

# Short Linkable Ring Signatures for E-voting, E-cash and Attestation

Patrick P. Tsang and Victor K. Wei

Department of Information Engineering  
The Chinese University of Hong Kong  
Shatin, Hong Kong  
{pktsang3,kwei}@ie.cuhk.edu.hk

**Abstract.** A ring signature scheme can be viewed as a group signature scheme with no anonymity revocation and with simple group setup. A *linkable* ring signature scheme additionally allows anyone to determine if two ring signatures have been signed by the same group member. Recently, Dodis et al. [19] gave the first short (constant-sized) ring signature scheme. We extend it to the first short linkable ring signature scheme, and reduce its security to a new candidate hard problem, the Link Decisional RSA (LD-RSA) Problem. We also extend [19]’s other schemes to a generic linkable ring signature scheme and a generic linkable group signature scheme. We discuss three applications of our schemes. Kiayias and Yung [22] constructed the first e-voting scheme which simultaneously achieves efficient tallying, public verifiability, and write-in capability for a typical voter distribution where only a small portion writes in. We construct an e-voting scheme based on our short linkable ring signature scheme which achieves the same even for all worst-case voter distribution. Direct Anonymous Attestation (DAA) [7] is essentially a ring signature with certain linking properties that can be naturally implemented using linkable ring signature schemes. The construction of an offline anonymous e-cash scheme using linkable ring signature schemes is also discussed.

## 1 Introduction

A *group signature* scheme [16] allows a member to sign messages anonymously on behalf of his group. The group manager is responsible to form the group and assign to the members the ability to sign. However, in the case of a dispute, the identity of a signature’s originator can be revealed (only) by a designated entity.

A *ring signature* scheme [29] can be viewed as a group signature scheme with no anonymity revocation and with simple group setup. Formation of a group is *spontaneous*: diversion group members can be totally unaware of being conscripted to the group. Applications include leaking secrets [29] and anonymous identification/authentication for ad hoc groups [6, 19].

*Linkable ring signatures* [23] are ring signatures, but with added linkability: such signatures allow anyone to determine if they are signed by the same group member (i.e. they are *linked*). If a user signs only once on behalf of a group, he

still enjoys anonymity similar to that in conventional ring signature schemes. If the user signs multiple times, anyone can tell that these signatures have been generated by the same group member. Applications include leaking sequences of secrets and e-voting [23]. Concepts similar to linkability also appeared in one-show credentials [8], linkable group signatures [26, 27], and generating rogue tags in direct anonymous attestations [7].

Early constructions of group signatures, ring signatures, and linkable ring signatures have large signature sizes, usually  $O(n)$  where  $n$  is the group size. Subsequent results incorporating techniques from coalition-resistant one-way accumulators reduced the sizes of state-of-the-art group signatures to a constant independent of group size. Consult [12, 1, 9, 4, 10, 28] for details. Essentially all ring signatures have size  $O(n)$ . Recently, Dodis, et al. [19] constructed the first short (constant-sized) ring signature scheme. In this paper, we extend their technique to construct linkable ring signatures of constant size. We also extend [19]’s generic ring (resp. group) signature constructions to their linkable version.

The constant-size linkable ring signature has many applications, including e-voting, e-cash, and attestations:

*E-Voting.* There are three basic paradigms for cryptographically secure ballot elections: (1) The *blind signature* [14] approach, (2) the *homomorphic encryption* [17] approach, and (3) the *mix-net* [13] approach.

Under the *blind signature* approach, the voters obtain a ballot from the authorities, certified but yet privacy-preserved. This enables them to embed any form of ballot (including write-ins). This approach requires the employment of an anonymous channel between the voter and the tallying authorities to hide the identity of the user at the “ballot casting stage.” Note that universal verifiability is missing and robustness is usually achieved by thresholding the authority.

Under the *homomorphic encryption* approach, the ballots are encrypted and then “compressed” via a homomorphic encryption scheme into a tally. This compression property allows fast tallying, and is what makes this approach attractive. However the drawback is that pure “compressible” homomorphic encryption is not suitable to deal with write-in ballots.

Under the *mix-net* approach, the tallying officials move the ballots between them and permute them in the process while changing their representation (e.g., partially decrypting them). Practical implementations of this approach in its fully robust form is still considered a slow tallying process.

*Offline Anonymous Electronic Cash (E-cash).* Most of the e-cash systems found in the literature makes use of *blind signatures*. In such systems, the users withdraw electronic coins, which consist of numbers generated by users and blindly signed by the bank. Each signature represents a given amount. These coins are then spent in shops which can authenticate them by using the public signature key of the bank. The users retain anonymity in any transaction since the coins they use have been blindly signed. Existing schemes of this category are fruitful, some of the important ones are: [14, 15, 5, 11, 21].

E-cash systems by *group signatures* is a relatively new approach. The idea is as follows: the group members in the group signature scheme forms a group of

users. The bank (acting as the GM) is capable of issuing electronic coins (which are actually the ability to sign) to the users. When a user spends, he/she signs a group signature for the shop. The anonymity inherited from the group signature scheme provides privacy for the users. Examples: [24, 30, 25].

*Direct Anonymous Attestation (DAA)*. In the context of the Trusted Computing Group (TCG)<sup>1</sup>, DAA is a solution to the following problem: The user of such a platform communicates with a verifier who wants to be assured that the user indeed uses a platform containing such a trusted hardware module, i.e., the verifier wants the trusted platform module (TPM) to authenticate itself. However, the user wants her privacy protected and therefore requires that the verifier only learns that she uses a TPM but not which particular one.

The first solution [20] has the drawback of requiring a TTP to be online in every transaction. Also, anonymity is lost when the TTP and the verifier collude. [7] solves the problem by making use of a group signature scheme variant based on the Camenisch-Lysyanskaya group signature scheme [8, 9]. Among other differences from the original scheme, the two crucial ones are (1) disabling anonymity revocation and (2) including a pseudonym in the signatures.

### Contributions.

- We extend the first short ring signature of Dodis, et al.[19] to the first short linkable ring signature, and reduce its security to a set of assumptions including a new candidate hard problem, Link Decisional RSA (LD-RSA) Problem.
- We also extend [19]’s generic ring (resp. group) signature constructions to their linkable version.
- Motivated by Kiayias and Yung [22] which presented the first e-voting scheme that simultaneously achieved efficient tallying, universal verifiability, and write-in capability for typical voter distribution where only a small portion writes in, we discuss that e-voting scheme constructed from linkable ring signatures [23] also achieve the same three properties even for all worst-case voter distributions.
- We discuss efficient implementation of direct anonymous attestation [7] using linkable ring signatures.
- We discuss construction of an offline anonymous e-cash scheme using linkable group signatures.

## 1.1 Paper Organization

The paper is organized as follows: In Sec. 2, we give some preliminaries. Then we define linkable ring signatures and notions of security in Sec. 3. Constructions and their security analysis are presented in Sec. 4. We discuss three applications in Sec. 5. Finally we conclude in Sec. 6.

<sup>1</sup> TCG develops and promotes open industry standard specifications for trusted computing hardware building blocks and software interfaces across multiple platforms, e.g. PC’s, PDA’s, and digital phones. This enables more secure data storage, online commerce transactions, etc, while protecting privacy and individual rights.

## 2 Preliminaries

We review literature results and introduce new terminologies.

**Strong RSA Assumption.** There exists no PPT algorithm which, on input a random  $\lambda$ -bit safe prime  $N$  and a random  $z \in QR(N)$ , returns  $u \in \mathbb{Z}_N^*$  and  $e \in \mathbb{N}$  such that  $e > 1$  and  $u^e = z \pmod{N}$ , with non-negligible probability and in time polynomial in  $\lambda$ .

**Simulation-Sound, Computationally Zero-Knowledge Proof System.** We adopt the definition of this concept from Bellare, et al. [2]. In a nutshell, it is a three-move zero-knowledge proof-of-knowledge system that is *simulation sound*, meaning that oracles specified in the security model can be successfully simulated, and that is *computational zero-knowledge*, meaning no PPT program can distinguish between real world and ideal world. For details, consult [2].

**Accumulator with One-Way Domain.** We adopt this concept introduced in Dodis, et al. [19]. Just a brief summary below. An *accumulator family* is a pair  $(\{F_i\}, \{X_i\})$  where  $F_i$  is a family of functions whose member is s.t.  $f : U_f \times X_i \rightarrow U_f$  and the accumulator family satisfies *efficient generation*, *efficient evaluation*, and *quasi-commutativity*. The last property means  $f(f(u, x_1), x_2) = f(f(u, x_2), x_1)$ . An accumulator is *collision resistant* if it is rare to have two different sequences accumulated to the same value.

An *accumulator with one-way domain* is a quadruple  $(\{F_i\}, \{X_i\}, \{Z_i\}, \{\mathcal{R}_i\})$  where  $(\{F_i\}, \{X_i\})$  is a collision-resistant accumulator, each  $\mathcal{R}_i$  is a relation over  $X_i \times Z_i$  satisfying *efficient verification*, *efficient sampling*, and *one-wayness*. For details consult the original paper [19].

In this paper, we need the following new candidate hard problem:

**Definition 1 (Link Decisional RSA (LD-RSA) Problem).** Let  $N = pq = (2p' + 1)(2q' + 1)$  be a sufficiently large safe prime. Let  $g \in QR(N)$ , with  $\text{order}(g) = p'q'$ . Let  $e_1$  and  $e_2$  be  $(\lambda + \epsilon)$ -bit primes,  $x = 2e_1e_2 + 1$ ,  $\tilde{y} = g^{e_1 + e_2}$ ,  $\tilde{y}'$  is a randomly generated element of  $QR(N)$ . Assume  $(x, \tilde{y})$  and  $(x, \tilde{y}')$  are picked with equal probability and given to a distinguisher. The Link Decisional RSA (LD-RSA) Problem is to distinguish the two cases with success probability non-negligibly over  $1/2$ .

In this paper, we have in mind  $\lambda$  being the security parameter, and  $\epsilon$  being a sufficiently small number relative to  $\lambda$ , and  $p', q'$  being at least  $\lambda + 2$  bits long.

We also need the following technical definition.

**Definition 2 (PK-bijectivity).** Let  $\mathcal{R} \subseteq \mathcal{X} \times \mathcal{Y}$  be a one-way efficiently samplable NP-relation. A mapping  $\theta : \mathcal{X} \rightarrow \mathcal{Z}$  is PK-bijjective if

1.  $(x, y) \in \mathcal{R}$  and  $(x', y) \in \mathcal{R}$  implies  $\theta(x) = \theta(x')$ .
2. It is rare to sample  $y_1 \neq y_2 \in \mathcal{Y}$  such that  $\theta(x_1) = \theta(x_2)$  for some  $x_1, x_2$  with  $(x_1, y_1), (x_2, y_2) \in \mathcal{R}$ .

*Remark:* In this paper,  $\mathcal{R}$  typically the *sk-pk* relation of the RSA system:  $\mathcal{R} = \{(p, q, n) | n = pq\}$ . The choice  $\theta = g^p$  is not PK-bijjective while  $\theta = g^{p+q}$

is. The choice  $g^{pq}$  is PK-bijective, but it will give away anonymity as we shall see later. The choice  $\mathcal{R} = \{(p, n) | p \text{ divides } n\}$  and  $\theta(sk) = g^p$  is not PK-bijective because  $n$  has two prime factors  $p$  and  $q$  and  $(p, n), (q, n) \in \mathcal{R}$  and  $\theta(p) \neq \theta(q)$ .

The above points out some common errors in designing linkable ring signatures that we have experienced. A related concept is to require that, given a random sample  $y_1$ , it is hard to compute  $y_2$  such that there exist  $x_1, x_2$ , satisfying  $(x_1, y_1), (x_2, y_2) \in \mathcal{R}, \theta(x_1) = \theta(x_2)$ . This stronger concept may be needed in further study of the current topic, but it is not needed in the present paper.

### 3 Security Model

We give our security model and define relevant security notions.

#### 3.1 Syntax

**Linkable Ring Signatures.** A linkable ring signature (LRS) scheme is a tuple  $(\text{Init}, \text{LRKg}, \text{LRSig}, \text{LRVf}, \text{Link})$ .

- $\text{Init}$  takes as input the security parameter  $1^\lambda$ , and outputs system-wide public parameters  $\text{param}$ . Typically,  $\text{param}$  includes an  $sk$ - $pk$  relation  $\mathcal{R}$  which is efficiently samplable, an one-way NP-relation, lengths of keys, ..., etc.
- $\text{LRKg}$  takes as inputs the security parameter  $1^\lambda$ , the group size  $n$ , and returns a tuple  $(gpk, gsk)$ , where  $gpk$  is group public key,  $gmsk$  is group master's secret key, and group secret key  $gsk$  is an  $n$ -vector with  $gsk[i]$  being the secret signing key for player  $i$ ,  $1 \leq i \leq n$ . Often  $gpk$  is also an  $n$  vector with  $(gsk[i], gpk[i]) \in \mathcal{R}$ .
- $\text{LRSig}$  takes inputs group public key  $gpk$ , a secret signing key  $gsk[i]$  and a message  $M$ , returns a linkable ring signature of  $M$ .
- $\text{LRVf}$  takes inputs the group public key  $gpk$ , a message  $M$ , and a candidate signature  $\sigma$  for  $M$ , returns either 1 or 0 for *valid* or *invalid*.
- $\text{Link}$  takes inputs two valid signatures  $\sigma$  and  $\sigma'$ , returns either 1 or 0 for *linked* or *unlinked*. It returns nothing or  $\perp$  if the signatures are not both valid.

**CORRECTNESS.** An LRS scheme is *correct* if (1)  $\text{LRVf}(gpk, M, \text{LRSig}(gpk, gsk[i], M))=1$ , and (2)  $\text{Link}(\text{LRSig}(gpk, gsk[i], M), \text{LRSig}(gpk, gsk[i], M'))=1$  for all  $gpk, gsk, i, M, M'$ . The two checks are sometimes called *verification correctness* and *linking correctness*, resp.

#### 3.2 Notions of Security

Security of LRS schemes has these aspects: unforgeability, anonymity and linkability. The following oracles define the attacker's capabilities.

- $sk_i \leftarrow \text{CO}(pk_i)$ . The *Corruption Oracle*, on input a public key  $pk_i \in \mathcal{Y}$  that is a query output of  $\text{LRKg}$ , returns the corresponding secret key  $sk_i \in \mathcal{X}$ .

- $\sigma' \leftarrow \mathcal{SO}(gpk, s, M)$ . The *Signing Oracle*, on input  $gpk$ , a designated signer  $s$ , returns a valid signature  $\sigma$  which is computationally indistinguishable from one produced by  $\text{LRSig}$  using the real secret key  $gsk[s]$ .

*Remark:* An alternative approach to exclude  $s$  from  $\mathcal{SO}$ 's input and have  $\mathcal{SO}$  randomly select the signer. We do not pursue that alternative here.

**Unforgeability.** Unforgeability for LRS schemes is defined in the following game between the Simulator  $\mathcal{S}$  and the Adversary  $\mathcal{A}$  in which  $\mathcal{A}$  is given access to oracles  $\mathcal{CO}$  and  $\mathcal{SO}$ :

1.  $\mathcal{S}$  generates and gives  $\mathcal{A}$  the system parameters  $\text{param}$ .
2.  $\mathcal{A}$  may query the oracles according to any adaptive strategy.
3.  $\mathcal{A}$  delivers  $gpk, M, \sigma_g$ .

$\mathcal{A}$  wins the game if: (1)  $\text{LRVf}(gpk, M, \sigma) = 1$ , (2) all public keys in  $gpk$  are query outputs of  $\text{LRKg}$ , none has been input to  $\mathcal{CO}$ , and (3)  $\sigma$  is not an output of  $\mathcal{SO}$  on any input containing  $M$ .  $\mathcal{A}$ 's *advantage* is its probability of winning.

**Definition 3 (unforgeability).** An LRS scheme is unforgeable if no PPT adversary  $\mathcal{A}$  has a non-negligible advantage in the game above.

**Anonymity.** Anonymity for LRS schemes is defined in the following game:

#### Game LA

1. (*Initialization Phase*)  $\mathcal{S}$  generates and gives  $\mathcal{A}$  the system parameters  $\text{param}$ .
2. (*Probe-1 Phase*)  $\mathcal{A}$  queries the oracles with arbitrary interleaving.
3. (*Gauntlet Phase*)  $\mathcal{A}$  gives  $\mathcal{S}$   $gpk, s$ , message  $M$ , where  $gpk[s]$  has never been queried to  $\mathcal{CO}$  and has never been the designated signer in any  $\mathcal{SO}$  query. Then  $\mathcal{S}$  flips a fair coin to select  $b \in \{\text{real}, \text{ideal}\}$ . Case  $b = \text{real}$ :  $\mathcal{S}$  queries  $\mathcal{CO}$  with  $gpk[s]$  to obtain  $gsk[s]$  and compute  $\sigma = \text{LRSig}(gpk, gsk[s], M)$ . Case  $b = \text{ideal}$ :  $\mathcal{S}$  computes  $\sigma = \mathcal{SO}(gpk, s, M)$ .
4. (*Probe-2 Phase*)  $\mathcal{A}$  receives  $\sigma$ , queries the oracles adaptively, except that  $gpk[s]$  cannot be queried to  $\mathcal{CO}$  or be designated signer in any  $\mathcal{SO}$  query.
5. (*End Game*)  $\mathcal{A}$  delivers an estimate  $\hat{b} \in \{\text{real}, \text{ideal}\}$  of  $b$ .

$\mathcal{A}$  wins the game if  $\hat{b} = b$ . Its *advantage* is its winning probability minus half.

**Definition 4 (Anonymity).** An LRS scheme is anonymous if for no PPT adversary has a non-negligible advantage in Game LA.

*Remark:* In this paper, the statistical distance between the real world and the ideal world is non-negligible. Therefore, we use a distinguishability test between real and ideal. The other popular approach, distinguishing between two possible signers, is not suitable. Our attacker model is not fully active due to restrictions that the *gauntlet* public key (i.e.  $gpk[s]$  in the Gauntlet Phase) cannot be queried.

**Linkability.** Linkability for LRS schemes is defined in the following game between the Simulator  $\mathcal{S}$  and the Adversary  $\mathcal{A}$ :

1.  $\mathcal{S}$  generates and gives  $\mathcal{A}$  the system parameters  $\text{param}$ .
2.  $\mathcal{A}$  queries the oracles according to any adaptive strategy.
3.  $\mathcal{A}$  delivers  $(gpk_i, M_i, \sigma_i)$ ,  $i=1,2$ .

$\mathcal{A}$  wins the game if (1) all public keys in  $gpk_1 \cup gpk_2$  are query outputs of  $\text{LRKg}$ , and at most one has been queried to  $\mathcal{CO}$ . (2)  $\text{LRVf}(gpk_i, M_i, \sigma_i)=1$ ,  $i=1,2$ . and (3)  $\text{Link}(\sigma_1, \sigma_2)=0$ . The Adversary's *advantage* is its probability of winning.

**Definition 5 (Linkability).** *An LRS scheme is linkable if no PPT adversary has a non-negligible advantage in winning the game above.*

**Security.** Summarizing we have:

**Definition 6 (Security of LRS Schemes).** *An LRS scheme is secure if it is unforgeable, anonymous, and linkable.*

## 4 Constructions of Linkable Ring Signature Schemes

We present several LRS scheme constructions, including one with  $O(1)$ -sized signatures, and all with provable security.

### 4.1 Constructions without Accumulators

We present several generic LRS scheme constructions with length  $O(n)$  bits. Let  $\mathcal{R} \subseteq \mathcal{X} \times \mathcal{Y}$  denote an efficiently-samplable, one-way NP-relation that is typical of  $sk$ - $pk$  relations. Let  $\theta_d : \mathcal{X} \rightarrow \tilde{\mathcal{Y}}$  denote a PK-bijective mapping w.r.t.  $\mathcal{R}$ .

**LRKg:** Upon input  $1^\lambda$ , output a pair of  $n$ -vectors  $(gpk, gsk)$  where  $(gsk[i], gpk[i]) \in \mathcal{R}$ , each  $i$ ,  $1 \leq i \leq n$ .

**LRSign:** Upon inputs  $gpk$ ,  $sk = gsk[s]$ , message  $M$ ,  $\mathcal{R}$ , and a PK-bijective mapping  $\theta_d$ , produces a linkable ring signature as follows (two different constructions below):

$$\text{(LRS1) } SPK\{sk : (\forall_{1 \leq i \leq n} (sk, gpk[i]) \in \mathcal{R}) \wedge (\tilde{y} = \theta_d(sk))\}(M), \text{ or} \quad (1)$$

$$\text{(LRS2) } SPK\{sk : \forall_{1 \leq i \leq n} ((sk, gpk[i]) \in \mathcal{R}) \wedge (\tilde{y}_i = \theta_d(sk))\}(M). \quad (2)$$

We adopt the notation of [12] in this paper.  $\text{LRVf}$  is straightforward.  $\text{Link}$  for LRS1 examines the values of the *linkability tag*,  $\tilde{y}$ , from both input signatures and outputs linked if and only if they are equal.

Theorem 1 in [19] implies the existence of efficient implementations of LRS1 and LRS2. The complexity of the signatures implied by the proof of the theorem is polynomially growing, but is high in practice. In the Appendix, we provide efficient instantiations of LRS1 and LRS2 use two popular mechanisms to achieve 1-out-of- $n$  proof-of-knowledge, namely Cramer, et al.'s partial proof-of-knowledge [18] or Rivest, et al.'s ring structure [29].

## 4.2 Generic Constructions with Accumulators

We construct short linkable ring signatures, using methods motivated by [19]. Define  $f(u, \{z_1, \dots, z_n\}) \doteq f(\dots(f(f(u, z_1), z_2), \dots, z_n))$ , the accumulation of  $z_1, \dots, z_n$ .

**LRSig:** Upon inputs  $gpk$ ,  $gsk[s]$ , message  $M$ , produces a ring signature as follows: (Let  $gpk=(y_1, \dots, y_n)$ , and  $gsk[s] = x$ .)

1. Compute  $v = f(u, \{y_1, \dots, y_n\})$ , Compute  $w = f(u, \{y_1, \dots, y_n\} \setminus \{y_s\})$ ,
2.  $\sigma =$

$$(LRS3) \text{ SPK}\{(w, y_s, x) : (y_s, x) \in \mathcal{R} \wedge f(w, y_s) = v \wedge \tilde{y} = \theta_d(x)\}(M) \quad (3)$$

## 4.3 The Short Linkable Ring Signature Scheme

The construction is further instantiated by the following parameter choices: Use the accumulator  $f(u, x) = u^x \bmod N$  where  $N = pq = (2p' + 1)(2q' + 1)$  is a sufficiently large safe prime. Let  $\mathcal{R}\{(2e_1e_2 + 1, (e_1, e_2)) : |e_1 - 2^\ell|, |e_2 - 2^\ell| < 2^\mu\}$ . The parameters  $\ell$  and  $\mu$  are selected according to methods in [9] to ensure security against coalition attacks. We have in mind  $\ell \approx \lambda/2$ ,  $\mu$  sufficiently small,  $p'$  and  $q'$  at least  $\lambda/2$  plus a few bits long. Let  $\theta(e_1, e_2) = g_\theta^{e_1 + e_2}$ , where  $g_\theta \in \mathcal{RQ}(N)$  and is fairly generated. Given the above instantiations, LRS3 becomes:

$$(LRS3') \text{ SPK}\{(w, x, e_1, e_2) : w^x = v \bmod N \wedge x = 2e_1e_2 + 1 \\ \wedge |e_1 - 2^\ell|, |e_2 - 2^\ell| < 2^\mu \wedge \tilde{y} = \theta((e_1, e_2))\}(M) \quad (4)$$

Writing down the  $\Sigma$ -protocol explicitly, LRS3' becomes

$$(LRS4) \text{ SPK}\{(r, w, x, e_1, e_2) : T_1 = g^r \wedge T_2 = g^x h^r \wedge T_3 = g^{e_2} s^r \\ \wedge T_4 = wy^r \wedge T_5 = g^{e_1} t^r \wedge T_1^x = g^{a_1} \wedge T_1^{e_2} = g^{a_2} \wedge T_4^x = vy^{a_1} \\ \wedge T_5^{2e_2} g = g^x t^{2a_2} \wedge |e_1 - 2^\ell|, |e_2 - 2^\ell| < 2^\mu \wedge \tilde{y} = g_\theta^{e_1 + e_2}\}(M) \quad (5)$$

where  $a_1 = xr$ ,  $a_2 = e_2r$ .

The above instantiates a  $O(1)$ -sized linkable ring signature scheme, provided the list of public keys  $y_1, \dots, y_n$  is not included in the signature.

**Dynamic group setting.** Our scheme and security model can be easily adapted for dynamic group settings. Without accumulators, the **Join** operation can be easily implemented for ring signature or linkable ring signature. The group membership manager simply maintains a list of all members' public keys. **Join** simply adds new member's public key to the list. In the presence of accumulators, the group membership manager maintains the accumulated value of all members' public keys. To do **Join**, simply accumulate the new member's public key.

**Discussions.** Linkable ring signatures where user key pairs are from cryptosystems other than RSA can be similarly constructed. We have in mind modifications to the usual form of public key for added security against potential coalition attacks. E.g., for user key pairs from DL we have in mind  $\mathcal{R} = \{(x, y = 2g^x + 1)\}$ , For pairings with  $\mathcal{R} = \{(x, \text{first coordinate of } P^x)\}, \dots$ , etc.



#### 4.4 Linkable Group Signatures

Our linkable ring signature scheme can be easily adapted to become a linkable group signature (LGS) scheme. To support openability of signatures to identify its signers, we change LRS3 to become

$$\begin{aligned} \text{(LGS1) } SPK\{(w, y_s, x, r) : (y_s, x) \in \mathcal{R} \wedge f(w, y_s) = v \wedge \tilde{y} = \theta_d(x) \\ \wedge ctxt = \text{Enc}(K_e, y_s, r)\}(M) \end{aligned} \quad (6)$$

where  $\text{Enc}$  is an IND-CCA2-secure encryption,  $K_e$  is its encryption key,  $r$  is the randomness used in the encryption.

The result is a short,  $O(1)$ -size group signature which also has linkability, i.e. two signatures by the same signer can be detected as such, without opening or otherwise creaking the signer's anonymity.

#### 4.5 Security Theorems

We have some theorems about the security of the generic construction, and then about the specific and short construction of linkable ring signatures.

**Theorem 1.** *If (3) (resp. (4), (5) ) is a simulation-sound, computational zero-knowledge proof system, then the LRS3 (resp. LRS3', LRS4) linkable ring signature scheme has unforgeability and anonymity.*

*Proof Sketch:* We follow proof lines similar to those in [2]. Soundness of the proof system implies unforgeability of the signature. Anonymity follows from the computational zero-knowledge.  $\square$

**Theorem 2.** *If (3) (resp. (4), (5) ) is a simulation-sound, computational zero-knowledge proof system, and  $\theta_d(\cdot)$  is PK-bijective, then the LRS3 (resp. LRS3', LRS4) signature scheme has linkability.*

*Proof Sketch:* Rewind twice, to the two crucial hash queries which produced the two signatures, to obtain two different user secret keys. The PK-bijectivity ensures that the two witnesses extracted are indeed from different users.  $\square$

**Theorem 3.** *The proof system (5) is simulation-sound, computational zero-knowledge provided the Strong RSA Assumption holds and the Link Decisional RSA (LD-RSA) Problem is hard, in the Random Oracle Model.*

*Proof Sketch:* Same as before. Note the accumulator being used is an accumulator with one-way domain (which is *coalition-resistant*) by [19, 1]. The simulation of the Signing Oracle  $\mathcal{SO}$  deserves special attention. It is done by randomly generating  $\tilde{y}'$  and simulating the SPK without the secret keys,  $e_1$  and  $e_2$ . Note that usually  $\tilde{y}' \neq \theta(e_1, e_2)$ . The simulation of the SPK is standard computational zero-knowledge: (1) Obtain challenge  $c$ 's. (2) Randomly generate  $\tilde{y}$ . (3) Compute response  $z$ 's. (4) Compute commitment  $t$ . The resulting signature

will be passed by LRVf as valid. Parts of the simulation of  $\mathcal{SO}$  use back-packing in the Random Oracle Model.

If  $\mathcal{A}$  is an anonymity attacker, then it can be used to solve the LD-RSA Problem by having  $\mathcal{S}$  do the following in the Gantlet Phase: Instead of flipping a coin  $b$ ,  $\mathcal{S}$  always uses the LD-RSA Problem instantiation,  $(x, \tilde{y}_{LD\text{RSA}})$  as the Gauntlet  $x = \text{gpk}[s]$  and  $\tilde{y}$  to simulate the SPK. Note  $\tilde{y}_{LD\text{RSA}}$  equals  $\theta(e_1, e_2)$  or randomly generated  $\tilde{y}'$  with equal probability. Simulating the SPK without  $e_1$  and  $e_2$  when  $\tilde{y}_{LD\text{RSA}} = \theta(x)$  is statistically indistinguishable from the real world, i.e. LRSig with  $e_1$  and  $e_2$ . Simulating the SPK without  $e_1$  and  $e_2$  is statistically indistinguishable from the ideal world because  $\tilde{y}_{LD\text{RSA}}$  is random. Then  $\mathcal{A}$ 's answer in the End Game solves the LD-RSA Problem.  $\square$

**Theorem 4.** *The LRS4 linkable ring signature constructed is secure provided the Strong RSA Assumption holds and the Link Decisional RSA (LD-RSA) Problem is hard, in the Random Oracle Model.*

*Proof Sketch:* Summarizing above theorems and proving  $\theta = g_\theta^{e_1+e_2}$  is indeed PK-bijective.  $\square$

## 5 Applications

### 5.1 E-voting

Remarkable advances in group/ring signatures in recent years have given new options to e-voting scheme constructions. In fact, many papers on group/ring signatures have included e-voting as applications. Using group/ring signatures contributes to a new paradigm of e-voting construction.

Nevertheless, none of the existing group/ring signature schemes gives raise to a reasonably good construction. First, most group signature schemes are unlinkable, which means double-voting cannot be detected (an exception: the one-show credential system). Secondly, and more importantly, anonymity revocation is an inherited property in group signatures/credential systems. Note that anonymity is of prime concern in e-voting. Nothing justifies to open a vote.

Linkable ring signature schemes partly solved the problem because they have: (1) double-voting detecting capability and (2) no anonymity revocation. However existing linkable ring signature schemes have signature sizes linear with the signing group, which makes them impractical when used in large-scale voting. Our short linkable ring signature scheme (LRS4) has constant signature size and is thus very practical in this sense.

**Construction.** We use the construction of an e-voting scheme from [23]. The main contribution of the current paper w.r.t. this e-voting scheme is that we have an  $O(1)$ -sized signature whereas [23] used an  $O(n)$ -sized signature, where  $n$  is the group size. We summarize the e-voting scheme below. For further details, see [23].

- (Registration.) Through a registration process, a list of the public keys of all eligible voters is published. Each voter can check if his public key is included. A number of independent registrants can be used to ensure that no ineligible entity is listed.
- (Vote Casting.) Each voter sends in a linkable ring signature on a message which states its selected candidate, from a prescribed candidate list or as a write-in candidate. The cast ballots can be listed in a public bulletin board for voter inspection.
- (Tallying.) Simply verify all received linkable ring signatures, drop the invalid or linked ones, and tally the remaining according to their signed messages.

Kiayias and Yung [22] hybridized homomorphic encryption and mix-net to achieves simultaneously (1) efficient tallying, (2) universal verifiability and (3) write-in capability under typical voter distribution where only a small proportion of voters write-in. Our e-voting scheme above achieves the same even under worst-case voter distributions: the proportion of voters who write in can vary from negligible to overwhelming. To write-in in our scheme, a voter simply sends in a linkable ring signature on the message which includes its write-in candidate.

If one worries about the group manager having too much power from knowing the factoring of  $N$ , then Boneh and Franklin’s [3] for generating  $N$  collaboratively among a number of servers, none of which knows the factoring of  $N$ , can be used. Nakanishi, et al. [26, 27] presented e-voting from linkable group signature. Our version of the linkable group signature can also be used to construct e-voting.

## 5.2 Direct Anonymous Attestation (DAA)

In essence, DAA (Direct Anonymous Attestation) [7] is a group signature without revocability, and with an additional feature of *rogue tagging*. Double signers can be detected, or linked, yet their identities are not revealed. When a double signer is detected, a *rogue tag* is produced to prevent it from signing again: future signatures (attestations) identified with a known rogue tag is not accepted. Double signers of different transactions with the same *basename*, *bsn*, are detected. But signing twice with different basenamespace is not detected.

The linkable ring signature is ideally suited to implementing DAA. It is a group signature without revocation. Its linkability feature can be used to detect double signers, and when linked output the linkability tag,  $\tilde{y} = g_\theta^{sk}$ , as the rogue tag. Future signatures whose  $\tilde{y}$  equals a known rogue tag is not accepted. The value  $g_\theta$  can be made a function of the basenamespace but not the transaction, e.g.  $g_\theta = \text{Hash}(\text{bsn}, \dots)$ . Then double signing on different transactions with same basenamespace is linked, while double signing on different basenamespace will not be linked.

Below, we highlight a few important points in implementing DAA from linkable ring signatures. Further details are straightforward from [7] and omitted.

- (*Setup for Issuer.*) The issuer acts as the GM. He initializes our short linkable ring signature scheme.

- (*Join Protocol.*) The TPM joins by first running  $\text{LRKg}$  of the linkable ring signature scheme in order to obtain a user key pair. It then submits the public key to the Issuer and retains the secret key. It also proves to the Issuer that the public key is correctly formed.
- (*DAA-Signing Protocol.*) The TPM signs a linkable ring signature by invoking the  $\text{LRS.Sign}$  algorithm.
- (*Verification Algorithm.*) This is exactly the same as  $\text{LRS.Verify}$ .
- (*Rogue Tagging.*) When a user secret key is found, it should be distributed to all potential verifiers. These verifiers can then put the key on their list of rogue keys.

### 5.3 E-cash

Our LRS scheme (LRS4) can also be used to construct an e-cash scheme. It serves as a new alternative to e-cash schemes of the “group signature approach”, as described in the introduction.

**Construction.** The Bank takes the role of the GM. We adopt the “group of coins” model: each user key pair represents a coin; the knowledge of a user secret key means the ability to spend a coin; and anonymity is among the group of coins issued. The Bank initializes our short linkable group signature scheme. Assume the shops and the users have their accounts established with the bank.

- (*Withdrawal*) To withdraw a coin, the user first runs  $\text{LRKg}$  to obtain a key pair. He keeps the secret key with himself and gives the public key to the bank. The bank debits the user’s account, and update the group public key by accumulating the new public key into the current group public key.
- (*Payment*) The user signs a linkable group signature, using his secret key, on the payment transcript, on behalf of the most up-to-date coin group (i.e. using the most up-to-date group public key). The shop verifies against the signature and accepts the payment if the signature is valid.
- (*Deposit*) The shop gives the bank the payment transcript, along with the associated linkable group signature. The bank verifies as the shop did and credits the shop’s account if the signature is valid. To detect double-spending, the bank goes through the deposit database to look for signatures that are linked.

Double spenders of the e-cash are detected as double signers of the linkable ring signature. However, methodologies differ after detection. In *non-accusatory* linkability, the suspect can only be *tagged* and prevented from further double spending afterwards. The drawbacks are time delay to effective tagging and small punishment for the offense. In *accusatory* linkability, the linking algorithm outputs a suspect. But there are issues of *non-slanderability* and *deniability* that can be quite subtle.

## 6 Conclusion

In this paper, we have presented generic LRS scheme constructions with provable security and the first short LRS scheme construction, with security reducible to the LD-RSA problem. We have shown a generic LGS scheme construction with provable security. Also, we have given the first bandwidth-conserving e-voting scheme that simultaneously achieves efficient tallying, universal verifiability, and write-in capability, even in the worst case of voter distribution. We have discussed how to implement DAA and e-cash schemes using LRS/LGS schemes.

## References

1. G. Ateniese, J. Camenisch, M. Joye, and G. Tsudik. A practical and provably secure coalition-resistant group signature scheme. In *CRYPTO 2000*, pages 255–270. Springer-Verlag, 2000.
2. M. Bellare, H. Shi, and C. Zhang. Foundations of group signatures: The case of dynamic groups. In *CT-RSA 2005*, 2005. To appear.
3. D. Boneh and M. Franklin. Efficient generation of shared RSA keys. In *CRYPTO 1997*, volume 1294 of *LNCS*, pages 425–439. Springer-Verlag, 1997.
4. Dan Boneh, Xavier Boyen, and Hovav Shacham. Short group signatures. In *Eurocrypt 2004*, volume 3027 of *LNCS*. Springer-Verlag, 2004.
5. S. Brands. Untraceable off-line cash in wallet with observers. In *Proceedings of the 13th annual international cryptology conference on Advances in cryptology*, pages 302–318. Springer-Verlag, 1994.
6. E. Bresson, J. Stern, and M. Szydło. Threshold ring signatures and applications to ad-hoc groups. In *Crypto'02*, volume 2442 of *LNCS*, pages 465–480. Springer-Verlag, 2002.
7. E. Brickell, J. Camenisch, and L. Chen. Direct anonymous attestation. Cryptology ePrint Archive, Report 2004/205, 2004. <http://eprint.iacr.org/>.
8. J. Camenisch and A. Lysyanskaya. Efficient non-transferable anonymous multi-show credential system with optional anonymity revocation. In *EUROCRYPT 2001*, volume 2045 of *LNCS*, pages 93–118. Springer-Verlag, 2001.
9. J. Camenisch and A. Lysyanskaya. A signature scheme with efficient protocols. In *SCN 2002*, volume 2576 of *LNCS*, pages 268–289. Springer-Verlag, 2003.
10. J. Camenisch and A. Lysyanskaya. Signature schemes and anonymous credentials from bilinear maps, 2004. To appear.
11. J. Camenisch, J.-M. Piveteau, and M. Stadler. An efficient fair payment system. In *Proceedings of the 3rd ACM conference on Computer and communications security*, pages 88–94. ACM Press, 1996.
12. J. Camenisch and M. Stadler. Efficient group signature schemes for large groups (extended abstract). In *CRYPTO'97*, pages 410–424. Springer-Verlag, 1997.
13. D. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Commun. ACM*, 24(2):84–90, 1981.
14. D. Chaum. Blind signatures for untraceable payments. In *Crypto'82*, pages 199–203. Plenum Press, 1982.
15. D. Chaum, A. Fiat, and M. Naor. Untraceable electronic cash. In *Proceedings on Advances in cryptology*, pages 319–327. Springer-Verlag, 1990.
16. D. Chaum and E. van Heyst. Group signatures. In *EUROCRYPT'91*, volume 547 of *LNCS*, pages 257–265. Springer-Verlag, 1991.

17. J. D. Cohen and M. J. Fischer. A robust and verifiable cryptographically secure election scheme. In *FOCS 85*, pages 372–382, 1985.
18. R. Cramer, I. Damgard, and B. Schoenmakers. Proofs of partial knowledge and simplified design of witness hiding protocols. In *CRYPTO'94*, pages 174–187. Springer-Verlag, 1994.
19. Y. Dodis, A. Kiayias, A. Nicolosi, and V. Shoup. Anonymous identification in ad hoc groups. In *EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 609–626. Springer-Verlag, 2004.
20. Trusted Computing Group. Trusted computing platform alliance (tcpa) main specification, version 1.1a. republished as trusted computing group (tcg) main specification, version 1.1b, 2001. <http://www.trustedcomputinggroup.org>.
21. D.-G. Han, H.-Y. Park, Y.-H. Park, S. Lee, D. H. Lee, and H.-J. Yang. A practical approach defeating blackmailing. In *ACISP'02*, pages 464–481. Springer-Verlag, 2002.
22. A. Kiayias and M. Yung. The vector-ballot e-voting approach. In *FC 2004*, volume 3110 of *LNCS*, pages 72–89. Springer-Verlag, 2004.
23. J. K. Liu, V. K. Wei, and D. S. Wong. Linkable spontaneous anonymous group signature for ad hoc groups (extended abstract). In *ACISP'04*, volume 3108 of *LNCS*, pages 325–335. Springer-Verlag, 2004.
24. A. Lysyanskaya and Z. Ramzan. Group blind digital signatures: A scalable solution to electronic cash. In *FC'98*, pages 184–197. Springer-Verlag, 1998.
25. G. Maitland and C. Boyd. Fair electronic cash based on a group signature scheme. In *ICICS'01*, volume 2229 of *LNCS*. Springer-Verlag, 2001.
26. T. Nakanishi, T. Fujiwara, and Watanabe H. A linkable group signature and its application to secret voting. In *4th Int'l Symp. on Communicatin Theory and Appl.*, 1997.
27. T. Nakanishi, T. Fujiwara, and Watanabe H. A linkable group signature and its application to secret voting. *Trans. of Information Processing Society of Japan*, 40(7):3085–3096, 1999.
28. L. Nguyen and R. Safavi-Naini. Efficient and provably secure trapdoor-free group signature schemes from bilinear pairings. In *ASIACRYPT 2004*, 2004. To appear.
29. R. L. Rivest, A. Shamir, and Y. Tauman. How to leak a secret. In *ASIACRYPT 2001*, pages 552–565. Springer-Verlag, 2001.
30. J. Traoré. Group signatures and their relevance to privacy-protecting off-line electronic cash systems. In *ACISP'99*, pages 228–243. Springer-Verlag, 1999.
31. Patrick P. Tsang, Victor K. Wei, Man Ho Au, Tony K. Chan, Joseph K. Liu, and Duncan S. Wong. Separable linkable threshold ring signatures. In *Indocrypt 2004*, 2004. To appear.

## A Linkable Ring Signatures without Accumulators

### A.1 CDS-type Linkable Ring Signatures

LRSig. Given public keys  $gpk = (y_1, \dots, y_n)$ , secret key  $x_s$  satisfying  $(x_s, y_s) \in \mathcal{R}$ , and message  $M$ , do:

1. For each  $i$ ,  $1 \leq i \leq n$ ,  $i \neq s$ , randomly generate  $\tilde{y}_i$  and simulate a conversation  $(t_i, c_i, z_i)$  for  $PoK\{x_i : (x_i, y_i) \in \mathcal{R} \wedge \tilde{y}_i = \theta(x_i)\}$ .

2. Randomly generate  $t_s$ . Compute  $c_0 = H(M, y_1, \dots, y_n, \text{nonce}, t_1, \dots, t_n)$ . Compute  $c_s$  such that the polynomial  $f$  interpolated by  $f(i) = c_i, 0 \leq i \leq n$ , has degree  $\leq n - 1$ .
3. Use  $x_s$  to compute the response  $z_s$  which completes the conversation  $(t_s, c_s, z_s)$  to  $PoK\{x_s : (x_s, y_s) \in \mathcal{R} \wedge \tilde{y}_s = \theta(x_s)\}$ .
4. Output the signature

$$\sigma = ((y_1, \tilde{y}_1), \dots, (y_n, \tilde{y}_n), (t_1, c_1, z_1), \dots, (t_n, c_n, z_n), \text{nonce}).$$

LRVf. Given  $gpk = (y_1, \dots, y_n), M, \sigma$ , verify that:

1. each  $(t_i, c_i, z_i)$  is a valid conversation for  $PoK\{x_i : (x_i, y_i) \in \mathcal{R} \wedge \theta(x_i) = \tilde{y}_i\}$ ;
2.  $c_i, 0 \leq i \leq n$ , interpolate a polynomial  $f$  satisfying  $f(i) = c_i$  for  $0 \leq i \leq n$  and  $\text{degree}(f) \leq n - 1$ ; and
3.  $c_0 = H(M, y_1, \dots, y_n, \text{nonce}, t_1, \dots, t_n)$ .

Link. Given two valid signatures:  $\sigma_i = ((y_1^{(i)}, \tilde{y}_1^{(i)}), \dots, (y_n^{(i)}, \tilde{y}_n^{(i)}), (t_1^{(i)}, c_1^{(i)}, z_1^{(i)}), \dots, (t_n^{(i)}, c_n^{(i)}, z_n^{(i)}), \text{nonce}^{(i)})$ , where  $i = 1, 2$ , look for  $\tilde{y}_j^{(1)} = \tilde{y}_j^{(2)}$  for some  $j$ . If found, output linked. Otherwise, output unlinked.

Signature size can be reduced by dropping or replacing some redundant terms. Discussions omitted. A variant is to make  $\tilde{y}_i = \tilde{y}_s$  for all  $i$ . No other alterations are made to LRSig, LRVf, or Link. Example: Some of the linkable ring signatures in [31] is instantiated by  $\mathcal{R} = \{(x, y) : y = g^x\}, \theta_d(x) = g_\theta^x$ .

## A.2 RST-type Linkable Ring Signatures.

LRSig. Given  $gpk = (y_1, \dots, y_n), x_s$  satisfying  $(x_s, y_s) \in \mathcal{R}$ , message  $M$ , do:

1. Randomly generate commitment  $t_s$ . For each  $i, 1 \leq i \leq n, i \neq s$ , randomly generate  $\tilde{y}_i$ . Set  $\tilde{y}_s = \theta_d(x_s)$ .
2. For  $i = s + 1, \dots, n$ , and then for  $i = 1, \dots, s - 1$ , compute  $c_i = H_i(M, y_1, \dots, y_n, t_{i-1})$  and simulate a conversation  $(t_i, c_i, z_i)$  for  $PoK\{x_i : (x_i, y_i) \in \mathcal{R} \wedge \theta(x_i) = \tilde{y}_i\}$ .
3. Compute  $c_s = H_i(M, y_1, \dots, y_n, t_{s-1})$  and use  $x_s$  to compute a conversation  $(t_s, c_s, z_s)$  for  $PoK\{x_s : (x_s, y_s) \in \mathcal{R} \wedge \theta(x_s) = \tilde{y}_s\}$ .
4. Output the signature  $\sigma = ((y_1, \tilde{y}_1), \dots, (y_n, \tilde{y}_n), (t_1, c_1, z_1), \dots, (t_n, c_n, z_n))$ .

LRVf. Given a candidate signature  $\sigma = ((y_1, \tilde{y}_1), \dots, (y_n, \tilde{y}_n), (t_1, c_1, z_1), \dots, (t_n, c_n, z_n))$ , verify that, for every  $i, 1 \leq i \leq n$ ,

1.  $(t_i, c_i, z_i)$  is a valid conversation of  $PoK\{x_i : (x_i, y_i) \in \mathcal{R} \wedge \theta(x_i) = \tilde{y}_i\}$ ;
2. and  $c_i = H_i(M, y_1, \dots, y_n, t_{i-1}), 1 \leq i \leq n$ .

Link. Same as the Link for the CDS-type signature, specified above.

There are also techniques to drop some redundant terms for length reduction. We omit those discussions. A variant is to make  $\tilde{y}_i = \tilde{y}_s$  for all  $i$ . Example: Some of the linkable ring signatures in [23] is instantiated by  $\mathcal{R} = \{(x, y) : y = g^x\}, \theta_d(x) = g_\theta^x$ .