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,kwwei}@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 (LRS) scheme additionally allows anyone to determine if two ring signatures have been signed by the same group member. Recently, Dodis et al. [19] gave a short (constant-sized) ring signature scheme. We extend it to the first short LRS scheme, and reduce its security to a new hardness assumption, the Link Decisional RSA (LD-RSA) Assumption. We also extend [19]'s other schemes to a generic LRS scheme and a generic linkable group signature scheme. We discuss three applications of our schemes. Kiayias and Yung [23] constructed the first e-voting scheme which simultaneously achieves efficient tallying, public verifiability, and write-in capability for a typical voter distribution under which only a small portion writes in. We construct an e-voting scheme based on our short LRS scheme which achieves the same even for all worst-case voter distribution. Direct Anonymous Attestation (DAA) [7] is essentially a ring signature scheme with certain linking properties that can be naturally implemented using LRS schemes. The construction of an offline anonymous e-cash scheme using LRS 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 [30] 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 [30] and anonymous identification/authentication for ad hoc groups [6, 19].

Linkable ring signatures [24] 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

* An extended abstract of this paper appeared in ISPEC 2005.

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 [24]. Concepts similar to linkability also appeared in one-show credentials [8], linkable group signatures [27, 28], and DAA [7].

Early constructions of (linkable) ring/group signature schemes have large signature sizes, which are usually $O(n)$ where n is the group size. Subsequent results incorporating various techniques reduced the sizes of state-of-the-art group signatures to a constant independent of group size. Consult [12, 1, 9, 4, 10, 29] for details. Essentially all ring signatures have sizes $O(n)$. Recently, Dodis, et al. [19] gave a short ring signature scheme construction. In this paper, we extend their technique to construct a short LRS scheme. We also extend [19]'s generic ring (resp. group) signature scheme constructions to their linkable version.

Tracing-by-linking versus tracing-by-escrowing in group signatures. The following two papers came to our attention after the completion of this research: Teranishi, et al. [31] and Wei [34]. They achieve *tracing-by-linking*, i.e. tracing the double signer's public key without identity escrowing to an Open Authority (OA). In comparison, tradtional group signatures use the *tracing-by-escrowing* technique, and give the Open Authority the unnecessary power to open an honest signer's identity even when there is no dispute to investigate. Comparing the two tracing-by-linking group signatures: [31]'s has smaller size. [34]'s has larger, but still $O(1)$, size, but it is more flexible and supports features such as tracing the double signer's secret key, tracing the double signer's identity without going through the public key, etc. Another paper containing tracing-by-linking ring signature is due to Tsang, et al. [33].

Constant-sized LRS schemes have many applications. We describe three of them briefly in the following.

E-Voting. There are three basic paradigms for cryptographically secure ballot elections. Under the *blind signature* [14] paradigm, the voters obtain ballots from the authorities, certified but 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* [17] paradigm, 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* [13] paradigm, 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].

E-cash systems by *group signatures* recently received much attention. 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: [25, 32, 26].

Direct Anonymous Attestation (DAA). In the context of the Trusted Computing Group (TCG), 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 [22] 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 short ring signature scheme construction of Dodis, et al.[19] to the first short LRS scheme construction, and reduce its security to a set of assumptions including a new hardness assumption, the Link Decisional RSA (LD-RSA) Assumption.
- We also extend [19]'s generic ring (resp. group) signature scheme constructions to their linkable version.
- Motivated by [23], who presented the first e-voting scheme that simultaneously achieved efficient tallying, universal verifiability, and write-in capability for typical voter distribution under which only a small portion writes in, we discuss that e-voting scheme constructed from LRS [24] schemes also achieve the same three properties even for all worst-case voter distributions.
- We discuss an efficient implementation of direct anonymous attestation [7] using linkable ring signatures, and the construction of an e-cash scheme using linkable group signatures.

1.1 Paper Organization

The paper is organized as follows: In Section 2, we give some preliminaries. Then we define linkable ring signatures and notions of security in Section 3.

Constructions and their security analysis are presented in Section 4. We discuss three applications in Section 5. We finally conclude the paper in Section 6.

2 Preliminaries

We review literature results and introduce new terminologies.

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

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. $f : U_f \times X_i \to U_f$ and the accumulator family satisfies *efficient generation*, *efficient evaluation*, and *quasi-commutativity.* 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].

Definition 1 (The Link Decisional RSA (LD-RSA) Assumption). *Let $N = pq = (2p' + 1)(2q' + 1)$ be a sufficiently large safe prime product. Let $g \in QR(N)$, with order $p'q'$. Let p_0, q_0, p_1, q_1 be four sufficiently large and distinct primes. Let b be a fair coin flip, $n_0 = p_0q_1$, $n_1 = p_1q_1$. Given n_0, n_1, $g^{p_b+q_b}$, the* LD-RSA Assumption *says that no PPT algorithm can compute b correctly with probability non-negligibly over 1/2.*

Definition 2 (PK-bijectivity). *Let $\mathcal{R} \subseteq \mathcal{X} \times \mathcal{Y}$ be a one-way efficiently samplable NP-relation. Let $\mathcal{X}_{\mathcal{R}} = \{x \in \mathcal{X} :$ there exists $y \in \mathcal{Y}, (x, y) \in \mathcal{R}\}$. A mapping $\theta : \mathcal{X}_{\mathcal{R}} \to \mathcal{Z}$ is* PK-bijective with respect to $\mathcal{R}$ *if it satisfies the first two of the following three properties. It is* special PK-bijective *if it satisfies all three of the following properties.*

1. *The mapping θ is one-way and bijective.*
2. *Let (x_0, y_0) and (x_1, y_1) be two random samples of $\mathcal{R}$ with $y_0 \neq y_1$. Let $b \in \{0, 1\}$ be a fair coin flip and $z = \theta(x_b)$. Then there is no PPT algorithm who can, given z, distinguish between the the two cases $b = 0$ and $b = 1$ with success probability non-negligibly over half.*

3. Let (x_0, y_0) *and* (x_1, y_1) *be any two samples of* $\mathcal{R}$ *with* $y_0 \neq y_1$. *Let* $b \in \{0, 1\}$ *be a fair coin flip and* $z = \theta(x_b)$. *Then there is no PPT algorithm who can, given* z, *distinguish between the the two cases* $b = 0$ *and* $b = 1$ *with success probability non-negligibly over half.*

The property of PK-bijectivity will be important to the L-anonymity of linkable ring signatures. Details later.

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 (Init, LRKg, LRSig, LRVf, Link).

- Init takes as input the security parameter 1^λ, and outputs system-wide public parameters param. Typically, param includes an *sk-pk* relation $\mathcal{R}$ which is efficiently samplable, an one-way NP-relation, lengths of keys, ..., etc.
- LRKg takes as inputs the security parameter 1^λ, the group size n, and returns a tuple (gpk, gsk), where gpk is group public key, $gmsk$ is group manager'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 $(gsk[i], gpk[i]) \in \mathcal{R}$.
- LRSign takes inputs group public key gpk, a secret signing key $gsk[i]$ and a message M, returns a linkable ring signature σ of M.
- LRVf takes inputs the group public key gpk, a message M, and a signature σ for M, returns either 1 or 0 for valid or invalid.
- Link takes as inputs two valid signatures σ and σ', 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) LRVf(gpk, M, LRSign(gpk, $gsk[i]$, M))=1, and (2) Link(LRSign(gpk, $gsk[i]$, M), LRSign(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, L-anonymity and linkability. The following oracles define the attacker's capabilities.

- $sk_i \leftarrow \mathcal{CO}(pk_i)$. The *Corruption Oracle*, on input a public key $pk_i \in \mathcal{Y}$ that is an output of 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 σ which is computationally indistinguishable from one produced by LRSign using the real secret key $gsk[s]$ on message M.

Remark: An alternative approach is 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 param.
2. $\mathcal{A}$ may query the oracles according to any adaptive strategy.
3. $\mathcal{A}$ delivers gpk, M, σ_g.

$\mathcal{A}$ wins the game if: (1) $\mathsf{LRVf}(gpk,M,\sigma_g)=1$, (2) all public keys in gpk are outputs of LRKg, none has been input to $\mathcal{CO}$, and (3) σ_g is not an output of $\mathcal{SO}$ on any input containing M as the message. $\mathcal{A}$'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.*

L-Anonymity. Anonymity for LRS schemes is defined in the following game in which $\mathcal{A}$ is given access to oracles $\mathcal{CO}$ and $\mathcal{SO}$:

Game LA

1. (*Initialization Phase*) $\mathcal{S}$ generates and gives $\mathcal{A}$ the system parameters 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 $b \in \{\mathsf{real}, \mathsf{ideal}\}$. Case b=real: $\mathcal{S}$ queries $\mathcal{CO}$ with $gpk[s]$ to obtain $gsk[s]$ and compute $\sigma = \mathsf{LRSign}\ (gpk,gsk[s],M)$. Case b=ideal: $\mathcal{S}$ computes $\sigma = \mathcal{SO}(gpk,s,M)$.
4. (*Probe-2 Phase*) $\mathcal{A}$ receives σ, 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 $\hat{b} \in \{\mathsf{real}, \mathsf{ideal}\}$ of b.

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

Definition 4 (L-Anonymity). *An LRS scheme is* L-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 $\mathcal{A}$ 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 param.

2. $\mathcal{A}$ queries the oracles according to any adaptive strategy.
3. $\mathcal{A}$ delivers (gpk_i, M_i, σ_i), i=1,2.

$\mathcal{A}$ wins the game if (1) all public keys in $gpk_1 \cup gpk_2$ are outputs of LRKg, and at most one has been queried to $\mathcal{CO}$. (2) $\mathsf{LRVf}(gpk_i, M_i, \sigma_i)$=1, i=1,2. and (3) $\mathsf{Link}(\sigma_1, \sigma_2)$=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, L-anonymous, and linkable.*

4 Constructions of Linkable Ring Signature Schemes

In this section we present several LRS scheme constructions.

4.1 Generic Constructions without Accumulators

We present two generic LRS scheme constructions with signature size $O(n)$. 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 $\theta_d : \mathcal{X} \to \tilde{\mathcal{Y}}$ denote a one-way bijective mapping. Two LRS schemes are constructed as follows:

- LRKg: Upon input 1^λ, 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 one-way bijective mapping θ_d, produces a linkable ring signature following either Eq (1) or Eq (2) below (thereby resulting in two different constructions):

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

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

  We adopt the notation of [12] in the above.
- LRVf: Straightforward. Accepts if the corresponding PoK is verified to be correct.
- Link: For LRS1, examine the *linkability tag*, $\tilde{y}$, from each of the two input signatures and outputs linked if and only if they are equal. For LRS2, compare the lists of linkability tags of the signatures and outputs linked if and only if they contain at least one tag in common.

Theorem 1 in [19] implies the existence of efficient implementations of LRS1 and LRS2. The complexity of the signatures implied by the theorem is polynomially growing, but is high in practice. In Appendix A, we provide efficient instantiations of LRS1 and LRS2 that use two popular mechanisms to achieve 1-out-of-n proof-of-knowledge, namely Cramer, et al.'s partial proof-of-knowledge [18] and Rivest, et al.'s ring structure [30].

4.2 Generic Construction with Accumulators

We construct a short LRS scheme, using methods motivated by [19]. Define $f(u, \{z_1, \cdots, z_n\}) \doteq f(\cdots(f(f(u, z_1), z_2), \cdots, z_n)$, the accumulation of $z_1, \cdots, z_n$. The signing algorithm of the scheme is given by the following:

– LRSign: Let gpk=($y_1, \cdots, y_n$), and $gsk[s] = x$. Upon inputs gpk, $gsk[s]$, message M, compute $v = f(u, \{y_1, \cdots, y_n\})$, and $w = f(u, \{y_1, \cdots, y_n\} \setminus \{y_s\})$. The signature is computed as $\sigma =$

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

4.3 The Short Linkable Ring Signature Scheme Construction

The generic construction in the previous section 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 product. Let $\mathcal{R}\{(2e_1e_2 + 1, (e_1, e_2)) : |e_1 - 2^\ell|, |e_2 - 2^\ell| < 2^\mu\}$. The parameters ℓ and μ are selected according to methods in [9] to ensure security against coalition attacks. We have in mind $\ell \approx \lambda/2$, μ sufficiently small, p' and q' at least $\lambda/2$ plus a few bits long. Let $\theta_d(e_1, e_2) = g_\theta^{e_1+e_2}$, where $g_\theta \in RQ(N)$ and is fairly generated. Given the above instantiations, LRS3 becomes:

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

Writing down the Σ-protocol explicitly, LRS3' becomes

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

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 $y_1, \cdots, y_n$ is not included in the signature.

4.4 Security Theorems

We give in this section theorems on the security of some of our constructions and leave the proofs in Appendix B.

Theorem 1. *Assume θ_d is a special PK-bijection. The linkable ring signature LRS3 (resp. LRS3', LRS4) is*

1. correct.
2. L-anonymous *if Eq. (3) (resp. (4), (5)) is a simulation-sound, computational zero-knowledge proof system.*

3. unforgeable *if Eq. (3) (resp. (4), (5)) is a sound non-interactive proof system.*
4. linkable *if Eq. (3) (resp. (4), (5)) is a sound non-interactive proof system.*

Theorem 2. *If one-way functions exist and θ_d is a special PK-bijection, then there exist an efficient instantiation of the linkable ring signature LRS3 (resp. LRS3', LRS4) which has correctness, unforgeability, linkability, and L-anonymity.*

The linkable ring signature LRS3 (resp. LRS3') instantiated by the proofs of [21, 20, 19] are asymptotically polynomial time. But they are too complex for practical system parameters. The linkable ring signature instantiation LRS4 is efficient even for practical system parameters. This construction closely followed the short ring signature of [19], and therefore inherits several of their properties. In particular, there is no rigorous proof of unforgeability and unlinkability. We have only been able to obtain the following security reduction of L-anonymity.

Theorem 3. *The linkable ring signature LRS4 has L-anonymity provided the DDH Assumption and the LD-RSA Assumption hold in the RO Model.*

4.5 Discussions

Common errors. If we change the mapping $\theta_d((e_1, e_2)) = g_\theta^{e_1+e_1}$ to $\theta_{d,2}((e_1, e_2)) = g_\theta^{e_1}$, then linkability is lost beause a single user in possession of an RSA secret (e_1, e_2) can produce two unlinked signatures with $\tilde{y} = g_\theta^{e_1}$ and $\tilde{y} = g_\theta^{e_2}$. If we replace with $\theta_{d,3}((e_1, e_2)) = g_\theta^{e_1 e_2}$, then L-anonymity is lost because the proof system LRS4 Eq. (5) is no longer computational zero-knowledge. However, a (probably) secure alternative choice is $\theta_{d,4}((e_1, e_2)) = (g_\theta^{e_1}, g_\theta^{e_2})$. Note that $\theta_{d,2}$ while $\theta_{d,3}$ are not special PK-bijective.

A related security requirement is to 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), $(x_2, y_2) \in \mathcal{R}$, $\theta_d(x_1) = \theta_d(x_2)$. This stronger concept may be needed in further study of the current topic, but it is not needed in the present paper.

It is straightforward to extend our LRS's to *linkable group signatures* [27, 28]. Simply also escrow the user identity (or the user public key) to an Open Authority (OA) in the signatures. The escrow can be done by verifiably encrypt the identity (or public key) to the OA by methods in [2], for example.

Dynamic group setting. We have used the static group setting, where memberships to the group remain unchanged during the course of signature generation and verification. Our scheme and security model can be adapted for dynamic group settings. The Join operation can be implemented. The group membership manager maintains a list of all members' public keys.

Other kinds of user key pairs. LRS schemes in which 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 DL user key pairs $\mathcal{R} = \{(x, y = 2g^x + 1)\}$, for pairings $\mathcal{R} = \{(x, \text{first coordinate of } P^x)\}$, etc.

5 Applications

We discuss in this section the application of LRS schemes to E-voting, offline anonymous electronic cash and direct anonymous attestation.

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 satisfactory construction. First, most group signature schemes are unlinkable, which means double-voting cannot be detected (an exception: the one-show credential system due to [8]). 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.

Previously proposed linkable ring signature schemes partly solved the problem because they have (1) double-voting detecting capability and (2) no anonymity revocation. However, all existing 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 [24]. The main contribution of the current paper w.r.t. this e-voting scheme is that we have an $O(1)$-sized signature whereas [24] used an $O(n)$-sized signature, where n is the group size. We summarize the e-voting scheme below. For further details, see [24].

- (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 [23] 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. [27, 28] 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 [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 basename 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_θ can be made a function of the basename but not the transaction, e.g. $g_\theta = Hash(\mathsf{bsn}, \cdots)$. Then double signing on different transactions with same basename is linked, while double signing on different basename 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 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 $\mathsf{LRS.Sign}$ algorithm.
- (*Verification Algorithm.*) This is exactly the same as $\mathsf{LRS.Verify}$.
- (*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 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 scheme. 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 systems 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–Verglag, 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. Szydlo. 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.
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 cryptographyically 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. U. Feige and A. Shamir. Zero knowledge proofs of knowledge in two rounds. In *CRYPTO'89*, pages 526–544, 1989.
21. O. Goldreich, S. Micali, and A. Wigderson. Proof that yields nothing but their validity or all languages in NP have zero-knowledge proof system. *Journal of the ACM*, 38(3):691–729, 1991.
22. Trusted Computing Group. Trusted computing platform alliance (tcpa) main specification, version 1.1a. republished as trusted computing group (tcg) main specifcation, version 1.1b, 2001. http://www.trustedcomputinggroup.org.
23. A. Kiayias and M. Yung. The vector-ballot e-voting approach. In *FC 2004*, volume 3110 of *LNCS*, pages 72–89. Springer-Verlag, 2004.

24. 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.
25. A. Lysyanskaya and Z. Ramzan. Group blind digital signatures: A scalable solution to electronic cash. In *FC'98*, pages 184–197. Springer-Verlag, 1998.
26. G. Maitland and C. Boyd. Fair electronic cash based on a group signature scheme. In *ICICS'01*, volume 2229 of *LNCS*. Springer-Verlag, 2001.
27. 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.
28. 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.
29. L. Nguyen and R. Safavi-Naini. Efficient and provably secure trapdoor-free goup signature schems from bilinear pairings. In *ASIACRYPT 2004*, volume 3329 of *LNCS*, pages 372–386. Springer-Verlag, 2004.
30. R. L. Rivest, A. Shamir, and Y. Tauman. How to leak a secret. In *ASIACRYPT 2001*, pages 552–565. Springer-Verlag, 2001.
31. I. Teranishi, J. Furukawa, and K. Sako. k-times anonymous authentication. In *Asiacrypt 2004*, volume 3329 of *LNCS*, pages 308–322. Springer-Verlag, 2004.
32. J. Traoré. Group signatures and their relevance to privacy-protecting off-line electronic cash systems. In *ACISP'99*, pages 228–243. Springer-Verlag, 1999.
33. 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*, volume 3348 of *LNCS*, pages 384–398. Springer-Verlag, 2004.
34. Victor K. Wei. Tracing-by-linking group signatures. Cryptology ePrint Archive, Report 2004/370, 2004. http://eprint.iacr.org/.

A Instantiating LRS1 and LRS2

A.1 CDS-type Linkable Ring Signatures

LRSign. Given public keys $gpk = (y_1, \cdots, y_n)$, 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 $c_0 = H(M, y_1, \cdots, y_n, nonce, t_1, \cdots, t_n)$. 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

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

LRVf. Given $gpk = (y_1, \cdots, y_n)$, M, σ, 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, $0 \le i \le n$, interpolate a polynomial f satisfying $f(i) = c_i$ for $0 \le i \le n$ and degree(f)$\le n-1$; and
3. $c_0 = H(M, y_1, \cdots, y_n, nonce, t_1, \cdots, t_n)$.

Link. Given two valid signatures: σ_i=$((y_1^{(i)}, \tilde{y}_1^{(i)}), \cdots, (y_n^{(i)}, \tilde{y}_n^{(i)}), (t_1^{(i)}, c_1^{(i)}, z_1^{(i)}), \cdots, (t_n^{(i)}, c_n^{(i)}, z_n^{(i)}), nonce^{(i)})$, 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 LRSign, LRVf, or Link. Example: Some of the linkable ring signatures in [33] is instantiated by $\mathcal{R} = \{(x, y) : y = g^x\}$, $\theta_d(x) = g_\theta^x$.

A.2 RST-type Linkable Ring Signatures.

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

1. Randomly generate commitment t_s. For each i, $1 \le i \le n$, $i \ne s$, randomly generate $\tilde{y}_i$. Set $\tilde{y}_s = \theta_d(x_s)$.
2. For $i = s+1, \cdots, n$, and then for $i = 1, \cdots, s-1$, compute $c_i = H_i(M, y_1, \cdots, y_n, t_{i-1})$ 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 $c_s = H_i(M, y_1, \cdots, y_n, t_{s-1})$ 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 $\sigma = ((y_1, \tilde{y}_1), \cdots, (y_n, \tilde{y}_n), (t_1, c_1, z_1), \cdots, (t_n, c_n, z_n))$.

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

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 $c_i = H_i(M, y_1, \cdots, y_n, t_{i-1})$, $1 \le i \le n$.

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 [24] is instantiated by $\mathcal{R} = \{(x, y) : y = g^x\}$, $\theta_d(x) = g_\theta^x$.

B Proof Sketches

Proof Sketch of Theorem 1: We follow approach in [2]. L-anonymity follows from the computational zero-knowledge. Unforgeability proof skecth is as follows: Rewind once, to the crucial hash query which produced the signature, to obtain a user secret key. Linkability proof sketch is as follows: Rewind twice, to the two crucial hash queries which produced the two signatures, to obtain two different user secret keys. The one-way bijection of θ_d ensures that the two witnesses extracted are indeed from different users. □

Proof Sketch of Theorem 2: Follows from [21, 20, 19]. □

Proof Sketch of Theorem 3: It is common that anonymity requires the DDH Assumption in group or ring signatures. We omit the details. Below, we reduce L-Anonymity to the LD-RSA Assumption, assuming the DDH Assumption holds.

Upon inputs gpk, s, and M, the simulation of the Signing Oracle $\mathcal{SO}$ is as follows: (1) Randomly generate the challenge c. (2) Randomly generate $\tilde{y}$. (3) Randomly generate the responses z's. (4) Compute commitments. (5) Use the Random Oracle to backpatch the hash output to c.

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 Gauntlet Phase of Game LA: Instead of flipping a coin b, $\mathcal{S}$ always uses the LD-RSA Problem instantiation, $(x, \tilde{y}_{LDRSA})$ as the Gauntlet $x = gpk[s]$ and $\tilde{y}$. Note $\tilde{y}_{LDRSA}$ equals $\theta_d(e_1, e_2)$ or randomly generated $\tilde{y}'$ with equal probability. Simulating the SPK without e_1 and e_2 when $\tilde{y}_{LDRSA} = \theta_d(e_1, e_2)$ is statistically indistinguishable from the real world, i.e. LRSign with e_1 and e_2. Simulating the SPK without e_1 and e_2 is statistically indistinguishable from the ideal world because $\tilde{y}_{LDRSA}$ is random. Then $\mathcal{A}$'s answer in the End Game solves the LD-RSA Problem. □