Anonymous Reputation Systems Achieving Full Dynamicity from Lattices

Ali El Kaafarani[1] Shuichi Katsumata[2] Ravital Solomon[1]

[1] University of Oxford, UK
ali.elkaafarani@maths.ox.ac.uk, ravital.solomon@maths.ox.ac.uk
[2] The University of Tokyo,
National Institute of Advanced Industrial Science and Technology (AIST)
shuichi_katsumata@it.k.u-tokyo.ac.jp

Abstract. In this work, we revisit the *Anonymous Reputation Systems* presented by Blömer et al. in (FC'15). An anonymous reputation system allows users to review/rate products that they have purchased. The main security guarantee that such systems ensure is *privacy*, *i.e.,* users are allowed to *anonymously* write reviews for any products which they have purchased. However, to avoid abuse/misuse cases, a *review-once-policy* is also enforced, *i.e.,* if a user tries to write a second review for the same product, his reviews will be publicly linkable. Therefore, the system manager can revoke this user from the system.

The contribution of this paper is threefold. First, we strengthen and reformalize the security model for reputation systems of Blömer et al. so that it captures more accurately real-life threats. In particular, our security model captures all possible framing scenarios including when the adversary tries to produce a review that links to another review produced by an honest user. Without this security notion, an adversary can exploit this vulnerability in order to revoke or partially de-anonymize a particular user. Second, our reputation system is fully dynamic so that users and items can be added and revoked at any time. This is an attractive and should possibly be a default feature for reputations systems to have, since the system manager will not know the users/items in the time of setup of the system. Finally, we propose the first construction of a reputation system based on lattice assumptions that are conjectured to be resistant to quantum attacks by incorporating a lattice-based tag scheme.

1 Introduction

Since 2000, a tremendous effort has been made to improve the state-of-the-art of reputation systems[3], trying to build the best possible system that helps both consumers and sellers establish mutual trust on the internet. A reputation system allows users to anonymously rate or review products that they bought

[3] In this paper, we will use the terms reputation systems and anonymous reputation systems interchangeably.

over the internet, which would help people decide what/whom to trust in this fast emerging e-commerce world. In 2000, Resnick et al. in their pioneering work [RKZF00] concluded their paper on reputation systems with an allusion to democracy. They envisioned what would Winston Churchill (British prime minister during WWII) comment on reputation systems as he did on democracy. They claim that he might say the following: *"Reputation systems are the worst way of building trust on the Internet, except for all those other ways that have been tried from time-to-time."* Sixteen years later, Zhai et al., in their interesting work [ZWC$^+$16], are still asking the intriguing and challenging question; *"Can we build an anonymous reputation system?"* This clearly shows how challenging and difficult it is to build a useful, secure, and deployable reputation system.

Why reputation systems? Because they simulate what used to happen before the internet era; people used to make decisions on what to buy and from whom, based on personal and corporate reputations. However, on the internet, users are dealing with total strangers, and reputation systems seem to be a suitable solution for building trust while maintaining privacy. Without a doubt, privacy has become a major concern for every internet user. Consumers want to rate products that they buy on the internet and yet keep their identities hidden. This is not merely paranoia; Resnick and Zeckhauser showed in [RZ02] that sellers on eBay discriminate against potential customers based on their review history. This discrimination could take the form of *"Sellers providing exceptionally good service to a few selected individuals and average service to the rest"*, as stated in [Del00]. Therefore, anonymity seems to be the right property for a reputation system to have. However, on the other hand, we cannot simply fully anonymize the reviews, since otherwise malicious users can for example create spam reviews for the purpose of boosting/reducing the popularity of specific products, thus defeating the purpose of a reliable reputation system. Therefore, reputation systems must also enforce *public linkability*, *i.e.,* if any user misuses the system by writing multiple reviews or rating multiple times on the same product, he will be detected, and therefore revoked from the system.

Different cryptographic tools have been used to realize reputation systems, including *Ring Signatures* (*e.g.,* [ZWC$^+$16]), *Signatures of Reputations* (*e.g.,* [BSS10]), *Group Signatures* (*e.g.,* [BJK15]), *Blockchains* (*e.g.,* [SKCD16]), *Mix-Nets* (*e.g.,* [ZWC$^+$16]), *Blind Signatures* (*e.g.,* [ACBM08]), etc., each of which improves on one or multiple aspects of reputation systems that are often complementary and incomparable. Other relevant works include a long line of interesting results presented in [Del00, JI02, KSGM03, DMS03, Ste06, ACBM08, Ker09, YSK$^+$09, GK11, VMG$^+$12, CSK13, MKKSM13, MK14].

Why group signatures. In this work, we choose to move forward and strengthen the state-of-the-art of reputation systems built from group signatures presented in [BJK15] in three orthogonal directions (see details in next paragraph). Undeniably, group signatures are considered to be one of the most well-established type of anonymous digital signatures, with a huge effort being made to generically formalize such an intriguing tool (see for instance, [CVH91, Cam97, AT99, BMW03, BBS04, BS04, CG04, BSZ05, BW06, BCC$^+$16, LNWX17]), and therefore, building a reputation system from group signatures seems one of the advanced and safe options.

Although anonymous reputation systems share some of their security properties with group signatures, they do have their unique setting that requires a different and more challenging security model. For instance, a unique security property that is required by reputation systems is *public-linkability*; adding public-linkability will surely effect the way we would define the *anonymity* and *non-frameability* properties. For example, public-linkability can be easily seen to harm the standard anonymity notion for group signatures. Furthermore, a new framing threat arises when using any linking technique within an anonymous system (see details in Section (3.2)). Therein lie the main subtleties of reputation systems' design, and that is why it has been difficult to define an acceptable security model for such systems so far even though reputation systems have been a hot topic for the last decade and one of the most promising applications of anonymous digital signatures.

Contribution. In our work, we substantially boost the line of work of reputation systems built from group signatures by providing a reputation system that affirmatively addresses three main challanges simultanouesly; namely, we give a rigorous security model, achieve full dynamicity (*i.e.,* users can join and leave at any moment), and equip this important topic with an alternative construction to be ready for the emerging post-quantum era. In more details, we first strengthen and re-formalize the security model for anonymous reputation systems presented in [BJK15] to fully capture all the real-life threats. In particular, we identify an essential security notion[4] uncalled in the presentation of [BJK15]; we capture and formalize the framing scenario where the adversary tries to produce a review that links to another review produced by an honest user. We believe this to be one of the central security notions to be considered in order to maintain a reliable anonymous reputation system, as an adversary otherwise can exploit this vulnerability for the purpose of revoking or partially de-anonymizing a particular user. Also, our security model captures the notion of tracing soundness. It is indeed an important security property as it ensures that even if *all* parties in the system are fully corrupt, no one but the actual reviewer/signer can claim authorship of the signature. Additionally, in our security model, we are able to put less trust in the managing authorities, namely, the tracing manager does not necessarily have to be honest as is the case with [BJK15]. Second, our reputation system is *fully dynamic* where users/items can be added and revoked at any time. This is an attractive and should possibly be a default feature for a reputation system to have, due to its dynamic nature, *i.e.,* the system manager will not have the full list of users and items that will be participating in the system upon

[4] We like to emphasize that the scheme of [BJK15] is secure according to their formalization, and we do not claim their scheme to be wrong in their proposed security model. We view one of our contribution as identifying a security hole which was not captured by the previous security model for reputation systems [BJK15], and providing a more complete treatment of them by building on the ideas of the most up-to-date security model for group signatures [BCC$^+$16].

the setup of the system. Finally, we give a construction of a reputation system that is secure w.r.t our strong security model based on lattice assumptions. To the best of our knowledge, this is the first reputation system that relies on non number-theoretic assumptions, and thereby not susceptible to quantum attacks.

2 Preliminaries

2.1 Lattices

For positive integers n, m such that $n \leq m$, an integer n-dimensional lattice Λ in $\mathbb{Z}^m$ is a set of the form $\{\sum_{i\in[n]} x_i \mathbf{b}_i | x_i \in \mathbb{Z}\}$, where $\mathbf{B} = \{\mathbf{b}_1, \cdots, \mathbf{b}_n\}$ are n linearly independent vectors in $\mathbb{Z}^m$. Let $D_{\mathbb{Z}^m,\sigma}$ be the discrete Gaussian distribution over $\mathbb{Z}^m$ with parameter $\sigma > 0$. In the following, we recall the definition of the Short Integer Solution (SIS) problem and the Learning with Errors (LWE) problem.

Definition 1 (SIS). *For integers $n = n(\lambda), m = m(n), q = q(n) > 2$ and a positive real β, we define the* short integer solution *problem $\mathsf{SIS}_{n,m,q,\beta}$ as the problem of finding a vector $\mathbf{x} \in \mathbb{Z}^m$ such that $\mathbf{A}\mathbf{x} = \mathbf{0} \mod q$ and $\|\mathbf{x}\|_\infty \leq \beta$ when given $\mathbf{A} \leftarrow \mathbb{Z}_q^{n\times m}$ as input.*

When $m, \beta = \mathsf{poly}(n)$ and $q > \sqrt{n}\beta$, the $\mathsf{SIS}_{n,m,q,\beta}$ problem is at least as hard as SIVP_γ for some $\gamma = \beta \cdot \tilde{O}(\sqrt{nm})$. See [GPV08, MP13].

Definition 2 (LWE). *For integers $n = n(\lambda), m = m(n), t = t(n)$, a prime integer $q = q(n) > 2$ such that $t < n$ and an error distribution over $\chi = \chi(n)$ over $\mathbb{Z}$ we define the decision* learning with errors *problem $\mathsf{LWE}_{n,m,q,\chi}$ as the problem of distinguishing between $(\mathbf{A}, \mathbf{A}^\top\mathbf{s} + \mathbf{x})$ from $(\mathbf{A}, \mathbf{b})$, where $\mathbf{A} \leftarrow \mathbb{Z}_q^{n\times m}$, $\mathbf{s} \leftarrow \chi^n$, $\mathbf{x} \leftarrow \chi^m$ and $\mathbf{b} \leftarrow \mathbb{Z}_q^m$. We also define the search* first-are-errorless learning with errors *problem $\mathsf{faeLWE}_{n,t,m,q,\chi}$ as the problem of finding a vector $\mathbf{s} \in \mathbb{Z}_q^n$ when given $\mathbf{b} = \mathbf{A}^\top\mathbf{s} + \mathbf{x} \mod q$ as input, where $\mathbf{A} \leftarrow \mathbb{Z}_q^{n\times m}$, $\mathbf{s} \leftarrow \chi^n$ and $\mathbf{x} \leftarrow \{0\}^t \times \chi^{m-t}$, i.e., the first t samples are noise-free.*

[ACPS09] showed that one can reduce the standard LWE problem where $\mathbf{s}$ is sampled from $\mathbb{Z}_q^n$ to the above LWE problem where the secret is distributed according to the error distribution. Furthermore, [ALS16] showed a reduction from $\mathsf{LWE}_{n-t,m,q,\chi}$ to $\mathsf{faeLWE}_{n,t,m,q,\chi}$ that reduces the advantage by at most 2^{n-t-1}. When $\chi = D_{\mathbb{Z},\alpha q}$ and $\alpha q > 2\sqrt{2n}$, the $\mathsf{LWE}_{n,m,q,\chi}$ is at least as (quantumly) hard as solving SIVP_γ for some $\gamma = \tilde{O}(n/\alpha)$. See [Reg05, Pei09, BLP$^+$13]. We sometimes omit the subscript m from $\mathsf{LWE}_{n,m,q,\chi}, \mathsf{faeLWE}_{n,m,q,\chi}$, since the hardness of the problems hold independently from $m = \mathsf{poly}(n)$. In the following, in case $\chi = D_{\mathbb{Z},\beta}$, we may sometimes denote $\mathsf{LWE}_{n,m,q,\beta}, \mathsf{faeLWE}_{n,m,q,\beta}$.

2.2 Tag Schemes

We recall here the lattice-based *linkable indistinguishable tag* ($\mathsf{LWE\text{-}LIT}$) scheme presented in [EE17]. Let m, ω, q be positive integers with $m = 3\omega$ and $q > 2$

a prime. Assume they are all implicitly a polynomial function of the security parameter n, where we provide a concrete parameter selection in our construction (see Section 4). Let $\mathcal{H} : \{0,1\}^* \to \mathbb{Z}_q^{m\times\omega}$ be a hash function modeled as a random oracle in the security proofs. Let $\mathcal{K} = \mathbb{Z}_q^m \cap [-\beta, \beta]^m$ be the key space for some positive integer $\beta < q$, $\mathcal{T} = \mathbb{Z}_q^m$ be the tag space, and $\mathcal{I} = \{0,1\}^*$ be the message space. Finally, let β' be some positive real such that $\beta > \beta'\omega(\sqrt{\log n})$. Then, the lattice-based linkable indistinguishable tag scheme is defined by the following three PPT algorithms $\mathsf{LIT} = (\mathsf{KeyGen}_{\mathsf{LIT}}, \mathsf{TAG}_{\mathsf{LIT}}, \mathsf{Link}_{\mathsf{LIT}})$:

$\mathsf{KeyGen}_{\mathsf{LIT}}(1^n)$: The *key generation* algorithm takes as input the security parameter 1^n, it samples a secret key $\mathsf{sk} \leftarrow \mathcal{D}_{\mathbb{Z}^m,\beta'}$ until $\mathsf{sk} \in \mathcal{K}$.[5] It then outputs sk.

$\mathsf{TAG}_{\mathsf{LIT}}(I, \mathsf{sk})$: The *tag generation* algorithm takes as input a message $I \in \mathcal{I}$ and a secret key $\mathsf{sk} \in \mathcal{K}$, and samples an error vector $\mathbf{e} \leftarrow \mathcal{D}_{\mathbb{Z}^\omega,\beta'}$. It then outputs a tag $\tau = \mathcal{H}(I)^\top \mathsf{sk} + \mathbf{e} \in \mathcal{T}$.

$\mathsf{Link}_{\mathsf{LIT}}(\tau_0, \tau_1)$: The *linking* algorithm takes as input two tags τ_0, τ_1, and outputs 1 if $\|\tau_0 - \tau_1\|_\infty \le 2\beta$ and 0 otherwise.

We require one additional algorithm only used during the security proof.

$\mathsf{IsValid}_{\mathsf{LIT}}(\tau, \mathsf{sk}, I)$: This algorithm takes as input a tag τ, a secret key sk and a message I, and outputs 1 if $\|\tau - \mathcal{H}(I)^\top \mathsf{sk}\|_\infty \le \beta$ and 0 otherwise.

The tag scheme (LIT) must satisfy two security properties, namely, the tag-indistinguishability and linkability. Informally speaking, tag-indistinguishability ensures that an adversary $\mathcal{A}$ cannot distinguish between two tags produced by two users (of his choice) even given access to a tag oracle. Linkability means that two tags must "link" together if they are produced by the same user on the same message. In the context of reputation systems, the messages associated to the tag will correspond to the items that the users buy. Therefore, when the users write two anonymous reviews on the same item, the tags will help us link the two reviews.

Tag-indistinguishability. A tag-indistinguishability for a LIT scheme is defined by the experiment in Fig. 1. We define the advantage of an adversary $\mathcal{A}$ breaking the tag-indistinguishability as follows:

$$\mathsf{Adv}_{\mathcal{A}}^{\mathsf{Tag}}(n) = \left|\Pr[\mathsf{Exp}_{\mathcal{A}}^{\mathsf{Tag},0}(n) = 1] - \Pr[\mathsf{Exp}_{\mathcal{A}}^{\mathsf{Tag},1}(n) = 1]\right|$$

We say that a LIT scheme is tag-indistinguishable if for all polynomial time adversary $\mathcal{A}$ the advantage is negligible.

The proof of the following Theorem 1 will be provided in the full version.

Theorem 1 (tag-indistinguishability). *For any efficient adversary $\mathcal{A}$ against the tag-indistinguishability experiment of the* LWE-LIT *scheme as defined*

[5] The expected number of samples required will be a constant due to our parameter selection. In particular, we have $\Pr[|x| > \beta'\omega(\sqrt{\log n})] = \mathsf{negl}(n)$ for $x \leftarrow D_{\mathbb{Z},\sqrt{2}\beta'}$.

```
Experiment: $\mathsf{Exp}_{\mathcal{A}}^{\mathsf{Tag},b}(n)$
$\mathsf{sk}_j \leftarrow \mathsf{KeyGen}_{\mathsf{LIT}}(1^n)$ for $j = 0, 1$.
$V_0, V_1 \leftarrow \emptyset$
$(I^*, \mathsf{st}) \leftarrow \mathcal{A}^{\mathcal{O}_{\mathsf{Tag}}(\cdot,\cdot),\mathcal{H}(\cdot)}(1^n)$
$\tau^* \leftarrow \mathsf{TAG}_{\mathsf{LIT}}(I^*, \mathsf{sk}_b)$
$b^* \leftarrow \mathcal{A}^{\mathcal{O}_{\mathsf{Tag}}(\cdot,\cdot),\mathcal{H}(\cdot)}(\tau^*, \mathsf{st})$
If either $(0, I^*)$ or $(1, I^*)$ was submitted
    to $\mathcal{O}_{\mathsf{Tag}}$  return 0
If $b^* = b$ return 1, else return 0
```

Fig. 1. Tag-indistinguishability

```
Oracle: $\mathcal{O}_{\mathsf{Tag}}(j, I)$
If $j \notin \{0, 1\}$ return $\perp$
If $\exists \tau$ such that $(I, \tau) \in V_j$ return $\tau$
Else $\tau \leftarrow \mathsf{TAG}_{\mathsf{LIT}}(I, \mathsf{sk}_j)$
$V_j \leftarrow V_j \cup \{(I, \tau)\}$
return $\tau$
```

Fig. 2. Description of the tag oracle

above, we can construct an efficient algorithm $\mathcal{B}$ *solving the* $\mathsf{LWE}_{m,\omega Q,q,\beta'/\sqrt{2}}$ *problem with advantage:*

$$\mathsf{Adv}_{\mathcal{B}}^{\mathsf{LWE}_{m,\omega Q,q,\beta'/\sqrt{2}}}(n) \geq \mathsf{Adv}_{\mathcal{A}}^{\mathsf{Tag}}(n) - \mathsf{negl}(n),$$

where Q *denotes the number of random oracle queries made by* $\mathcal{A}$*. In particular, assuming the hardness of* $\mathsf{LWE}_{m,\omega Q,q,\beta'/\sqrt{2}}$*, the advantage of any efficient adversary* $\mathcal{A}$ *is negligible.*

Linkability. A linkability of a LIT scheme is defined by the experiment in Fig. 3. We define the advantage of an adversary $\mathcal{A}$ breaking the linkability as $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{Link}}(n) = \Pr[\mathsf{Exp}_{\mathcal{A}}^{\mathsf{Link}}(n) = 1]$. We say that a LIT scheme is non-linkable if for all adversary $\mathcal{A}$ the advantage is negligible.

```
Experiment: $\mathsf{Exp}_{\mathcal{A}}^{\mathsf{Link}}(n)$
$(\tau_0, \tau_1, I, \mathsf{sk}) \leftarrow \mathcal{A}^{\mathcal{H}(\cdot)}(1^n)$
If $\mathsf{IsValid}_{\mathsf{LIT}}(\tau_b, \mathsf{sk}, I) = 1$ for $b \in \{0, 1\}$ and
    $\mathsf{Link}_{\mathsf{LIT}}(\tau_0, \tau_1) = 0$ return 1
Else return 0
```

Fig. 3. Linkability

Theorem 2 (Linkability). *For* any *adversary* $\mathcal{A}$ *against the linkability experiment of the* $\mathsf{LWE\text{-}LIT}$ *scheme as defined above, the advantage* $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{Link}}(n)$ *is negligible.*

Proof. Suppose, towards a contradiction, that an adversary $\mathcal{A}$ wins the linkability experiment. In particular, $\mathcal{A}$ outputs $(\tau_0, \tau_1, I, \mathsf{sk})$ such that the following three conditions hold: $\|\tau_0 - \mathcal{H}(I)^\top \mathsf{sk}\| \leq \beta$, $\|\tau_1 - \mathcal{H}(I)^\top \mathsf{sk}\| \leq \beta$, and $\|\tau_0 - \tau_1\| > 2\beta$. From the first two inequalities, we have

$$\begin{aligned}\|\tau_0 - \tau_1\| &= \|(\tau_0 - \mathcal{H}(I)^\top \mathsf{sk}) + (-\tau_1 + \mathcal{H}(I)^\top \mathsf{sk})\| \\ &\leq \|\tau_0 - \mathcal{H}(I)^\top \mathsf{sk}\| + \|\tau_1 - \mathcal{H}(I)^\top \mathsf{sk}\| \leq 2\beta,\end{aligned}$$

by the triangular inequality. However, this contradicts the third inequality.

2.3 Group Signatures

In a group signature, a group member can anonymously sign on behalf of the group, and anyone can then verify the signature using the group's public key without being able to tell which group member signed it. A group signature has a group manager who is responsible for generating the signing keys for the group members. There are two types of group signatures: the static type and the dynamic type. In the static type [BMW03], the group members are fixed at the setup phase. In this case, the group manager can additionally trace a signature and reveal which member has signed it. In the dynamic type [BSZ05, BCC+16], users can join/leave the system at anytime. Now a group has two managers; the group manager and a separate tracing manager who can open signatures in case of misuse/abuse. Briefly speaking, a group signature has three main security requirements; *anonymity*, *non-frameability*, and *traceability*. Anonymity ensures that an adverary cannot tell which group member has signed the message given the signature. Non-frameability ensures that an adversary cannot produce a valid signature that traces back to an honest user. Finally, traceability ensures that an adversary cannot produce a valid signature that does not trace to an user.

In our work, we build on the recent lattice-based fully dynamic group signature scheme of [LNWX17] to construct our reputation system. We briefly sketch how the group signature scheme of [LNWX17] works; a group manager maintains a Merkle-tree in which he stores members' public keys in the leaves where the exact position are given to the signers at join time. The leaves will be hashed to the top of the tree using an accumulator instantiated using a lattice-based hash function. The relevant path to the top of the tree will be given to each member where the top of the tree itself is public. In order to sign, a group member has to prove in zero-knowledge that; first, he knows the pre-image of a public key that has been accumulated in the tree, and that he also knows of a path from that position in the tree to its root. Additionally, they apply the Naor-Yung double-encryption paradigm [NY90] with Regev's LWE-based encryption scheme [Reg05] to encrypt the identity of the signer (twice) w.r.t the tracer's public key to prove anonymity. To summarize, a group signature would be of the form (Π, c_1, c_2), where Π is the zero-knowledge proof that the signer is indeed a member of the group (*i.e.,* his public key has been accumulated into the Merkle-tree), and the encrypted identity in both c_1 and c_2 is a part of the path that he uses to get to the root of the Merkle-tree. Note that this implies that the ciphertexts (c_1, c_2) are bound to the proof Π.

3 Syntax and Security Definitions

We formalize the syntax of reputation systems following the sate-of-the-art formalization of dynamic group signatures of [BCC+16]. We briefly explain the two major differences that distinguish between a reputation system from a group signature scheme. First, a reputation system is in essence a *group of group signature schemes* run in parallel, where we associate each item uniquely to one instance of the group signature scheme. Second, we require an additional algorithm Link in order to publicly link signatures (i.e., reviews), which is the core functionality provided by reputation systems. We now define reputation systems by the following PPT algorithms:

$\mathsf{RepSetup}(1^n) \rightarrow \mathsf{pp}$: On input of the security parameter 1^n, the setup algorithm outputs public parameters pp.

$\mathsf{KeyGen}_{\mathsf{GM}}(\mathsf{pp}) \leftrightarrow \mathsf{KeyGen}_{\mathsf{TM}}(\mathsf{pp})$: This an interactive protocol between the group manager GM and the tracing manager TM. If completed successfully, $\mathsf{KeyGen}_{\mathsf{GM}}$ outputs the GM's key pair $(\mathsf{mpk}, \mathsf{msk})$ and $\mathsf{KeyGen}_{\mathsf{TM}}$ outputs the TM's key pair $(\mathsf{tpk}, \mathsf{tsk})$. Set the system public key to be $\mathsf{gpk} := (\mathsf{pp}, \mathsf{mpk}, \mathsf{tpk})$.

$\mathsf{UKgen}(1^n) \rightarrow (\mathsf{upk}, \mathsf{usk})$: On input of the security parameter 1^n, it outputs a key pair $(\mathsf{upk}, \mathsf{usk})$ for a user. We assume that the key table containing the various users' public keys upk is publicly available.

$\mathsf{Join}(\mathsf{info}_{t_{\mathsf{current}}}, \mathsf{gpk}, \mathsf{upk}, \mathsf{usk}, \mathsf{item}) \leftrightarrow \mathsf{Issue}(\mathsf{info}_{t_{\mathsf{current}}}, \mathsf{msk}, \mathsf{upk}, \mathsf{item})$: This is an interactive protocol between a user upk and the GM. Upon successful completion, the GM issues an identifier $\mathsf{uid}_{\mathsf{item}}$ associated with item to the user who then becomes a member of the group that corresponds to item[6]. The final state of the Issue algorithm, which would always include the user public key upk, is stored in the user registration table reg at index $(\mathsf{item}, \mathsf{uid}_{\mathsf{item}})$ which is made public. Furthermore, the final state of the Join algorithm is stored in the secret group signing key $\mathsf{gsk}[\mathsf{item}][\mathsf{uid}_{\mathsf{item}}]$.

$\mathsf{RepUpdate}(\mathsf{gpk}, \mathsf{msk}, R, \mathsf{info}_{t_{\mathsf{current}}}, \mathsf{reg}) \rightarrow (\mathsf{info}_{t_{\mathsf{new}}}, \mathsf{reg})$: This algorithm is run by the GM to update the system info. On input of the group public key gpk, GM's secret key msk, a list R of active users' public keys to be revoked, the current system info $\mathsf{info}_{t_{\mathsf{current}}}$, and the registration table reg, it outputs a new system info $\mathsf{info}_{t_{\mathsf{new}}}$ while possibly updating the registration table reg. If no changes have been made, output $\perp$.

$\mathsf{Sign}(\mathsf{gpk}, \mathsf{gsk}[\mathsf{item}][\mathsf{uid}_{\mathsf{item}}], \mathsf{info}_{t_{\mathsf{current}}}, \mathsf{item}, \mathsf{M}) \rightarrow \Sigma$: On input of the system's public key gpk, user's group signing key $\mathsf{gsk}[\mathsf{item}][\mathsf{uid}_{\mathsf{item}}]$, system info $\mathsf{info}_{t_{\mathsf{current}}}$ at epoch t_{current}, an item, and message M, it outputs a signature Σ. If the user owning $\mathsf{gsk}[\mathsf{item}][\mathsf{uid}_{\mathsf{item}}]$ is not an active member at epoch t_{current}, the algorithm outputs $\perp$.

$\mathsf{Verify}(\mathsf{gpk}, \mathsf{info}_{t_{\mathsf{current}}}, \mathsf{item}, \mathsf{M}, \Sigma) \rightarrow 1/0$: On input of the system's public key gpk, system info $\mathsf{info}_{t_{\mathsf{current}}}$, an item, a message M, and a signature Σ, it outputs 1 if Σ is valid signature on M for item at epoch t_{current}, 0 otherwise.

$\mathsf{Trace}(\mathsf{gpk}, \mathsf{tsk}, \mathsf{info}_{t_{\mathsf{current}}}, \mathsf{reg}, \mathsf{item}, \mathsf{M}, \Sigma) \rightarrow (\mathsf{uid}_{\mathsf{item}}, \Pi_{\mathsf{Trace}})$: On input of the system's public key gpk, the TM's secret key tsk, the system information $\mathsf{info}_{t_{\mathsf{current}}}$, the user registration table reg, an item, a message M, and a signature Σ, it outputs the identifier of the user $\mathsf{uid}_{\mathsf{item}}$ who produced Σ

[6] Here our syntax assumes that the items to be reviewed have been already communicated to the GM from the respective service providers. We merely do this to make our presentation simple and we emphasize that our construction is general in the sense that the GM does not need to know neither the number of items nor the items themselves ahead of time. Items can dynamically be added/removed from the system by GM when it is online.

and a proof $\mathit{\Pi}_{\mathsf{Trace}}$ that attests to this fact. If the algorithm cannot trace the signature to a particular group member, it returns $\perp$.

$\mathsf{Judge}(\mathsf{gpk}, \mathsf{uid}_{\mathsf{item}}, \mathit{\Pi}_{\mathsf{Trace}}, \mathsf{info}_{t_{\mathsf{current}}}, \mathsf{item}, \mathsf{M}, \mathit{\Sigma}) \rightarrow 1/0$: On input of the system's public key gpk, a user's identifier $\mathsf{uid}_{\mathsf{item}}$, a tracing proof $\mathit{\Pi}_{\mathsf{Trace}}$ from the Trace algorithm, the system info $\mathsf{info}_{t_{\mathsf{current}}}$, an item, a message M and signature $\mathit{\Sigma}$, it outputs 1 if $\mathit{\Pi}_{\mathsf{Trace}}$ is a valid proof that $\mathsf{uid}_{\mathsf{item}}$ produced $\mathit{\Sigma}$ and 0 otherwise.

$\mathsf{Link}(\mathsf{gpk}, \mathsf{item}, (m_0, \mathit{\Sigma}_0), (m_1, \mathit{\Sigma}_1)) \rightarrow 1/0$: On input of the system's public key gpk, an item, and two message-signature pairs, it returns 1 if the signatures were produced by the same user on behalf of the group that corresponds to item, 0 otherwise.

$\mathsf{IsActive}(\mathsf{info}_{t_{\mathsf{current}}}, \mathsf{uid}_{\mathsf{item}}, \mathsf{reg}, \mathsf{item}) \rightarrow 1/0$: this algorithm will only be used in the security games. On input of the system $\mathsf{info}_{t_{\mathsf{current}}}$, a user's identifier $\mathsf{uid}_{\mathsf{item}}$, the user registration table reg, and an item, it outputs 1 if $\mathsf{uid}_{\mathsf{item}}$ is an active member of the group for item at epoch t_{current} and 0 otherwise.

3.1 Discussion on the Security Model of FC'15 Reputation System

Blömer et al. [BJK15] constructed an anonymous reputation system from group signatures based on number-theoretical assumptions. In their work, they claim to formalize reputation systems following the formalization of partially dynamic group signature schemes presented by Bellare et al. [BSZ05], *i.e.,* they have two managers, the *group manger* and *key issuer*[7]. However, one can notice that the security model is in fact strictly weaker than that of [BSZ05]; the major difference being the assumption that the opener/tracer is *always* honest. Furthermore, in their public-linkability property, the key issuer (the GM in our case) is assumed to be honest. Another observation, which we believe to be of much bigger concern, is that their security notion for reputation systems does not fully capture all the real-life threats. In particular, their strong-exculpability property (which is essentially the notion of non-frameability), does not capture the framing scenario where the adversary outputs a signature that *links* to an honest user; it only captures the scenario where the adversary outputs a signature that traces to an honest user. Note that the former attack scenario does not exist in the context of group signatures since no tag schemes are being used there, *i.e.,* the whole notion of linkability does not exist. However, it is a vital security requirement in the reputation system context as an adversary could try to generate a review that links to an honest user's review so that the GM may decide to revoke or de-anonymize the honest user. In our work, we provide a formal definition of reputation systems that models more accurately these real-life threats, which in particular, solve the aforementioned shortcomings of [BJK15].

3.2 Security Definitions

We provide a formal security definition following the experiment type definition of [BCC+16, LNWX17] for fully dynamic group signatures, which originates

[7] Note that [BJK15] does not completely follow the notation used in [BSZ05], *i.e.,* their group manager is in fact the tracer in [BSZ05].

to [BSZ05]. Anonymity, non-frameability and public-linkability are provided in Fig. 4, whereas the rest of the security experiment together with the oracles used therein are provided in the full version of the paper. One of the main differences between theirs and ours is that, we require the *public-linkability* property, which does not exist in the group signature setting. Moreover, the existence of the tag scheme further affects the anonymity and non-frameability properties, which are depicted in Fig. 4; for the former, an adversary should not be allowed to ask for signatures by the challenge users on the challenge item, otherwise he could trivially win the game by linking the signatures. In the latter, an additional attack scenario is taken into consideration, *i.e.,* when an adversary outputs a review that links to an honest user's review.

In our formalization, we only require TM to be honest in the anonymity experiment, which is inevitable as otherwise the adversary could trivially win the game. Also, our public linkability holds unconditionally, and therefore, GM can be assumed to be corrupt there. We now present the security properties of our reputation system.

Correctness A reputation system is correct if reviews produced by honest, non-revoked users are always accepted by the Verify algorithm and if the honest tracing manager can always identify the signer of such signatures where his decision will be accepted by a Judge. Additionally, two reviews produced by the same user on the same item should always link.

Anonymity A reputation system is anonymous if for any PPT adversary the probability of distinguishing between two reviews produced by any two honest signers is negligible even if the GM and all other users are corrupt, and the adversary has access to the Trace oracle.

Non-frameability A reputation system is non-frameable if for any PPT adversary it is unfeasible to generate a valid review that traces or links to an honest user even if it can corrupt all other users and chose the keys for GM and TM.

Traceability A reputation system is traceable if for any PPT adversary it is unfeasible to produce a valid review that cannot be traced to an active user at the chosen epoch, even if it can corrupt any user and can choose the key of TM[8].

Public-Linkability A reputation system is publicly linkable if for any (*possibly inefficient*) adversary it is unfeasible to output two reviews for the same item that trace to the same user but does not link. This should hold even if the adversary can chose the keys of GM and TM.

Tracing Soundness A reputation system has tracing soundness if no (*possibly inefficient*) adversary can output a review that traces back to two different signers even if the adversary can corrupt all users and chose the keys of GM and TM.

[8] The group manager GM is assumed to be honest in this game as otherwise the adversary could trivially win by creating dummy users.

```
Experiment: $\mathsf{Exp}^{\mathsf{Anon}\text{-}b}_{\mathsf{rep\text{-}sys},\mathcal{A}}(n)$
$\mathsf{pp} \leftarrow \mathsf{RepSetup}(1^n)$; $\mathsf{HUL}, \mathsf{CUL}, \mathsf{BUL}, \mathsf{SL}, \mathsf{CL} := \emptyset$
$(\mathsf{st}, \mathsf{info}, \mathsf{mpk}, \mathsf{msk}) \leftarrow \mathcal{A}^{(\cdot \leftrightarrow \mathsf{KeyGen}_{\mathsf{TM}}(\mathsf{pp}))}(\mathsf{pp})$
if $\mathsf{KeyGen}_{\mathsf{TM}}$ did not accept or $\mathcal{A}$'s output is not well-formed, return 0
$\mathsf{gpk} := (\mathsf{pp}, \mathsf{mpk}, \mathsf{tpk})$
$b^* \leftarrow \mathcal{A}^{\mathsf{AddU},\mathsf{CrptU},\mathsf{SndToU},\mathsf{RevealU},\mathsf{Trace},\mathsf{MReg},\mathsf{Chal_b},\mathsf{Sign}}(\mathsf{st}, \mathsf{gpk})$
if $|\mathsf{CL}| \neq 1$ return 0, otherwise, let $\mathsf{CL} = \{(\mathsf{uid}_0, \mathsf{uid}_1, \mathsf{item}^*, \mathsf{M}, \Sigma)\}$
if $(-, \mathsf{uid}_b, -, \mathsf{item}^*, -, -) \in \mathsf{SL}$ for $b = 0$ or $1$, return 0
else return $b^*$
```

```
Experiment: $\mathsf{Exp}^{\mathsf{non\text{-}frame}}_{\mathsf{rep\text{-}sys},\ \mathcal{A}}(n)$
$\mathsf{pp} \leftarrow \mathsf{RepSetup}(1^n)$; $\mathsf{HUL}, \mathsf{CUL}, \mathsf{BUL}, \mathsf{SL} := \emptyset$
$(\mathsf{st}, \mathsf{info}, \mathsf{msk}, \mathsf{mpk}, \mathsf{tsk}, \mathsf{tpk}) \leftarrow \mathcal{A}(\mathsf{pp})$
if $\mathcal{A}$'s output is not well-formed, return 0
Set $\mathsf{gpk} := (\mathsf{pp}, \mathsf{mpk}, \mathsf{tpk})$
$(\mathsf{uid}^*_{\mathsf{item}^*}, \Pi^*_{\mathsf{Trace}}, \mathsf{info}_{t^*}, \mathsf{item}^*, \mathsf{M}^*, \Sigma^*) \leftarrow \mathcal{A}^{\mathsf{CrptU},\mathsf{SndToU},\mathsf{RevealU},\mathsf{Sign},\mathsf{MReg}}(\mathsf{st}, \mathsf{gpk})$
$X \leftarrow \mathsf{RUser}(\mathsf{item}^*, \mathsf{uid}_{\mathsf{item}^*})$
if $X = \perp$ return 0, else $\mathsf{upk}^* := X$
if $\mathsf{Verify}(\mathsf{gpk}, \mathsf{info}_{t^*}, \mathsf{item}^*, \mathsf{M}^*, \Sigma^*) = 0$, return 0
if $\exists(\mathsf{upk}, \mathsf{uid}_{\mathsf{item}^*}, t, \mathsf{item}^*, \mathsf{M}, \Sigma) \in \mathsf{SL}$ s.t. $\mathsf{uid}_{\mathsf{item}^*} \in \mathsf{HUL}[\mathsf{upk}] \wedge \mathsf{upk} \notin \mathsf{BUL}$
    $\wedge\ \mathsf{Link}(\mathsf{gpk}, \mathsf{item}^*, (\mathsf{M}^*, \Sigma^*), (\mathsf{M}, \Sigma)) = 1$, return 1
if $\mathsf{Judge}(\mathsf{gpk}, \mathsf{uid}^*_{\mathsf{item}^*}, \Pi^*_{\mathsf{Trace}}, \mathsf{info}_{t^*}, \mathsf{item}^*, \mathsf{M}^*, \Sigma^*) = 1 \wedge \mathsf{uid}^*_{\mathsf{item}^*} \in \mathsf{HUL}[\mathsf{upk}^*]$
    $\wedge\ \mathsf{upk}^* \notin \mathsf{BUL} \wedge (\mathsf{upk}^*, \mathsf{uid}^*_{\mathsf{item}^*}, t^*, \mathsf{item}^*, \mathsf{M}^*, \Sigma^*) \notin \mathsf{SL}$, return 1
else return 0
```

```
Experiment: $\mathsf{Exp}^{\mathsf{Public\text{-}Link}}_{\mathsf{rep\text{-}sys},\ \mathcal{A}}(n)$
$\mathsf{pp} \leftarrow \mathsf{RepSetup}(1^n)$; $\mathsf{CUL} := \emptyset$
$(\mathsf{st}, \mathsf{info}, \mathsf{msk}, \mathsf{mpk}, \mathsf{tsk}, \mathsf{tpk}) \leftarrow \mathcal{A}(\mathsf{pp})$
if $\mathcal{A}$'s output is not well-formed, return 0
Set $\mathsf{gpk} := (\mathsf{pp}, \mathsf{mpk}, \mathsf{tpk})$
$(\mathsf{item}, \mathsf{uid}_{\mathsf{item}}, \mathsf{info}_t, \{(\mathsf{M}_b, \Sigma_b, \Pi_{\mathsf{Trace},b})\}_{b=0,1}) \leftarrow \mathcal{A}^{\mathsf{CrptU},\mathsf{MReg}}(\mathsf{st}, \mathsf{gpk})$
if $\mathsf{Verify}(\mathsf{gpk}, \mathsf{info}_t, \mathsf{item}, \mathsf{M}_b, \Sigma_b) = 0$ for $b = 0$ or $1$, return 0
if $\mathsf{Link}(\mathsf{gpk}, \mathsf{item}, (\mathsf{M}_0, \Sigma_0), (\mathsf{M}_1, \Sigma_1)) = 1$, return 0
if $\mathsf{Judge}(\mathsf{gpk}, \mathsf{uid}_{\mathsf{item}}, \Pi_{\mathsf{Trace},b}, \mathsf{info}_t, \mathsf{item}, \mathsf{M}_b, \Sigma_b) = 0$ for $b = 0$ or $1$, return 0
else return 1
```

Fig. 4. Security Experiments for the Reputation System-1

4 Our Lattice-Based Reputation System

Intuition behind our scheme. It is helpful to think of our reputation system as a *group of group signatures* managed by a global group manager (or call it a system manager), whom we refer to as a group manager GM for simplicity. This group manager shares the managerial role with the tracing manager TM who is only called for troubleshooting, *i.e.,* to trace users who misused the system. The group manager maintains a set of groups, each corresponding to a product/item owned by a certain service provider. Users who bought a certain item are eligible to become a member of the group that corresponds to that item, and can therefore write *one* anonymous review for that item. Every user in the system will have his own pair of public-secret key $(\mathsf{upk}, \mathsf{usk})$. When he wants to join the system for a particular item, he would engage in the Join-Issue protocol with GM, after which, he would be assigned a position $\mathsf{uid} = \mathsf{bin}(j) \in \{0,1\}^{\ell}$ in the Merkle-tree that corresponds to the item in question, and his public key will be accumulated in that tree. Here, j (informally) denotes the j-th unique user to have bought the corresponding item. The user can now get his witness w_j that attests to the fact that he is indeed a consumer of the item, on which he is then ready to write a review for that item. Technically speaking, he needs to provide a non-interactive zero-knowledge argument of knowledge for a witness to the following relation $\mathcal{R}_{\mathsf{Sign}}$:

$$\begin{aligned}\mathcal{R}_{\mathsf{Sign}} = \big\{&(\mathbf{A}, \mathbf{u}, \mathcal{H}_{\mathsf{Tag}}(\mathsf{item}), \tau, c_1, c_2, \mathbf{B}, \mathbf{P}_1, \mathbf{P}_2), (\mathbf{p}, w_j, \mathbf{x}, \mathbf{e}, \mathsf{uid}_{\mathsf{item}}, \mathbf{r}_1, \mathbf{r}_2) :\\ &\mathbf{p} \neq \mathbf{0}^{nk} \wedge \mathsf{TVerify}_{\mathbf{A}}(\mathbf{p}, w_j, \mathbf{u}) = 1 \wedge \mathbf{A} \cdot \mathbf{x} = \mathbf{G} \cdot \mathbf{p} \bmod q \wedge\\ &\big(\mathsf{Enc}_{\mathsf{Regev}}\big((\mathbf{B}, \mathbf{P}_1, \mathbf{P}_2), \mathsf{uid}_{\mathsf{item}}; (\mathbf{r}_1, \mathbf{r}_2)\big) = (c_1, c_2)\\ &\wedge \tau = \mathcal{H}_{\mathsf{Tag}}(\mathsf{item})^{\top}\mathbf{x} + \mathbf{e}\big\}.\end{aligned}$$

As can be seen, the signer encrypts his uid and computes a tag for the item in question. This tag ensures that he can only write one review for each item, otherwise his reviews will be publicly linkable and therefore detectable by GM. Regarding the verification, anyone can then check the validity of the signature by simply running the verify algorithm of the underlying NIZKAoK proof system. In any misuse/abuse situation, TM can simply decrypt the ciphertext attached to the signature to retrieve the identity of the signer. TM also needs to prove correctness of opening (to avoid framing scenarios) via the generation of a NIZKAoK for the following relation $\mathcal{R}_{\mathsf{Trace}}$:

$$\mathcal{R}_{\mathsf{Trace}} = \{(c_1, c_2, \mathsf{uid}_{\mathsf{item}}, \mathbf{B}, \mathbf{P}_1), (\mathbf{S}_1, \mathbf{E}_1) : \mathsf{Dec}_{\mathsf{Regev}}\big((\mathbf{S}_1, \mathbf{E}_1), (c_1, c_2)\big) = \mathsf{uid}_{\mathsf{item}}\}$$

Finally, for public linkability, we require that any two given signatures (Σ_0, Σ_1) for the same item can be publicly checked to see if they are linkable, *i.e.,* check that were produced by the same reviewer. This can be done simply by feeding the tags τ_0 and τ_1 of the two signatures, to the $\mathsf{Link}_{\mathsf{LIT}}$ algorithm of the underlying LIT scheme. If $\mathsf{Link}_{\mathsf{LIT}}$ returns 0, then Σ_0 and Σ_1 were not produced by the same user, and therefore are legitimate reviews from two different users. Otherwise, in the case it returns 0, we know that some user reviewed twice for the same item; the GM asks TM to trace those signatures and find out who generated them and GM will then revoke the traced user from the system.

4.1 Our construction

Underlying Tools. In our construction, we use the multi-bit variant of the encryption scheme of Regev [KTX07, PVW08], which we denote by $(\mathsf{KeyGen}_{\mathsf{Regev}}, \mathsf{Enc}_{\mathsf{Regev}}, \mathsf{Dec}_{\mathsf{Regev}})$. We also employ the lattice-based tag scheme $(\mathsf{KeyGen}_{\mathsf{LIT}}, \mathsf{TAG}_{\mathsf{LIT}}, \mathsf{Link}_{\mathsf{LIT}})$ provided in Section 2.2. We assume that both schemes share the same noise distribution χ (see below). We also use a lattice-based accumulator (TSetup, $\mathsf{TAcc}_{\mathbf{A}}$,$\mathsf{TVerify}_{\mathbf{A}}$,$\mathsf{TUpdate}_{\mathbf{A}}$) [LNWX17]. Finally, we use a Stern-like zero-knowledge proof system where the commitment scheme of [KTX08] is used internally. More details on these building blocks can be found in the full version of the paper.

Construction. The proposed reputation system consists of the following PPT algorithms:

$\mathsf{RepSetup}(1^n)$: On input of the security parameter 1^n, it outputs the public parameters,

$$\mathsf{pp} = (N, n, q, k, m, m_E, \omega, \ell, \beta, \chi, \kappa, \mathcal{H}_{\mathsf{Tag}}, \mathcal{H}_{\mathsf{Sign}}, \mathcal{H}_{\mathsf{Trace}}, \mathbf{A}).$$

Where, $N = 2^\ell = \mathsf{poly}(n)$ is the number of potential users, $q = \mathcal{O}(n^{1.5}), k = \lceil \log_2 q \rceil, m = 2nk, m_E = 2(n+\ell)k, \omega = 3m, \beta = \sqrt{n} \cdot \omega(\log n)$, and a $\beta/\sqrt{2}$-bounded noise distribution χ. Moreover, $\mathcal{H}_{\mathsf{Tag}} : \{0,1\}^* \to \mathbb{Z}_q^{m \times \omega}$ is the hash function used for the tag scheme, and $\mathcal{H}_{\mathsf{Sign}}, \mathcal{H}_{\mathsf{Trace}} : \{0,1\}^* \to \{1,2,3\}^\kappa$ are two hash functions used for the NIZKAoK proof systems for $\mathcal{R}_{\mathsf{Sign}}$ and $\mathcal{R}_{\mathsf{Trace}}$, where $\kappa = \omega(\log n)$. Finally, $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}$.

$\mathsf{KeyGen}_{\mathsf{GM}}(\mathsf{pp}) \leftrightarrow \mathsf{KeyGen}_{\mathsf{TM}}(\mathsf{pp})$: This is for the group manager and tracing manager to set up their keys and publish the system's public information. The group manager samples $\mathsf{msk} \leftarrow \{0,1\}^m$, and sets $\mathsf{mpk} := \mathbf{A} \cdot \mathsf{msk} \mod q$. On the other hand, TM runs $(\mathsf{pk}_{\mathsf{Enc}}, \mathsf{sk}_{\mathsf{Enc}}) \leftarrow \mathsf{KeyGen}_{\mathsf{Regev}}(1^\mathsf{n})$ and sets $\mathsf{tpk} := \mathsf{pk}_{\mathsf{Enc}} = (\mathbf{B}, \mathbf{P}_1, \mathbf{P}_2)$ and $\mathsf{tsk} := \mathsf{sk}_{\mathsf{Enc}} = (\mathbf{S}_1, \mathbf{E}_1)$. GM receives tpk from TM and creates an empty reg table. Namely, $\mathsf{reg}[\mathsf{item}][\mathsf{bin}(j)][1] = \mathbf{0}^{nk}$and $\mathsf{reg}[\mathsf{item}][\mathsf{bin}(j)][2] = 0$ for $j = 1, \cdots, N-1$ and all item in the system, *i.e.,* it is epoch 0 and no users have joined the system yet[9]. Here, GM maintains multiple local counters c_{item} to keep track of the registered users for each item, which are all set initially to 0. Finally, GM outputs $\mathsf{gpk} = (\mathsf{pp}, \mathsf{mpk}, \mathsf{tpk})$ and $\mathsf{info} = \emptyset$.

$\mathsf{UKgen}(1^n)$: This algorithm is run by the user. It samples $\mathbf{x} \leftarrow \mathsf{KeyGen}_{\mathsf{LIT}}(1^n)$ where $\mathbf{x} \in [-\beta, \beta]^m$ and sets $\mathsf{usk} := \mathbf{x}$. It then computes $\mathsf{upk} := \mathbf{p} = \mathsf{bin}(\mathbf{A}\mathbf{x} \bmod q) \in \{0,1\}^{nk}$. Hereafter, the user is identified by his public key upk.

$\mathsf{Join} \leftrightarrow \mathsf{Issue}$: A user $(\mathsf{upk}, \mathsf{usk}) = (\mathbf{p}, \mathbf{x})$ requests to join the group that corresponds to item at epoch t. He sends $\mathbf{p}$ to GM. If GM accepts the request, it issues an identifier for this user, *i.e.,* $\mathsf{uid}_{\mathsf{item}} = \mathsf{bin}(c_{\mathsf{item}}) \in \{0,1\}^\ell$. The user's signing key for item is then set to $\mathsf{gsk}[\mathsf{uid}_{\mathsf{item}}][\mathsf{item}] = (\mathsf{uid}_{\mathsf{item}}, \mathbf{p}, \mathbf{x})$. Now, GM updates the Merkle tree via $\mathsf{TUpdate}_{\mathsf{item},\mathbf{A}}(\mathsf{uid}_{\mathsf{item}}, \mathbf{p})$, and sets $\mathsf{reg}[\mathsf{item}][\mathsf{uid}_{\mathsf{item}}][1] := \mathbf{p}$, $\mathsf{reg}[\mathsf{item}][\mathsf{uid}_{\mathsf{item}}][2] := t$. Finally, it increments the counter $c_{\mathsf{item}} := c_{\mathsf{item}} + 1$.

$\mathsf{RepUpdate}(\mathsf{gpk}, \mathsf{msk}, R, \mathsf{info}_{t_{\mathsf{current}}}, \mathsf{reg})$: This algorithm is be run by GM. Given a set R of users to be revoked, it first retrieves all the $\mathsf{uid}_{\mathsf{item}}$ associated to

[9] Recall that for simplicity of presentation, we assume the all items are provided to the GM. Our scheme is general enough so that the items can dynamically be added/removed from the system by GM.

each $\mathsf{upk} = \mathbf{p} \in R$. It then runs $\mathsf{TUpdate}_{\mathsf{item},\mathbf{A}}(\mathsf{reg}[\mathsf{item}][\mathsf{uid}_{\mathsf{item}}][1],\mathbf{0}^{nk})$ for all the retrieved $\mathsf{uid}_{\mathsf{item}}$. It finally recomputes $\mathbf{u}_{t_{\mathsf{new}},\mathsf{item}}$ and publishes

$$\mathsf{info}_{\mathsf{new}} = \left\{ \left(\mathbf{u}_{t_{\mathsf{new}},\mathsf{item}}, W_{\mathsf{item}} \right) \right\}_{\mathsf{item}},$$

where, $W_{\mathsf{item}} = \{w_{i,\mathsf{item}}\}_i$ and $w_{i,\mathsf{item}} \in \{0,1\}^\ell \times (\{0,1\}^{nk})^\ell$ is the witness that proves that $\mathsf{upk}_i = \mathbf{p}_i$ is accumulated in $\mathbf{u}_{t_{\mathsf{new}},\mathsf{item}}$. Here, the first ℓ-bit string term of the witness refers to the user identifier $\mathsf{uid}_{\mathsf{item}}$ associated to item.

$\mathsf{Sign}(\mathsf{gpk}, \mathsf{gsk}[\mathsf{item}][\mathsf{uid}_{\mathsf{item}}], \mathsf{info}_{t_{\mathsf{current}}}, \mathsf{item}, \mathsf{M})$: If $\mathsf{info}_{t_{\mathsf{current}}}$ does not contain a witness $w_{i,\mathsf{item}}$ with the first entry being $\mathsf{uid}_{\mathsf{item}} \in \{0,1\}^\ell$, return $\perp$. Otherwise, the user downloads $\mathbf{u}_{t_{\mathsf{current},\mathsf{item}}}$ and his witness $w_{i,\mathsf{item}}$ from $\mathsf{info}_{t_{\mathsf{current}}}$. Then, it computes $(c_1, c_2) \leftarrow \mathsf{Enc}_{\mathsf{Regev}}(\mathsf{tpk}, \mathsf{uid}_{\mathsf{item}})$ and the tag $\tau \leftarrow \mathsf{TAG}_{\mathsf{LIT}}(\mathsf{item}, \mathbf{x})$, where recall $\mathsf{usk} = \mathbf{x}$. Finally, it generates a NIZKAoK $\mathit{\Pi}_{\mathsf{Sign}} = (\{\mathsf{CMT}_i\}_{i=1}^{\kappa}, \mathsf{CH}, \{\mathsf{RSP}\}_{i=1}^{\kappa})$ for the relation R_{Sign} , where

$$\mathsf{CH} = \mathcal{H}_{\mathsf{Sign}}\big(\mathsf{M}, \{\mathsf{CMT}_i\}_{i=1}^{\kappa}, \mathbf{A}, \mathbf{u}, \mathcal{H}_{\mathsf{Tag}}(\mathsf{item}), \tau, c_1, c_2, \mathbf{B}, \mathbf{P}_1, \mathbf{P}_2\big) \in \{1,2,3\}^{\kappa},$$

and outputs the signautre $\mathit{\Sigma} = (\mathit{\Pi}_{\mathsf{Sign}}, \tau, c_1, c_2)$.

$\mathsf{Verify}(\mathsf{gpk}, \mathsf{info}_{t_{\mathsf{current}}}, \mathsf{item}, \mathsf{M}, \mathit{\Sigma})$: It verifies if $\mathit{\Pi}_{\mathsf{Sign}}$ is a valid proof. If so it outputs 1 and otherwise it outputs 0.

$\mathsf{Trace}(\mathsf{gpk}, \mathsf{tsk}, \mathsf{info}_{t_{\mathsf{current}}}, \mathsf{reg}, \mathsf{item}, \mathsf{M}, \mathit{\Sigma})$: It first runs $\mathsf{uid}_{\mathsf{item}} \leftarrow \mathsf{Dec}_{\mathsf{Regev}}((\mathbf{S}_1, \mathbf{E}_1), (c_1, c_2))$. Then, it generates a NIZAoK proof $\mathit{\Pi}_{\mathsf{Trace}}$ for the relation R_{Trace}.

$\mathsf{Judge}(\mathsf{gpk}, \mathsf{uid}_{\mathsf{item}}, \mathit{\Pi}_{\mathsf{Trace}}, \mathsf{info}_{t_{\mathsf{current}}}, \mathsf{item}, \mathsf{M}, \mathit{\Sigma})$: It verifies if $\mathit{\Pi}_{\mathsf{Trace}}$ is a valid proof. If so it outputs 1 and otherwise it outputs 0.

$\mathsf{Link}(\mathsf{gpk}, \mathsf{item}, (\mathsf{M}_0, \mathit{\Sigma}_0), (\mathsf{M}_1, \mathit{\Sigma}_1))$: It parses $\mathit{\Sigma}_0$ and $\mathit{\Sigma}_1$ and outputs $b \leftarrow \mathsf{Link}_{\mathsf{LIT}}(\tau_0, \tau_1)$, where $b = 1$ when it is linkable and 0 otherwise.

4.2 Security Analysis

We show that our reputation system is secure. Each of the following theorems correspond to the security definitions provided in Section 3.2, except for the correctness which can be easily checked to hold. Here, we only provide the high-level overview of some of the proofs that we believe to be of interest, and defer the formal proofs to the full version of the paper. The parameters that appear in the theorems are as provided in the above construction.

Theorem 3 (Anonymity). *Our reputation system is anonymous, assuming the hardness of the decision* $\mathsf{LWE}_{n,q,\chi}$ *problem.*

Proof Overview. We proceed in a sequence of hybrid experiments to show that $|\mathsf{Exp}^{\mathsf{Anon}\text{-}0}_{\mathsf{rep}\text{-}\mathsf{sys},\mathcal{A}}(n) - \mathsf{Exp}^{\mathsf{Anon}\text{-}1}_{\mathsf{rep}\text{-}\mathsf{sys},\mathcal{A}}(n)| \leq \mathsf{negl}$ for any PPT algorithm. The high level strategy is similar to the anonymity proof for the dynamic group signature scheme provided in [LNWX17], Lemma 2. Namely, for the challenge signature, we swap the user identifier $\mathsf{uid}_{\mathsf{item}}$ embedded in the ciphertexts (c_1, c_2) and the user's secret key usk embedded in the tag τ. The main difference between the proof of [LNWX17] is that for our reputation system we have to swap the tag in the challenge signature. For this, we use the tag indistinguishability property of the underlying tag scheme $\mathsf{LWE\text{-}LIT}$ presented in Theorem 1. This modification in the experiments are provided in Exp_5 of our proof.

Theorem 4 (Non-Frameability). *Our Reputation System is non-frameable, assuming the hardness of the* $\mathsf{SIS}_{n,m,q,1}$ *problem of the search* $\mathsf{faeLWE}_{m,n,q,\chi}$ *(or equivalently the search* $\mathsf{LWE}_{m-n,q,\chi}$*) problem.*

Proof Overview. For an adversary to win the experiment, he must output a tuple $(\mathsf{uid}^*_{\mathsf{item}^*}, \Pi^*_{\mathsf{Trace}}, \mathsf{info}_{t^*}, \mathsf{item}^*, \mathsf{M}^*, \Sigma^*)$ such that (informally): (i) the pair (M^*, Σ^*) links to some other message-signature pair (M, Σ) corresponding to item^* of an honest non-corrupt user or (ii) the proof Π^*_{Trace} traces the signature Σ^* back to some honest non-corrupt user. Since the latter case (ii) essentially captures the non-frameability of fully dynamic group signatures, the proof follows similarly to [LNWX17], Lemma 3. However, for case (i), we must use a new argument, since this is a security notion unique to reputation systems. In particular, we aim to embed a search LWE problem into the tag of the message-signature pair (M, Σ) of an honest non-corrupt user (where the simulator does not know the secret key usk) for which the adversary outputs a linking signature forgery (M^*, Σ^*). Due to the special nature of our LWE tag scheme, we can prove that if the signatures link, then the two secret keys $\mathsf{usk}, \mathsf{usk}^*$ embedded in the tags must be the same. Therefore, by extracting usk^* from the adversary's forgery, we can solve the search LWE problem. However, the problem with this approach is that since the simulator does not know usk, he will not be able to provide the adversary with this particular user's public key upk, which is defined as $\mathbf{A} \cdot \mathsf{usk} \mod q$. Our final idea to overcome this difficulty is by relying on the so called *first-are-error-less* LWE problem [BLP+13, ALS16], which is proven to be as difficult as the standard LWE problem. Namely, the simulator will be provided with $\mathbf{A} \cdot \mathsf{usk}$ as the error-less LWE samples and uses the remaining non-noise-less LWE samples to simulate the tags.

Theorem 5 (Public Linkability). *Our reputation system is unconditionally public-linkable.*

Proof Overview. We show that no such (possibly inefficient) adversary exists by assuming the linkability property of our underlying tag scheme $\mathsf{LWE\text{-}LIT}$ presented in Theorem 2, which holds unconditionally. Our strategy is to prove by contradiction. Assuming that an adversary winning the public-linkability experiment exists, we obtain two signatures Σ_0, Σ_1 on item such that the two tags τ_0, τ_1 associated with the signatures does not link, but the two tags embed the same user secret key usk (which informally follows from the $\Pi_{\mathsf{Trace},b}$ provided by the adversary). Then, by extracting the usk from the signatures produced by the adversary, we can use $(\tau_0, \tau_1, I = \mathsf{item}, \mathsf{sk} = \mathsf{usk})$ to win the linkability experiment of the tag scheme. Thus a contradiction.

The following two theorems follow quite naturally from the proofs of the dynamic group signatures schemes of [LNWX17]. At a high level, this is because the following security notions captures threats that should hold regardless of the presence of tags.

Theorem 6 (Traceability). *Our reputation system is traceable assuming the hardness of the* $\mathsf{SIS}_{n,m,q,1}$ *problem.*

Theorem 7 (Tracing Soundness). *Our reputation system is unconditionally tracing sound.*

Acknowledgement. We would like to thank the anonymous reviewers of FC 2018 for their valuable comments. The first author was funded by a research grant from the UK Government. The second author was partially supported by JST CREST Grant Number JPMJCR1302 and JSPS KAKENHI Grant Number 17J05603.

References

ACBM08. Elli Androulaki, Seung Choi, Steven Bellovin, and Tal Malkin. Reputation systems for anonymous networks. In *Privacy Enhancing Technologies*, pages 202–218, 2008. 2

ACPS09. Benny Applebaum, David Cash, Chris Peikert, and Amit Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In *CRYPTO*, pages 595–618. 2009. 4

ALS16. Shweta Agrawal, Benoît Libert, and Damien Stehlé. Fully secure functional encryption for inner products, from standard assumptions. In *Crypto*, pages 333–362, 2016. 4, 15

AT99. Giuseppe Ateniese and Gene Tsudik. Some open issues and new directions in group signatures. In *International Conference on Financial Cryptography*, pages 196–211, 1999. 3

BBS04. Dan Boneh, Xavier Boyen, and Hovav Shacham. Short group signatures. In *Crypto*, pages 41–55, 2004. 3

BCC$^+$16. Jonathan Bootle, Andrea Cerulli, Pyrros Chaidos, Essam Ghadafi, and Jens Groth. Foundations of fully dynamic group signatures. In *ACNS*, pages 117–136, 2016. 3, 7, 9

BJK15. Johannes Blömer, Jakob Juhnke, and Christina Kolb. Anonymous and publicly linkable reputation systems. In *Financial Cryptography*, pages 478–488, 2015. 2, 3, 9

BLP$^+$13. Zvika Brakerski, Adeline Langlois, Chris Peikert, Oded Regev, and Damien Stehlé. Classical hardness of learning with errors. In *STOC*, pages 575–584, 2013. 4, 15

BMW03. Mihir Bellare, Daniele Micciancio, and Bogdan Warinschi. Foundations of group signatures: Formal definitions, simplified requirements, and a construction based on general assumptions. In *EUROCRYPT*, pages 614–629, 2003. 3, 7

BS04. Dan Boneh and Hovav Shacham. Group signatures with verifier-local revocation. In *Proceedings of the 11th ACM conference on Computer and communications security*, pages 168–177, 2004. 3

BSS10. John Bethencourt, Elaine Shi, and Dawn Song. Signatures of reputation. pages 400–407, 2010. 2

BSZ05. Mihir Bellare, Haixia Shi, and Chong Zhang. Foundations of group signatures: The case of dynamic groups. In *CT-RSA*, pages 136–153, 2005. 3, 7, 9, 10

BW06. Xavier Boyen and Brent Waters. Compact group signatures without random oracles. In *EUROCRYPT*, pages 427–444, 2006. 3

Cam97. Jan Camenisch. Efficient and generalized group signatures. In *EUROCRYPT*, pages 465–479, 1997. 3

CG04. Jan Camenisch and Jens Groth. Group signatures: Better efficiency and new theoretical aspects. In *SCN*, pages 120–133, 2004. 3

CSK13. Sebastian Clauß, Stefan Schiffner, and Florian Kerschbaum. k-anonymous reputation. In *ACM SIGSAC*, pages 359–368, 2013. 2

CVH91. David Chaum and Eugène Van Heyst. Group signatures. In *EUROCRYPT*, pages 257–265, 1991. 3

Del00. Chrysanthos Dellarocas. Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior. In *ACM conference on Electronic commerce*, pages 150–157, 2000. 2

DMS03. Roger Dingledine, Nick Mathewson, and Paul Syverson. Reputation in p2p anonymity systems. In *Workshop on economics of peer-to-peer systems*, 2003. 2

EE17. Rachid El Bansarkhani and Ali El Kaafarani. Direct anonymous attestation from lattices. *IACR Cryptology ePrint Archive*, 2017. 4

GK11. Michael T Goodrich and Florian Kerschbaum. Privacy-enhanced reputation-feedback methods to reduce feedback extortion in online auctions. In *ACM conference on Data and application security and privacy*, pages 273–282, 2011. 2

GPV08. Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *STOC*, pages 197–206. ACM, 2008. 4

JI02. Audun Josang and Roslan Ismail. The beta reputation system. In *Proceedings of the 15th bled electronic commerce conference*, pages 2502–2511, 2002. 2

Ker09. Florian Kerschbaum. A verifiable, centralized, coercion-free reputation system. In *ACM workshop on Privacy in the electronic society*, pages 61–70, 2009. 2

KSGM03. Sepandar D Kamvar, Mario T Schlosser, and Hector Garcia-Molina. The eigentrust algorithm for reputation management in p2p networks. In *Proceedings of the 12th international conference on World Wide Web*, pages 640–651, 2003. 2

KTX07. Akinori Kawachi, Keisuke Tanaka, and Keita Xagawa. Multi-bit cryptosystems based on lattice problems. In *PKC*, pages 315–329, 2007. 12

KTX08. Akinori Kawachi, Keisuke Tanaka, and Keita Xagawa. Concurrently secure identification schemes based on the worst-case hardness of lattice problems. In *ASIACRYPT*, pages 372–389, 2008. 13

LNWX17. San Ling, Khoa Nguyen, Huaxiong Wang, and Yanhong Xu. Lattice-based group signatures: Achieving full dynamicity with ease. In *ACNS*, pages 293–312, 2017. 3, 7, 9, 13, 14, 15

MK14. Antonis Michalas and Nikos Komninos. The lord of the sense: A privacy preserving reputation system for participatory sensing applications. In *IEEE SCC)*, pages 1–6, 2014. 2

MKKSM13. Arash Molavi Kakhki, Chloe Kliman-Silver, and Alan Mislove. Iolaus: Securing online content rating systems. In *Proceedings of the 22nd international conference on World Wide Web*, pages 919–930, 2013. 2

MP13. Daniele Micciancio and Chris Peikert. Hardness of SIS and LWE with small parameters. In *CRYPTO*, pages 21–39. 2013. 4

NY90. M. Naor and M. Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In *STOC*, pages 427–437, 1990. 7

Pei09. Chris Peikert. Public-key cryptosystems from the worst-case shortest vector problem. In *STOC*, pages 333–342, 2009. 4

PVW08. Chris Peikert, Vinod Vaikuntanathan, and Brent Waters. A framework for efficient and composable oblivious transfer. In *CRYPTO*, pages 554–571, 2008. 12

Reg05. Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *STOC*, pages 84–93, 2005. 4, 7

RKZF00. Paul Resnick, Ko Kuwabara, Richard Zeckhauser, and Eric Friedman. Reputation systems. *Communications of the ACM*, (12):45–48, 2000. 2

RZ02. Paul Resnick and Richard Zeckhauser. Trust among strangers in internet transactions: Empirical analysis of ebay's reputation system. In *The Economics of the Internet and E-commerce*, pages 127–157, 2002. 2

SKCD16. Kyle Soska, Albert Kwon, Nicolas Christin, and Srinivas Devadas. Beaver: A decentralized anonymous marketplace with secure reputation. Cryptology ePrint Archive, Report 2016/464, 2016. 2

Ste06. Sandra Steinbrecher. Design options for privacy-respecting reputation systems within centralised internet communities. *Security and Privacy in Dynamic Environments*, pages 123–134, 2006. 2

VMG$^+$12. Bimal Viswanath, Mainack Mondal, Krishna P Gummadi, Alan Mislove, and Ansley Post. Canal: Scaling social network-based sybil tolerance schemes. In *Proceedings of the 7th ACM european conference on Computer Systems*, pages 309–322, 2012. 2

YSK$^+$09. Haifeng Yu, Chenwei Shi, Michael Kaminsky, Phillip B Gibbons, and Feng Xiao. Dsybil: optimal sybil-resistance for recommendation systems. In *Security and Privacy, 2009 30th IEEE Symposium on*, pages 283–298, 2009. 2

ZWC$^+$16. Ennan Zhai, David Isaac Wolinsky, Ruichuan Chen, Ewa Syta, Chao Teng, and Bryan Ford. Anonrep: Towards tracking-resistant anonymous reputation. In *NSDI*, pages 583–596, 2016. 2