2}, T_2 \xleftarrow{R} \mathbb{G}_{p_1}$$

We define the advantage of an algorithm ($\mathcal{A}$) in breaking Assumption 1 to be:

$$Adv1_{\mathcal{G},\mathcal{A}}(\lambda) := |\Pr[\mathcal{A}(D, T_1) = 1] - \Pr[\mathcal{A}(D, T_2) = 1]|$$

Definition 1: We say that $\mathcal{G}$ satisfies Assumption 1 if $Adv1_{\mathcal{G},\mathcal{A}}(\lambda)$ is a negligible function of λ for any polynomial time algorithm $\mathcal{A}$.

Assumption 2: Given a group generator ($\mathcal{G}$), we define the following distribution:

$$G = (N = p_1p_2p_3, \mathbb{G}, \mathbb{G}_T, e) \xleftarrow{R} \mathcal{G}, \alpha, s \xleftarrow{R} \mathbb{Z}_N,$$
$$g \xleftarrow{R} \mathbb{G}_{p_1}, g_2, X_2, Y_2 \xleftarrow{R} \mathbb{G}_{p_2}, g_3 \xleftarrow{R} \mathbb{G}_{p_3},$$
$$D = (G, g, g_2, g_3, g^{\alpha}X_2, g^{s}Y_2),$$
$$T_1 = e(g, g)^{\alpha s}, T_2 \xleftarrow{R} \mathbb{G}_T$$

We define the advantage of an algorithm ($\mathcal{A}$) in breaking Assumption 2 to be:

$$Adv2_{\mathcal{G},\mathcal{A}}(\lambda) := |\Pr[\mathcal{A}(D, T_1) = 1] - \Pr[\mathcal{A}(D, T_2) = 1]|$$

Definition 2: We say that $\mathcal{G}$ satisfies Assumption 2 if $Adv2_{\mathcal{G},\mathcal{A}}(\lambda)$ is a negligible function of λ for any polynomial time algorithm $\mathcal{A}$.

Assumption 3: Given a group generator ($\mathcal{G}$), we define the following distribution:

$$G = (N = p_1p_2p_3, \mathbb{G}, \mathbb{G}_T, e) \xleftarrow{R} \mathcal{G},$$
$$g, X_1 \xleftarrow{R} \mathbb{G}_{p_1}, g_2 \xleftarrow{R} \mathbb{G}_{p_2}, X_3 \xleftarrow{R} \mathbb{G}_{p_3},$$
$$D = (G, g, g_2, X_1X_3),$$
$$T_1 \xleftarrow{R} \mathbb{G}_{p_1}, T_2 \xleftarrow{R} \mathbb{G}_{p_1p_3}$$

We define the advantage of an algorithm ($\mathcal{A}$) in breaking Assumption 3 to be:

$$Adv3_{\mathcal{G},\mathcal{A}}(\lambda) := |\Pr[\mathcal{A}(D, T_1) = 1] - \Pr[\mathcal{A}(D, T_2) = 1]|$$

Definition 3: We say that $\mathcal{G}$ satisfies Assumption 3 if $Adv3_{\mathcal{G},\mathcal{A}}(\lambda)$ is a negligible function of λ for any polynomial time algorithm $\mathcal{A}$.

Assumption 4: Given a group generator ($\mathcal{G}$), we define the following distribution:

$$G = (N = p_1p_2p_3, \mathbb{G}, \mathbb{G}_T, e) \xleftarrow{R} \mathcal{G},$$
$$g, X_1 \xleftarrow{R} \mathbb{G}_{p_1}, X_2, Y_2 \xleftarrow{R} \mathbb{G}_{p_2}, g_3, Y_3 \xleftarrow{R} \mathbb{G}_{p_3},$$
$$D = (G, g, X_1X_2, g_3, Y_2Y_3),$$
$$T_1 \xleftarrow{R} \mathbb{G}, T_2 \xleftarrow{R} \mathbb{G}_{p_1p_3}$$

We define the advantage of an algorithm ($\mathcal{A}$) in breaking Assumption 4 to be:

$$Adv4_{\mathcal{G},\mathcal{A}}(\lambda) := |\Pr[\mathcal{A}(D, T_1) = 1] - \Pr[\mathcal{A}(D, T_2) = 1]|$$

Definition 4: We say that $\mathcal{G}$ satisfies Assumption 4 if $Adv4_{\mathcal{G},\mathcal{A}}(\lambda)$ is a negligible function of λ for any polynomial time algorithm $\mathcal{A}$.

III. OVERVIEW OF OUR CONSTRUCTION

We provide an overview of our construction. Firstly, we describe a design principle for an ID-based ring signature from HIBE. We would like to stress that this is not a generic construction as the security requirement of ID-based ring signature differs significantly from HIBE. Nonetheless, given any HIBE that satisfies a certain condition, one could construct an ID-based ring signature scheme. Next, we describe the methodology for proving our proposed scheme.

A. Relationship between HIBE and ID-based Ring Signature

a) Hierarchical ID-Based Encryption and Signatures.: Any $(n+1)$-level HIBE can be converted into a n-level Hierarchical ID-based Signature (HIBS) [11]. Specifically, an n-level HIBS ($\sigma_{\vec{I},m}$) from signer of identity vector $\vec{I} = (ID_1, \ldots, ID_n)$ on message m is the decryption key of the $(n+1)$-level HIBE on identity vector $\vec{I} = (ID_1, \ldots, ID_n, m)$. To verify signature $\sigma_{\vec{I},m}$, the verifier creates ciphertext C which is the encryption of random value V under the $(n+1)$-level identity vector $\vec{I} = (ID_1, \ldots, ID_n, m)$ in the HIBE. If signature $\sigma_{\vec{I},m}$ in the n-level HIBS can decrypt ciphertext C and output V correctly, the verifier accepts $\sigma_{\vec{I},m}$.

b) HIBS and ID-Based Ring Signatures.: We first observe the similarities between HIBS and ID-based ring signature schemes. Both of them are ID-based signatures. They also include a number of identities in the signature. In a HIBS (converted from a HIBE), a signature in fact is a secret key generated by a node delegated to his child where the message is set to be the identity of the child. If this key is indistinguishable from another key generated by the PKG directly (or from any previous nodes in the upper level), the signature still provides some measure of signer ambiguity. As the keys are indistinguishable, the signature can actually be generated by any part of the prefix of the embedded identity. For instance, a signature on message m on identity vector $\vec{I} = (ID_1, ID_2, ID_3, ID_4, ID_5)$ can be created from signers of 5 different identities: $\vec{I}_1 = (ID_1)$, $\vec{I}_2 = (ID_1, ID_2)$, $\vec{I}_3 = (ID_1, ID_2, ID_3)$, $\vec{I}_4 = (ID_1, ID_2, ID_3, ID_4)$, $\vec{I}_5 = (ID_1, ID_2, ID_3, ID_4, ID_5)$. In other words, it already shares the characteristic of a ring-signature. That is, it is created by 1-out-of-5 possible signers.

However, there is a crucial difference: consider a signing key of identity $\vec{I} = (ID_1)$. While it can be used to create signatures on behalf of any identity vector of the form $(ID_1, *, *, *)$, it can only place its identity as the first element of the vector. Specifically, it can create signatures on behalf of (ID_1, ID_2), (ID_1, ID_2, ID_3) but not (ID_2, ID_1) nor (ID_3, ID_2, ID_1). On the other hand, the identity of the actual signer in a ring signature can be placed in any position in the list of signers. We can see that an ID-based ring signature cannot be obtained directly from a HIBS.

c) Our Approach.: We need to apply a special 'rotational system' to eliminate the ordering in HIBS. Suppose it is possible to modify an HIBS in such a way that the key for the identity vector $\vec{I} = (ID)$ could be used to sign on behalf of any identity vector $\vec{I}'$ containing ID at any position. Then the signature on behalf of an n-level HIBS of signer with identity $(ID_1, \ldots, ID_n)$ can be treated as an ID-based ring signature from n possible signers of identities (ID_1), (ID_2), ..., (ID_n) because it could have been created by the signer having a key of identity vector $\vec{I} = (ID_i)$ for $i \in [n]$. Of course, none of the existing HIBS would support this feature. For if this is the case, an adversary could simply request the key of identity vector $\vec{I} = (ID_2)$ and use it to forge a signature on identity vector $\vec{I}^* = (ID_1, ID_2)$ and thus breaking the unforgeability of the HIBS.

While this modification could lead to a trivial attack in the HIBS security model, it can still be proven secure in the ring signature model. The reason is that the adversary is not allowed to possess any keys in the ring of signers of the forged signature. In other words, the forgery of a signature on a ring (ID_1, ID_2) is not a valid attack if the adversary has asked the key for identity ID_1 or ID_2, as the adversary is not allowed to do so. Now the remaining questions are (1) whether or not such a modification is feasible in existing HIBS and (2) whether the modified scheme is secure in ID-based ring signature model. We answer both questions affirmatively.

We answer the first question by showing how some existing HIBE schemes can be modified to have the above property. We use an existing HIBE as a starting point. Recently, Lewko and Waters [41] proposed an unbounded HIBE (referred to as LW11-HIBE hereafter). LW11-HIBE supports an arbitrary number of levels. That is, the maximum depth of the hierarchy does not need to be fixed at the set-up stage. In addition, the hierarchical structure of the secret keys is not embedded in the system parameter. Rather, special encoding is applied to the identity vector to deploy the hierarchy. LW11-HIBE requires that for any identity vector $\vec{I} = (ID_1, \ldots, ID_j)$ that is not a prefix of another identity vector $\vec{I}' = (ID'_1, \ldots, ID'_\ell)$, $ID_j \neq ID'_k$ for $k \in [\ell]$. One encoding method which ensures this property is proposed. For instance, an identity vector $(ID_1, ID_2, \ldots, ID_k)$ would be encoded as $(ID_1, ID_1\|ID_2, \ldots, ID_1\|ID_2\|\cdots\|ID_k)$. If we remove this encoding, a 1-level key for identity $\vec{I} = (ID)$ can be used to easily produce a key for n-level identity $(ID_1, \ldots, ID_n)$ as long as $ID = ID_i$ for some i.

The final question is, after the modification, is the scheme secure in the ring signature model? We present an overview of the answer in the next subsection.

B. Security of Our Scheme following the Design Principle

d) On Anonymity.: Recall that a signature in HIBS, and thus our ring signature, is an $(n+1)$-level decryption key of the underlying HIBE. The perfect anonymity of the resulting ring signatures follows closely from the delegation invariance of the HIBE scheme. That is, a key for an $(n+1)$-level identity created by the PKG is indistinguishable from the key generated from any previous nodes in the upper level. It is straightforward to see after the modification, an n-level key can be created from one of these n possible 1-level keys with the same distribution. This will be clear when we present our construction.

e) On Unforgeability.: Our scheme is derived from the HIBE scheme [41] the security of which is analyzed using a new methodology, often called the dual system encryption methodology, for proving the security of encryption systems [56]. Before explaining the setting of our proof, it will be necessary to review some basic concepts in a dual system encryption. In such a paradigm, keys and ciphertexts can take on one of two forms: normal (type-N) or semi-functional (type-S). A private key or ciphertext is of type-N if it is created normally from the system parameters following the correct key generation and encryption algorithm. A type-S key or ciphertext is of a special structure that only appears in the security proof. A type-S key will be able to decrypt all ciphertext of type-N but

it will fail to decrypt a ciphertext of type-S. On the other hand, a type-N key will be able to decrypt both type-S and type-N ciphertexts. One important feature is that keys and ciphertexts in the two forms are indistinguishable from each other.

Dual system encryption opens up a new possibility in security proof by reduction. Reduction in the setting in which all the keys and the challenge ciphertext returned by the simulator are of type-S are often relatively straightforward. A sequence of games is established to demonstrate that the success probability of the adversary in the game where all keys and the challenge ciphertext it received are of type-S is equivalent to its success probability in the normal game where all keys and challenge ciphertexts are of type-N.

A similar idea applies to digital signatures. [56] described how the dual system technique can be applied to demonstrate the security of the signature schemes converted from the IBE in the same paper. Inspired by this and the dual system encryption proof in [40], [41], we prove the security of our ring signature scheme as follows. There are two types of signature and key: type-N and type-S. Under normal system operation, all signatures and keys are of type-N. The forgery can be classified into two types according to the type of signature that the forger outputs and we describe the reduction proof for each of these. Looking ahead, our scheme follow this structure. Let $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_3$ be subgroups of a cyclic group ($\mathbb{G}$). A type-N key or signature consists only of elements in $\mathbb{G}_1$ while a type-S component contains terms of all $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_3$.

- **Type-S Forger:** Suppose the forger is of type-S. The simulator in this case only outputs type-N signatures and keys. Thus, it can be set up using elements in $\mathbb{G}_1$ only. Finally, the forger outputs something that has an element in $\mathbb{G}_2$. The simulator can then make use of this element in $\mathbb{G}_2$ to decide if the element in the problem instance has a $\mathbb{G}_2$ component by pairing it with the forged signature.
- **Type-N Forger:** Suppose that the forger is of type-N. The simulator in this case first uses a game-hopping technique to change the key and the signatures that it gives to the adversary from type-N to type-S in each query until finally everything given to the forger from the simulator is of type-S. Specifically, in the first game, all keys and signatures given to the adversary are of type-N. In the second game, the first key or signature given to the adversary is of type-S and the remaining are of type-N. In the j-th game, the keys and signatures given to the adversary before the j-th query are of type-S. In the final game, all keys and signatures given to the adversary are of type-S. This is possible due to the indistinguishability of type-N and type-S keys and signatures so the simulator can switch one answer in a game. In the final game, the reduction is simple. Given a type-N forgery consisting only of elements from $\mathbb{G}_1$, the simulator can distinguish if the problem instance is a random element in $\mathbb{G}$ or not.

IV. Security Models

A. Definition

An ID-based $(1, n)$ ring signature scheme is a tuple of probabilistic polynomial-time (PPT) algorithms below:

- **Setup**. On input of a unary string (1^λ) where λ is a security parameter, the algorithm outputs master secret key α and list of system parameters (param) that includes λ and the descriptions of user secret key space $\mathcal{SK}$, message space $\mathcal{M}$, identity space $\mathcal{ID}$ as well as signature space $\mathcal{SG}$.
- **Extract**. On input system parameters param, identity $ID \in \mathcal{ID}$ for a user and master secret key α, the algorithm outputs the user's secret key ($sk_{ID} \in \mathcal{SK}$). When we say identity ID corresponds to user secret key sk_{ID} or vice versa, we mean the pair (ID, sk_{ID}) is an input-output pair of **Extract** with respect to param and α.
- **Sign**. On input param, group size n of length polynomial in λ, set $\mathcal{L} = \{ID_i \in \mathcal{ID} | i \in [n]\}$ of n user identities, message $m \in \mathcal{M}$, and secret key $\{sk_{ID'} \in \mathcal{SK} | ID' \in \mathcal{L}\}$, the algorithm outputs an ID-based $(1, n)$ ring signature ($\sigma \in \mathcal{SG}$).
- **Verify**. On input param, group size n of length polynomial in λ, set $\mathcal{L} = \{ID_i \in \mathcal{ID} | i \in [n]\}$ of n user identities, message $m \in \mathcal{M}$, signature $\sigma \in \mathcal{SG}$, it outputs either valid or $\mathsf{invalid}$.

Correctness. An ID-based $(1, n)$ ring signature scheme should satisfy verification correctness – signatures signed by honest signers are verified to be invalid with negligible probability.

B. Security Requirement

A secure ID-based $(1, n)$ ring signature scheme should be unforgeable and anonymous, defined in a similar way to that of a traditional ring signature scheme.

Unforgeability: It should not be possible for an adversary to forge any signature from the identities of the group members alone. We specify a security model which captures the following two attacks:

1) Adaptive chosen message attack
2) Adaptive chosen identity attack

An adaptive chosen message attack allows an adversary to obtain message-signature pairs on demand during the forging attack. An adaptive chosen identity attack allows the adversary to forge a signature with respect to a group chosen by the adversary. To support an adaptive chosen message attack, we provide the adversary with the following oracle queries.

- **Extraction oracle** $(\mathcal{EO})$**:** On input ID, $sk_{ID} \leftarrow$ **Extract**(param, ID) is returned.
- **Signing oracle** $(\mathcal{SO})$**:** $\mathcal{A}$ chooses set of n identities $\mathcal{L} = \{ID_i\}_{i \in [n]}$ and message m, the oracle outputs a valid ID-based $(1, n)$ ring signature denoted by $\sigma \leftarrow$ **Sign**$(\mathsf{param}, n, \mathcal{L}, m)$. The signing oracle may query the extraction oracle during its operation.

Let $(\mathsf{param}, \alpha) \leftarrow \mathbf{Setup}$. An adversary $(\mathcal{A})$ with oracles $\mathcal{EO}$ and $\mathcal{SO}$ *succeeds* if it outputs $(\mathcal{L}, m, \sigma) \leftarrow \mathcal{A}^{\mathcal{SO}, \mathcal{EO}}(\mathsf{param})$, such that it satisfies **Verify**$(\mathsf{param}, \mathcal{L}, m, \sigma) = \mathsf{valid}$, where $\mathcal{L} \subseteq \mathcal{ID}$, and with restriction that $(\mathcal{L}, m)$ should not be in the set of oracle queries and replies between $\mathcal{A}$ and $\mathcal{SO}$, and $\mathcal{A}$ is not allowed to make an Extraction query on any identity $ID \in \mathcal{L}$.

The advantage of adversary $\mathcal{A}$ is defined to be

$$\mathbf{Adv}_{\mathcal{A}} = \Pr[\mathcal{A} \text{ succeeds}]$$

Definition 5 (Unforgeability): An ID-based $(1, n)$ ring signature scheme is unforgeable against adaptive chosen message and chosen identity attack if all polynomial time adversaries achieve at most a negligible advantage in the above security game.

Note that we cannot achieve unforgeability in the stronger sense that the adversary produces a different signature on the same message and the same list of identities, as described in [1], [44] since our proposed scheme does not possess this stronger level of security.

Anonymity: It should not be possible for an adversary ($\mathcal{A}$) to tell the identity of the signer with a probability larger than $1/n$, where n is the cardinality of the ring, even assuming that the adversary has unlimited computing resources. We further allow the adversary to choose the randomness used in the generation of the public parameters, the master secret key, as well as the secret keys corresponding to the challenged identities. It is formally defined in the following game:

1) The adversary sends a randomness ω_{Setup} to the simulator. The simulator generates $(\mathsf{param}, \alpha) \leftarrow \mathbf{Setup}$ using randomness ω_{Setup}. It gives $\mathcal{A}$ (param, α).[6]
2) $\mathcal{A}$ outputs message m, two identities ID_0, ID_1, list of identities $\mathcal{L}$ such that $ID_0, ID_1 \in \mathcal{L}$ and two randomness ω_0, ω_1.
3) The simulator randomly chooses a bit ($b \in_R \{0, 1\}$), and generates secret key for ID_b using randomness ω_b. It uses this secret key to generate a ring signature for message m and ring $\mathcal{L}$. It gives $\mathcal{A}$ the signature.
4) $\mathcal{A}$ outputs a bit (b') and succeeds if $b' = b$.

The advantage of an adversary $\mathcal{A}$ is defined to be

$$\mathbf{Adv}_{\mathcal{A}} = \Pr[\mathcal{A} \text{ succeeds}] - 1/2$$

Definition 6 (Anonymity against chosen key randomness): An ID-based ring signature scheme is said to be unconditional anonymous against chosen key randomness attack if the advantage of $\mathcal{A}$ is zero.

V. The Proposed 1-out-of-n ID-based Ring Signature Scheme

A. Construction

We use the setting from the ID-based cryptosystem in LW11-HIBE.

Setup. Choose a bilinear group ($\mathbb{G}$) of order $N = p_1p_2p_3$ (where p_1, p_2, p_3 are distinct primes). Let ℓ be the length of the identities. Let $H_0 : \{0,1\}^* \to \mathbb{Z}_N, H_1 : \{0,1\}^* \times \{0,1\}^* \to \mathbb{Z}_N$ be two collision-resistant hash functions. Choose $\alpha \in \mathbb{Z}_N$ as the master secret key and $g, h, u, v, w \in \mathbb{G}_{p_1}$. The public parameters are published as

$$\mathsf{param} = \{N, g, h, u, v, w, e(g,g)^{\alpha}, H_0, H_1\}.$$

The master secret key is α.

Extract. To generate a private key for $ID \in \{0,1\}^{\ell}$, randomly generate $r, y, \in_R \mathbb{Z}_N$, compute $\mathsf{ID} = H_0(ID)$ and

$$A = g^{\alpha}w^{y}, \qquad B = g^{y}, \qquad C = v^{y}(u^{\mathsf{ID}}h)^{r}, \qquad D = g^{r}.$$

[6] $\mathcal{A}$ can check if $\mathcal{S}$ is computing the keys using randomness ω_{Setup}.

Output the private key for user with identity ID as $SK_{ID} = \{A, B, C, D\}$.

Sign. Let $\mathcal{L} = \{ID_1, \ldots, ID_n\}$ be the list of n identities to be included in the ring signature. We assume the user with identity ID' is the actual signer, where $ID' \in \mathcal{L}$. W.l.o.g. we also assume that ID' is in the π-th position in $\mathcal{L}$, where $\pi \in \{1, \ldots, n\}$. That is, $ID_{\pi} = ID'$. To sign message $m \in \{0,1\}^*$, compute $\mathsf{ID}_i = H_0(ID_i)$ for $i = 1$ to n. Further compute $\mathsf{ID}_{n+1} = H_1(m, \mathcal{L})$. Then, execute the following using private key $SK_{ID'} = \{A, B, C, D\}$:

1) Randomly generate $y_i, r_i, \lambda_i \in_R \mathbb{Z}_N$ for $i = 1$ to $n+1$ subject to the constraint that
$$\lambda_1 + \ldots + \lambda_n + \lambda_{n+1} = 0 \tag{1}$$
2) FOR $i = 1$ to $n+1$ Do BEGIN
   - IF $i \neq \pi$ THEN
$$A_i = g^{\lambda_i}w^{y_i}, \qquad B_i = g^{y_i},$$
$$C_i = v^{y_i}(u^{\mathsf{ID}_i}h)^{r_i}, \qquad D_i = g^{r_i}$$
   ELSE // $i = \pi$
$$A_{\pi} = Ag^{\lambda_{\pi}}w^{y_{\pi}}, \quad B_{\pi} = Bg^{y_{\pi}},$$
$$C_i = Cv^{y_{\pi}}(u^{\mathsf{ID}_{\pi}}h)^{r_{\pi}}, \quad D_i = Dg^{r_{\pi}}$$
   END IF
   END
3) Output signature $\sigma = \left\{A_i, B_i, C_i, D_i\right\}_{i=1}^{n+1}$.

Verify. Given signature $\sigma = \left\{A_i, B_i, C_i, D_i\right\}_{i=1}^{n+1}$ for list of identities $\mathcal{L} = \{ID_1, \ldots, ID_n\}$ on message m, first compute $\mathsf{ID}_i = H_0(ID_i)$ for $i = 1$ to n and $\mathsf{ID}_{n+1} = H_1(m, \mathcal{L})$. Then randomly generate $s, t_1, \ldots, t_{n+1} \in_R \mathbb{Z}_N$ and check whether

$$\prod_{i=1}^{n+1} \frac{e\left(g^s, A_i\right) \cdot e\left(g^{t_i}, C_i\right)}{e\left(w^s v^{t_i}, B_i\right) \cdot e\left((u^{\mathsf{ID}_i}h)^{t_i}, D_i\right)} \stackrel{?}{=} e(g,g)^{\alpha s} \tag{2}$$

Output valid if the equality holds. Otherwise output invalid.

B. Feasibility Study

It is known that schemes working in groups equipped with composite order pairing, including the LW11-HIBE on which our proposal is based, are much slower than those with prime order pairing. For instance, [30] estimates that the former is 50 times slower. We provide an estimation of the performance of our scheme based on the latest results in the realization of groups with composite order pairing [34]. Table II provides an estimation of the timing of each operation in our proposal (for ring size $n = 10$) based on the benchmark results from [34] for various operations.

The benchmark of [34] for exponentiations and pairing are from groups equipped with composite order pairing, where the group order is of 3072-bit and is a product of three primes. This setting offers a security comparable to 128-bit AES. The software implementation of [34] is executed on a 2.6 GHz Intel

TABLE II
COMPLEXITY ANALYSIS OF OUR CONSTRUCTION

Algorithm	Computation Cost	Estimation (when n = 10)
Setup	1 Pairing	1.27s
Extract	7 $\mathbb{G}$-Exponentiations	3.85s
Sign	7*(n+1) $\mathbb{G}$-Exponentiations	42.35s
Verify	(4n + 6) $\mathbb{G}$-Exponentiations, 1 $\mathbb{G}_T$-Exponentiations, (4n+4) Pairing	44.1s

Celeron 64 bit PC with 1 GB RAM and Ubuntu 10.04.4 LTS OS. The estimation shows that while our proposal is not fast, it is at least feasible for a small ring size.

Recent work on improving systems based on composite order pairing has been promising. These include transforming schemes based on composite order pairing into schemes requiring prime order pairing only [30], [39]. Improved timing is made possible based on optimised implementation of composite-order bilinear pairings with the help of the processing power of GPU [62]. It is hoped that our proposal would become practical for larger ring sizes in the near future.

C. Security Analysis

Correctness. Our construction is correct as shown in the following:

$$\begin{aligned}
&\prod_{i=1}^{n+1}\left(\frac{e(g^s,A_i)\cdot e(g^{t_i},C_i)}{e(w^s v^{t_i},B_i)\cdot e((u^{\mathsf{ID}_i}h)^{t_i},D_i)}\right)\\
=\;& \frac{e(g^s,\prod_{i=1}^{n+1}A_i)}{e(w^s,\prod_{i=1}^{n+1}B_i)}\prod_{i=1}^{n+1}\left(\frac{e(g^{t_i},C_i)}{e(v^{t_i},B_i)e((u^{\mathsf{ID}_i}h)^{t_i},D_i)}\right)\\
=\;& \frac{e(g^s,g^{\alpha}w^{y+\sum_{i=1}^{n+1}y_i})}{e(w^s,g^{y+\sum_{i=1}^{n+1}y_i})}\prod_{i=1}^{n+1}\left(\frac{e(g,C_i)}{e(v,B_i)e((u^{\mathsf{ID}_i}h),D_i)}\right)^{t_i}\\
=\;& e(g,g)^{\alpha s}\left(\frac{e(g,C)}{e(v,B)e(u^{\mathsf{ID}_\pi}h,D)}\right)^{t_\pi}\cdot\\
&\prod_{i=1}^{n+1}\left(\frac{e(g,v^{y_i}(u^{\mathsf{ID}_i}h)^{r_i})}{e(g,v^{y_i})e(g,(u^{\mathsf{ID}_i}h)^{r_i})}\right)^{t_i}\\
=\;& e(g,g)^{\alpha s}\left(\frac{e\big(g,v^{y}(u^{\mathsf{ID}_\pi}h)^{r}\big)}{e\big(v,g^{y}\big)e\big((u^{\mathsf{ID}_\pi}h),g^{r}\big)}\right)^{t_\pi}\\
=\;& e(g,g)^{\alpha s}
\end{aligned}$$

Theorem 1: Our construction satisfies definition 5 (i.e. our scheme is unforgeable) under Assumptions 1, 2, 3 and 4.

Proof Overview: We divide the signature into two types, namely, type-N and type-S. A signature $(\{A_i,B_i,C_i,D_i\}_{i=1}^{n+1})$ is of type-N if $A_i = \Psi_1(A_i)$, $B_i = \Psi_1(B_i)$ for $i = 1$ to $n+1$, where n is the size of the ring. In other words, a type-N signature is a signature such that all components A_i, B_i are from group $\mathbb{G}_{p_1}$ only. A signature is of type-S if it is not of type-N. In the normal system operation, all signatures are of type-N. Likewise, a private key (A,B,C,D) is of type-N if both $A = \Psi_1(A)$ and $B = \Psi_1(B)$. Otherwise, a private key is of type-S.

In proof part I, we show the adversary cannot output a forgery of type-S. Then, in proof part II, we show that the adversary cannot output a forgery of type-N.

Proof part II is more involved and is achieved via the game-hopping technique. The first game is the game of unforgeability. In the next game, Game_0, we make the restriction that any distinct hash values produced by the adversary are also distinct modulo p_3. That is, the adversary cannot produce two identities ID and ID' such that $ID \neq ID'$ but $H_0(ID) = H_0(ID') \bmod p_3$. Furthermore, the adversary cannot produce two distinct rings of identities and messages $(\mathcal{L},m) \neq (\mathcal{L}',m')$ such that $H_1(\mathcal{L},m) = H_1(\mathcal{L}',m') \bmod p_3$. (From here on, the term restricted adversary refers to one which works within this restriction.) In Game_ι, the first ι queries are answered so that the reply is of type-S. For instance, if the κ-th query is an extract query and that $\kappa < \iota$, the key returned to the adversary is of type-S. Likewise, if the κ-th query is a signature query and that $\kappa < \iota$, the signature returned to the adversary is of type-S. Otherwise, the key or signature returned to the adversary is of type-N. We first show that the adversary in Game_ι for $\iota = 0$ to k is always restricted. Then we show the behavior of the restricted adversary between $\mathsf{Game}_{\iota-1}$ and Game_ι is the same for $\iota = 1$ to k where k is the number of queries made by the adversary. Finally, we show that probability of the adversary wining Game_k is negligible with another reduction. This completes proof part II.

To provide a consistent view to the adversary in the proof part II, the distributions of the type-S keys and type-S signatures provided to the adversary in Game_ι for $\iota = 1$ to k must be identical. Specifically, the type-S keys and signatures provided to the adversary in proof part II satisfy the following distribution. Let $\mathfrak{e} \in_R \mathbb{G} \neq 1_{\mathbb{G}}$ be an element in $\mathbb{G}$ such that $\Psi_1(\mathfrak{e}) = 1_{\mathbb{G}_{p_1}}$. Let $\mathfrak{a},\mathfrak{b} \in_R \mathbb{Z}_N$ be two constants that is chosen at the beginning of the simulation. Any type-S key (A,B,C,D) given to the adversary in the simulation satisfies the following equations:

$$\begin{aligned}
\Psi_2(A)\Psi_3(A) &= \mathfrak{e}^{\mathfrak{a}}\\
\Psi_2(B)\Psi_3(B) &= \mathfrak{e}\\
\Psi_2(C)\Psi_3(C) &= \mathfrak{e}^{\mathfrak{b}}\\
\Psi_2(D)\Psi_3(D) &= 1_{\mathbb{G}}
\end{aligned}$$

In other words, all type-S keys given to the adversary in a game always contain the same element in subgroup $\mathbb{G}_{p_2}$ and $\mathbb{G}_{p_3}$. Furthermore, only components A, B and C (but not D) contain the those subgroup elements. In addition, for each type-S signature $\{A,B,C,D\}_{i=1}^{n+1}$ given to the adversary within the same simulation, there exists a random value $y' \in_R \mathbb{Z}_N$ such that

$$\begin{aligned}
\Psi_2(A_{n+1})\Psi_3(A_{n+1}) &= \mathfrak{e}^{y'\mathfrak{a}}\\
\Psi_2(B_{n+1})\Psi_3(B_{n+1}) &= \mathfrak{e}^{y'}\\
\Psi_2(C_{n+1})\Psi_3(C_{n+1}) &= \mathfrak{e}^{y'\mathfrak{b}}\\
\Psi_2(D_{n+1})\Psi_3(D_{n+1}) &= 1_{\mathbb{G}}
\end{aligned}$$

and that for $i = 1$ to n:

$$\begin{aligned}\Psi_2(A_i)\Psi_3(A_i) &= 1_{\mathbb{G}}\\ \Psi_2(B_i)\Psi_3(B_i) &= 1_{\mathbb{G}}\\ \Psi_2(C_i)\Psi_3(C_i) &= 1_{\mathbb{G}}\\ \Psi_2(D_i)\Psi_3(D_i) &= 1_{\mathbb{G}}\end{aligned}$$

In other words, all type-S signatures given to the adversary in a game contain different elements in subgroup $\mathbb{G}_{p_2}$ and $\mathbb{G}_{p_3}$ in signature component A_{n+1}, B_{n+1}, C_{n+1}. All other signature components are from $\mathbb{G}_{p_1}$.

Part I: Assume the adversary ($\mathcal{A}$) outputs a forgery of type-S, we show how to construct a simulator ($\mathcal{S}$) that breaks Assumption 1.

- *Setup.* $\mathcal{S}$ is given (g, X_3, T) and its task is to determine if $\Psi_2(T) = 1$ or not. In fact, we only require a weaker version of assumption 1 when component X_3 is not necessary. $\mathcal{S}$ randomly picks $\alpha, a, b, c, d \in_R \mathbb{Z}_N$, computes $h = g^a, u = g^b, v = g^c, w = g^d$, chooses two hash functions H_0, H_1 and gives
$$\mathsf{param} = \{N, g, h, u, v, w, e(g,g)^\alpha, H_0, H_1\}$$
to $\mathcal{A}$.
- *Query.* $\mathcal{S}$ is in possession of the master secret key (α) and can thus answer all the queries correctly.
- *Forgery.* Finally, $\mathcal{A}$ outputs a signature $(\{A_i, B_i, C_i, D_i\}_{i=1}^{n+1})$ on ring $\mathcal{L}$ and message m such that $|\mathcal{L}| = n$. $\mathcal{S}$ first computes $\mathsf{ID}_{n+1} = H_1(m, \mathcal{L})$ and $\mathsf{ID}_i = H_0(ID_i)$. We have, for any $s, t_1, \ldots, t_{n+1} \in_R \mathbb{Z}_N$, that
$$\prod_{i=1}^{n+1} \frac{e\Big(g^s, A_i\Big) \cdot e\Big(g^{t_i}, C_i\Big)}{e\Big(w^s v^{t_i}, B_i\Big) \cdot e\Big((u^{\mathsf{ID}_i} h)^{t_i}, D_i\Big)} = e(g,g)^{\alpha s}$$
- Since $\mathcal{A}$ outputs a type-S forgery, there exists an index (j) such that $A_j \neq \Psi_1(A_j)$ or $B_j \neq \Psi_1(B_j)$. We show how it can be used to test if T contains a component in $\mathbb{G}_{p_2}$. Note that values p_2 and p_3 are completely hidden from the view of $\mathcal{A}$. That is, $\mathcal{A}$ cannot tell whether $\mathcal{S}$ is trying to use $\mathcal{A}$ to test if $\Psi_2(T) = 1$ or $\Psi_3(T) = 1$. Without loss of generality, we assume either A_j or B_j contains an element in $\mathbb{G}_{p_2}$.
- $\mathcal{S}$ checks if
$$\prod_{i=1}^{n+1} \frac{e\Big(T, A_i\Big) \cdot e\Big(g^{t_i}, C_i\Big)}{e\Big(T^d v^{t_i}, B_i\Big) \cdot e\Big((u^{\mathsf{ID}_i} h)^{t_i}, D_i\Big)} \stackrel{?}{=} e(g,T)^\alpha$$
If $T \in \mathbb{G}_{p_1}$, there exists s such that $T = g^s$ and the above equation holds. Otherwise, there exists s, k such that $T = g^s g_2^k$ where g_2 is a generator of $\mathbb{G}_{p_2}$. In that case the equation holds if and only if
$$\left(\frac{e(g_2, \prod_{i=1}^{n+1} \Psi_2(A_i))}{e(g_2^{d \bmod p_2}, \prod_{i=1}^{n+1} \Psi_2(B_i))}\right)^k = 1$$
This happens with negligible probability since the value $d \bmod p_2$ is information theoretically hidden from $\mathcal{A}$. All $\mathcal{A}$ can infer that d is $d \bmod p_1$ which is unrelated to $d \bmod p_2$.

■

Part II: In this part of the proof, we assume the forgery is of type-N. Furthermore, we assume $\mathcal{A}$ makes a total of k extract and signature queries. Recall that the proof is done via the game-hopping technique and the sequence of the game is recalled below.

$$\mathsf{Game}_0, \ldots, \mathsf{Game}_k$$

Below we first show that $\mathcal{A}$ is restricted in Game_ι for $\iota = 0$ to k under Assumptions 3 and 4. Assume, on the contrary, $\mathcal{A}$ produces two values ID and ID' such that $\mathsf{ID} \neq \mathsf{ID}'$ and $\mathsf{ID} = \mathsf{ID}' \bmod p_3$. ID can be $H_0(ID)$ or $H_1(m, \mathcal{L})$ for some identity ID or ring of identities $\mathcal{L}$ and message m.

Let $P = \gcd(\mathsf{ID} - \mathsf{ID}', N)$. P will be a non-trivial factor of N and a multiple of p_3. In other words, $P \in \{p_3, p_1p_3, p_2p_3\}$. Denote by Q the value of N/P. We consider two cases and describe a reduction for each of them.

1) $(P, Q) = (p_3, p_1p_2) \vee (p_2p_3, p_1)$. In this case we construct a simulator ($\mathcal{S}$) that breaks assumption 3 as follows.
   - *Setup.* $\mathcal{S}$ receives (g, g_2, X_1X_3, T) as input and its goal is to tell if $\Psi_3(T) = 1$. $\mathcal{S}$ randomly chooses $\alpha, a, b, c, d \in_R \mathbb{Z}_N$, two hash functions H_0, H_1 and gives $\mathsf{param} = \{N, g, h = g^a, u = g^b, v = g^c, w = g^d, e(g,g)^\alpha, H_0, H_1\}$ to $\mathcal{A}$.
   - *Queries.* For the jth query such that $j > \iota$, $\mathcal{S}$ uses master secret key α to create a key or signature of type-N. For $j \leq \iota$, $\mathcal{S}$ computes a type-S key by randomly generating $y, r \in_R \mathbb{Z}_N$ and
   $$\begin{aligned} A &= g^\alpha (X_1X_3g_2)^{dy}, & B &= (X_1X_3g_2)^y,\\ C &= (X_1X_3g_2)^{cy}(u^{\mathsf{ID}}h)^r, & D &= g^r.\end{aligned}$$
   Likewise, a type-S signature can be created by transforming a type-N signature $(\{A_i, B_i, C_i, D_i\}_{i=1}^{n+1})$ as follows using $y'_{n+1} \in_R \mathbb{Z}_N$:
   $$\begin{aligned} A_{n+1} &:= A_{n+1}(X_1X_3g_2)^{dy'_{n+1}},\\ B_{n+1} &:= B_{n+1}(X_1X_3g_2)^{y'_{n+1}},\\ C_{n+1} &:= C_{n+1}(X_1X_3g_2)^{cy'_{n+1}},\\ D_{n+1} &:= D_{n+1}.\end{aligned}$$
   This is a signature of type-S. Note that only components A_{n+1}, B_{n+1}, C_{n+1} contain elements in $\mathbb{G}_{p_2}$ and $\mathbb{G}_{p_3}$ as will be the case in the subsequent proof.
   - *Output.* $\mathcal{S}$ first checks if it is the case that $(P, Q) = (p_3, p_1p_2) \vee (p_2p_3, p_1)$ via testing if $g^Q = 1$ and $(X_1X_3)^Q \neq 1$. This check ensures that $p_1 \mid Q$ and $p_3 \nmid Q$. Finally, $\mathcal{S}$ tests if $T^Q = 1$ to tell if $\Psi_3(T) = 1$ or not.
2) $(P, Q) = (p_1p_3, p_2)$. In this case, we construct a simulator ($\mathcal{S}$) that breaks Assumption 4 as follows.
   - *Setup.* $\mathcal{S}$ receives $(g, g_3, X_1X_2, Y_2Y_3, T)$ as input and its goal is to tell if $\Psi_2(T) = 1$. $\mathcal{S}$ randomly chooses $\alpha, a, b, c, d \in_R \mathbb{Z}_N$, two hash functions H_0, H_1 and gives $\mathsf{param} = \{N, g, h = g^a, u = g^b, v = g^c, w = g^d, e(g,g)^\alpha, H_0, H_1\}$ to $\mathcal{A}$. $\mathcal{S}$ keeps random value $\psi \in_R \mathbb{Z}_N$ secret.
   - *Queries.* For the jth query such that $j > \iota$, $\mathcal{S}$ uses master secret key α to create a key or signature of

type-N. For $j \leq \iota$, $\mathcal{S}$ computes a type-S key by randomly generating $y, r \in_R \mathbb{Z}_N$ and

$$\begin{aligned} A &= g^{\alpha} w^{y} (Y_2 Y_3)^{\psi y}, \\ B &= (g Y_2 Y_3)^{y}, \\ C &= v^{y} (Y_2 Y_3)^{cy} (u^{\mathsf{ID}} h)^{r}, \\ D &= g^{r}. \end{aligned}$$

Likewise, a type-S signature can be created by transforming a type-N signature ($\{A_i, B_i, C_i, D_i\}_{i=1}^{n+1}$) as follows, using $y'_{n+1} \in_R \mathbb{Z}_N$:

$$\begin{aligned} A_{n+1} &\coloneqq A_{n+1} (Y_2 Y_3)^{\psi y'_{n+1}}, \\ B_{n+1} &\coloneqq B_{n+1} (Y_2 Y_3)^{y'_{n+1}}, \\ C_{n+1} &\coloneqq C_{n+1} (Y_2 Y_3)^{c y'_{n+1}}, \\ D_{n+1} &\coloneqq D_{n+1}. \end{aligned}$$

This is a signature of type-S. Note that only components A_{n+1}, B_{n+1}, C_{n+1} contain elements in $\mathbb{G}_{p_2}$ and $\mathbb{G}_{p_3}$ as will be the case in the subsequent proof.

- *Output.* $\mathcal{S}$ first checks if it is the case that $(P, Q) = (p_1 p_3, p_2)$ via testing if $g^{Q} \neq 1$ and $g_3^{Q} \neq 1$. This check ensures that $p_1 \nmid Q$ and $p_3 \nmid Q$. Since $PQ = N$ and both P and Q are non-trivial factors of N, it implies $P = p_1 p_3$. $\mathcal{S}$ tests if $T^{P} = 1$ to tell if $\Psi_2(T) = 1$ or not.

As a second step of the proof, we need to show that the behavior of a restricted adversary in $\mathsf{Game}_{\iota-1}$ and Game_{ι} is the same for $\iota = 1$ to k.

We state the following oracle lemma developed and proven in [41]. Let $\mathcal{O}_0$ and $\mathcal{O}_3$ be two oracles whose specifications are as follows[7]

- *Init.* Both $\mathcal{O}_0$ and $\mathcal{O}_3$ initialize themselves as follows. It picks random elements $g, u, v, w \in_R \mathbb{G}_{p_1}$, $g_2 \in_R \mathbb{G}_{p_2}$, $g_3 \in_R \mathbb{G}_{p_3}$ and random exponents $\psi, \sigma, a', b', s, \delta, y \in_R \mathbb{Z}_N$ and outputs $\{N, g, h, u, v, w, g^{s} g_2^{\gamma}, w^{y} (g_2 g_3)^{y\psi}, g^{y} (g_2 g_3)^{y}, v^{y} (g_2 g_3)^{y\sigma}\}$
- Both oracles support two types of queries:
  1) *CT-Query.* On input $I \in \mathbb{Z}_N$, both oracles randomly choose $t \in_R \mathbb{Z}_N$ and return
  $$\{w^{s} g_2^{\delta} v^{t}, g^{t}, (u^{I} h)^{t}\}$$
  2) *CK-Query.* On input $I \in \mathbb{Z}_N$,
  $\mathcal{O}_0$ randomly chooses $r, y' \in_R \mathbb{Z}_N$ and returns
  $$\{w^{y'}, g^{y'}, v^{y'} (u^{I} h)^{r}, g^{r}\}$$
  $\mathcal{O}_3$ randomly chooses $r, y' \in_R \mathbb{Z}_N$ and returns
  $$\{w^{y'} (g_2 g_3)^{y'\psi}, g^{y'} (g_2 g_3)^{y'}, v^{y'} (g_2 g_3)^{y'\sigma} (u^{I} h)^{r}, g^{r}\}$$

Let $\mathcal{D}$ be a PPT algorithm that is given access to an oracle $\mathcal{O} \in \{\mathcal{O}_0, \mathcal{O}_3\}$ and $\mathcal{D}$ makes only one CK-Query and an arbitrary number, say q, of CT-Query to $\mathcal{O}$. Let I be the input of the CK-Query and I_i^* for $i = 1$ to q be the input of the CT-Query. Further suppose $I \neq I_i^* \bmod p_3$ for all i. The oracle lemma states that under Assumption 3 and 4, $\mathcal{D}$ cannot distinguish if it is interacting with $\mathcal{O}_0$ or $\mathcal{O}_3$.

[7]The oracles are labeled as 0 and 3 to be consistent with [41].

Assume there exists $\mathcal{A}$ that behaves differently in $\mathsf{Game}_{\iota-1}$ and Game_{ι}, we show how to construct a simulator ($\mathcal{S}$) that distinguishes oracle $\mathcal{O}_0$ and $\mathcal{O}_3$, thus either breaking Assumption 3 or 4.

- *Setup.* $\mathcal{S}$ is given an oracle $\mathcal{O}$ and its task is to determine if $\mathcal{O}$ is $\mathcal{O}_0$ or $\mathcal{O}_3$. $\mathcal{S}$ receives from $\mathcal{O}$ the following elements:

  $$\{N, g, h, u, v, w, g^{s} g_2^{\gamma}, w^{y} (g_2 g_3)^{y\psi}, g^{y} (g_2 g_3)^{y}, v^{y} (g_2 g_3)^{y\sigma}\}.$$

  $\mathcal{S}$ randomly picks $\alpha \in_R \mathbb{Z}_N$, two hash functions H_0, H_1 and gives
  $$\mathsf{param} = \{N, g, h, u, v, w, e(g, g)^{\alpha}, H_0, H_1\}$$
  to $\mathcal{A}$.
- *Query.* $\mathcal{S}$ is required to return a key or a signature of type-S or type-N depending on value j.
  - For the jth query such that $j > \iota$, $\mathcal{S}$ uses master secret key α to create a key or a signature of type-N.
  - For $j < \iota$ and the jth query being a key query for identity ID such that $\mathsf{ID} = H_0(ID)$, $\mathcal{S}$ computes a type-S key by randomly generating $y', r \in_R \mathbb{Z}_N$ and
  $$\begin{aligned} A &= g^{\alpha} (w^{y} (g_2 g_3)^{\psi y})^{y'}, & B &= (g^{y} (g_2 g_3)^{y})^{y'}, \\ C &= (v^{y} (g_2 g_3)^{y\sigma})^{y'} (u^{\mathsf{ID}} h)^{r}, & D &= g^{r}. \end{aligned}$$
  - For $j < \iota$ and the jth query being a signature query for ring $\mathcal{L} = \{ID_1, \ldots, ID_n\}$ and message m such that $\mathsf{ID}_i = H_0(ID_i)$ for $i = 1$ to n and $\mathsf{ID}_{n+1} = H_1(m, \mathcal{L})$, $\mathcal{S}$ randomly picks $\lambda_1, \ldots, \lambda_{n+1} \in_R \mathbb{Z}_N$ subject to the constraint that $\sum_{i=1}^{n+1} \lambda_i = \alpha$. It also randomly chooses $y_1, \ldots, y_n, y', r_1, \ldots, r_{n+1} \in_R \mathbb{Z}_N$ and computes a type-S signature ($\{A_i, B_i, C_i, D_i\}_{i=1}^{n+1}$) as follows:
  $$\begin{aligned} A_i &= g^{\lambda_i} w^{y_i}, & B_i &= g^{y_i}, \\ C_i &= v^{y_i} (u^{\mathsf{ID}_i} h)^{r_i}, & D_i &= g^{r_i} \qquad \text{for } i = 1 \text{ to } n \end{aligned}$$
  $$\begin{aligned} A_{n+1} &= g^{\lambda_{n+1}} (w^{y} (g_2 g_3)^{y\psi})^{y'}, \\ B_{n+1} &= (g^{y} (g_2 g_3)^{y})^{y'}, \\ C_{n+1} &= (v^{y} (g_2 g_3)^{y\sigma})^{y'}, \\ D_{n+1} &= g^{r_{n+1}}. \end{aligned}$$
  - For $j = \iota$ and the jth query being a key query for identity ID such that $\mathsf{ID} = H_0(ID)$, $\mathcal{S}$ issues a CK-Query to $\mathcal{O}$ on input ID and obtains four elements denoted as (T_0, T_1, T_2, T_3). $\mathcal{S}$ computes a key by
  $$\begin{aligned} A &= g^{\alpha} T_0, & B &= T_1, \\ C &= T_2, & D &= T_3. \end{aligned}$$
  - For $j = \iota$ and the jth query being a signature query for ring $\mathcal{L} = \{ID_1, \ldots, ID_n\}$ and message m such that $\mathsf{ID}_i = H_0(ID_i)$ for $i = 1$ to n and $\mathsf{ID}_{n+1} = H_1(m, \mathcal{L})$, $\mathcal{S}$ issues a CK-Query to $\mathcal{O}$ on input ID_{n+1} and obtains four elements denoted as (T_0, T_1, T_2, T_3). $\mathcal{S}$ randomly picks $\lambda_1, \ldots, \lambda_{n+1} \in_R \mathbb{Z}_N$ subject to the constraint that $\sum_{i=1}^{n+1} \lambda_i = \alpha$. It also randomly chooses $y_1, \ldots, y_n, r_1, \ldots, r_n \in_R \mathbb{Z}_N$ and computes
  $$\begin{aligned} A_i &= g^{\lambda_i} w^{y_i}, & B_i &= g^{y_i}, \\ C_i &= v^{y_i} (u^{\mathsf{ID}_i} h)^{r_i}, & D_i &= g^{r_i} \qquad \text{for } i = 1 \text{ to } n \end{aligned}$$

$$A_{n+1} = g^{\lambda_{n+1}} T_0, \qquad B_{n+1} = T_1,$$
$$C_{n+1} = T_2, \qquad D_{n+1} = T_3.$$

- *Forgery.* Finally, $\mathcal{A}$ outputs a signature $(\{A_i, B_i, C_i, D_i\}_{i=1}^{n+1})$ on ring $\mathcal{L}$ and message m such that $|\mathcal{L}| = n$. $\mathcal{S}$ first computes $\mathsf{ID}_{n+1} = H_1(m, \mathcal{L})$ and $\mathsf{ID}_i = H_0(ID_i)$. We have, for any $s, t_1, \ldots, t_{n+1} \in_R \mathbb{Z}_N$, that

$$\prod_{i=1}^{n+1} \frac{e\Big(g^s, A_i\Big) \cdot e\Big(g^{t_i}, C_i\Big)}{e\Big(w^s v^{t_i}, B_i\Big) \cdot e\Big((u^{\mathsf{ID}_i} h)^{t_i}, D_i\Big)} = e(g,g)^{\alpha s}$$

- *Output.* Note that if $\mathcal{O} = \mathcal{O}_0$, $\mathcal{S}$ is playing $\mathsf{Game}_{\iota-1}$ with $\mathcal{A}$. Otherwise, $\mathcal{S}$ is playing Game_{ι} with $\mathcal{A}$. Any difference in $\mathcal{A}$'s success probability would lead to $\mathcal{S}$ distinguishing the oracles. However, it is possible that while $\mathcal{A}$'s success probability remains the same, it changes its forgery type from type-N to type-S. Thus, $\mathcal{S}$ has to detect if $\mathcal{A}$ is still outputting the type-N forgery as well. To do this, $\mathcal{S}$ issues $n+1$ CT-Query to $\mathcal{O}$ on input ID_i for $i = 1$ to $n+1$. Due to the setting of the game, none of these ID_i will have been input to the CK-Query since $\mathcal{A}$ is not allowed to query any of the keys from the ring of the forged ring signature. At the same time, due to the collision resistant property, ID_{n+1} has not been input to CK-Query as well. Let $T_{1,i}, T_{2,i}, T_{3,i}$ be the elements received by $\mathcal{S}$ from $\mathcal{O}$ on the CT-Query of ID_i for $i = 1$ to $n+1$. $\mathcal{S}$ checks if

$$\prod_{i=1}^{n+1} \frac{e\Big(g^s g_2^{\gamma}, A_i\Big) \cdot e\Big(T_{2,i}, C_i\Big)}{e\Big(T_{1,i}, B_i\Big) \cdot e\Big(T_{3,i}, D_i\Big)} \stackrel{?}{=} e(g, g^s g_2^{\gamma})^{\alpha}$$

  If the forgery is of type-N, the above equation holds. Otherwise, the above equation will not hold since the $\mathbb{G}_{p_2}$ component in the $T_{1,i}, T_{2,i}, T_{3,i}$ is never shown to the adversary. Thus, if $\mathcal{A}$ behalves differently in $\mathsf{Game}_{\iota-1}$ and Game_{ι}, $\mathcal{S}$ can use $\mathcal{A}$ to distinguish $\mathcal{O}_0$ from $\mathcal{O}_3$. This implies breaking Assumption 3 or 4 according to the oracle lemma.

Finally, we present a reduction of $\mathcal{A}$ that produces a type-N forgery in Game_k to simulator $\mathcal{S}$ that breaks Assumption 2.

- *Setup.* $\mathcal{S}$ is given $(g, g_2, g_3, g^{\alpha} X_2, g^s Y_2, T)$ and its task is to determine if $T = e(g,g)^{\alpha s}$ or not. $\mathcal{S}$ randomly picks $a, b, c, d \in_R \mathbb{Z}_N$, $h = g^b, u = g^a, v = g^c, w = g^d \in_R \mathbb{G}_{p_1}$, two hash functions H_0, H_1 and gives

$$\mathsf{param} = \{N, g, h, u, v, w, e(g, g^{\alpha} X_2), H_0, H_1\}$$

  to $\mathcal{A}$. Note that $\mathcal{S}$ does not know master secret key α.
- *Extract Query.* To answer an extract query on identity ID such that $\mathsf{ID} = H_0(ID)$, $\mathcal{S}$ chooses $y, r, f \in_R \mathbb{Z}_N$ and computes

$$\begin{aligned} A &= (g^{\alpha} X_2)^{d+1} w^y (g_2 g_3)^{f(d+1)}, \\ B &= g^{\alpha} X_2 g^y (g_2 g_3)^f, \\ C &= (g^{\alpha} X_2)^c v^y (u^{\mathsf{ID}} h)^r (g_2 g_3)^{fc}, \\ D &= g^r. \end{aligned}$$

  This is a key of type-S.
- *Signature Query.* To answer a signature query on ring $\mathcal{L} = \{ID_1, \ldots, ID_n\}$ on message m such that $\mathsf{ID}_i = H_0(ID_i)$ for $i = 1$ to n and $\mathsf{ID}_{n+1} = H_1(m, \mathcal{L})$, $\mathcal{S}$ chooses $f, \lambda_1, r_1, y_1, \ldots, \lambda_n, y_n, r_n, \lambda'_{n+1}, y'_{n+1}, r_{n+1} \in_R \mathbb{Z}_N$ subject to the constraint that $\lambda_1 + \ldots + \lambda_n + \lambda'_{n+1} = 0$ and computes for $i = 1$ to n:

$$\begin{aligned} A_i &= g^{\lambda_i} w^{y_i}, & B_i &= g^{y_i}, \\ C_i &= v^{y_i} (u^{\mathsf{ID}_i} h)^{r_i}, & D &= g^{r_i}. \end{aligned}$$

  $\mathcal{S}$ then computes the following:

$$\begin{aligned} A_{n+1} &= (g^{\alpha} X_2)^{d+1} g^{\lambda'_{n+1}} w^{y'_{n+1}} (g_2 g_3)^{f(d+1)}, \\ B_{n+1} &= g^{\alpha} X_2 g^{y'_{n+1}} (g_2 g_3)^f, \\ C_{n+1} &= (g^{\alpha} X_2)^c v^{y'_{n+1}} (u^{\mathsf{ID}_{n+1}} h)^r (g_2 g_3)^{fc}, \\ D_{n+1} &= g^{r_{n+1}}. \end{aligned}$$

  This is a signature of type-S.
- *Forgery.* Finally, $\mathcal{A}$ outputs a signature $(\{A_i, B_i, C_i, D_i\}_{i=1}^{n+1})$ on ring $\mathcal{L}$ and message m such that $|\mathcal{L}| = n$. $\mathcal{S}$ first computes $\mathsf{ID}_{n+1} = H_1(m, \mathcal{L})$ and $\mathsf{ID}_i = H_0(ID_i)$. We have, for any $s, t_1, \ldots, t_{n+1} \in_R \mathbb{Z}_N$, that

$$\prod_{i=1}^{n+1} \frac{e\Big(g^s, A_i\Big) \cdot e\Big(g^{t_i}, C_i\Big)}{e\Big(w^s v^{t_i}, B_i\Big) \cdot e\Big((u^{\mathsf{ID}_i} h)^{t_i}, D_i\Big)} = e(g,g)^{\alpha s}$$

- $\mathcal{S}$ randomly generates $t_1, \ldots t_{n+1} \in_R \mathbb{Z}_N$ and computes

$$e(g,g)^{\alpha s} := \prod_{i=1}^{n+1} \frac{e\Big(g^s Y_2, A_i\Big) \cdot e\Big(g^{t_i}, C_i\Big)}{e\Big((g^s Y_2)^d v^{t_i}, B_i\Big) \cdot e\Big((u^{\mathsf{ID}_i} h)^{t_i}, D_i\Big)}$$

  This is the case since $A_i = \Psi_1(A_i)$ and $B_i = \Psi_1(B_i)$. $\mathcal{S}$ can then test if $T = e(g,g)^{\alpha s}$ or not and break Assumption 2.

■

Theorem 2: Our scheme satisfies definition 6 (i.e. our scheme is anonymous).

Proof: The proof is straightforward. The challenger constructs the master key as well as signing keys SK_{ID_0} and SK_{ID_1} for identities ID_0 and ID_1 following the Setup and Extract algorithm based on randomness $\omega_{Setup}, \omega_0, \omega_1$ specified by the adversary.

For any challenge signature $\sigma := \{A_i, B_i, C_i, D_i\}_{i=1}^{n+1}$ created using SK_{ID_b} on message m and ring $\mathcal{L}$, there exists values $\mathcal{R}_0 := \{\lambda_{i,0}, y_{i,0}, r_{i,0}\}_{i=1}^{n+1}$ and $\mathcal{R}_1 := \{\lambda_{i,1}, y_{i,1}, r_{i,1}\}_{i=1}^{n+1}$ such that σ is created from signing key SK_{ID_0} using randomness $\mathcal{R}_0$ or SK_{ID_1} using randomness $\mathcal{R}_1$. In addition, $\mathcal{R}_0$ and $\mathcal{R}_1$ have identical distributions. Thus, even computationally unbounded adversary cannot identify the actual signer with probability better than random guessing. ■

VI. Conclusion

In this paper, we presented a secure identity-based ring signature scheme in the standard model. Our scheme is provably fully secure in the standard model without relying on any interactive assumptions and enjoy unconditional anonymity. Further, we also observed the relationship between HIBE and identity-based ring signature schemes.

The size of our signature grows linearly with the number of users in the ring. Constructing constant size (or at least, sub-linear size) ID-based ring signatures in the standard model remains an open problem. Another interesting open problem is to improve the efficiency of our scheme. For example, to construct a scheme with the same functionality and security as ours while working in the prime order pairing setting.

REFERENCES

[1] M. Abe, M. Ohkubo, and K. Suzuki. 1-out-of-n Signatures from a Variety of Keys. In *ASIACRYPT 2002*, volume 2501 of *Lecture Notes in Computer Science*, pages 415–432. Springer, 2002.

[2] N. Attrapadung and B. Libert. Functional encryption for public-attribute inner products: Achieving constant-size ciphertexts with adaptive security or support for negation. *Journal of Mathematical Cryptology*, 5(2):115–158, 2012.

[3] M. H. Au, J. K. Liu, T. H. Yuen, and D. S. Wong. ID-Based Ring Signature Scheme Secure in the Standard Model. In *IWSEC*, volume 4266 of *Lecture Notes in Computer Science*, pages 1–16. Springer, 2006.

[4] M. Bellare, A. Boldyreva, and A. Palacio. An Uninstantiable Random-Oracle-Model Scheme for a Hybrid-Encryption Problem. In C. Cachin and J. Camenisch, editors, *EUROCRYPT*, volume 3027 of *Lecture Notes in Computer Science*, pages 171–188. Springer, 2004.

[5] M. Bellare, D. Micciancio, and B. Warinschi. Foundations of Group Signatures: Formal Definitions, Simplified Requirements, and a Construction Based on General Assumptions. In *EUROCRYPT 2003*, volume 2656 of *Lecture Notes in Computer Science*, pages 614–629. Springer, 2003.

[6] M. Bellare, C. Namprempre, and G. Neven. Security Proofs for Identity-Based Identification and Signature Schemes. In *EUROCRYPT*, volume 3027 of *Lecture Notes in Computer Science*, pages 268–286. Springer, 2004.

[7] A. Bender, J. Katz, and R. Morselli. Ring Signatures: Stronger Definitions, and Constructions Without Random Oracles. In *TCC*, volume 3876 of *Lecture Notes in Computer Science*, pages 60–79. Springer, 2006.

[8] D. Boneh and X. Boyen. Efficient Selective-ID Secure Identity-Based Encryption Without Random Oracles. In *EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 223–238. Springer, 2004.

[9] D. Boneh, X. Boyen, and E.-J. Goh. Hierarchical Identity Based Encryption with Constant Size Ciphertext. In *EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 440–456. Springer, 2005.

[10] D. Boneh and M. Franklin. Identity-Based Encryption from the Weil Pairing. In *CRYPTO 2001*, volume 2139 of *Lecture Notes in Computer Science*, pages 213–229. Springer, 2001.

[11] D. Boneh and M. K. Franklin. Identity-Based Encryption from the Weil Pairing. *SIAM J. Comput.*, 32(3):586–615, 2003.

[12] D. Boneh, E.-J. Goh, and K. Nissim. Evaluating 2-DNF Formulas on Ciphertexts. In *TCC*, volume 3378 of *Lecture Notes in Computer Science*, pages 325–341. Springer, 2005.

[13] D. Boneh and M. Hamburg. Generalized identity based and broadcast encryption schemes. In *ASIACRYPT*, volume 5350 of *Lecture Notes in Computer Science*, pages 455–470. Springer, 2008.

[14] X. Boyen. Mesh signatures. In M. Naor, editor, *EUROCRYPT*, volume 4515 of *Lecture Notes in Computer Science*, pages 210–227. Springer, 2007.

[15] Z. Brakerski and Y. T. Kalai. A Framework for Efficient Signatures, Ring Signatures and Identity Based Encryption in the Standard Model. *IACR Cryptology ePrint Archive*, 2010:86, 2010.

[16] N. Chandran, J. Groth, and A. Sahai. Ring Signatures of Sub-linear Size Without Random Oracles. In *ICALP*, volume 4596 of *Lecture Notes in Computer Science*, pages 423–434. Springer, 2007.

[17] B. K. Chaurasia and S. Verma. Conditional Privacy through Ring Signature in Vehicular Ad-hoc Networks. *Transactions on Computational Science*, 6750:147–156, 2011.

[18] Y. Chen, W. Susilo, and Y. Mu. Convertible Identity-Based Anonymous Designated Ring Signatures. *IJSN*, 1(3/4):218–225, 2006.

[19] Y. Chen, W. Susilo, and Y. Mu. Identity-Based Anonymous Designated Ring Signatures. In *IWCMC*, pages 189–194. ACM, 2006.

[20] H.-Y. Chien. Highly Efficient ID-Based Ring Signature from Pairings. In *APSCC*, pages 829–834. IEEE, 2008.

[21] S. S. M. Chow and K.-K. R. Choo. Strongly-Secure Identity-Based Key Agreement and Anonymous Extension. In J. A. Garay, A. K. Lenstra, M. Mambo, and R. Peralta, editors, *ISC*, volume 4779 of *Lecture Notes in Computer Science*, pages 203–220. Springer, 2007.

[22] S. S. M. Chow, L. C. Hui, and S. Yiu. Identity Based Threshold Ring Signature. In *ICISC 2004*, volume 3506 of *Lecture Notes in Computer Science*, pages 218–232. Springer, 2004.

[23] S. S. M. Chow, J. K. Liu, V. K. Wei, and T. H. Yuen. Ring Signatures Without Random Oracles. In *ASIACCS 06*, pages 297–302. ACM Press, 2006.

[24] S. S. M. Chow, J. K. Liu, and D. S. Wong. Robust receipt-free election system with ballot secrecy and verifiability. In *NDSS*. The Internet Society, 2008.

[25] S. S. M. Chow, R. W. C. Lui, L. C. K. Hui, and S.-M. Yiu. Identity Based Ring Signature: Why, How and What Next. In *EuroPKI*, volume 3545 of *Lecture Notes in Computer Science*, pages 144–161. Springer, 2005.

[26] S. S. M. Chow, W. Susilo, and T. H. Yuen. Escrowed linkability of ring signatures and its application. In *VIETCRYPT 2006*, volume 4341 of *Lecture Notes in Computer Science*, pages 175–192. Springer, 2006.

[27] S. S. M. Chow, S.-M. Yiu, and L. C. K. Hui. Efficient Identity Based Ring Signature. In *ACNS 2005*, volume 3531 of *Lecture Notes in Computer Science*, pages 499–512, 2005. Also available at Cryptology ePrint Archive, Report 2004/327.

[28] C.-K. Chu, J. K. Liu, J. W. Wong, Y. Zhao, and J. Zhou. Privacy-preserving Smart Metering with Regional Statistics and Personal Enquiry Services. To appear in ASIACCS 2013, 2013.

[29] A. L. Ferrara, M. Green, S. Hohenberger, and M. Ø. Pedersen. Practical Short Signature Batch Verification. In *CT-RSA*, volume 5473 of *Lecture Notes in Computer Science*, pages 309–324. Springer, 2009.

[30] D. M. Freeman. Converting pairing-based cryptosystems from composite-order groups to prime-order groups. In H. Gilbert, editor, *EUROCRYPT*, volume 6110 of *Lecture Notes in Computer Science*, pages 44–61. Springer, 2010.

[31] C. Gentry. Practical Identity-Based Encryption Without Random Oracles. In *EUROCRYPT 2006*, volume 4404 of *Lecture Notes in Computer Science*, pages 445–464. Springer, 2006.

[32] C. Gentry and A. Silverberg. Hierarchical ID-Based Cryptography. In *ASIACRYPT 2002*, volume 2501 of *Lecture Notes in Computer Science*, pages 548–566. Springer, 2002.

[33] J. Groth and A. Sahai. Efficient non-interactive proof systems for bilinear groups. In N. P. Smart, editor, *EUROCRYPT*, volume 4965 of *Lecture Notes in Computer Science*, pages 415–432. Springer, 2008.

[34] A. Guillevic. Comparing the pairing efficiency over composite-order and prime-order elliptic curves. Cryptology ePrint Archive, Report 2013/218, 2013. http://eprint.iacr.org/.

[35] J. Han, Q. Xu, and G. Chen. Efficient ID-based Threshold Ring Signature scheme. In *EUC (2)*, pages 437–442. IEEE Computer Society, 2008.

[36] J. Herranz. Identity-based Ring Signatures from RSA. *Theor. Comput. Sci.*, 389(1-2):100–117, 2007.

[37] J. Herranz and G. Sáez. New Identity-Based Ring Signature Schemes. In *ICICS*, volume 3269 of *Lecture Notes in Computer Science*, pages 27–39. Springer, 2004.

[38] J. Horwitz and B. Lynn. Toward Hierarchical Identity-Based Encryption. In *EUROCRYPT 2002*, volume 2332 of *Lecture Notes in Computer Science*, pages 466–481. Springer, 2002.

[39] A. B. Lewko. Tools for simulating features of composite order bilinear groups in the prime order setting. In D. Pointcheval and T. Johansson, editors, *EUROCRYPT*, volume 7237 of *Lecture Notes in Computer Science*, pages 318–335. Springer, 2012.

[40] A. B. Lewko and B. Waters. New Techniques for Dual System Encryption and Fully Secure HIBE with Short Ciphertexts. In *TCC*, volume 5978 of *Lecture Notes in Computer Science*, pages 455–479. Springer, 2010.

[41] A. B. Lewko and B. Waters. Unbounded HIBE and Attribute-Based Encryption. In *EUROCRYPT*, volume 6632 of *Lecture Notes in Computer Science*, pages 547–567. Springer, 2011.

[42] C.-Y. Lin and T.-C. Wu. An Identity-based Ring Signature Scheme from Bilinear Pairings. In *AINA (2)*, pages 182–186. IEEE Computer Society, 2004.

[43] J. K. Liu, V. K. Wei, and D. S. Wong. Linkable Spontaneous Anonymous Group Signature for Ad Hoc Groups (Extended Abstract). In *ACISP*, volume 3108 of *Lecture Notes in Computer Science*, pages 325–335. Springer, 2004.

[44] J. K. Liu and D. S. Wong. Linkable Ring Signatures: Security Models and New Schemes (Extended Abstract). In *ICCSA 2005*, volume 3481 of *LNCS*, pages 614–623. Springer-Verlag, 2005.

[45] J. K. Liu, T. H. Yuen, and J. Zhou. Forward secure ring signature without random oracles. In *ICICS*, volume 7043 of *Lecture Notes in Computer Science*, pages 1–14. Springer, 2011.

[46] H. K. Maji, M. Prabhakaran, and M. Rosulek. Attribute-based signatures. In *CT-RSA*, volume 6558 of *Lecture Notes in Computer Science*, pages 376–392. Springer, 2011.

[47] L. Nguyen. Accumulators from Bilinear Pairings and Applications. In *CT-RSA 2005*, volume 3376 of *Lecture Notes in Computer Science*, pages 275–292. Springer, 2005.

[48] T. Okamoto and K. Takashima. Efficient attribute-based signatures for non-monotone predicates in the standard model. In *Public Key Cryptography*, volume 6571 of *Lecture Notes in Computer Science*, pages 35–52. Springer, 2011.

[49] R. L. Rivest, A. Shamir, and Y. Tauman. How to Leak a Secret. In *ASIACRYPT 2001*, volume 2248 of *Lecture Notes in Computer Science*, pages 552–565. Springer, 2001.

[50] S. Schäge and J. Schwenk. A CDH-Based Ring Signature Scheme with Short Signatures and Public Keys. In *Financial Cryptography*, volume 6052 of *Lecture Notes in Computer Science*, pages 129–142. Springer, 2010.

[51] H. Shacham and B. Waters. Efficient Ring Signatures Without Random Oracles. In *Public Key Cryptography*, volume 4450 of *Lecture Notes in Computer Science*, pages 166–180. Springer, 2007.

[52] A. Shamir. Identity-Based Cryptosystems and Signature Schemes. In *CRYPTO 1984*, volume 196 of *LNCS*, pages 47–53, 1984.

[53] N. P. Smart and B. Warinschi. Identity Based Group Signatures from Hierarchical Identity-Based Encryption. In H. Shacham and B. Waters, editors, *Pairing*, volume 5671 of *Lecture Notes in Computer Science*, pages 150–170. Springer, 2009.

[54] P. P. Tsang, M. H. Au, J. K. Liu, W. Susilo, and D. S. Wong. A Suite of Non-pairing ID-Based Threshold Ring Signature Schemes with Different Levels of Anonymity (Extended Abstract). In *ProvSec*, volume 6402 of *Lecture Notes in Computer Science*, pages 166–183. Springer, 2010.

[55] B. Waters. Efficient Identity-Based Encryption Without Random Oracles. In *EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 114–127. Springer, 2005.

[56] B. Waters. Dual System Encryption: Realizing Fully Secure IBE and HIBE under Simple Assumptions. In S. Halevi, editor, *CRYPTO*, volume 5677 of *Lecture Notes in Computer Science*, pages 619–636. Springer, 2009.

[57] J. Xu, Z. Zhang, and D. Feng. A Ring Signature Scheme Using Bilinear Pairings. In *WISA 2004*, volume 3325 of *Lecture Notes in Computer Science*, pages 163–172. Springer, 2004.

[58] T. H. Yuen, J. K. Liu, M. H. Au, W. Susilo, and J. Zhou. Threshold Ring Signature without Random Oracles. In *ASIACCS '11*, pages 261–267. ACM, 2011.

[59] T. H. Yuen, J. K. Liu, X. Huang, M. H. Au, W. Susilo, and J. Zhou. Forward secure attribute-based signatures. In *ICICS*, volume 7618 of *Lecture Notes in Computer Science*, pages 167–177, 2012.

[60] F. Zhang and X. Chen. Cryptanalysis and Improvement of an ID-Based Ad-hoc Anonymous Identification Scheme at CT-RSA 05. Cryptology ePrint Archive, Report 2005/103, 2005. eprint.iacr.org/.

[61] F. Zhang and K. Kim. ID-Based Blind Signature and Ring Signature from Pairings. In *ASIACRYPT 2002*, volume 2501 of *Lecture Notes in Computer Science*, pages 533–547. Springer, 2002.

[62] Y. Zhang, C. J. Xue, D. S. Wong, N. Mamoulis, and S. M. Yiu. Acceleration of composite order bilinear pairing on graphics hardware. In T. W. Chim and T. H. Yuen, editors, *ICICS*, volume 7618 of *Lecture Notes in Computer Science*, pages 341–348. Springer, 2012.

APPENDIX A
ANALYSIS OF THE PROOF OF THE SECOND SCHEME IN [3]

We point out a flaw in the unforgeability proof of the second scheme in [3]. We use the notation as in that paper to avoid any confusion.

Let $n-1$ be the maximum number of users allowed in the ring signature. In the proof of unforgeability, the simluator is given the n-DHI* problem instance $g, g^x, \ldots, g^{x^n}$ and is asked to output $g^{x^{n+1}}$. The secret key of each user contains the following elements:

$$\{a_i, b_i, c_{i,1}, \ldots, c_{i,i-1}, c_{i,i+1}, \ldots, c_{i,n}\}_{i=1}^{n}$$

The flaw appeared in the extraction oracle. The authors in [3] have written down the simulation of the *first part* of a user secret key. The generation of $a_1, b_1, c_{1,2}, \ldots, c_{1,n}$ is correctly simulated. The authors then claimed that

> *The computation for* $(a_i, b_i, c_{i,j})$ *where* $1 \le i \le n-1$ *are similar and hence are omitted.*

However, when we try to generate the second part of the secret key in a *similar* way, we have encountered a problem. The second part of the secret key contains $a_2, b_2, c_{2,1}, c_{2,3}, \ldots, c_{2,n}$. Using the way that the authors used to generate $c_{1,2}, \ldots, c_{1,n}$, we have

$$c_{2,1} = \hat{u}_1^{r_2} = \left(g^{\gamma_1 - x^n}\right)^{\left(\frac{x}{\mathsf{id}-\mathsf{id}_2^*} + \tilde{r}_2\right)}$$

where γ_1, $\tilde{r}_2$ are chosen by the simulator, id is the identity submitted by the adversary to the extraction oracle, id_2^* is one of the challenged identities submitted by the adversary at the beginning and g^{x^n} is an element in the problem instance. The simulator needs to expand the terms in order to compute this element, as it does not have the knowledge of x. After expansion, we have

$$g^{\frac{\gamma_1 x}{\mathsf{id}-\mathsf{id}_2^*} - \frac{x^{n+1}}{\mathsf{id}-\mathsf{id}_2^*} + \gamma_1 \tilde{r}_2 - x^n \tilde{r}_2} = \left(g^x\right)^{\frac{\gamma_1}{\mathsf{id}-\mathsf{id}_2^*}} \cdot \left(g^{x^{n+1}}\right)^{-\frac{1}{\mathsf{id}-\mathsf{id}_2^*}} \cdot g^{\gamma_1 \tilde{r}_2} \cdot \left(g^{x^n}\right)^{-\tilde{r}_2}$$

However, the term $g^{x^{n+1}}$ should be unknown to the simulator (in fact, it is actually the term required to output). That is, this part of the secret key cannot be computed in this way (the *similar* way as in the first part). The authors did not mention how to resolve this obstacle in the proof. We believe this cannot be solved easily or in any trivial way. Thus we conclude that the unforgeability proof of the scheme is incorrect (or at least, incompleted), though we have not found any concrete attack on the scheme itself.