A Privacy-aware Decentralized and Personalized Reputation System

Samiran Bag, Muhammad Ajmal Azad and Feng Hao

School of Computing Science, Newcastle University
Newcastle Upon Tyne, United Kingdom

Abstract

Reputation systems enable consumers to evaluate the trustworthiness of business entities (retailers, sellers) over the marketplace. In electronic marketplaces, the reputation of an business entity (retailer, seller) is computed by aggregating the "trust-scores" assigned to her by the parties who have had transactions with her. Most reputation systems designed for online marketplaces use all the available trust-scores to compute the reputation of business entity. However, in some scenarios, the consumer may wish to compute the reputation of a business entity by considering the trust-scores from a set of trustworthy participants, however, she does not want to disclose the identities of the users she trusts. There are two privacy protection challenges in the design of this kind of personalized reputation system: 1) protecting the set of trusted users of participants, and 2) protecting the trust-scores assigned by the participants in the trusted set. In this paper, we present a novel framework for computing the personalized global reputation of a business entity by considering the trust-scores from a set of trusted participants without disclosing identities of participants in the trusted set and their trust-scores. To this extent, the participants share cryptograms of their trust-scores for the business entity to the decentralized public bulletin board or tally center. These encrypted trust-scores are then used by the requester to compute the personalized reputation score of the business entity without leaking private information of participants in the system. We have analyzed the security and privacy properties of the scheme for the malicious adversarial model. The protocol has a linear message complexity, which proves that the system can be deployed in a real setup where such personalized recommendations may be required in practice. Furthermore, the system ensures correctness, privacy and security of trust-scores of participants in the trusted set under the malicious adversarial model.

Keywords: Personalized Reputation System, Secure Multiparty Computation, Reputation among Trusted peers, Personalized Recommendation, Online Marketplaces

1. Introduction

Online marketplaces have gained popularity over recent years. In an online marketplace, consumers get to purchase products and services from different retailers on the same platform. The marketplace is responsible for the management of consumers' transactions, building a bridge between consumers and retailers and ensuring the integrity of each transaction. The marketplace asks consumers to provide trust scores for the retailers they have interacted with, once the products or services have been delivered to them. The marketplace then aggregates the trust scores collected from the users, corresponding to a particular retailer to compute the aggregate trust or reputation score of the retailer or service provider[1]. The aggregated reputation score can assist consumers in deciding whether or not to interact with that retailer. Thus, the Reputation system plays a vital role in the interest of the consumers, protecting them from getting involved with perfidious entities in the marketplace [1]. Various online marketplaces (Amazon, eBay, Airbnb, Uber) collect feedbacks in the form of binary (0,1) or discrete (1 star to 5 star) ratings of their retailers from the users who have interacted with those retailers. The aggregate rating score is then shown at the retailer's web profile to delineate her trustworthiness. A high reputation scores implies that the retailer is trustworthy with respect to her past transactions and a low score suggests that she is somewhat not trustworthy.

The reputation systems used by many popular online marketplaces like eBay, Amazon, Uber, and Airbnb operate in a centralized setting, i.e. users submit feedbacks directly to the centralized system. The centralized system then combines these feedback scores into the global trust or reputation score. In this case, the centralized reputation system knows the private information of users (feedback values, identity of rater). Thus, this scheme only achieves weak privacy preservation goal. Furthermore, the centralized system requires that users should trust them for the protection of their private information such as rating scores, which normally is not the case in any realistic

*Email: {samiran.bag, Muhammad.Azad, feng.hao}@ncl.ac.uk

[1]The terms entity, business entity, retailer, and service provider refer to the business entity and are interchangeable. Similarly, the terms users, consumers, customers are referring the consumers and are also interchangeable.

setup as people are often reluctant to trust the centralized system with their sensitive information due to concern of misuse [2, 3, 4, 5]. Furthermore, the rating score in a centralized system is exposed to other users which also discourages users from submitting negative ratings because of fear of retaliation. In the context of online marketplaces, the major privacy concern is to hide the feedback of consumers, as such feedback can be used to learn the private information of the user. If the online marketplaces publish the rating scores along with real or pseudonymous identity of users, then the adversary (insurance companies, agents) can easily infer the purchasing patterns of consumers as well as their likes and dislikes [2]. Further, the centralized systems can also become malicious to sell private data of consumers for the financial benefits or can be attacked by the adversary for learning the private information of consumers. Recently, it has been shown that an adversary could learn the private information of consumers by correlating the information of consumers from the eBay network and their Facebook social network [2]. Furthermore, if ratings of a certain consumer for certain special medical products, medical specialists (sex specialists or other medical specialists), or certain marketplaces (providing specific services) are shown on the marketplace's website, then the adversary or another consumer could interpret which products the target consumer has recently bought or which specialist doctors he has visited recently. Exposure of this information would reveal sensitive information (such as particular health issues, purchased products, watched movies etc.). The privacy of feedback provider can be protected by encrypting the feedback values [6, 7, 8, 9, 10], but it comes with inherent challenges of verifiability and accountability.

The existing reputation systems for online marketplaces have three major shortcomings: 1) the raters submit their ratings to the centralized system in a plain form which makes users feel uncomfortable while rating others, specifically providing negative ratings, 2) the reputation systems do not consider all the available ratings while computing the aggregated reputation of the service provider, and 3) the aggregated reputation score is not verifiable. In a personalized and verifiable reputation system, any user of the marketplace may wish to know the aggregate reputation score of a particular service provider by considering the trust scores from a certain set of trusted users without disclosing the set of trusted users and without learning private ratings of participants in trusted set. Furthermore, the computed reputation value should be publicly verifiable without relying on any trusted system.

In this paper, we present the design of a novel decentralized and personalized reputation aggregation system called **PrivRep** that considers trust scores received from a set of trusted feedback providers while computing the aggregate reputation of a retailer or service provider. The design of PrivRep enables privacy protection of feedback providers. The system neither reveals the identities of the users in the set of trusted users nor divulges their rating scores. The architecture of PrivRep consists of three key components: the raters (users), the marketplace (Reputation Engine) and the public bulletin board or the tally system. The design of PrivRep is based on the semantics of homomorphic cryptographic system that enables feedback providers to collaborate in a secure and privacy-preserving fashion. To this extent, the feedback providers (users) homomorphically encrypt their rating scores and post them on the public bulletin board, which is digitally signed and can be traced back to its source. In addition to the cryptograms of rating scores, the feedback providers also provide non-interactive zero-knowledge proofs to prove that the rating scores provided by them are within the permissible range. The use of NIZK proof restricts the users from acting maliciously to circumvent the reputation system by providing out-of-bound ratings. The reputation engine then computes personalized reputation by utilizing the cryptograms in a personalized manner. The protocol meets privacy and correctness requirements under the standard malicious adversarial model (in which feedback providers not only try to provide out-of-range ratings but also act curiously to learn the private information of others).

The system comes with a unique trait that allows the Reputation Engine (RE) to consider the feedbacks only from a premeditated set of trustworthy feedback providers without revealing whether their ratings are counted during the computation of the final reputation or not. This freedom, however, does not allow the reputation engine to infringe the privacy of the users. It only allows a RE to disregard the feedbacks of suspicious users without letting them know about it. It is reasonable to assume that the RE is owned by the marketplace and would calculate the reputation of any service provider correctly as this has direct implications on her own eminence. It is thus in the interest of the reputation engine to only include the ratings of trustworthy feedback providers while computing the reputation of a service provider, discarding the ratings provided by other users with suspicious intent. Our scheme provides the required tool for achieving this sort of personalization in a privacy-preserving manner.

In summary, this paper makes the following contributions.

1. We present a decentralized and personalized reputation aggregation system that computes personalized reputation of particular service provider.
2. We analyze the privacy and security properties of our scheme under the malicious adversarial model.
3. We empirically evaluate the communication and computation overheads of the system.

The rest of the paper is organized as follows. In Section 2, we discuss the related work and compare our approach with other systems. In Section 3, we provide a background on the reputation systems and the homomorphic encryption technique used for developing our proposed protocol. In Section 4, we present the overview of the proposed approach and detail its operations. In Section 6, we analyze

the security and privacy properties of the proposed system. In Section 7, we empirically evaluate the computation and communication overheads of the system and compare it with other closely related systems in 8. We conclude the paper in Section 9.

2. Related work

A number of proposals have been presented in the context of privacy-preserving reputation management in a peer to peer network and online marketplace. These systems can be clustered into two major classes: systems that ensure the privacy of consumers through the use of a trusted third party systems, and systems that ensure privacy via the use of homomorphic cryptographic systems.

The centralized reputation system (used in many online marketplaces, such as Amazon, eBay, Alibaba, uber etc.) collects feedbacks from users in the form of rating scores (for example 0, 1, -1 in eBay, 0-5 star ratings in amazon etc.) and the free text comments. These feedback scores are then added together to yield the aggregate reputation of retailers over the marketplace. Section 3.1 provides a detailed discussion on the reputation systems used in online marketplaces. In the centralized system, the consumer has to trust the marketplace for the protection of his private data, which is a major privacy concern for the consumers. The centralized system addresses privacy of users through the use of pseudonymous identities [11, 8, 12]. However, pseudonymous identities cannot provide absolute privacy protection as user ratings can be correlated with user data from other sources to infer what a target user is buying in a particular online marketplace. Further, the feedback scores are exposed (in plaintext form) to a trusted system and other users as well. This may cause users to shy away from providing negative rating because of the fear of retaliation from other feedback providers [13]. Although a trusted party ensures privacy protection to some extent, it still poses serious privacy threat when the third party itself becomes malicious or gets compromised. Further, in online marketplaces and P2P (peer to peer) networks, it is more important to protect feedback ratings rather than the identity of the user. The privacy of users can also be protected by a onetime anonymization [14, 15, 9, 16] but anonymization would not provide any meaningful recommendation to other users of the system. Furthermore, the anonymization can also be subject to de-anonymization by correlating information from multiple sources [4, 3].

Several decentralized reputation systems have been proposed for reputation management. Largely, these systems are based on the use of secure multiparty computation and differential privacy. In [17] Schaub et al. proposed a blockchain-based reputation system in which consumers submit their unlinkable ratings of the marketplace or service provider they have interacted recently with. The credentials are issued by the service provider itself thus is prone to be misused by the malicious service provider to increase his reputation via feedback stuffing. Further, the system does not provide any personalized aggregation. In [18] Blömer et al. presented a secure and anonymous reputation system based on the semantics of group signature schemes developed by Boneh et al. [19]. The system provides anonymity by generating anonymous signed ratings. However, if the consumer provides a rating for the same product more than once, then his identity can be de-anonymized along with the ratings. In [20] Busom et al. protected ratings of the feedback provider through anonymization and by allowing only authorized consumers to participate in the rating process.

In [21] Hasan et al. proposed a system based on an additive homomorphic cryptography semantics and Zero-Knowledge proofs. The privacy of a given user can be preserved even in the presence of a large number of malicious users. However, the system requires a preselected set of users for protecting the privacy of the users. In [22] Pavlov et al. presented three different protocols that protect privacy by hiding the values of the responses submitted by the feedback providers. These protocols operate within a minimal system complexity where raters are honest but curious. The system has a computationally complexity of $O(n^3)$ messages for witness selection within a malicious adversarial model.

In [23] Dolev et al. presented a distributed approach for calculating a user's trustworthiness in specific time window by aggregating scores upon the request of an initiator. The scheme operates in a distributed manner, where each user calculates its trust value privately and independently. Though the protocol does not depend on a trusted third party, it does require the protocol initiator to be honest for ensuring the preservation of privacy and security properties of the protocol. In [24] Dimitriou et al. proposed a decentralized protocol that preserves privacy of feedback providers under the semi-honest adversarial model. The protocol allows participants to securely present their ratings in a way that preserves the privacy of individual ratings. In [6] Clark et al. proposed a delegation protocol for the users who want to leave the network. The designed protocol enables exiting participants to delegate their task of providing feedback scores to a set of other users in the network without affecting the privacy of the user who is delegating and the user who acted as the delegate. In [25, 26, 27] Azad et al. presented a decentralized reputation aggregation protocol for the weighted aggregation of reputation scores provided by the participants. Though the protocol considers the trust weights, it includes all the participants in the aggregation process. In [28] Visan et al. proposed a secure protocol for computing the reputation of peers in a peer to peer network using cryptographic systems and the anonymous identities.

A few proposals have also been proposed targeting decentralized marketplaces. In [29], Stefanos et al. proposed a reputation system for the decentralized marketplaces where the trust value the user gives to another user is quantifiable and is expressed as a bitcoin wallet. The system is completely decentralized and is based on the seman-

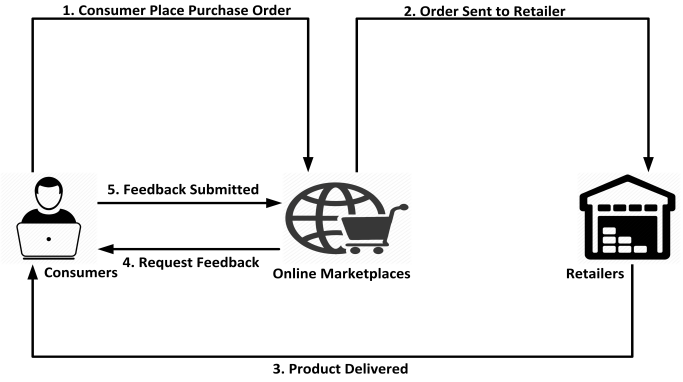

Figure 1: Transaction and Reputation Flow work for the Online Marketplaces.

tics of risks that how much a user would lose if she trusts someone and she behaves selfishly later on. However, it does not provide any information about which user is reputed in the network and how users' feedback is aggregated in the network. In [7] a decentralized anonymous marketplace is proposed that uses public ledger based consensus to aggregate the reputation of retailers while preserving the privacy of users. In [30] Moreno-Sanchez et al. proposed a secure privacy-preserving protocol that protects privacy of receiver and its value in a credit networks with the use of transaction obliviousness.

The existing systems have some limitations. Firstly, they require all the users to be online at the time of aggregation process (because they are primarily designed for P2P networks), therefore are not suitable for the online marketplaces. Secondly, they are secure under honest but curious adversarial model (rater provides honest rating). Therefore, they do not provide any defense when raters are collaborating with the service provider to increase her reputation in a malicious model. Thirdly, these systems are not providing personalized aggregation based on the interest or relationship network of the user requesting the aggregation.

To the best of our knowledge, no work has been presented that computes the personalized reputation of service providers in a marketplace while ensuring the privacy of feedback providers. The work presented in this paper has the following properties: 1) it considers feedback scores provided by a set of trusted users without disclosing who is trusted and who is not, 2) the system protects the privacy of participants under the malicious adversary model, 3) the protocol operations are performed in a decentralized way without no central trusted party learning individual user feedback, and 4) the system has a linear computational complexity which in itself is an improvement over other systems. Furthermore, the system can also be applied in a decentralized marketplace for the reputation management [31, 32].

3. Preliminaries

In this section, we discuss different aspects of a reputation system and describe cryptographic primitives that are essential in the design of the PrivRep reputation management system.

3.1. Marketplace Reputation Systems

Reputation represents collective information about the trustworthiness of users and retailers in the online marketplace. The reputation of a given retailer or user at the marketplace is computed as the sum or average of the feedback ratings assigned to retailers by their buyers based on the past transactions [33, 34]. In the marketplace the feedback ratings can be represented on the scale of {0, -1, 1} (eBay marketplace), 0-5 star (Amazon marketplace) that is assigned to each attribute describing the performance of a retailer. The attributes can be the information about whether the retailer has delivered the product on time, whether the product is the same as what is listed on the retailer's page and quality of services etc. Figure 1 shows the flow of events that take place in the marketplace when a consumer submits the purchase order to a particular retailer. When the product is delivered to the consumer, the online marketplace asks the consumer for the feedback against her recent interaction with the retailer. The user reports a feedback rating for the retailer and the marketplace then adds this trust score to the aggregated reputation of the retailer and displays this value on the web-page designated for the retailer.

Reputation systems can assist consumers to evaluate the trustworthiness of other retailers or users (consumer to consumer marketplaces or P2P) before making the transaction with a retailer or a user. This would also boost the sales for a particular retailer as well as people's trust on the marketplace if the reputation is well conceived and accurate in its functionality. The reputation system can be implemented as a centralized system or a distributed system depending on the requirement. In electronic marketplaces, the reputation systems are the centralized ones, where all the data regarding the rating history of the user is held by the central system. This is the normal practice that is being used in popular reputation systems (Amazon, eBay, Airbnb, uber etc.). On the other hand in P2P network, the reputation systems can be distributed [35, 36, 37] where user ratings are retained within the peers and used on-demand upon request from other peers. Table 1 presents a list of various reputation systems used in popular online marketplaces and P2P networks, classified according to their implementation-styles.

3.2. Secure Multiparty Computation

Secure Multiparty Computation enables computation of a mathematical function over inputs from multiple data providers, in a secure and privacy-preserving way. The

System	Architecture	Rating	Personalized
Amazon	Centralized	0-5	No
eBay	Centralized	0,1,-1	No
Bizrate	Centralized	0,1	No
Uber	Centralized	0-5	No
Airbnb	Centralized	0-5	No
Bazzar	Decentralized	0-5	No
Gunetella	Distributed	0,1	Yes
PrivRep	Decentralized	0,1 (0-5)	Yes

Table 1: Comparison of online reputation systems

computation would not reveal anything about the private data held by other parties involved in the computation. Secure Multiparty Computation (SMC) has been applied in a wide range of privacy-related problems like online e-voting [38, 39], statistical data analysis with privacy [40, 41, 42, 43], and privacy preserving user analytics [44, 45]. In SMC, we have a number of parties, say $p_1, p_2, \ldots, p_n$, each with a private input $x_1, x_2, \ldots, x_n$, respectively. The parties would like to compute a function (sum, average etc.) of their inputs, say $f(x_1, x_2, \ldots, x_n)$ without revealing their individual inputs to others. The system performs computation over the ciphertext and the final result matches the computation performed over the corresponding plain text.The SMC computation consists of three major algorithms: Key generation — responsible for generating public and private keys given security parameters, Encryption — responsible for generating the cipher text corresponding to the plain text using the public key, Decryption — responsible for deciphering the final result from the encrypted data using the private key.In this paper, we consider the additively homomorphic encryption system that does not rely on the trusted system for cryptographic operations while ensuring privacy of participants.

3.3. Cryptographic Building Blocks

The cryptographic primitive underpinning PrivRep uses the feedback randomization technique proposed for the decentralized and verifiable electronic voting [39, 46]. Let $N = \{1, 2, \ldots, n\}$ be the set of users in the network holding the feedback scores (0,1) for a certain retailer at the marketplace or online business. Let there be a DSA (or ECDSA)-like multiplicative cyclic group $\mathbb{Z}_q^*$, where q is a large prime. Let p be another large prime such that $p \mid q-1$. Let there be a subgroup G of order p of the group $\mathbb{Z}_q^*$, and let g be a generator of G. In order to provide the feedback for the business entity or retailer, the feedback provider first gets the unique token from the marketplace or business entity, then it generates a random value (private key) $Sk_i \in \mathbb{Z}_p$ for $i \in N$. The user then generates and publishes the public keys Pk to the PBB as follows.

$$Pk_i = g^{Sk_i}$$

When all the registered users in the system have generated and published their public keys on the public bulletin board (PBB), the encryption key (restructured key) of the user is computed as follows:

$$Y_i = \prod_{j \in N, j<i} Pk_j \Big/ \prod_{j \in N, j>i} Pk_j$$

The computation of Y_i as above ensures that

$$\prod_{i \in N} Y_i^{Sk_i} = 1. \quad (1)$$

Equation 1 ensures that $Y_i^{Sk_i}$ can be used as a randomizer for computing the secret feedbacks. This property is crucial to the design of our system. Anyone in the system is able to compute Y_i based on the published Pk_j values of other users P_j; $i \in [1, n] \setminus \{i\}$ and her own secret key Sk_i. The $Y_i's$ are used in our scheme to randomize the feedbacks provided by the users. This randomization technique serves as an encryption method that allows the RE to compute the aggregate of all feedbacks without learning anything about the individual feedbacks. We do not need any trusted party to de-randomize the encrypted feedbacks as simple multiplication of the encrypted feedbacks will do that as Equation 1 shows. This property is key to designing our decentralized scheme for reputation management.

3.4. Non-Interactive Zero-Knowledge Proof

A non-interactive zero-knowledge proof (NIZK), is a zero-knowledge proof of the statement where the sender (prover) can prove to the receiver (verifier) that a given statement is true, without revealing any information other than the fact that the statement is indeed correct. In PrivRep, we use NIZK proofs to prove the knowledge of secret parameters chosen by different participants, as well as to prove the well-formedness of different cryptograms. For generating a non-interactive proof we have applied the Fiat-Shamir heuristic to a standard zero-knowledge proof [47].

3.5. Public bulletin board

PBB is an implementation of a public authenticated channel. It is the storage that holds the public keys, encrypted feedback scores and other public information from the participants. Specifically, the PBB has the following information: the participant's public key, the identity of the business entity, encrypted feedback scores assigned by the participant, and the associated NIZK proofs and other crucial data associated with the protocol. The participants of the protocol have write access to the PBB, whereas, anyone else can only read data from it. Every datagram posted on the PBB has an associated signature that serves to prove the data authenticity. The adversary having access to the PBB would not be able to learn the individual feedback scores; instead the adversary would only learn the aggregated statistics about the business entity, like the RE and all the users. The PBB can also be a tally server held

by the marketplace since all operations performed by the PBB are based on public data inputs and are publicly verifiable. The use of PBB is common in privacy-preserving recommendation systems [48, 42] as well as in electronic voting systems [38, 39, 49].

3.6. Problem Definition

Here, the problem is to compute personalized reputation of the retailer in the online marketplace in a privacy-preserving and decentralized fashion. There are two distinct ways of viewing the problem. First, let us assume that the marketplace wishes to calculate the reputation of any service provider on the basis of the quantitative feedbacks provided by the users. The marketplace wants every user to participate in the process but wants to exclude the feedbacks provided by some users she does not consider trustworthy. These unreliable users may be floated by the retailer or a contender of the retailer to influence the outcome of the protocol in favor/ against the service provider in question. The marketplace can identify fake users by analyzing their purchasing patterns. Now, one trivial solution to this problem is to allow only trustworthy users to participate in the computation of the reputation. However, in order to do that, the marketplace will have to reveal the identities of trustworthy users which will, in turn, reveal the identities of suspicious users too. The marketplace cannot presumably be 100% accurate in deciding whether a user is trustworthy or not. Hence, revealing the list of users whom she considers suspicious may not go down well with the community of users. For example, a user who has recently joined the marketplace may be deemed as 'not-a-trustworthy-user' by the marketplace. On the other hand, an old-timer with a clean record can be considered as a trustworthy user. As such, other retailers may be skeptical about engaging with a user who is not deemed as trustworthy. This imposes the need to hide the identities of suspicious users. Therefore, the marketplace will need to allow all the users to participate in the above computation. However, the marketplace may be inclined to perform the above computation in such a fashion that gives her a degree of freedom to exclude the feedbacks of some of the 'not-trustworthy' users while keeping their identities secret. Our personalized reputation just serves this purpose. It allows every user to give feedbacks corresponding to a service provider. It also gives a degree of freedom to the marketplace to exclude suspicious users' feedbacks without requiring to reveal anything to any of the users and more importantly, without being able to compromise their privacy.

An alternative way of viewing this problem is the following. Let, U be the set of users, and R be the set of retailers in the marketplace. The users purchase products from the retailer via marketplace and leave feedback rating to the marketplace for their transactions. Let there be a new user $u_i \in U$ who wishes to calculate the personalized reputation of a particular retailer. The personalized reputation presents the aggregated view about the retailer according to the trusted friends of u_i. To this extent, u_i has to expose a list of his friends F to the marketplace for the personalized aggregation. However, u_i wants to ensure that neither the marketplace nor other users should be able to learn the list of her trusted friends. The proposed decentralized approach computes the personalized reputation of the retailer in a privacy-preserving way without disclosing the identities of the trusted friends of u_i.

In this paper, we have modeled the solution so as to address the first problem where the marketplace computes the personalized reputation of a service provider. However, it is trivial to adapt the scheme to yield a potential solution for the second problem too.

4. PrivRep System Design

This section presents an overview of the PrivRep system. Here, we describe various components of the PrivRep system, the adversarial model and the assumptions made in this paper.

4.1. System Components

Our Reputation system is composed of three different types of entities: users, service providers and the reputation engine (RE). The users are buyers who rate the service providers. The owner of the marketplace is the controller of the reputation engine. For example, in an online marketplace like Amazon and eBay, the users are the buyers, the retailers are service providers, and the marketplace itself (Amazon/eBay) is the handler of the reputation engine. Users rate their interacted services on a scale of 0 to 1. The reputation engine aggregates all the ratings from participants and calculates an overall personalized reputation of a service provider in the marketplace.

4.2. Threat Model

As discussed above, there are three different components in the system: user, the reputation engine, and the service provider. Our system should have a built-in mechanism to preserve the privacy of all users i.e. the system must ensure that the ratings provided by the users will remain private to the individual users and only the aggregate reputation which is calculated from these private ratings should be known publicly. This should hold true even in the case when the adversary has compromised the reputation engine and a number of users. The reputation engine should ensure that no service provider can artificially increase its rating by creating a large number of fake users and using them to influence the outcome of the reputation algorithm in its favor.

Similarly, it is also required that no malicious service provider can use feedback stuffing to reduce the rating of a contender. In order to achieve this, the scheme gives some degree of freedom to the reputation engine to exclude ratings of suspicious users without letting them about their exclusion. For example, if there are n users, the reputation

engine may choose Δ users whom she considers trustworthy and discard the ratings given by the remaining $n - \Delta$ users. Hence, the scheme provides freedom to the RE to choose some Δ users who will ultimately have a say in the actual calculation of the reputation of a service provider. The scheme assumes that the reputation engine does have such a mechanism to identify the set of users [50]. The scheme protects the privacy of users whose ratings are discarded. It also protects the ratings of trusted users whose rating are counted, as long as the overall reputation computed from them does not breach that privacy. If Δ is large enough, a user can be assured that her privacy will be preserved even when the overall reputation is published.

4.3. Assumptions

Every entity involved in the protocol agrees on a cyclic group G of p elements, p being a prime number. G is publicly known. Decisional Diffie-Hellman problem is assumed to be hard in G. Also, there is a publicly known generator $g \in G$. There is a publicly accessible bulletin board (similar to the one used in [39, 42, 49]), where the participants of the protocol can upload any information they may like to share with everyone else. This bulletin board acts as a notice board where only participants of the protocol can make a post. Each such posted data is digitally signed by the user who owns it. The user's public key can be used to check the authenticity of the posted data. Again, the bulletin board is 'append-only'. Hence, no user is allowed to overwrite any information posted either by herself or by any other user.

5. Protocol Operations

The feedback represents the trust-value a participant intends to assign to the service provider. The RE wants to compute the personalized reputation of a service provider by considering the ratings of a pre-selected set of trusted users. A rating is either 0 or 1. Though, here we consider only binary inputs from the users, the reputation scheme can be easily extended to allow any rating between 0 and a small integer. Our protocol allows the RE to execute an MPC protocol in a network comprising n users in such a fashion that only the ratings of a finite subset of the n users will be included in the final tally and the ratings of all other users will be discarded secretly and none of the n users will have any clue about whether her rating is counted or excluded. In order to make it fair, the RE needs to prove to the users that the number of actual users whose ratings will be counted is equal to a given number, say Δ. Also, the RE will have the privilege of choosing the users whose ratings will be counted at the end. We call them 'trusted users' throughout this paper. Let us assume there are n users, designated as $P_i : i \in [n]$. The RE will be able to calculate this tally $S = \sum_{i=1}^{n} I(i) \cdot v_i$, where $v_i \in \{0, 1\}$ is P_i's secret rating for the service provider SP and

$$I(i) = \begin{cases} 1, & \text{if } i \in \Gamma \\ 0, & \text{if } i \in [n] \setminus \Gamma \end{cases} \tag{2}$$

Here, Γ is the set of indices of the users trusted by the RE. Hence, $\Gamma \subseteq [n]$. Thus, the RE wants to calculate $S = \sum_{i \in \Gamma} v_i$, where v_i, the secret of of $P_i, i \in [n]$ is defined below:

$$v_i = \begin{cases} 1, & \text{if } P_i \text{ recommends the service provider} \\ 0, & \text{if } P_i \text{ does not recommend the service provider} \end{cases}$$

Upon completion of the process, the tally S will be computed as described in later sections. Once S is computed, the overall reputation/rating of the service provider SP on a scale of 1 to 10 can be calculated as : $\lfloor 10 * S/\Delta \rceil$.

Now, we begin describing the protocol operations. There is a public bulletin board which is a publicly accessible web-page on which every participant of the protocol posts their public information including public keys and/or encrypted ratings, Zero Knowledge proofs etc. Any information posted on the bulletin board by any participant cannot be overwritten. Again, those who do not participate in the protocol have only read access to the bulletin board. The protocol works as follow.

The RE generates two random values $\omega_1, \omega_2 \in \mathbb{Z}_p$ and publishes $\sigma_1 = g^{\omega_1}$ and $\sigma_2 = g^{\omega_2}$. The RE also publishes NIZK proofs of knowledge of $\omega_i = \log_g \sigma_i$ for $i = 1, 2$. This NIZK proofs can be constructed using Schnorr signature scheme [39]. Every user $P_i, i \in [n]$ generates a random $a_{1i}, b_{1i} \in \mathbb{Z}_p$, and publishes $g^{a_{1i}}$ and $g^{b_{1i}}$ on the bulletin board. They also publish NIZK proofs of knowledge of a_{1i}, b_{1i} using Schnorr's signature method. Thereafter, the RE generates two values as equation 3 and 4 :

$$g^{a_{2i}} = g^{(I(i) - a_{1i}\omega_1)/\omega_2} = (g^{I(i)}/g^{a_{1i}\omega_1})^{1/\omega_2} \tag{3}$$

and

$$g^{b_{2i}} = g^{(1 - b_{1i}\omega_1)/\omega_2} = (g/g^{b_{1i}\omega_1})^{1/\omega_2} \tag{4}$$

where $I(i)$ is the value from equation 2. Further, the RE also publishes Non-interactive Zero Knowledge Proof as

$$PW_i^1 \left[g^{\sum_{j=1}^{2} a_{ji}\omega_j} \in \{1, g\} : g^{a_{1i}}, g^{a_{2i}}, g^{\omega_1}, g^{\omega_2} \right]$$

This NIZK proof comprises a witness to the fact that $g^{\sum_{j=1}^{2} a_{ji}\omega_j}$ is either 1 or g, given $g^{a_{1i}}, g^{a_{2i}}, g^{\omega_1}$ and g^{ω_2}. The RE publishes this proof for all users $P_i, i \in [n]$. The RE also publishes another NIZK proof as

$$PW_{\text{total}} \left[g^{\sum_{i=1}^{n} \sum_{j=1}^{2} a_{ji}\omega_j} = g^{\Delta} : g^{a_{1i}}, g^{a_{2i}}, g^{\omega_1}, g^{\omega_2} \right]$$

This proof consists of a witness to the fact that $| \{i : g^{\sum_{j=1}^{2} a_{ji}\omega_j} = g\} | = \Delta$, given $g^{a_{1i}}, g^{a_{2i}}, \forall\, i \in [n], g^{\omega_1}$ and

g^{ω_2}. In other words PW_{total} proves that Δ is the total number of users whose ratings count at the end. The RE publishes a final NIZK proof to prove the fact that $\forall\, i \in [n]$, $g^{\sum_{j=1}^{2} b_{ji}\omega_j}$ equals g, given $g^{b_{1i}}, g^{b_{2i}}, i \in [n], g^{\omega_1}$ and g^{ω_2}. This proof is constructed as:

$$PW_i^2 \left[g^{\sum_{j=1}^{2} b_{ji}\omega_j} = g : g^{b_{1i}}, g^{b_{2i}}, g^{\omega_1}, g^{\omega_2} \right], \forall\, i \in [n]$$

The detail description for constructing these NIZK proofs is provided in the Appendix section.

In the next step, every user $P_i, i \in [n]$ generates a secret key $K_i = (x_{1i}, x_{2i})$ and publishes $Pub_i = (g^{x_{1i}}, g^{x_{2i}})$. The user publishes NIZK proofs of knowledge of the secret keys using Schnorr signature scheme [39]. Once all the users have published their public keys, then every user $P_i, i \in [n]$, computes the restructured keys (Y_{1i}, Y_{2i}), where

$$Y_{ji} = g^{y_{ji}} = \prod_{k=1}^{i-1} g^{x_{jk}} / \prod_{k=i+1}^{n} g^{x_{jk}};\ j = 1, 2 \tag{5}$$

Note that, the restructured keys can be computed by any user after downloading the public keys of other users from the bulletin board. The user P_i then computes a vector $C_i = (c_{1i}, c_{2i})$, where $c_{ji} = Y_{ji}^{x_{ji}} g^{b_{ji}\alpha_i} g^{a_{ji}v_i} = g^{x_{ji}y_{ji}} g^{b_{ji}\alpha_i} g^{a_{ji}v_i}, j = 1, 2,\ \alpha_i \in_R \mathbb{Z}_p$ is chosen by P_i and v_i is the secret rating of P_i. The P_i uploads the feedback $(\langle C_i, g^{\alpha_i} \rangle)$ and a NIZK proof of well-formedness as:

$$PW_i^3 \left[C_i : g^{x_{1i}}, g^{x_{2i}}, g^{y_{1i}}, g^{y_{2i}}, g^{b_{1i}}, g^{b_{2i}}, g^{\alpha_i}, g^{b_{1i}}, g^{b_{2i}} \right]$$

This NIZK proof provides a witness to the fact that each $c_{ji}, j = 1, 2$ is of the form $g^{x_{ji}y_{ji}} g^{b_{ji}\alpha_i} g^{a_{ji}v_i}$, where $g^{x_{1i}}, g^{x_{2i}}, g^{y_{1i}}, g^{y_{2i}}, g^{b_{1i}}, g^{b_{2i}}, g^{\alpha_i}, g^{b_{1i}}, g^{b_{2i}}$ are given and $v_i \in \{0, 1\}$. In other words, the NIZK proof proves that either of the two statements is correct but not both.

1) $c_{1i} = g^{x_{1i}y_{1i}} g^{b_{1i}\alpha_i} \wedge c_{2i} = g^{x_{2i}y_{2i}} g^{b_{2i}\alpha_i}$

2) $c_{1i} = g^{x_{1i}y_{1i}} g^{b_{1i}\alpha_i} g^{a_{1i}} \wedge c_{2i} = g^{x_{2i}y_{2i}} g^{b_{2i}\alpha_i} g^{a_{2i}}$

It is easy to see that the first statement corresponds to the case: $v_i = 0$, whereas the second one corresponds to the case: $v_i = 1$. The detailed construction of this NIZK proof is provided in the Appendix section.

Now, after the last user P_n (say) has uploaded her cryptogram C_n, anyone can compute $C = (\hat{c}_1, \hat{c}_2)$, where

$$\hat{c}_j = \prod_{i=1}^{n} c_{ji} \tag{6}$$

$$= \prod_{i=1}^{n} g^{x_{ji}y_{ji}} g^{a_{ji}v_i} g^{b_{ji}\alpha_i} \tag{7}$$

$$= g^{\sum_{i=1}^{n} a_{ji}v_i} g^{\sum_{i=1}^{n} b_{ji}\alpha_i} \tag{8}$$

Then, RE calculates an intermediate variable as

$$L = (\hat{c}_1)^{\omega_1} (\hat{c}_2)^{\omega_2} \tag{9}$$

$$= g^{\sum_{i=1}^{n} v_i(a_{1i}\omega_1 + a_{2i}\omega_2)} g^{\sum_{i=1}^{n} \alpha_i(b_{1i}\omega_1 + b_{2i}\omega_2)} \tag{10}$$

Note that $a_{1i}\omega_1 + a_{2i}\omega_2 = 1, \forall\, i \in \Gamma$ and $a_{1i}\omega_1 + a_{2i}\omega_2 = 0, \forall\, i \in [n] \setminus \Gamma$. Also $b_{1i}\omega_1 + b_{2i}\omega_2 = 1, \forall\, i \in \Gamma$. Hence,

$$L = g^{\sum_{i \in \Gamma} v_i} \prod_{i=1}^{n} g^{\alpha_i} = g^S \prod_{i=1}^{n} g^{\alpha_i} \tag{11}$$

The RE publishes L along with a NIZK proof of knowledge $PW_T\,[L : \hat{c}_1, \hat{c}_2, g^{\omega_1}, g^{\omega_2}]$. This proof provides a witness to the fact that $L = (\hat{c}_1)^{\omega_1} (\hat{c}_2)^{\omega_2}$, given $\hat{c}_1, \hat{c}_2, g^{\omega_1}$ and g^{ω_2}. Since the values of g^{α_i} are publicly known for all $i \in [1, n]$, anyone can calculate $g^S = L / \prod_{i=1}^{n} g^{\alpha_i}$, and from this can find the value of S through brute force search. Brute force search will be feasible, since, $S \in [0, \Delta]$.

The algorithmic steps of the protocol are outlined below. There are two phases of the protocol: the feedback collection phase and the tally phase. Figure 2 depicts the steps of the protocol. These steps are described below:

Phase I : Collecting Feedbacks.

Step I: The RE selects random $\omega_1, \omega_2 \in \mathbb{Z}_p$ and publishes $\sigma_1 = g^{\omega_1}$ and $\sigma_2 = g^{\omega_2}$. She also publishes NIZK proofs of knowledge of $\log_g \sigma_1$ and $\log_g \sigma_2$ computed using Schnorr signature protocol [39].

Step II: Each user $P_i, i \in [n]$ selects random $a_{1i}, b_{1i} \in \mathbb{Z}_p$ and publishes $g^{a_{1i}}$ and $g^{b_{1i}}$. P_i also publishes NIZK proofs of knowledge of the two random numbers computed using Schnorr signature protocol [39].

Step III: $\forall\, i \in [n]$ the RE computes $g^{a_{2i}} = (g^{I(i)} / g^{a_{1i}\omega_1})^{1/\omega_2}$ and $g^{b_{2i}} = (g^{1-I(i)} / g^{b_{1i}\omega_1})^{1/\omega_2}$, where $I(i)$ is described above. The RE publishes $g^{a_{2i}}$ and $g^{b_{2i}}$ along with the non-interactive zero knowledge proofs $PW_i^1 \left[g^{\sum_{j=1}^{2} a_{ji}\omega_j} \in \{1, g\} : g^{a_{1i}}, g^{a_{2i}}, g^{\omega_1}, g^{\omega_2} \right]$, and $PW_i^2 \left[g^{\sum_{j=1}^{2} b_{ji}\omega_j} = g : g^{a_{1i}}, g^{a_{2i}}, g^{\omega_1}, g^{\omega_2} \right], \forall\, i \in [n]$. The RE also publishes another NIZK proof $PW_{\text{total}} \left[g^{\sum_{i=1}^{n} \sum_{j=1}^{2} a_{ji}\omega_j} = g^{\Delta} : g^{b_{1i}}, g^{b_{2i}}, g^{\omega_1}, g^{\omega_2} \right]$. This Zero-knowledge proof proves that the total number of trusted users is Δ.

Step IV: Each user P_i selects a random key $(x_{1i}, x_{2i}) \in \mathbb{Z}_p^2$ and publishes the public key $Pub_i = (g^{x_{1i}}, g^{x_{2i}})$ along with the NIZK proof of knowledge of the corresponding secret key computed by means of Schnorr signature protocol [39].

Step V: Each user P_i selects random $\alpha_i \in \mathbb{Z}_p$ and publishes a feedback $\langle C_i, g^{\alpha_i} \rangle$, where $C_i = (c_{1i}, c_{2i})$, $c_{ji} = Y_{ji}^{x_{ji}} g^{b_{ji}\alpha_i} g^{a_{ji}v_i} = g^{x_{ji}y_{ji}} g^{b_{ji}\alpha_i} g^{a_{ji}v_i}, j = 1, 2,\ Y_{ji}$ is as defined in equation 5 and v_i is the secret rating of P_i. This feedback is published by P_i. P_i also publishes a NIZK proof $PW_i^3 \left[C_i : g^{x_{1i}}, g^{x_{2i}}, g^{y_{1i}}, g^{y_{2i}}, g^{b_{1i}}, g^{b_{2i}}, g^{\alpha_i}, g^{b_{1i}}, g^{b_{2i}} \right]$ as mentioned above.

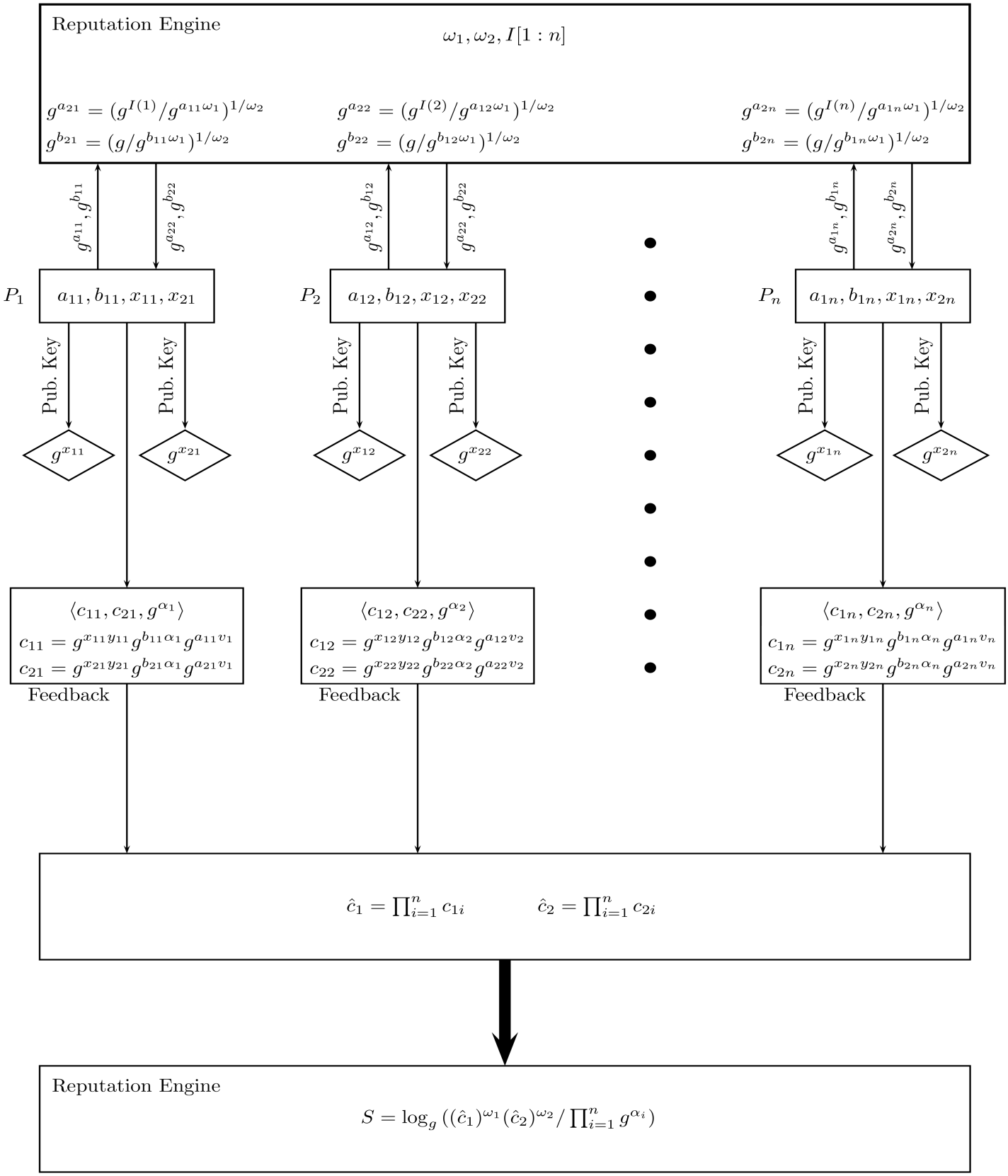

Figure 2: Flow diagram of the PrivRep protocol

Phase II : Computing the Overall Reputation.

Step I: The RE computes $C = (\hat{c}_1, \hat{c}_2)$, where $\hat{c}_j = \prod_{i=1}^{n} c_{ji} = \prod_{i=1}^{n} g^{x_{ji}y_{ji}} g^{a_{ji}v_i} g^{b_{ji}\alpha_i} = g^{\sum_{i=1}^{n} a_{ji}v_i} g^{\sum_{i=1}^{n} b_{ji}\alpha_i}$, for $j = 1, 2$. Then, the RE calculates $L = (\hat{c}_1)^{\omega_1}(\hat{c}_2)^{\omega_2} = g^S \prod_{i=1}^{n} g^{\alpha_i}$. Then the RE publishes L along with a NIZK proof of well-formedness of L.

Step II: Anyone can calculate $g^S = L/(\prod_{i=1}^{n} g^{\alpha_i})$ and from it can compute S via brute force search technique.

6. Security and Privacy Analysis

In this section, we prove the security properties of our scheme.

- The privacy of an untrusted user cannot be breached even if the RE and all the other users collude against her.
- Users cannot find whether they are considered as trusted or untrusted by the reputation system.
- The privacy of a trusted user is preserved if the overall tally does not allow the RE to deduce her rating. The tally has to be made public at the end, or the RE has to be able to compute the tally. If the tally on its own does not expose the rating of a trusted user, then her rating cannot be deduced. However, if everyone colludes against a trusted user, then the tally will be sufficient to find her rating which can be obtained by subtracting the tally by the partial tally of all colluding user. However, if the adversary cannot find her rating from the tally itself (i.e. if the adversary cannot collude with all $\Delta - 1$ trusted users), then her privacy cannot be infringed upon.

Note that, we have distinguished between the privacy of trusted and untrusted users. The ratings of untrusted users are not included in the tally but the ratings of the trusted ones are. So, the tally which is made public at the end will be based upon the ratings of trusted users. Like any election if the adversary colludes with all the $\Delta - 1$ other trusted users, finding the rating of the non-colluding user is trivial. However, since the rating of an untrusted user is not included in the tally and since our cryptographic protocol does not leak any data, the rating of an untrusted user cannot be found even if everyone colludes against her.

In Section 6.1, we show the first property of our privacy-preserving reputation scheme. In Section 6.2, we show that the second property is satisfied by our scheme, and lastly in Section 6.3 we show that the third property holds for our scheme.

6.1. Privacy of an Untrusted User

The main result of our first privacy-preserving property is Lemma 3. In this lemma, we show that no adversary will be able to compromise the secret rating of a user whose rating is made to be discarded. That is, if the RE chosen not include the rating of a certain user, her rating can not be deduced by any means. We use Assumption 3 to prove Lemma 3. In Lemma 2 and 1, we prove that Assumption 3 follows directly from the DDH (Decisional Diffie–Hellman) assumption. Hence, Lemma 3 holds under the DDH assumption.

Assumption 1. ***DDH Assumption [51]:*** *Given g, g^a, g^b and $\Omega \in \{g^{ab}, R\}$, where $R \xleftarrow{\$} G$ it is hard to distinguish whether $\Omega = g^{ab}$ or $\Omega = R$.*

Assumption 2. *Given g, g^a, g^b and $\Omega \in \{g^{ab}g^a, g^{ab}\}$ it is hard to find whether $\Omega = g^{ab}g^a$ or $\Omega = g^{ab}$.*

Lemma 1. *Assumption 1 implies assumption 2.*

Proof. According to assumption 1, $(g, g^a, g^b, g^{ab}) \overset{c}{\approx} (g, g^a, g^b, R)$. Similarly, $(g, g^a, g^b, g^{ab}g^a) \overset{c}{\approx} (g, g^a, g^b, R' * g^a) \overset{c}{\approx} (g, g^a, g^b, R')$. Thus, $(g, g^a, g^b, g^{ab}) \overset{c}{\approx} (g, g^a, g^b, R) \overset{c}{\approx} (g, g^a, g^b, R') \overset{c}{\approx} (g, g^a, g^b, g^{ab}g^a)$. □

Assumption 3. *Given $g, \omega_1, \omega_2, g^{a_1}, g^{a_2}, g^{b_1}, g^{b_2}, g^{\alpha}$, satisfying $\omega_1 a_1 + \omega_2 a_2 = 0, \omega_1 b_1 + \omega_2 b_2 = 1$, it is computationally hard to distinguish A and B, where A and B are given on Table 2.*

$g^{b_1\alpha}g^{a_1}$ $g^{b_2\alpha}g^{a_2}$	$g^{b_1\alpha}$ $g^{b_2\alpha}$
A	B

Table 2: Values of A and B in assumption 3.

Lemma 2. *Assumption 2 implies assumption 3.*

Proof. Let us assume that there is an adversary $\mathcal{A}$ that can distinguish between A and B. We show how we can construct an adversary $\mathcal{A}'$ which with the help of $\mathcal{A}$ can break the security of the assumption 2. $\mathcal{A}'$ receives g, g^a, g^b, g^c and $\Omega \in \{g^{ab}g^c, g^{ab}\}$ as input. $\mathcal{A}'$ has to find whether $\Omega = g^{ab}g^c$ or $\Omega = g^{ab}$. $\mathcal{A}'$ works as follows:
it implicitly sets $b_1 = a, \alpha = b, a_1 = c$. It selects random $\omega_1, \omega_2 \in \mathbb{Z}_p$ and produces $\Omega' = \frac{g^{\alpha/\omega_2}}{\Omega^{\omega_1/\omega_2}}$. Then $\mathcal{A}'$ sends (Ω, Ω') to $\mathcal{A}$. Note that if $\Omega = g^{ab}g^c = g^{b_1\alpha}g^{a_1}$, then $\Omega' = g^{b_2\alpha}g^{a_2}$. Alternately, if $\Omega = g^{ab} = g^{b_1\alpha}$, then $\Omega' = g^{b_2\alpha}$. Now, if $\mathcal{A}$ can identify the correct value of (Ω, Ω'), then $\mathcal{A}'$ can identify the correct Ω.

□

Lemma 3. *The secret rating of an untrusted user cannot be compromised.*

Proof. We show that an adversary $\mathcal{A}$ will not be able to find the rating of an untrusted user that does not reveal its rating. We assume that P_κ is an untrusted user whose rating $\mathcal{A}$ aims to find. $\mathcal{A}$ colludes with everyone to learn the rating v_i. $\mathcal{A}$ plugs in its suitable secret keys and ratings on behalf of all the colluding users. The secret feedback of P_i is (C_i, g^{α_i}), where $C_i = (c_{1i}, c_{2i})$, $c_{ji} = g^{x_{ji}y_{ji}} g^{b_{ji}\alpha_i} g^{a_{ji}v_i}, j = 1, 2$. Now $c_{j\kappa} = g^{x_{j\kappa}y_{j\kappa}} g^{b_{j\kappa}\alpha_\kappa} g^{a_{j\kappa}v_\kappa}$, $\forall\, j \in \{1, 2\}$. It is easy to see that $g^{x_{j\kappa}y_{j\kappa}} = 1/\left(\prod_{k\in[n]\setminus\{\kappa\}} g^{x_{jk}y_{jk}}\right)$. So, $\mathcal{A}$ can calculate $g^{x_{j\kappa}y_{j\kappa}}$ with the help of other users. Hence, $\mathcal{A}$ will be able to find v_κ only if it can distinguish between the two items A and B in Table 3

$g^{b_{1\kappa}\alpha_\kappa} g^{a_{1\kappa}}$ $g^{b_{2\kappa}\alpha_\kappa} g^{a_{2\kappa}}$	$g^{b_{1\kappa}\alpha_\kappa}$ $g^{b_{2\kappa}\alpha_\kappa}$
A	B

Table 3: Values of A and B in Lemma 3.

Using assumption 3, we can say that A and B are hard to be distinguished. □

6.2. Privacy of the Reputation Engine

In this section, we show that no user can deduce whether or not their rating is going to be included in the tally. That is the users cannot find whether they are trusted or not. The Lemma 6 proves that no one except the RE can find whether a certain user is trusted or untrusted. Lemma 6 is based on the assumption 5. In Lemma 5, we show that assumption 5 follows from assumption 4. Again, In Lemma 4, we prove that assumption 4 follows from the DDH assumption.

Assumption 4. *Given g, g^a, g^b and a $\Omega \in \{\Omega_1, \Omega_2\}$, where $\Omega_1 = g^{b/a}$ and $\Omega_2 = R$, it is hard to decide whether $\Omega = \Omega_1$ or $\Omega = \Omega_2$.*

Lemma 4. *Assumption 1 implies assumption 4.*

Proof. The proof can be found in [52]. □

Assumption 5. *Given g, g^a, g^b, m and a challenge $\Omega \in \{\Omega_1, \Omega_2\}$, where $\Omega_1 = (1/g^{am})^{1/b}$ and $\Omega_2 = (g/g^{am})^{1/b}$, it is hard to decide whether $\Omega = \Omega_1$ or $\Omega = \Omega_2$.*

Lemma 5. *Assumption 4 implies assumption 5.*

Proof. From Assumption 4, we can write $(g, g^a, g^b, m, (1/g^{am})^{1/b}) = (g, g^a, g^b, m, (g^{a/b})^{-m}) \overset{c}{\approx} (g, g^a, g^b, m, (R)^{-m}) \overset{c}{\approx} (g, g^a, g^b, m, R) \overset{c}{\approx} (g, g^a, g^b, m, g^{1/b} * R) \overset{c}{\approx} (g, g^a, g^b, m, g^{1/b}(g^{a/b})^{-m}) \overset{c}{\approx} (g, g^a, g^b, m, (g/g^{am})^{1/b})$ □

Lemma 6. *The users cannot find whether they are trusted or untrusted.*

Proof. We show that if a user can find whether it is trusted or untrusted, then it can break the security of assumption 5. The user P_i chooses a $g^{a_{1i}}$ and receives $g^{a_{2i}}$ which equals either $(1/g^{\omega_1 a_{1i}})^{1/\omega_2}$ or $(g/g^{\omega_1 a_{1i}})^{1/\omega_2}$. Now, according to assumption 5, these two values should be indistinguishable. Hence, the lemma holds. □

6.3. Privacy of Trusted Users

Here, we prove that the rating of a trusted user whose rating counts, cannot be deduced if it cannot be inferred from the tally itself. Here, we assume that the RE has colluded with some of the users to deduce the rating of a particular user $\hat{P}$. From the discussion of Section 6.1, we know that if $\hat{P}$ is untrusted, that is, if the RE chooses not to include the rating of $\hat{P}$, then the rating of $\hat{P}$ cannot be compromised. Alternately, if $\hat{P}$ is a trusted user then its rating is included in the tally. Now, Lemma 12 says that if there exists another honest trusted user $\tilde{P}$, such that the ratings of $\hat{P}$ and $\tilde{P}$ are different, that is one of them is 0 and the other one is 1, then the RE will not be able to deduce the rating of either $\hat{P}$ or $\tilde{P}$. Thus, if the RE colludes with all but two trusted users, then if the ratings of two trusted uncompromised users are different, the RE will not be able to find their ratings. So, the result of Lemma 12 amounts to saying that the adversary will not be able to deduce the rating of a trusted user if the tally does not allow her to infer it. Lemma 12 is based on the validity of assumption 10. In Lemma 11, we show that assumption 10 follows from the DDH assumption. We also require assumption 6, 7, 8, 9, and Lemma 7, 8, 9, 10 to show the validity of Lemma 11.

Assumption 6. *Given $g, g^{a_1}, g^{a_2}, g^{b_1}, g^{b_2}, g^{r_1}, g^{r_2}$ and a challenge $\Omega \in \{g^{a_1b_1+a_2b_2}g^{r_1}, g^{a_1b_1+a_2b_2}g^{r_2}\}$, it is hard to decide whether $\Omega = g^{a_1b_1+a_2b_2}g^{r_1}$ or $\Omega = g^{a_1b_1+a_2b_2}g^{r_2}$.*

Lemma 7. *Assumption 1 implies Assumption 6.*

Proof. $(g, g^{a_1}, g^{a_2}, g^{b_1}, g^{b_2}, g^{r_1}, g^{r_2}, g^{a_1b_1+a_2b_2}g^{r_1})$
$\overset{c}{\approx} (g, g^{a_1}, g^{a_2}, g^{b_1}, g^{b_2}, g^{r_1}, g^{r_2}, g^{a_1b_1} * g^{a_2b_2}g^{r_1})$
$\overset{c}{\approx} (g, g^{a_1}, g^{a_2}, g^{b_1}, g^{b_2}, g^{r_1}, g^{r_2}, R * g^{a_2b_2}g^{r_1})$
$\overset{c}{\approx} (g, g^{a_1}, g^{a_2}, g^{b_1}, g^{b_2}, g^{r_1}, g^{r_2}, R)$
$\overset{c}{\approx} (g, g^{a_1}, g^{a_2}, g^{b_1}, g^{b_2}, g^{r_1}, g^{r_2}, R * g^{r_2})$
$\overset{c}{\approx} (g, g^{a_1}, g^{a_2}, g^{b_1}, g^{b_2}, g^{r_1}, g^{r_2}, R * g^{a_2b_2}g^{r_2})$
$\overset{c}{\approx} (g, g^{a_1}, g^{a_2}, g^{b_1}, g^{b_2}, g^{r_1}, g^{r_2}, g^{a_1b_1} * g^{a_2b_2}g^{r_2})$
$\overset{c}{\approx} (g, g^{a_1}, g^{a_2}, g^{b_1}, g^{b_2}, g^{r_1}, g^{r_2}, g^{a_1b_1+a_2b_2}g^{r_2})$. □

Assumption 7. *Given $g, \omega, \omega', g^{a_1}, g^{a_2}, g^{b_1}, g^{b_2}, g^{r_1}, g^{r_2}, g^{a_1'}, g^{a_2'}$, $g^{r_1'}, g^{r_2'}$, such that $\omega a_1 + \omega' a_1' = \omega a_2 + \omega' a_2' = \omega b_1 + \omega' b_1' = \omega b_2 + \omega' b_2' = \omega r_1 + \omega' r_1' = \omega r_2 + \omega' r_2' = 1$, and a challenge $\Omega \in \{\Omega_1, \Omega_2\}$, where $\Omega_1 = (g^{a_1b_1+a_2b_2}g^{r_1}, g^{a_1'b_1+a_2'b_2}g^{r_1'})$ and $\Omega_2 = (g^{a_1b_1+a_2b_2}g^{r_2}, g^{a_1'b_1+a_2'b_2}g^{r_2'})$, it is hard to decide whether $\Omega = \Omega_1$ or $\Omega = \Omega_2$.*

Lemma 8. *Assumption 6 implies assumption 7.*

Proof. Let us assume $\mathcal{A}$ is the adversary against assumption 7. We show how $\mathcal{A}$ can be used to construct another adversary $\mathcal{A}'$ against assumption 6. $\mathcal{A}'$ works as follows: it receives $g, g^{a_1}, g^{a_2}, g^{b_1}, g^{b_2}, g^{r_1}, g^{r_2}$ and a challenge $\Omega \in \{g^{a_1b_1+a_2b_2}g^{r_1}, g^{a_1b_1+a_2b_2}g^{r_2}\}$. Then it generates two random $\omega_1, \omega_2 \in \mathbb{Z}_p$ and computes the following items:
$g^{a_1'} = (g/(g^{a_1})^{\omega})^{1/\omega'}$, $g^{a_2'} = (g/(g^{a_2})^{\omega})^{1/\omega'}$,
$g^{r_1'} = (g/(g^{r_1})^{\omega})^{1/\omega'}$, $g^{r_2'} = (g/(g^{r_2})^{\omega})^{1/\omega'}$
$A = \Omega, A' = \frac{(g^{b_1}g^{b_2}g)^{1/\omega'}}{\Omega^{\omega/\omega'}}$.
Note that if $\Omega = g^{a_1b_1+a_2b_2}g^{r_1}$ then $(A, A') = (g^{a_1b_1+a_2b_2}g^{r_1}, g^{a_1'b_1+a_2'b_2}g^{r_1'})$. Alternately, if $\Omega = g^{a_1b_1+a_2b_2}g^{r_2}$ then $(A, A') = (g^{a_1b_1+a_2b_2}g^{r_2}, g^{a_1'b_1+a_2'b_2}g^{r_2'})$. Now, $\mathcal{A}'$ sends $g, \omega, \omega', g^{a_1}, g^{a_2}, g^{b_1}, g^{b_2}, g^{r_1}, g^{r_2}, g^{a_1'}, g^{a_2'}, g^{r_1'}, g^{r_2'}$ and (A, A') to $\mathcal{A}$. If $\mathcal{A}$ can identify the correct (A, A'), then $\mathcal{A}'$ can use the same to identify the correct Ω. □

Assumption 8. *Given g, g^a, g^b, g^c and $\Omega \in \{g^{-ab}, g^{-ab}g^c\}$ it is hard to decide whether $\Omega = g^{-ab}$ or $\Omega = g^{-ab}g^c$.*

Lemma 9. *Assumption 1 implies assumption 8.*

Proof. If there exists an adversary $\mathcal{A}$ against assumption 8, then it could be to construct an adversary $\mathcal{A}'$ who with the help of $\mathcal{A}$ can break the security of assumption 1. $\mathcal{A}$ works as follows:
it receives as input g, g^a, g^b and a challenge $\Omega \in \{g^{ab}, g^{ab}g^c\}$. It computes $g^{-c} = 1/g^c, \Omega' = \Omega^{-1}$. It now sends g^a, g^b, g^{-c} and Ω^{-1} to $\mathcal{A}$. Note that, if $\Omega = g^{ab}$, $\Omega' = g^{-ab}$ and if $\Omega = g^{ab}g^c$, then $\Omega' = g^{-ab}g^{-c}$. Now if $\mathcal{A}$ can distinguish the two possible values of Ω', $\mathcal{A}$ can use that to find whether $\Omega = g^{ab}$ or $\Omega = g^{ab}g^c$. □

Assumption 9. *Given $g, g^a, g^b, g^c, g^d, g^e, g^f, g^h$ and $\Omega \in \{\Omega_1, \Omega_2\}$, where $\Omega_1 = (g^{-ab}g^hg^{cd}, g^{ab}g^{ef}), \Omega_2 = (g^{-ab}g^{cd}, g^{ab}g^hg^{ef})$, it is hard to decide whether $\Omega = \Omega_1$ or $\Omega = \Omega_2$.*

Lemma 10. *Assumption 8 implies assumption 9.*

Proof. According to assumption 8, given g, g^a, g^b and g^h, $g^{-ab} \overset{c}{\approx} g^{-ab}g^h$. Thus, $\Omega_1 = (g^{-ab}g^hg^{cd}, g^{ab}g^{ef}) = (g^{-ab}g^hg^{cd}, \frac{g^hg^{ef}}{g^{-ab}g^h}) \overset{c}{\approx} (g^{-ab}g^{cd}, \frac{g^hg^{ef}}{g^{-ab}}) = \Omega_2$. □

Assumption 10. *Given $g, A = (g^{a_1}, g^{a_2}), B = (g^{b_1}, g^{b_2}), \mathcal{C}_1 = (g^{c_{11}}, g^{c_{21}}), \mathcal{C}_2 = (g^{c_{12}}, g^{c_{22}}), \Sigma_1 = g^{\alpha_1}, \Sigma_2 = g^{\alpha_2}, \mathcal{R}_1 = (g^{r_{11}}, g^{r_{21}}), \mathcal{R}_2 = (g^{r_{12}}, g^{r_{22}}), \omega = (\omega_1, \omega_2)$ such that $\sum_{i=1}^2 \omega_i r_{ij} = \sum_{i=1}^2 \omega_i c_{ij} = 1, j \in \{1, 2\}$, and $\Omega \in \{\Omega_1, \Omega_2\}$, where $\Omega_1 = (U_1, U_2), \Omega_2 = (V_1, V_2)$,
$U_i = (g^{-a_ib_i}g^{c_{i1}}g^{r_{i1}\alpha_1}, g^{a_ib_i}g^{r_{i2}\alpha_2})$,
$V_i = (g^{-a_ib_i}g^{r_{i1}\alpha_1}, g^{a_ib_i}g^{c_{i2}}g^{r_{i2}\alpha_2})$. It is hard to decide whether $\Omega = \Omega_1$ or $\Omega = \Omega_2$.*

Lemma 11. *Assumption 1 implies assumption 10.*

Proof. Using assumption 9, we can write
$U_i = (g^{-a_ib_i}g^{c_{i1}}g^{r_{i1}\alpha_1}, g^{a_ib_i}g^{r_{i2}\alpha_2}) \overset{c}{\approx} (g^{-a_ib_i}g^{r_{i1}\alpha_1}, g^{a_ib_i}g^{c_{i1}}g^{r_{i2}\alpha_2}) = (g^{-a_ib_i}g^{r_{i1}\alpha_1}, \frac{g^{r_{i1}\alpha_1+r_{i2}\alpha_2}g^{c_{i1}}}{g^{-a_ib_i}g^{r_{i1}\alpha_1}})$. Using assumption 7, we can write $(g^{-a_ib_i}g^{r_{i1}\alpha_1}, \frac{g^{r_{i1}\alpha_1+r_{i2}\alpha_2}g^{c_{i1}}}{g^{-a_ib_i}g^{r_{i1}\alpha_1}}) \overset{c}{\approx} (g^{-a_ib_i}g^{r_{i1}\alpha_1}, \frac{g^{r_{i1}\alpha_1+r_{i2}\alpha_2}g^{c_{i2}}}{g^{-a_ib_i}g^{r_{i1}\alpha_1}}) = (g^{-a_ib_i}g^{r_{i1}\alpha_1}, g^{a_ib_i}g^{c_{i2}}g^{r_{i2}\alpha_2}) = V_i, \forall\, i \in \{1, 2\}$. Since, assumption 9 and 7 can be reduced to assumption 1, the lemma holds. □

Lemma 12. *No PPT adversary, having an alliance with the RE, can distinguish between two bulletin boards where the ratings of two trusted users P_γ and P_δ ($\gamma < \delta$), who rated the service provider differently, are mutually interchanged.*

Proof. Let us assume that A and B are two bulletin boards where v_γ and v_δ are interchanged. Since, P_γ and P_δ have rated the service provider differently, $v_\gamma + v_\delta = 1$. We show that if there exists an adversary $\mathcal{A}$ that can distinguish between A and B, it could be used to construct an adversary $\mathcal{A}'$ against assumption 10. $\mathcal{A}'$ works as under:
It receives as input $g, A = (g^{a_1}, g^{a_2}), B = (g^{b_1}, g^{b_2}), \mathcal{C}_1 = (g^{c_{11}}, g^{c_{21}}), \mathcal{C}_2 = (g^{c_{12}}, g^{c_{22}}), \Sigma_1 = g^{\alpha_1}, \Sigma_2 = g^{\alpha_2}, \mathcal{R}_1 = (g^{r_{11}}, g^{r_{21}}), \mathcal{R}_2 = (g^{r_{12}}, g^{r_{22}}), \omega = (\omega_1, \omega_2)$ such that $\sum_{i=1}^2 \omega_i r_{ij} = \sum_{i=1}^2 \omega_i c_{ij} = 1, j \in \{1, 2\}$, and a challenge $\Omega \in \{\Omega_1, \Omega_2\}$, where $\Omega_1 = \{U_i : i = 1, 2\}, \Omega_2 = \{V_i : i = 1, 2\}, U_i = (g^{-a_ib_i}g^{c_{i1}}g^{r_{i1}\alpha_1}, g^{a_ib_i}g^{r_{i2}\alpha_2}), V_i = (g^{-a_ib_i}g^{r_{i1}\alpha_1}, g^{a_ib_i}g^{c_{i2}}g^{r_{i2}\alpha_2})$. $\mathcal{A}'$ has to identify whether $\Omega = \Omega_1$ or $\Omega = \Omega_2$. $\mathcal{A}'$ lets $\mathcal{A}$ choose a subset $\Lambda \subseteq [n]\backslash\{\gamma, \delta\}$. $\mathcal{A}$ selects all secret keys and ratings for all users $\{P_i : i \in \Lambda\}$. For the users $P_i : i \in [n] \setminus (\Lambda \cup \{\gamma, \delta\})$, $\mathcal{A}'$ selects the secret keys and ratings randomly. Then $\mathcal{A}'$ sets $X_\gamma = (g^{a_1}, g^{a_2})$ and $X_\delta = (g^{b_1}, g^{b_2})$. That is $\mathcal{A}'$ implicitly sets $(x_{1\gamma}, x_{2\gamma}) = (a_1, a_2)$ and $(x_{1\delta}, x_{2\delta}) = (b_1, b_2)$. We will see later that $\mathcal{A}'$ implicitly sets $a_{j\gamma} = c_{j1}$ and $a_{j\delta} = c_{j2}, j = 1, 2$. $\mathcal{A}'$ also implicitly sets $b_{j\gamma} = r_{j1}$ and $b_{j\delta} = r_{j2}, j = 1, 2$. For all other $n-2$ compromised users, either $\mathcal{A}'$ or $\mathcal{A}$ chooses $g^{a_{1i}}$ randomly and sets $g^{a_{2i}} = (g^{\theta_i}/g^{a_{1i}\omega_1})^{1/\omega_2}$, where $\theta_i \in \{0, 1\}$ and is chosen by either $\mathcal{A}'$ or $\mathcal{A}$. Similarly they choose $g^{b_{1i}}$ randomly and set $g^{b_{2i}} = (g/g^{b_{1i}\omega_1})^{1/\omega_2}$. Both $\mathcal{A}'$ and $\mathcal{A}$ know the values of a_{1i}, b_{1i}, ω_1 and ω_2, which allow them to compute $g^{a_{2i}}$ and $g^{b_{2i}}$ from randomly chosen $g^{a_{1i}}$ and $g^{b_{1i}}$ respectively. $\mathcal{A}$ generates the cryptograms for all the users with indices from the set Λ. It generates secret key $(x_{1i}, x_{2i}) \in \mathbb{Z}_p^2$ and publishes the public key $X_i = (g^{x_{1i}}, g^{x_{2i}})$ for all $i \in \Lambda$. $\mathcal{A}$ also produces cryptogram $C_i = (c_{1i}, c_{2i})$, for all $i \in \Lambda$, where $c_{ji} = g^{x_{ji}y_{ji}}g^{a_{ji}v_i}g^{b_{ji}\alpha_i}, j \in \{1, 2\}$, $g^{y_{ji}} = g^{\sum_{k=1}^{i-1} x_{jk} - \sum_{k=i+1}^{n} x_{jk}} = \prod_{k=1}^{i-1} g^{x_{jk}} / \prod_{k=i+1}^{n} g^{x_{jk}}$ and $\alpha_i \in \mathbb{Z}_p$. $\mathcal{A}'$ also does the same thing for all users $P_i : i \in [n] \setminus (\Lambda \cup \{\gamma, \delta\})$. Let, $\Omega = (w_1, w_2), \omega_j = (w_j^1, w_j^2), \forall\, j \in \{1, 2\}$. Then $\mathcal{A}'$ generates C_γ and C_δ as follows: $c_{j\gamma} = K_{j1} * w_j^1$ and $c_{j\delta} = K_{j2} * w_j^2, \forall\, j \in \{1, 2\}$.
Here, $K_{j1} = (g^{a_j})^{\sum_{k=1}^{\gamma-1} x_{jk} - \sum_{k=\gamma+1}^{\delta-1} x_{jk} - \sum_{k=\delta+1}^{n} x_{jk}}$ and $K_{j2} = (g^{b_j})^{\sum_{k=1}^{\gamma-1} x_{jk} + \sum_{k=\gamma+1}^{\delta-1} x_{jk} - \sum_{\delta+1}^{n} x_{jk}}$. Since, the secret keys of users $P_i : i \in [n] \setminus \{\gamma, \delta\}$ are chosen by either $\mathcal{A}$ or $\mathcal{A}'$, they know the values of $x_{jk}, \forall\, j \in \{1, 2\}$ and $\forall\, k \in [n] \setminus \{\gamma, \delta\}$. Hence, they can calculate the values of $K_{j1}, K_{j2}, \forall\, j \in \{1, 2\}$, and thus can compute C_γ and C_δ. Note that, if $\Omega = \Omega_1$, then C_γ corresponds to the rating $v_\gamma = 1$ and C_δ corresponds to the rating $v_\delta = 0$. On the other hand if $\Omega = \Omega_2$, then $v_\gamma = 0$ and $v_\delta = 1$. If $\mathcal{A}$ can

Entity	Computational overhead (number of exponentiations)				Communication overhead		
	Initialization	Feedback	NIZK Proof	Tallying	Initialization	Feedback	NIZK Proof
User	4	5	26	-	4	3	34
RE	$2n+3$	-	$15n+10$	2	$4n+2$	-	$18n+18$

Table 4: Protocol Overhead

Operation	Computation Cost	Communication Cost
Setup	-	$4n+2$
Key	-	$2n$
Feedback	-	$3n$
NIZKP	$56n+18$ exponentiations	$52n+18$
Tallying	1 exponentiation	1

Table 5: Cost for Public Verification

distinguish between these two cases, then $\mathcal{A}'$ can use the same to identify whether $\Omega = \Omega_1$ or $\Omega = \Omega_2$. Hence, the result holds. □

7. Evaluation

In this section, we present the computation and bandwidth complexity of protocol and provide the benchmark for each cryptographic operation.

7.1. Complexity Evaluation

In this section, we calculate the computational and the communication overhead of our scheme. In our scheme, the most expensive operation is the exponentiation. We measure the computational overheads in terms of the number of exponentiations done by an entity. First, we measure the computational overhead for the RE. The RE selects two random elements ω_1 and ω_2 and publishes g^{ω_1} and g^{ω_2}, which requires two exponentiations. Again, computing $g^{a_{2i}}$ and $g^{b_{2i}}$ from $g^{a_{1i}}$ and $g^{b_{1i}}$ requires one exponentiation each, provided g^{1/ω_2} has been computed beforehand. The RE computes them for all $i \in [n]$. Thus, the total number of exponentiations required will be $2n$. The RE computes the NIZK proofs of knowledge of ω_1 and ω_2 which require one exponentiation each. Again the RE computes NIZK proofs of well-formedness of $g^{a_{2i}}$ and $g^{b_{2i}}$ for all $i \in [n]$. In order for generating NIZK proof of well-formedness of $g^{a_{2i}}$, the RE needs to do at most $4+7=11$ exponentiations (4 for the correct proof and 7 for the simulated proof). Similarly for producing the NIZK proof of well-formedness of $g^{b_{2i}}$, the RE needs to do 4 exponentiations. Thus, the RE needs to do 15 exponentiations for constructing the proof of well-formedness of $g^{a_{2i}}$ and $g^{b_{2i}}$ for any $i \in [n]$. So, for all users, the total number of exponentiation comes to be $15n$. The RE also produces NIZK proof $PW_{\text{total}}\left[g^{\sum_{i=1}^{n}\sum_{j=1}^{2} a_{ji}\omega_j} = g^{\Delta} : g^{a_{1i}}, g^{a_{2i}}, g^{\omega_1}, g^{\omega_2}\right]$ which requires 4 exponentiations. Then, at the end of the protocol, the RE needs to do 2 exponentiations in order to calculate L. The RE also generates NIZK proof of well-formedness of L which requires 4 exponentiations. So the grand total of the number of exponentiations required for computing all NIZK proofs is equal to $15n+10$ which is $O(n)$. Note that, we did not count the overhead of checking the NIZK proofs in the calculation of this overhead. These are separately calculated below.

Now we measure the overhead on each of the users who participate in the protocol as users. In order to compute $g^{x_{1i}}, g^{x_{2i}}, g^{a_{1i}}, g^{b_{1i}}$ the user needs to do 4 exponentiations. Each user does 1 exponentiation to compute g^{α_i} and 4 others in order to compute the feedback C_i discussed in section 5. The zero knowledge proof of knowledge of the secret key requires 2 exponentiations. The zero-knowledge proofs of feedback well-formedness and knowledge of secret parameters require 26 exponentiations. So, the total number of exponentiation needed by one user is 35. It can be seen that the communication overhead at the RE is dominated by the size of the NIZK proofs. The RE consumes a bandwidth of $18n+18$ for communicating all NIZK proofs to the bulletin board. Each of the users requires communicating 34 items as part of NIZK argument. Hence a user requires constant communication bandwidth for posting all the information at her disposal to the bulletin board. Hence, the total communication overhead of executing the entire protocol comes to be $O(n)$. Table 6 provides the computational and communication overheads in a tabular format. Furthermore, Table 7 shows the overhead on a public verifier for verifying all the arguments posted on the bulletin board including the tally. A public verifier needs to do $56n+18$ exponentiations for verifying all the NIZK arguments posted on the bulletin board which includes setup parameters, feedbacks etc. Again if the RE posts the tally and L, the public verifier can verify the tally by doing one exponentiation.

7.2. Microbenchmarks

The cryptographic operations of PrivRep include two major operations: generating cryptograms of responses and generating non-interactive zero-knowledge proofs. Each of these operations involves a number of exponentiation operations. We now evaluate the computational efficiency and bandwidth overhead of PrivRep crypto operations by

Entity	Computational overhead (number of exponentiations)				Communication overhead		
	Initialization	Feedback	NIZK Proof	Tallying	Initialization	Feedback	NIZK Proof
User	40 msec	50 msec	260 msec	-	1.7kb	1.3 kb	16.2 kb
RE	23 msec	-	160 msec	2 msec	1.8kb	-	8.7 kb

Table 6: Protocol Overhead for one user providing feedback for 10 retailers.

Operation	Computation Cost	Communication Cost
Setup	-	1.8 kb
Key	-	0.8 kb
Feedback	-	1.3 kb
NIZKP	578 msec	24.8kb
Tallying	1 msec	88 kb

Table 7: Cost for Public Verification for 10 users providing the responses for 10 retailers.

providing the microbenchmarks for these cryptographic operations. We have implemented the protocol operations in Java on a Windows 10 computer system powered by a 2.5 GHz Intel Core i7 processor. We have used the standard elliptic curve NIST P-256 for generating the cryptographic parameters of our protocol. The cost for generating the encrypted response and zero-knowledge proof for one user is presented in Table 6. The computational time required for generating the cryptograms and the associated zero-knowledge proof turns out to be within an acceptable range and lies below 0.4 sec in total. However, the NIZK operations are the most expensive operations among all and require around 0.26 sec. At the RE's side, the computational time is also acceptable and all the operations are carried out in less than 0.1 sec. The cost of public verification is also not high, and the verifier can confirm the accuracy of the shared responses and protocol operation in around 0.578 sec for 10 users and 10 retailers as shown in Table 7. We observe that the computational time linearly increases with the number of retailers and is independent of a number of feedback providers in the system. The computational time is acceptable even for a large number of retailers as a user only interacts with few retailers only in a real scenario.

Table 6 and 7 present the bandwidth overheads of our protocol at the user's and reputation engine's side. It can be seen that the communication overhead is slightly high for the reputation engine and the user. This high communication cost is due to the need to transfer large number of NIZK proofs as each of them require a large number of arguments to be generated by the prover.

8. Comparison with other Systems

Among the existing protocols for reputation aggregation, none addresses the problem of secure aggregation among the selected peers while protecting the identities of peers in the 'trusted peers' list without the use of trusted setup and other trusted entities. The simplest way to compute the personalized reputation is to send the Secure Sum [53] request to the preselected 'trusted' peers in a distributed secure multiparty computation model. The selected peers compute the sum by encrypting the responses and adding some random noise which is later subtracted by the protocol initiator. The secure sum protocol discloses the identities of 'trusted peers' and cannot ensure privacy if some peers collude with each other to know the values of target peers. However, our protocol ensures the privacy under honest but curious as well as malicious model and also eradicating the need for randomization. The computational complexity of the above simple protocol is $O(n)$, which is the same as the protocol discussed in this paper. The protocol presented in [21] protects the privacy of users under the honest but curious model and has a computational complexity of $O(n)$. In contrast, our protocol not only provides personalization but also ensures privacy protection under the malicious model. [14] uses anonymous identities to protect the privacy of the user, however, our system does not need any anonymous identities for privacy protection. In [6] Clark et al. proposed a delegation protocol for the users who wants to leave the network by delegating their unfinished task to other trusted users and has the complexity of $O(mn)$. The system also considers personalization through the use of collaborative filtering. In contrast, our protocol has the complexity of $O(m)$ and can also handle the delegation without relying on other trusted peers. In [54] Erkin et al. use semi-trusted party and data packing for personalized reputation and recommendation and have communication complexity of $O(T.M \log N)$. The system is not purely decentralized in its operation and privacy may be compromised if semi-trusted parties get corrupted. On the other hand, our proposal is decentralized and is not reliant on any trusted peers or centralized peers for its operation.

Our work can be compared with the decentralized reputation aggregation scheme based on the blockchain system [17]. Though the system presented in [17] preserves the privacy of users in the malicious model but it does not consider personalized aggregation. In contrast, our approach is not only decentralized in its operation, but it also considers personalization of aggregation of ratings

provided by the users.

9. Conclusion

In this paper, we proposed a novel scheme that allows personalized computation of reputation of service providers in an online marketplace. The proposed approach is based on the semantics of secure multiparty computation and the non-interactive zero knowledge proof of knowledge. The personalized reputation of a service provider is computed by utilizing the private feedback values submitted by the feedback providers who have had interactions with the service providers. The protocol operations are performed in a decentralized and privacy-preserving manner. We proved that no computationally bounded adversary will learn anything about the secret inputs (ratings), other than what it can learn from the output of the protocol, i.e. the overall reputation score of a service provider. We believe our system can be used by any centralized or decentralized marketplace acting as the reputation engine to compute the reputation of its users and retailers in a privacy-preserving way.

Acknowledgment

The research in this paper was supported by the ERC Starting Grant, No. 306994. We thank the anonymous reviewers for their valuable comments and suggestions towards improving the quality of this paper.

References

[1] R. Kerr and R. Cohen, "Smart cheaters do prosper: Defeating trust and reputation systems," in *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 2*, 2009, pp. 993–1000.

[2] T. Minkus and K. W. Ross, *I Know What You're Buying: Privacy Breaches on eBay*, Cham, 2014, pp. 164–183.

[3] A. Narayanan and V. Shmatikov, "Robust de-anonymization of large sparse datasets," in *2008 IEEE Symposium on Security and Privacy (IEEE s&p 2008)*, May 2008, pp. 111–125.

[4] ——, "De-anonymizing social networks," in *Proceedings of the 2009 30th IEEE Symposium on Security and Privacy (IEEE s&p)*, 2009, pp. 173–187.

[5] P. Resnick and R. Zeckhauser, "Trust among strangers in internet transactions: Empirical analysis of eBay's reputation system," in *The Economics of the Internet an E-Commerce*, ser. Advances in Applied Microeconomics, 2002, vol. 11.

[6] M. R. Clark, K. Stewart, and K. M. Hopkinson, "Dynamic, privacy-preserving decentralized reputation systems," *IEEE Transactions on Mobile Computing*, vol. 16, no. 9, pp. 2506–2517, 2017.

[7] K. Soska, A. Kwon, N. Christin, and S. Devadas, "Beaver: A decentralized anonymous marketplace with secure reputation," *IACR Cryptology ePrint Archive*, 2016.

[8] J. Bethencourt, E. Shi, and D. Song, "Signatures of reputation: Towards trust without identity," in *Proceedings of the 14th International Conference on Financial Cryptography and Data Security*, ser. FC'10, 2010, pp. 400–407.

[9] E. Zhai, D. I. Wolinsky, R. Chen, E. Syta, C. Teng, and B. Ford, "Anonrep: Towards tracking-resistant anonymous reputation," in *Proceedings of the 13th Usenix Conference on Networked Systems Design and Implementation*, ser. NSDI'16, 2016, pp. 583–596.

[10] B. Palanisamy, L. Liu, Y. Zhou, and Q. Wang, "Privacy-preserving publishing of multilevel utility-controlled graph datasets," *ACM Transactions on Internet Technologies.*, vol. 18, no. 2, pp. 1–24, Feb. 2018.

[11] M. Kinateder and S. Pearson, "A privacy-enhanced peer-to-peer reputation system," in *Proceedings of International Conference on Electronic Commerce and Web Technologies*, 2003, pp. 206–215.

[12] S. Schiffner, S. Clauß, and S. Steinbrecher, *Privacy and Liveliness for Reputation Systems*, Berlin, Heidelberg, 2010, pp. 209–224.

[13] M. T. Goodrich and F. Kerschbaum, "Privacy-enhanced reputation-feedback methods to reduce feedback extortion in online auctions," in *Proceedings of Conference on Data and Application Security and Privacy*, 2011.

[14] E. Androulaki, S. G. Choi, S. M. Bellovin, and T. Malkin, "Reputation systems for anonymous networks," in *Proceedings of 8th International Symposium Privacy Enhancing Technologies (PETS)*, N. Borisov and I. Goldberg, Eds., 2008, pp. 202–218.

[15] A. Singh and L. Liu, "Trustme: Anonymous management of trust relationships in decentralized p2p systems," in *Proceedings of the 3rd International Conference on Peer-to-Peer Computing*, 2003.

[16] S. Clauß, S. Schiffner, and F. Kerschbaum, "K-anonymous reputation," in *Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security*, ser. ASIA CCS '13, 2013, pp. 359–368.

[17] A. Schaub, R. Bazin, O. Hasan, and L. Brunie, "A trustless privacy-preserving reputation system," in *Proceedings of 31st IFIP TC 11 International Conference, (SEC)*, 2016, pp. 398–411.

[18] J. Blömer, J. Juhnke, and C. Kolb, "Anonymous and publicly linkable reputation systems," in *Proceedings of 19th International Conference Financial Cryptography and Data Security (FC)*, 2015, pp. 478–488.

[19] D. Boneh, X. Boyen, and H. Shacham, in *Proceedings of 24th Annual International Cryptology Conference on Advances in Cryptology*, 2004, pp. 41–55.

[20] N. Busom, R. Petrlic, F. Sebé, C. Sorge, and M. Valls, "A privacy-preserving reputation system with user rewards," *Journal of Network and Computer Applications*, vol. 80, no. Supplement C, pp. 58 – 66, 2017.

[21] O. Hasan, L. Brunie, E. Bertino, and N. Shang, "A decentralized privacy preserving reputation protocol for the malicious adversarial model," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 6, pp. 949–962, June 2013.

[22] E. Pavlov, J. S. Rosenschein, and Z. Topol, "Supporting privacy in decentralized additive reputation systems," in *Proceedings of Second International Conference Trust Management*, 2004, pp. 108–119.

[23] S. Dolev, N. Gilboa, and M. Kopeetsky, "Efficient private multiparty computations of trust in the presence of curious and malicious users," *Journal of Trust Management*, vol. 1, no. 1, p. 8, Jun 2014.

[24] T. Dimitriou and A. Michalas, "Multi-party trust computation in decentralized environments in the presence of malicious adversaries," *Ad Hoc Networks*, vol. 15, pp. 53–66, Apr. 2014.

[25] M. A. Azad and S. Bag, "Decentralized privacy-aware collaborative filtering of smart spammers in a telecommunication network," in *Proceedings of the 32nd Symposium on Applied Computing*, 2017, pp. 1711–1717.

[26] M. A. Azad, S. Bag, S. Tabassum, and F. Hao, "privy: Privacy preserving collaboration across multiple service providers to combat telecoms spam," *IEEE Transactions on Emerging Topics in Computing*, vol. PP, no. 99, pp. 1–1, 2017.

[27] M. A. Azad, S. Bag, and F. Hao, "M2m-rep: Reputation of machines in the internet of things," in *Proceedings of the 12th International Conference on Availability, Reliability and Security, ARES-17*. ACM, 2017, pp. 28:1–28:7.

[28] A. Visan, F. Pop, and V. Cristea, "Decentralized trust management in peer-to-peer systems," in *Proceedings of 10th Interna-*

tional Symposium on Parallel and Distributed Computing, July 2011, pp. 232–239.

[29] O. Stefanos, T. Litos, and D. Zindros, "Trust is risk: A decentralized financial trust platform," *Proceedings of 21st Financial Cryptography and Data Security (FC)*, 2017.

[30] P. Moreno-Sanchez, A. Kate, M. Maffei, and K. Pecina, "Privacy preserving payments in credit networks: Enabling trust with privacy in online marketplaces." in *Proceedings of 22nd Network and Distributed System Security Symposium (NDSS)*, 2015.

[31] "Openbazaar: Online marketplace," 2018. [Online]. Available: https://docs.openbazaar.org/.

[32] A. Post, V. Shah, and A. Mislove, "Bazaar: Strengthening user reputations in online marketplaces," in *Proceedings of the 8th USENIX Conference on Networked Systems Design and Implementation*, 2011.

[33] A. Jøsang, R. Ismail, and C. Boyd, "A survey of trust and reputation systems for online service provision," *Decision Support Systems*, vol. 43, no. 2, pp. 618–644, Mar. 2007.

[34] F. Hendrikx, K. Bubendorfer, and R. Chard, "Reputation systems: A survey and taxonomy," *Journal of Parallel and Distributed Computing*, vol. 75, pp. 184 – 197, 2015.

[35] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The eigentrust algorithm for reputation management in p2p networks," in *Proceedings of the 12th International Conference on World Wide Web*, 2003, pp. 640–651.

[36] M. Gupta, P. Judge, and M. Ammar, "A reputation system for peer-to-peer networks," in *Proceedings of the 13th International Workshop on Network and Operating Systems Support for Digital Audio and Video*, 2003.

[37] K. Walsh and E. G. Sirer, "Experience with an object reputation system for peer-to-peer filesharing," in *Proceedings of the 3rd Conference on Networked Systems Design & Implementation - Volume 3*, ser. NSDI'06, 2006.

[38] P. Y. A. Ryan, D. Bismark, J. Heather, S. Schneider, and Z. Xia, "Prêt à voter: A voter-verifiable voting system," *Transaction on Information Security*, vol. 4, no. 4, pp. 662–673, Dec. 2009.

[39] F. Hao, P. Y. A. Ryan, and P. Zielinski, "Anonymous voting by two-round public discussion," *IET Information Security*, vol. 4, no. 2, pp. 62–67, 2010.

[40] E.Shi, R.Chow, T. h. Hubert Chan, D. Song, and E.Rieffel, "Privacy-preserving aggregation of time-series data," in *Proceedings of 18th Network and Distributed System Security Symposium (NDSS)*, 2011.

[41] H. Corrigan-Gibbs and D.Boneh, "Prio: Private, robust, and scalable computation of aggregate statistics," in *Proceedings of 14th USENIX Symposium on Networked Systems Design and Implementation (NSDI 17)*, 2017, pp. 259–282.

[42] L. Melis, G. Danezis, and E. D. Cristofaro, "Efficient private statistics with succinct sketches," in *Proceedings of 23rd Network and Distributed System Security Symposium (NDSS)*, 2015.

[43] M. Burkhart, M. Strasser, D. Many, and X. Dimitropoulos, "SEPIA: Privacy-preserving aggregation of multi-domain network events and statistics," in *Proceedings of the 19th USENIX Conference on Security*, 2010, pp. 15–15.

[44] I. E. Akkus, R. Chen, M. Hardt, P. Francis, and J. Gehrke, "Non-tracking web analytics," in *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, 2012, pp. 687–698.

[45] U. Erlingsson, V. Pihur, and A. Korolova, "Rappor: Randomized aggregatable privacy-preserving ordinal response," in *Proceedings of the 2014 ACM Conference on Computer and Communications Security*, 2014, pp. 1054–1067.

[46] F.Hao, M. N. Kreeger, B. Randell, D.Clarke, S. F. Shahandashti, and P. H.-J. Lee, "Every vote counts: Ensuring integrity in large-scale electronic voting," in *2014 Electronic Voting Technology Workshop/Workshop on Trustworthy Elections (EVT/WOTE 14)*, 2014.

[47] A. Fiat and A. Shamir, "How to prove yourself: Practical solutions to identification and signature problems," in *Proceedings of Advances in Cryptology — CRYPTO' 86*, 1987, pp. 186–194.

[48] F. Kerschbaum, "A verifiable, centralized, coercion-free reputation system," in *Proceedings of the 8th ACM Workshop on Privacy in the Electronic Society*, ser. WPES '09, 2009, pp. 61–70.

[49] B. Adida, "Helios: Web-based open-audit voting," in *Proceedings of the 17th Conference on Security Symposium*, 2008, pp. 335–348.

[50] Z. Malik and A. Bouguettaya, "Evaluating rater credibility for reputation assessment of web services," in *Proceedings of the 8th International Conference on Web Information Systems Engineering*, 2007, pp. 38–49.

[51] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, Sep. 2006.

[52] F. Bao, R. H. Deng, and H. Zhu, "Variations of diffie-hellman problem," in *Proceedings of 5th International Conference Information and Communications Security*, S. Qing, D. Gollmann, and J. Zhou, Eds., 2003, pp. 301–312.

[53] C. Clifton, M. Kantarcioglu, J. Vaidya, X. Lin, and M. Y. Zhu, "Tools for privacy preserving distributed data mining," *SIGKDD Explorations Newsletter*, vol. 4, no. 2, pp. 28–34, Dec. 2002.

[54] Z. Erkin, T. Veugen, T. Toft, and R. L. Lagendijk, "Generating private recommendations efficiently using homomorphic encryption and data packing," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 1053–1066, 2012.

Appendix

Zero Knowledge Proofs of Well-formedness

The section presents NIZK proofs for the Scheme presented in Section 5.

Proof I: $PW_i^1\left[g^{\sum_{j=1}^{2} a_{ji}\omega_j} \in \{1, g\} : g^{a_{1i}}, g^{a_{2i}}, g^{\omega_1}, g^{\omega_2},\right]$
This proof comprises a witness to the statement $\sigma \equiv g^{\sum_{j=1}^{2} a_{ji}\omega_j} \in \{1, g\}$, where $g^{a_{1i}}, g^{a_{2i}}, g^{\omega_1}, g^{\omega_2}$ are given. For sake of clarity, we write a_{ji} as $a_j, j = 1, 2$. So $\sigma \equiv (g^{a_1\omega_1+a_2\omega_2} = 1) \vee (g^{a_1\omega_1+a_2\omega_2} = g)$. Hence, σ is an 'OR' statement and either of the two constituent statements is true. Let us assume that the first statement is true, that is: $g^{a_1\omega_1+a_2\omega_2} = 1$. Now, a non-interactive zero-knowledge proof for σ will include a real proof for the true statement and a simulated proof for the wrong statement. The prover selects $r_1, r_2 \in_R \mathbb{Z}_p$, and generates two commitments: $com_{11} = g^{r_1}, com_{12} = g^{r_2}, com_{13} = (g^{a_1}g)^{r_1}(g^{a_2}g)^{r_2}$. Then the prover selects $ch_2, res_{21}, res_{22} \in_R \mathbb{Z}_p$, and generates $com_{21} = g^{res_{21}}(g^{\omega_1})^{ch_2}$, $com_{22} = g^{res_{22}}(g^{\omega_2})^{ch_2}$ and $com_{23} = (g^{a_1})^{res_{21}}(g^{a_2})^{res_{22}}g^{ch_2}$. Let the grand challenge of the NIZK statement be ch. Let, $ch_1 = ch - ch_2$. The prover generates two responses: $res1i = r_{1i} - ch_1 * \omega_i, i = 1, 2$. Now, the verification equations are as below:

1. $g^{res_{ij}} \stackrel{?}{=} \frac{com_{ij}}{(g^{\omega_j})^{ch_i}} \forall\, i, j \in \{1, 2\}$.
2. $(g^{a_1})^{res_{11}}(g^{a_2})^{res_{12}} \stackrel{?}{=} com_{13}$.
3. $(g^{a_1})^{res_{21}}(g^{a_2})^{res_{22}} \stackrel{?}{=} \frac{com_{23}}{g^{ch_2}}$

If the above 6 equations hold, then the NIZK proof is genuine. In a similar way, NIZK proof can be generated

when $g^{a_1\omega_1+a_2\omega_2} = g$. Here, we skip this due to space restriction.

Proof II: $PW_{\text{total}}\left[g^{\sum_{i=1}^{n}\sum_{j=1}^{2} a_{ji}\omega_j} = g^{\Delta} : g^{a_{1i}}, g^{a_{2i}}, g^{\omega_1}, g^{\omega_2}\right]$
This proof consists of a witness to the fact that the total number of feedback providers whose feedback actually get counted is given by Δ. The proof is as following:
Select $r_1, r_2 \in_R \mathbb{Z}_p$. Generate these 3 commitments $com_1 = (\prod_{i=1}^{n} g^{a_{1i}})^{r_1}(\prod_{i=1}^{n} g^{a_{2i}})^{r_2}$, $com_2 = g^{r_1}$ and $com_3 = g^{r_2}$. Let the challenge be ch. Then generate two responses $res_1 = r_1 - \omega_1 * ch, res_2 = r_2 - \omega_2 * ch$. The verifier accepts the proof if the following relations hold:

1. $(\prod_{i=1}^{n} g^{a_{1i}})^{res_1}(\prod_{i=1}^{n} g^{a_{1i}})^{res_2} \stackrel{?}{=} \frac{com_1}{(g^{\Delta})^{ch}}$
2. $g^{res_1} \stackrel{?}{=} \frac{com_2}{(g^{\omega_1})^{ch}}$
3. $g^{res_2} \stackrel{?}{=} \frac{com_3}{(g^{\omega_2})^{ch}}$

Proof III: $PW_i^2\left[g^{\sum_{j=1}^{2} b_{ji}\omega_j} = g : g^{b_{1i}}, g^{b_{2i}}, g^{\omega_1}, g^{\omega_2}\right]$
For the sake of clarity we write b_{ji} as $b_j, j = 1, 2$. Generate three commitments $com_1 = (g^{b_1})^{r_1}(g^{b_2})^{r_2}$, $com_2 = g^{r_1}$ and $com_3 = g^{r_2}$. Let the challenge be ch. Generate three responses $res_1 = r_1 - \omega_1 * ch, res_2 = r_2 - \omega_2 * ch$. The verifier accepts the proof if the following relations hold:

1. $(g^{b_1})^{res_1}(g^{b_2})^{res_2} \stackrel{?}{=} \frac{com_1}{g^{ch}}$
2. $g^{res_1} \stackrel{?}{=} \frac{com_2}{(g^{\omega_1})^{ch}}$
3. $g^{res_2} \stackrel{?}{=} \frac{com_3}{(g^{\omega_2})^{ch}}$

Proof IV: Feedback Well-formedness
$PW_i^3\left[C_i : g^{x_{1i}}, g^{x_{2i}}, g^{y_{1i}}, g^{y_{2i}}, g^{b_{1i}}, g^{b_{2i}}, g^{\alpha_i}, g^{b_{1i}}, g^{b_{2i}}\right]$
Here, we show how the feedback providers construct a NIZK proof of feedback well-formedness. The feedback $C_i = \langle c_{1i}, c_{2i}, g^{\alpha}\rangle$ where either of the two statements holds:
1) $c_{1i} = g^{x_{1i}y_{1i}}g^{b_{1i}\alpha_i} \wedge c_{2i} = g^{x_{2i}y_{2i}}g^{b_{2i}\alpha_i}$
2) $c_{1i} = g^{x_{1i}y_{1i}}g^{b_{1i}\alpha_i}g^{a_{1i}} \wedge c_{2i} = g^{x_{2i}y_{2i}}g^{b_{2i}\alpha_i}g^{a_{2i}}$
That is the feedback provider has to prove that either of the two statements stated above is true. For sake of clarity we write c_{ji} as c_j, x_{ji} as x_j, y_{ji} as y_j, b_{ji} as b_j, a_{ji} as $a_j, \forall\, j \in \{1, 2\}$ and α_i as α. We need to construct a proof for the statement;
$\sigma \equiv (c_1 = g^{x_1y_1}g^{b_1\alpha} \wedge c_2 = g^{x_2y_2}g^{b_2\alpha}) \vee (c_1 = g^{x_1y_1}g^{b_1\alpha}g^{a_1}$
$\wedge\, c_2 = g^{x_2y_2}g^{b_2\alpha}g^{a_2})$.
The given inputs are these: $g^{x_1}, g^{y_1}, g^{x_2}, g^{y_2}, g^{\alpha}, g^{a_1}, g^{a_2}, g^{b_1}$ and g^{b_2}. Only one of the two statements above is true. Let us assume that the first statement is true, that is $(c_1 = g^{x_1y_1}g^{b_1\alpha} \wedge c_2 = g^{x_2y_2}g^{b_2\alpha})$. Hence, the prover will have to provide a real proof for the first statement and a simulated proof for the second statement. The prover selects random r_1, r_2 and computes 3 commitments $com_{11} = g^{r_1}$, $com_{12} = g^{r_2}$, $com_{13} = (g^{y_1})^{r_1}(g^{b_1})^{r_2}$, $com'_{11} = g^{r'_1}$, $com'_{12} = g^{r'_2}$, $com'_{13} = (g^{y_2})^{r'_1}(g^{b_2})^{r'_2}$. Then the prover selects random $ch_2, res_{21}, res_{22}, res'_{21}, res'_{22} \in_R \mathbb{Z}_p$ and computes these commitments:
$com_{21} = g^{res_{21}}(g^{x_1})^{ch_2}$, $com_{22} = g^{res_{22}}(g^{\alpha})^{ch_2}$,
$com_{23} = (g^{y_1})^{res_{21}}(g^{b_1})^{res_{22}}(c_1/g^{a_1})^{ch_2}$ and
$com'_{21} = g^{res'_{21}}(g^{x_2})^{ch_2}$, $com'_{22} = g^{res'_{22}}(g^{\alpha})^{ch_2}$,
$com'_{23} = (g^{y_2})^{res'_{21}}(g^{b_2})^{res'_{22}}(c_2/g^{a_2})^{ch_2}$.
Now let ch be the grand challenge of the NIZK proof, obtained by feeding all the above parameters into a hash function. Let, $ch_1 = ch - ch_2$. The prover computes $res_{11} = r_1 - x_1 * ch_1, res_{12} = r_2 - \alpha * ch_1, res'_{11} = r'_1 - x_2 * ch_1, res'_{12} = r'_2 - \alpha * ch_1$. The verification equations are as follows:

1. $g^{res_{i1}} \stackrel{?}{=} \frac{com_{i1}}{(g^{x_1})^{ch_i}}, i = 1, 2$
2. $g^{res_{i2}} \stackrel{?}{=} \frac{com_{i2}}{(g^{\alpha})^{ch_i}}, i = 1, 2$
3. $g^{res'_{i1}} \stackrel{?}{=} \frac{com'_{i1}}{(g^{x_2})^{ch_i}}, i = 1, 2$
4. $g^{res'_{i2}} \stackrel{?}{=} \frac{com'_{i2}}{(g^{\alpha})^{ch_i}}, i = 1, 2$
5. $(g^{y_1})^{res_{11}}(g^{b_1})^{res_{12}} \stackrel{?}{=} \frac{com_{13}}{c_1^{ch_1}}$
6. $(g^{y_1})^{res_{21}}(g^{b_1})^{res_{22}} \stackrel{?}{=} \frac{com_{23}}{(c_1/g^{a_1})^{ch_2}}$
7. $(g^{y_2})^{res'_{11}}(g^{b_2})^{res'_{12}} \stackrel{?}{=} \frac{com'_{13}}{c_2^{ch_1}}$
8. $(g^{y_2})^{res'_{21}}(g^{b_2})^{res'_{22}} \stackrel{?}{=} \frac{com'_{23}}{(c_2/g^{a_2})^{ch_2}}$

If all the above 12 equations satisfy, the NIZK statement is true. Similarly, the prover can generate a NIZK proof statement if the second statement is true, that is: $(c_1 = g^{x_1y_1}g^{b_1\alpha}g^{a_1} \wedge c_2 = g^{x_2y_2}g^{b_2\alpha}g^{a_2})$. Here, we omit this due to space restriction.

Proof V: $PW_T\left[L : \hat{c}_1, \hat{c}_2, g^{\omega_1}, g^{\omega_2}\right]$ This proof consists of a witness to the fact that $L = \hat{c}_1^{\omega_1}\hat{c}_2^{\omega_2}$, given g^{ω_1} and g^{ω_2}. It can be easily seen that this proof is similar to Proof III discussed above.

Proof VI (Schnorr Signature): We have used this Schnorr signature zero knowledge proof [39] many times in our paper. Here we provide a generic construction for this kind of zero knowledge proofs. Here, the prover wants to prove knowledge of a secret variable x, where $X = g^x$ is given. We show how the prover can construct this NIZK proof. The prover selects random $r \in_R \mathbb{Z}_p$ and generates a commitment $com = g^r$. Let, the random challenge be ch. The prover generates a response $res = r - ch * x$. The verification equation is as follows:

$$g^{res} \stackrel{?}{=} \frac{com}{X^{ch}}$$