privy: Privacy Preserving Collaboration Across Multiple Service Providers to Combat Telecom Spams

Muhammad Ajmal Azad, Samiran Bag, Shazia Tabassum and Feng Hao *Senior Member, IEEE*

Abstract—Nuisance or unsolicited calls and instant messages come at any time in a variety of different ways. These calls would not only exasperate recipients with the unwanted ringing, impacting their productivity, but also lead to a direct financial loss to users and service providers. Telecommunication Service Providers (TSPs) often employ standalone detection systems to classify call originators as spammers or non-spammers using their behavioral patterns. These approaches perform well when spammers target a large number of recipients of one service provider. However, professional spammers try to evade the standalone systems by intelligently reducing the number of spam calls sent to one service provider, and instead distribute calls to the recipients of many service providers. Naturally, collaboration among service providers could provide an effective defense, but it brings the challenge of privacy protection and system resources required for the collaboration process. In this paper, we propose a novel decentralized collaborative system named privy for the effective blocking of spammers who target multiple TSPs. More specifically, we develop a system that aggregates the feedback scores reported by the collaborating TSPs without employing any trusted third party system, while preserving the privacy of users and collaborators. We evaluate the system performance of privy using both the synthetic and real call detail records. We find that privy can correctly block spammers in a quicker time, as compared to standalone systems. Further, we also analyze the security and privacy properties of the privy system under different adversarial models.

Index Terms—Privacy Preservation, Telephony Spam, Decentralized Collaboration, Secure Multi-party Computation

1 Introduction

New technologies bring many advantages and benefits to the society and business organizations that employ them but unfortunately at the same time have attracted criminals and scammers for fraud and criminal activities. Fraudsters can adopt various ways to reach their targets, e.g., via social networks (Facebook, Twitter, on-line blogs etc.), email and the telephony (VoIP, Fixed landline, and Mobile). During the past few years, telephone networks have become one of the most preferred media for doing business and personal communication all over the world. However, the unprecedented growth of telephony has also attracted advertisers, criminals and fraudsters to use this medium for marketing products and defrauding users [1]. Recent statistics on the telephony frauds and scams show that subscribers and telecommunication companies lose about $38.1 billion to the fraudsters annually. Further, answering unwanted calls would result in an estimated waste of 20 million man hours and a loss of $475 million annually [2].

Researchers have proposed many standalone techniques for detecting spammers in the telecommunication and VoIP (Voice over Internet Protocol) networks [3]. These approaches are mainly based on the white and black lists [4], analyzing speech content exchanged between the calling and the called parties [5], [6], based on analyzing the social behaviour of user towards others in the network [7]–[10], and sender verification through CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) and Turing test [11]. The standalone solutions could also be implemented as multistage systems [8], [12] where multiple independent modules could collaborate internally for the effective detection. Standalone systems normally use information from the single service provider, hence are prone to evade by the sophisticated spammers by simply controlling the number of spam calls per service provider. However, for a high financial gain, spammers target large number of subscribers suppressed across many service providers. This makes collaborative systems a natural choice for the timely detection of intelligent spamming users [13],[14],[15], [16], whereby a set of telecommunication service providers share their information about the behavior of their user in their networks, respectively. However, TSPs are not comfortable in sharing information of their customers to another entity because they are concerned about privacy of their customers, and their own network configurations. Other challenges a collaborative system has are, the decentralization (having collaboration without any third party trusted system) and the overheads required for the collaboration.

In recent years, several collaborative systems have been proposed to enable service providers to share their private data to the trusted centralized systems for statistics aggregation and meaningful decisions [17]–[20] [21]. These systems rely on a centralized trusted third party for the aggregation and analysis of the feedback submitted by the collaborating TSP, thereby prone to be attack by the adver-

- *Muhammad Ajmal Azad, Samiran Bag and Feng Hao are with School of Computing, Newcastle University, United Kingdom. E-mail: {muhammad.azad,samiran.bag,feng.hao}@ncl.ac.uk*
- *Shazia Tabassum is with INESCTEC Porto, University of Porto, Portugal. Email: shazia.tabassum@inesctec.pt*

Manuscript received June 08, 2017; revised September 13, 2017, Accepted October 10,2017.

sary at the centralized system for the sensitive and private information. However, service providers want a complete control over the private data of their customers and are not willing to share this data with the trusted centralized system because of privacy concerns and regulatory affairs [22]. The alternative decentralized solutions such as [14], [23], [24] [13] are not dependent on the trusted centralized authority for statistics aggregation but these systems require a set of pre-trusted users for protecting the privacy of others. Furthermore, these systems do not consider trust weights of collaborators and have high system resources. We believe that service providers would agree on the collaboration process if collaboration process ensures privacy of collaborator's data under an honest but curious and a malicious adversary models in a fully decentralized way.

In this paper, we proposed a novel privacy preserving decentralized collaborative system named privy, where the service provider wants to compute the aggregated statistics from the feedback values provided by collaborating participants without learning the feedback values of collaborating TSPs. The proposed approach allows each collaborating TSP to publish the encrypted feedback values on the append only public bulletin board (BB). Service providers or end-users having access to the BB would only learning the aggregated weighted sum of the feedback score submitted scores, without inferring any other information submitted by collaborators. Participants would only learn the final weighted aggregated score of the user, thus individual feedback scores remain anonymous to participants. The privy system consists of two well-known entities: a public BB and the participating service providers. The TSP monitors the behavior of their customers using locally recorded data (signaling or call record database, or collects direct feedback from their users), computes the local aggregate score of the user, and reports encryption of the score to the BB. The BB is an append only table that contains the identity of the user, the encrypted score assigned by the service provider, and the non-interactive zero-knowledge (NIZK) proof to ensure that the encrypted feedback score is indeed within the prescribed range.

Our privy system ensures privacy preservation of collaborators and their customers. Specifically, the system protects privacy of participants for the following adversarial models:

Privacy against curious participants : the curious participants alone or in collusion with other participants would learn nothing about the feedback of other participants or target participant.

Ensure correctness against malicious participants: the proposed system not only achieves full privacy protection, but also operates correctly under malicious participants through the use of efficient non-interactive zero-knowledge proof.

This paper extends our previous work presented at SAC'17 [25], which does not support weighted feedback, i.e. all collaborators have equal impact on the final aggregation. Weighted feedback can differentiate feedback of collaborating TSPs by giving more weight to the directly connected and trusted TSPs than other TSPs. To enable this, the aggregation system needs to know the weights and scores of collaborators secretly. For the weighted reputation aggregation, in this paper we introduce another phase conducted by the trusted setup for the secure weighted reputation aggregation without revealing weights as well as the feedback scores of users assigned by the collaborating TSP. This paper also presents detailed security proofs and privacy analysis. Further, it presents a proof-of-concept implementation of cryptographic operations with detailed performance evaluation.

In summary, this paper makes the following contributions.

- It presents a decentralized aggregation protocol based on the secure multi-party computation that provides aggregated statistics without revealing any private information provided by the participants. Specifically, a participating TSP publishes local feedback about his user to the BB and any other participant can find the aggregate behavior of the user by simply aggregating the encrypted score without learning any individual score. In our protocol, we do not trust any entity (BB or protocol setup) thus achieve strong notions of privacy and security under malicious and honest but curious models. Further, the protocol has a small computation and communication overhead, and can easily handle a large number of participants and a large number of feedbacks.
- It presents a complete description of privacy and security proofs. Further, it also describes a proof-of-concept implementation, and reports the evaluation results for different crypto operations.
- This paper evaluates performance of the collaborative privy system on a realistic dataset that has been generated using the probability distribution learned from real anonymized call detail records. The results demonstrate that our proposed system has a better detection accuracy than the traditional stand-alone systems.

The novelty of the privy system is based on its underlying cryptographic protocol. The underlying cryptographic protocol of the privy system can be used in any multi-party collaborative system with the intention to securely compute weighted sum of feedback scores provided by the parties. We applied the approach for detecting spammers in telecom networks because of the following reasons. First, telecom spams are increasing rapidly, and it is necessary to have a mechanism that convinces service providers to collaborate in achieving effective and early detection of spammers. Second, it is possible to convince a large number of TSPs for the collaboration because they are under the control of one regulatory authority of the country. The proposed approach can be used for aggregating rating and recommendation in online marketplaces with a minor modification. For example, changing the NIZK proof for handling the ratings presented at the scale of 0-5 or 1 star to 5 stars. This involves representing the NIZK proof to prove that users have selected only one value out of all possible values.

The remainder of this paper is organized as follows. Section 2 analyzes the related work; Section 3 presents the motivation for this work and defines the problem; Section 4 describes an overview of privy system. Section 5 describes the system architectures of privy system and its cryptography operations. Section 6 analyzes the security and pri-

vacy properties of the system. Section 7 presents prototype implementation and methodology used for analyzing the performance of the system. Section 8 shows results of our experiments. Section 9 presents the deployment challenges and limitations of the system. Section 10 concludes the paper. The appendix (supplementary material) provides the security proofs and assumptions considered in this paper.

2 State of the Art

To date, anti-SPIT (SPam over Internet Telephony) systems have taken two main methods for identifying spammers: 1) content-based anti-SPIT systems [5], [6] and 2) identity-based anti-SPIT systems [4], [7], [8], [26]–[28]. Content-based systems process and analyze the speech content exchanged between callers and the callee for blocking malicious callers involved in spreading unwanted content; whereas the identity-based systems use the identity of the user for characterizing the behavior of user towards others. The identity-based systems can be further grouped into: a black or white list [4], a reputation-based system [7], [8], [26]–[29], caller authentication in the form of CAPTCHA [11], and Honeynets [30], [31]. A standalone SPIT solution can also be deploy by combining several individual systems in the form of multistage systems [8], [12].

Very few works have been reported on the collaborative spam detection in the VoIP and telecommunication networks to improve the detection accuracy and the detection time. These approaches mainly carry out collaboration among different standalone modules, in the form of multistage systems [8], [12], [32]. However, collaboration among multiple modules would considerably add delay in the call setup time. The complete information about caller's calling behavior mostly resides in his home network and this information could be exchanged to the service provider of callee using SIP (Session Initiation Protocol) signaling tags [33] in order to evaluate the performance of SPIT detection system deployed in the home service provider of the caller. However, this system requires direct trust relationship between TSPs and only rate detection system rather than end-users. SPACEDIVE [34] performs collaborative intrusion detection in the network domain by matching and correlating the local and remote rules at the individual and across the different components.

A distributed cooperative detection method has been proposed in [35] for identifying the SPIT callers by having internal collaboration among several VoIP servers within a particular TSP, but the proposal has not provided any mechanism for the collaboration between TSPs.

Several collaborative systems have been proposed for detecting spammers and intruders in the email network and the Internet domain. The collaborative email spam detection systems identify spammers by sharing the message content among the collaborators [36], [37] or relying on the trusted third party centralized system for the aggregation of the reputation and statistical scores [14], [17]–[20]. Both the content-based and centralized trusted reputation systems have access to the private information of collaborators thus pose serious privacy concerns for the collaborators and their customers. In [23], a privacy-aware data aggregation protocol is proposed for the anomaly detection where private data from the multiple collaborators is exchanged to the semi-trusted system which respond back the collaborators with the aggregated statistics. In [13], a distributed decentralized system is proposed that uses the semantics of secure multi-party computation and secret sharing for aggregating the security alerts and traffic logs among several collaborating operators. The system is decentralized, but the collaborators can collude with each other to breach the privacy of some specific target collaborator. Furthermore, [13] did not consider the effect of weights while aggregating the statistics. In [16], a controlled data sharing protocol is proposed where collaborators agreed on the set of information used for the collaboration. The system requires trusted relationship between collaborators and is prone to be circumvent by the malicious collaborators.

Electronic cash (e-cash) based reputation protocols [38]–[41] have also been proposed for computing the reputation of users by using anonymous identities and imposing some fee for posting the feedback. However, these systems require trusted third parties for some operations: in particular, a central bank for handling the transactions. For example, Repcoin [39] requires a trusted third party and the reputation providers for holding the reputation coins of every user in the system. The user in the Repcoin system has to choose either a public identity or the random pseudonym for the communication.

The disadvantages of previous e-cash based anonymous reputation systems are that they rely upon the trusted third party systems, do not support both positive and negative feedbacks and require anonymized identities. Further, the use of anonymous credentials would not help in detecting misbehaving users across multiple service providers. In contrast, the privy system is completely decentralized in its operations and does not require a set of anonymous identities or channels to protect the privacy of its users.

3 Motivation and Problem Definition

In this section, we describe the motivation for this work and define the problem this paper is addressing.

3.1 Motivation

The standalone systems are typically placed within the TSP and consider locally recorded data of a TSP for analyzing the behaviour of users and detecting malicious users. Since, there is no cooperation among TSPs, no data is passed to the other TSPs except call handling messages. Standalone anti-SPIT systems may have a high false negative rate and prolonged detection when spammers are making a very low-rate spam calls to the recipients of several TSPs without overwhelming any single TSP. Particularly, standalone systems could manage detecting the low-rate spammers over the time (after receiving enough number of calls from the caller), and when the number of spam calls from the same source within a TSP spikes. However, this detection is too late, as the spammer has already reached a large number of subscribers in a TSP and across several TSPs. This prolonged detection is because of unavailability of enough information about behavior of the sender within the network. The stand-alone systems could improve their detection rate and

minimize detection time by combining several detection approaches into a single multistage system or asking the caller to solve the CAPTCHA challenge. However, these implementations have following limitations. First, CAPTCHA test involves the caller to solve the challenge, which is not only resource demanding but is also intrusive to the caller. Second, multistage systems require the call request to pass through many detection components thus would increase the call setup delay. Third, multistage systems still require relatively large number of calls from the same source for the final decision, which still allows spammers to reach many subscribers.

Observing call patterns across several TSPs could improve the detection accuracy and detection time but it brings the challenge of privacy preservation that so far has restrained TSPs from participating in the collaboration process. The existing collaborative system [33] requires collaboration between home and visiting SPs through the exchange of message tags at the time of call initiation but it only provides feedback about the capability of detection system placed in the home network of the caller. Furthermore, this system requires a pre-defined trust relationship between involved parties. The effective collaborative detection systems need to have following properties: 1) collaboration among TSPs should be carried out without establishing a direct trust relationships between collaborators, 2) information exchanged between collaborators should not be resource demanding regarding network and system resources, 3) the sensitive information should be exchanged in such a way that it should not be used by the adversary for inferring private information not revealed, 4) the system should have high true positive rate and small false positive rate, and 5) the system should be decentralized i.e. no trusted third party employed for mediating the functions.

3.2 Problem Definition

There are N users $U = \{u_1, u_2, ..., u_k\}$ directly registered with the TSP for the telecommunication services. Every user U_i has a unique identity that he uses to make and receive the call. The service provider has access to different type of information when the user initiates or receives the call. This information can be grouped into on-line information (signaling or call setup messages) and the off-line information the call detail record (CDR). The TSP logs every call transaction of his customer in a CDR for user calling identity, and normally use the CDRs for billing and network management purposes. The CDR can also be used for other purposes (user churn analysis [42], outbreak of diseases [43], detecting criminals, spam detection etc.). A typical call detail record includes following important fields: the identity of the caller, identity of the callee, date and time of the call, a call duration of the call etc. CDRs contain private information of users such as who contact whom and who is a friend of whom, thus has serious privacy concerns if not strongly protected. Therefore, TSPs are not willing to exchange these private records to any trusted third party, and held well protected within their premises and under strong authentication and anonymization. However, TSP want to participate in collaboration for effective spam detection without revealing any private information. TSP could have collaboration with the exchange of anonymized CDRs; however, exchanging CDRs have following limitations. 1) The anonymized CDRs would not provide the functions that TSP is looking for (e.g. spam detection in our case) because of anonymized identity, 2) communication overhead required for exchanging CDRs is extensively very high, and 3) the exchanging CDRs to any trusted entity are prohibited by law due to privacy concerns.

Normally, TSPs have a standalone detection system that analyzes the calling behavior (legitimate or non-legitimate) of the particular user using his call records or in some cases gets the direct feedback (negative or positive) from his called users. Assume that there are n $TSP = \{TSP_1, TSP_2, \ldots, TSP_n\}$ agreed on the collaboration and each of them is equipped with a standalone detection system. They can locally rank caller within their networks using CDR or direct feedback from called users of the caller. The collaborating TSP could assign an input score $i \in \{0, 1\}$ such that $i = 0$ if the user in question is a suspected spammer, and $i = 1$ if the user is not a spammer. The TSPs wish to participate in collaboration without revealing their trust and weight scores. The privacy preserving collaborative detection system would allow an aggregator or Protocol Initiator (PI) [1](Specially designated TSP) to securely compute a weighted sum of the scores assigned to the user by all collaborating TSPs without learning the individual scores. Let us assume there are n collaborating TSPs and the score assigned by TSP_i to a particular user is $s_i \in \{0, 1\}$. The goal is to privately compute the weighted average of scores reported by the collaborating entities as $A = \sum_{i=1}^{n} w_i s_i / (\sum_{i=1}^{n} w_i)$ where w_i is the trust weight of collaborating TSP, and s_i is the score of the user. Once the aggregated score A is computed, the protocol initiator uses it to decide whether the certain user is a spammer or not. In privy system, the score s_i remain secret to the TSPs, and the weights w_i remain secret to the protocol initiator. The protocol initiator only learns the aggregated value of feedback scores.

4 Privacy Goals and Adversarial Model

We discuss privacy goals of the system and provide our adversarial model.

4.1 Privacy Preservation Goals

The primary goal of privy system is to ensure the unlinkability of feedback scores provided by the TSP and the anonymity of the TSP. Let us assume there are two service providers $TSP1$ and $TSP2$. They hold on the call detail records and compute the reputation of their clients/users. This can be represented as a vector i.e. $TSP1 = \{(u1, v1), (u2, v2), (u3, v3) \ldots (un, vn)\}$ and $TSP = \{(u1, v1), (u2, v2), (u3, v3) \ldots (un, vn)\}$ respectively, where u represents identity of the user, and v represents feedback value assigned to the user by the TSP. TSPs would like to collaborate without revealing any information about what score they have assigned to their users and how they rated their fellow service providers. The privy system ensures that the privacy of users and has following properties: 1) the global scores are computed in such a

1. The terms protocol initiator and aggregator are Interchangeable.

way that the feedback values provided by TSPs remain unlinkable and anonymous through out the process, 2) the design ensures that adversary would not be able to manipulate the scores provided by TSPs, and 3) entities involve in the collaboration would only learn the aggregate statistics without knowing the individual statistics. The privy system achieves these entire goals with the use of homomorphic encryption and the publicly available bulletin board.

4.2 Adversarial Model

The privy system consists of three major components. The end users, collaborating service providers and the public bulletin board. The description of each component is presented in section 5.1. Here, we define adversarial model of privy system by discussing the role of each component in the system.

4.2.1 Users

We assume that users trust the service providers for recording their call detail records and performing analysis on the data. We assume that users are trustworthy in providing feedback to the service providers if required.

4.2.2 Telecommunication Service Providers

We assume that TSPs are potentially malicious in providing feedback to the public BB. This means that malicious TSPs can send false reputation feedback to the BB to increase or decrease aggregated reputation of the particular user. This would possibly lead to an attack on the collaborative reputation system by falsely increasing reputation of spammers. We assume that the protocol setup and the initiator do not collude with each other. The TSP may collude with either the setup or the initiator but not with both at the same time. Moreover, the TSP might also collude with a subset of other TSPs except the TSP acting as the protocol initiator (PI). The design mechanism of the privy system and the use of non-interactive zero knowledge proofs have the ability to mitigate the effect of these malicious behaviors. In terms of TSPs protecting the user private data, we assume that TSPs are honest in protecting the users private data and keeping private data under strong authentication credentials.

4.2.3 Analyst or Aggregator

We assume that the analyst or aggregator having access to the public BB is honest-but-curious (HBC); in other words, it performs operations honestly, but may try to learn the private information of service providers and their users during the aggregation process. We assume that the aggregator does not collude with the protocol setup, TSPs and clients.

We assume that all the communication between entities is encrypted and carried out over a secure channel.

5 PRIVY SYSTEM DESIGN

In this section, we describe system architecture, work flow and crypto operations of the privy system.

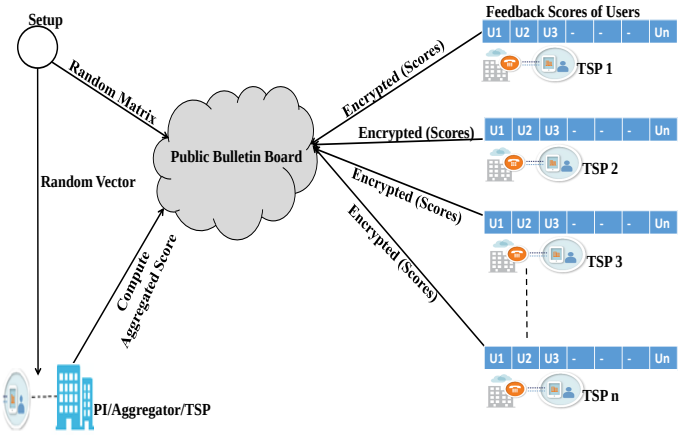


Fig. 1: Building Block of Decentralized Collaborative Aggregation System.

5.1 privy System Architecture and Design

The privy system consists of three major components: users, telecommunication service providers and the public BB, each with their own role in the system. Additionally, the system also has a setup phase. Figure 1 provides a high-level overview of our construction.

Users (U) Users access services provided by the telecommunication service providers. To enable access they need a calling identity from the service provider that can be either a unique IP address or a unique telephone number. The users actions are recorded in the call detail records that can be used to assign reputation scores to the users. Users are assumed not to be always on-line while their reputation scores are computed.

Telecommunication Service Provider (TSP) The service provider allows users to perform transactions (provided they supply the correct information and have enough rights). The service provider records the transactions of users in call detail records for billing and network management, and apply strong authentication. A service provider performs the following operations: 1) it computes the reputation of a user using his past call transactions; 2) it creates cryptograms and zero knowledge proofs of user's feedback values and publish those on the BB.

Public Bulletin Board (BB) The public BB holds the information provided by the collaborating TSPs during the course of collaboration. Specifically, it holds the identity of the user, encrypted feedback assigned to the user by the service provider, the public key of the service provider and the zero knowledge proof to prove that the encrypted feedback is indeed within the prescribed range. The public BB is an append-only database and it ensures that the analyst or collaborator gets the same information, and the data posted to it cannot be modified or deleted.

Setup The setup builds a random square matrix of size equal to the number of service providers, such that the discrete logarithm of the elements of matrix is not known to the protocol initiator. The setup also generates a special vector of dimension equal to the number of TSPs. Based on the matrix generated by the setup; the protocol initiator generates the public matrix of parameters that are used

by the other TSPs to compute the encrypted feedbacks. Once the setup has provided the matrix and the vector to the protocol initiator, its job finishes and it may go off-line thereafter.

The privy could also have an analyst or an aggregator that analyzes the information from BB and computes aggregated scores on behalf of the TSP; however, it is not a necessary component.

A typical privy session consists of several rounds of messages exchanged between BB and the TSPs. All participating TSPs participate in these rounds.

Further detail on the event flow of system is provided in a section 5.4.

We considered simple statistics that we called a reputation score of a user in the TSP. Each TSP encrypts the feedback value and posts it to the public BB. The collaborating TSP or the protocol initiator then homomorphically computes the aggregated weighted average of the user from the encrypted scores without learning scores provided by the collaborating TSP. The TSP exchanges information with the public BB in the following format [User-ID, ENC(LR), NIZK,PK_P]. The User-ID is an identity of the user and can be either an IP-address or a telephone number. The ENC(LR) is an encrypted feedback value of the user. It can be either zero or one. NIZK proof is a non-interactive zero knowledge proof that serves to prove the well-formedness of reported feedback and PK_P is a public key of the collaborator. We used the telephone number, as the identity of that user as service providers do not support anonymous identity. Whenever a TSP is uncertain about reputation of a particular user, it would then initiate the collaborative protocol for computing the aggregate global reputation of that user with other collaborators. At the end of the aggregation cycle, the TSP has the following information [User-ID, GR], where GR is the aggregated weighted reputation of the user. The TSP then uses this reputation score along with a predefined or automated threshold to place the user in a black or white list database.

5.2 Assumptions

We considered following assumptions in the design of privy system: 1) we assume that the collaborating TSPs have standalone system for monitoring the behavior of its users, and wish to have aggregated behavior of suspected user, 2) the protocol setup is assumed to be honest but curious and does not collude with the TSPs, 3) we assume that there exists a group G of p elements in which the Decisional Diffie-Hellman (DDH) assumption is intractable. All the security properties of our scheme depend upon the intractability of DDH problem in G. We shall establish in section 6 that the scheme is privacy preserving only if DDH assumption holds in G.

Assumption 1. **[DDH assumption]** Let G be a multiplicative group of finite order. The DDH assumption says, given g, g^a, g^b and a $\Omega \in \{g^{ab}, R\}$, it is computationally hard to decide whether $\Omega = g^{ab}$ or $\Omega = R$.

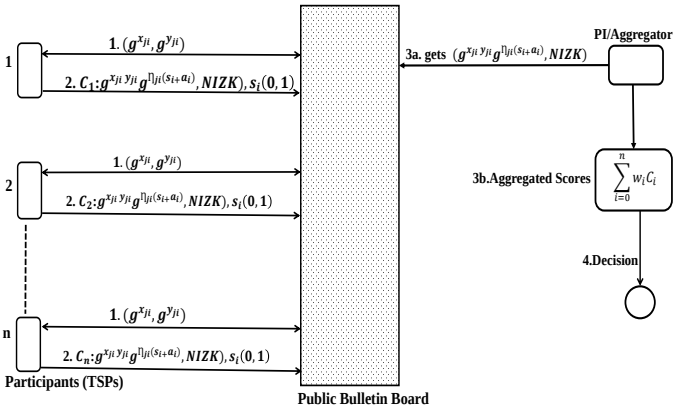


Fig. 2: Workflow of privy System.

5.3 Notations

Notations used in our system are summarized in Table 1 and algorithmic flow is presented in an algorithm 1.

5.4 Algorithm Workflow

We now describe the workflow of privy system in detail. The messages exchanged between BB, collaborators and the protocol initiators is presented in a figure 2 and each step is described as following.

1. Collaborating TSP would compute the secret and the public keys. It keeps the secret key and publishes the public key on the public BB. For submitting the encrypted scores to the public BB, the collaborator first computes the restructured key using the public keys of collaborating TSPs published at the BB.

2. The collaborating TSP then generates the cryptograms of its feedback score using his secret key and the restructured key. The information that is being posted on BB consist of two parts: the cryptogram of the feedback score and the zero-knowledge proof of well-formedness.

3. In the aggregation phase, the protocol initiator aggregates the reputation of user by requesting the encrypted values and NIZK proof from the public BB. The protocol initiator first checks the validity of zero knowledge proof, and then computes the final weighted aggregated reputation score of the user.

4. Finally, the protocol initiator or TSP compares the final aggregated score with the fixed or automatic threshold to place the user in the black or white list database.

5.5 privy Crypto Operations

With the system design in place, this section describes crypto operations of the privy system. As stated, in privy system, each collaborating TSP computes the weighted reputation of users without revealing information about two parameters: 1) the feedback scores assigned by the TSP, and 2) the trust weights of the collaborating TSP. The operations of privy are completely decentralized, however there exist a trusted setup for few operations. Specifically, the trusted setup and the collaborator wishing to have aggregated score performs the following functionalities:

TABLE 1: Symbols and abbreviations used the in privy System.

Symbols	Description
$TSP_1, TSP_2, \ldots, TSP_n$	Telecommunication Service Providers
$U_1, U_2, \ldots, U_u$	Set of Users registered in a TSP
$(x_{1i}, x_{2i}, \ldots, x_{ni})$	Secret key of TSP_i
$(g^{y_{1i}}, g^{y_{2i}}, \ldots, g^{y_{ni}})$	restructured key of TSP_i
pub_i	public key of TSP_i
TSP_a	Protocol Initiator or requester
G	cyclic Group of p elements in which DDH problem is hard
M	$n \times n$ matrix generated by setup
r_{ij}	elements of M
N	$n \times n$ matrix generated by TSP_a using $\mathcal{G}_M$
η_{ij}	discrete logarithm of elements of N
K	secret key chosen by TSP_a
ω	secret vector known to TSP_a
s_i	secret score of TSP_i
w_i	weight assigned to the score of $TSP_i, 1 \leq w_i \leq \tau$
α_i	random element generated by each TSP_i for generating the encrypted feedback
τ	max. weight assigned to the score of operators
NIZK	non-interactive zero knowledge
$[n]$	the set $\{1, 2, \ldots, n\}$
$\mathbf{c}_i$	encrypted score (feedback) of TSP_i

Trusted Setup: The setup chooses G, a finite multiplicative group of p elements. Let g be a random generator of g. It computes a random full rank $n \times n$ matrix $M = [r_{ij}]$, where, $i \in \{1, 2, \ldots, n\}, j \in \{1, 2, \ldots, n\}$ and $M \in \mathbb{Z}_p^{n \times n}$. Since, the rank of M is n, there exists $\omega \in \mathbb{Z}_p^n$, satisfying $\omega^T M = (1, 1, \ldots, 1) \mod p$.

ω can be computed uniquely by solving $\omega^T M = (1, 1, \ldots, 1) \mod p$. Let the solution be $\omega = (\omega_1, \omega_2, \ldots, \omega_n)$. The setup computes the power matrix $\mathcal{G}_M = g^M = [g^{r_{ij}}]$ ($\mathcal{G}_M \in G^{n \times n}$) and a vector $g^\omega = \{g_1^{\omega_1}, g_2^{\omega_2}, \ldots, g_n^{\omega_n}\}$ and post them on the public BB, where $g_1, g_2, \ldots, g_n$ are n random generators of G. Finally, the setup sends ω secretly to the protocol initiator.

Protocol Initiator: The protocol initiator selects a random $K \in \mathbb{Z}_p$ and publishes the matrix $N = [g^{\eta_{ij}}] = [g^{Kw_i r_{ij}}]$ on the public BB:

$$N = \begin{pmatrix} (g^{r_{11}})^{Kw_1} & (g^{r_{12}})^{Kw_2} & (g^{r_{13}})^{Kw_3} & \ldots & (g^{r_{1n}})^{Kw_n} \\ (g^{r_{21}})^{Kw_1} & (g^{r_{22}})^{Kw_2} & (g^{r_{23}})^{Kw_3} & \ldots & (g^{r_{2n}})^{Kw_n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ (g^{r_{n1}})^{Kw_1} & (g^{r_{n2}})^{Kw_2} & (g^{r_{n3}})^{Kw_3} & \ldots & (g^{r_{nn}})^{Kw_n} \end{pmatrix} \quad (1)$$

In the above equation, w_i is the weight assigned to TSP_i for all $i \in [n]$. The protocol initiator also publishes g^K on the BB. Later, we shall see that this step is necessary to make it impossible for the setup to breach the privacy of TSPs on its own. The protocol initiator also publishes a non-interactive zero knowledge proof of well-formedness of N. This proof shows that $N = g^{Kw_i r_{ij}}$, given $g^K, g^{r_{ij}}, \forall i, j \in \{1, 2, \ldots, n\}$. We assume that weights w_i are less than some constant τ. So, the NIZK proof should be constructed to validate the fact that $N_{ij} = g^{\eta_{ij}} \in \{(g^{Kr_{ij}})^{w_i} : 1 \leq w_i \leq \tau\}$. The protocol initiator also publishes a NIZK proof of the fact that $g^{\sum_{i=1}^n \eta_{ij}\omega_i} \in \{g^K, g^{2K}, \ldots, g^{\tau K}\}, \forall j \in \{1, 2, \ldots, n\}$. We have discussed both the NIZK proofs in the Appendix.

Key Generation: In order to provide the encrypted scores to the BB, the collaborating TSP has to generate the secret and the public keys. Let G be a finite group of p elements in which Decisional Diffie Hellman (DDH) problem is hard and g be a random generator of G. For each user u_j, the TSP uniformly choses the random value x from $[1, p-1]$ as his secret key and computes the public key as following: Pub_i=$g^{x_{ji}}$:$j \in \{1, 2, \ldots, n\}$. The value of x is kept secret within the TSP, and the value of Pub_i is published to the BB. The TSP who wish to provide the feedback first need to compute the restructured public key y_{ji} using public keys from public BB as following:

$$g^{y_{ji}} = \frac{\prod_{k=1}^{i-1} g^{x_{jk}}}{\prod_{k=i+1}^{n} g^{x_{jk}}} \quad (2)$$

We call it a restructured key because it is constructed by multiplying all the public keys before i and dividing it from all the public keys after i. Note that anyone can compute the $g^{y_{ji}}$ based on the published values of $Pub_k, k \in [n]$.

Submitting Scores: Once the TSP has computed the restructure key, the next step is encrypting the feedback scores and posting it to the public BB. The feedback score is $s_i \in \{0, 1\}$ is encrypted by multiplying the exponentiations of secret and restructured key with the exponentiations of feedback score and assigned weights. The scores are encrypted as following:

$$c_{ji} = g^{x_{ji}y_{ji}} g^{\eta_{ji}(s_i + \alpha_i)} \quad (3)$$

NIZK Proof of TSP Data Validity : The collaborating TSP has to provide a non-interactive zero-knowledge proof (NIZK) of knowledge that their provided encrypted reputation scores represent a valid input. Therefore, when a TSP provides the encrypted feedback, it should also compute and attach the NIZK of feedback correctness to public BB. The use of NIZK ensures that reported encrypted scores are not out of prescribe range without learning the score value. The TSP i encrypts the score as (C_i, g^{α_i}), where $C_i = (c_{1i}, c_{2i}, \ldots, c_{ni})$, $c_{ji} = g^{x_{ji}y_{ji}} g^{\eta_{ji}(s_i+\alpha_i)}$. The NIZK proof should provide witness exactly one of the two following statements is true.

1) $c_{1i} = g^{x_{ji}y_{ji}} g^{\eta_{1i}\alpha_i} \wedge c_{2i} = g^{x_{2i}y_{2i}} g^{\eta_{2i}\alpha_i} \wedge \cdots \wedge c_{ni} = g^{x_{ni}y_{ni}} g^{\eta_{ni}\alpha_i}$.

2)$c_{1i} = g^{x_{ji}y_{ji}} g^{\eta_{1i}(1+\alpha_i)} \wedge c_{2i} = g^{x_{2i}y_{2i}} g^{\eta_{2i}(1+\alpha_i)} \wedge \cdots \wedge c_{ni} = g^{x_{ni}y_{ni}} g^{\eta_{ni}(1+\alpha)}$.

Our NIZK proofs use Sigma protocol. For simplicity, we first present the well-formedness statement as in the form of

logical 'AND' statements, and then for each sub-statement, we generate a separate NIZK proof. The detailed description of NIZK is presented in the Appendix.

Score Aggregation: As the values of $g^{x_{jk}}, j \in \{1,2,\ldots,n\}, k \in \{1,2,\ldots,n\}$ are publicly available on the BB, anyone (TSP or protocol initiator) can compute $g^{y_{ji}}$, for any i and j, without having knowledge of y_{ji}. The service providers TSP_a can compute the weighted aggregated score of any user as following:

$$\mathbf{C} = (\prod_{j=1}^{n} c_{1j}, \prod_{j=1}^{n} c_{2j}, \ldots, \prod_{j=1}^{n} c_{nj}). \tag{4}$$

Let's denote $\mathbf{C} = (\mathbf{C}_1, \mathbf{C}_2, \ldots, \mathbf{C}_n)$ Then,

$$\mathbf{C}_i = \prod_{j=1}^{n} g^{y_{ij}x_{ij}} g^{\eta_{ij}(s_j+\alpha_j)}$$

Now, $\prod_{j=1}^{n} g^{y_{ij}x_{ij}} = \prod_{i=1}^{n}(\prod_{k=1}^{j-1} g^{x_{ik}} / \prod_{k=j+1}^{n} g^{x_{ik}})^{x_{ij}} = g^{\phi_i}$ (say)
where, $\phi_i = \sum_{j=1}^{n}(\sum_{k=1}^{j-1} x_{ik} - \sum_{k=j+1}^{n} x_{ik})x_{ij} = 0$, according to Proposition 1 [44].
Thus,

$$\mathbf{C}_i = g^{\sum_{j=1}^{n} \eta_{ij}(s_j+\alpha_j)}, \forall i \in \{1,2,\ldots,n\}.$$

$$\mathbf{C} = (g^{\sum_{i=1}^{n} \eta_{1i}(s_i+\alpha_i)}, g^{\sum_{i=1}^{n} \eta_{2i}(s_i+\alpha_i)}, \ldots, g^{\sum_{i=1}^{n} \eta_{ni}(s_i+\alpha_i)})$$

$$\mathbf{C} = (g^{K\sum_{i=1}^{n} r_{1i}w_i(s_i+\alpha_i)}, g^{K\sum_{i=1}^{n} r_{2i}w_i(s_i+\alpha_i)}, \ldots,$$
$$g^{K\sum_{i=1}^{n} r_{ni}w_i(s_i+\alpha_i)}).$$

Now, TSP_a computes the weighted product of the elements of $\mathbf{C}$ as:

$$L = \prod_{k=1}^{n} (\mathbf{C}_k)^{\omega_k} \tag{5}$$

$$= \prod_{k=1}^{n} (g^{K\sum_{i=1}^{n} r_{ki}w_i(s_i+\alpha_i)})^{\omega_k} \tag{6}$$

$$= g^{K\sum_{k=1}^{n}\sum_{i=1}^{n} \omega_k r_{ki} w_i(s_i+\alpha_i)} \tag{7}$$

Hence,

$$L = g^{K\sum_{i=1}^{n} w_i(s_i+\alpha_i)\sum_{k=1}^{n} \omega_k r_{ki}} \tag{8}$$

As mentioned before, $\omega^T M = (\mathbf{1})^T$ then

$$L = g^{K\sum_{i=1}^{n} w_i(s_i+\alpha_i)}$$
$$= \left(g^S \prod_{i=1}^{n} (g^{\alpha_i})^{w_i}\right)^K$$

So,

$$g^S = L^{1/K} / \prod_{i=1}^{n} (g^{\alpha_i})^{w_i} \tag{9}$$

Since, values of $g^{\alpha_i}, i \in \{1,2,\ldots,n\}$ are public and $K, w_i, 1 \le i \le n$ are known to TSP_a, it can compute g^S from the weighted product L. From g^S, a brute force search would yield the value of S which can be used for computing the weighted average reputation score by simply dividing S by $\sum_{i=1}^{n} w_i$.

Proposition 1. Let, $x_1, x_2, \ldots, x_n \in \mathbb{Z}_p^n$. Then $\sum_{i=1}^{n}(\sum_{j=1}^{i-1} x_j - \sum_{j=i+1}^{n} x_j)x_i = 0$.

proof:

$$\sum_{i=1}^{n}(\sum_{j=1}^{i-1} x_j - \sum_{j=i+1}^{n} x_j)x_i$$
$$= \sum_{i=1}^{n}\sum_{j=1}^{i-1} x_i x_j - \sum_{i=1}^{n}\sum_{j=i+1}^{n} x_i x_j$$
$$= \sum_{i=1}^{n}\sum_{j<i} x_i x_j - \sum_{i=1}^{n}\sum_{j>i} x_i x_j$$
$$= \sum_{j=1}^{n}\sum_{j>i} x_i x_j - \sum_{i=1}^{n}\sum_{j>i} x_i x_j.$$

as, $\sum_{j=1}^{n}\sum_{j>i} x_i x_j = \sum_{j=1}^{n}(x_1 + x_2 + \ldots + x_{j-1})x_j$
$= x_1\sum_{j=2}^{n} x_j + x_2\sum_{j=3}^{n} x_j + \cdots + x_{i-1}\sum_{j=i+1}^{n} x_j + \cdots +$
$x_{n-1}x_n$
$= \sum_{i=1}^{n}\sum_{j>i} x_i x_j$

Hence, $\sum_{i=1}^{n}(\sum_{j=1}^{i-1} x_j - \sum_{j=i+1}^{n} x_j)x_i = 0$.

Algorithm 1 Weighted Reputation Aggregation and Detection.

1: INPUT: There are n collaborating TSPs, each having u users. The $TSPs$ assigned a feedback score to each user as $s_i \in \{0,1\}$,and has a trust weight for other TSPs.
2: OUTPUT: Weighted Reputation Score (WRS) of the user U_κ, WRS= $\frac{\sum_{j=1}^{n} s_j w_j}{\sum_{j=1}^{n} w_j}$
3: **Each Collaborating TSP_i publishes scores to BB (lines 4-15):**
4: **procedure** KEY GENERATION(G,p)
5: $Pub_i = g^{x_{ji}} : j \in \{1,2,\ldots,n\}$ and $x_i \in_R [1, q-1]$
6: **end procedure**
7: **procedure** GENERATING RESTRUCTURED KEY ()
8: $y_{ji} = \sum_{k=1}^{i-1} x_{jk} - \sum_{k=i+1}^{n} x_{jk}$ and $s_i \in \{0,1\}$
9: **end procedure**
10: **procedure** ENCRYPTING AND PUBLISHING FEEDBACK SCORES ()
11: $Encryptedscores(c_{ji}) = g^{x_{ji}y_{ji}} g^{\eta_{ji}(s_i+\alpha_i)}$, where α_i is a random nonce
12: Generating NIZK proof that feedback score is either 0 or 1 as:
13: 1) $c_{1i} = g^{x_{ji}y_{ji}} g^{\eta_{1i}\alpha_i} \wedge c_{2i} = g^{x_{2i}y_{2i}} g^{\eta_{2i}\alpha_i} \wedge \cdots \wedge c_{ni} = g^{x_{ni}y_{ni}} g^{\eta_{ni}\alpha_i}$.
14: 2) $c_{1i} = g^{x_{ji}y_{ji}} g^{\eta_{1i}(1+\alpha_i)} \wedge c_{2i} = g^{x_{2i}y_{2i}} g^{\eta_{2i}(1+\alpha_i)} \wedge \cdots \wedge c_{ni} = g^{x_{ni}y_{ni}} g^{\eta_{ni}(1+\alpha)}$.
15: **end procedure**
16: **The Protocol Initiator computes weighted aggregated score of any user(U_k) as (lines 17-20):**
17: **procedure** WEIGHTED REPUTATION
$$L = g^{K\sum_{i=1}^{n} w_i(s_i+\alpha_i)}$$
$$= \left(g^S \prod_{i=1}^{n}(g^{\alpha_i})^{w_i}\right)^K$$
18: TSP computes S from the g^S using brute force search
19: Finally TSP computes WRS by dividing S by the sum of all weights.
20: **end procedure**
21: **The Protocol Initiator/TSP classifies U_k as (lines 22-30):**
22: **procedure** CLASSIFICATION AND DETECTION(TSP)
23: **if** $WRS(U_\kappa) >$ Threshold **then**
24: Place U_κ in a White-list
25: **else**
26: **if** $WRS(U_\kappa) <$ Threshold **then**
27: Place U_κ in a Black-list
28: **end if**
29: **end if**
30: **end procedure**

5.6 Protocol Summary

The important steps of the privy encryption are outlined below:

1) **Setup:** The trusted setup computes $n \times n$ full rank matrix $M = [r_{ij}]$, $i \in \{1,2,\ldots,n\}, j \in \{1,2,\ldots,n\}$, and $M \in \mathbb{Z}_p^{n\times n}$. There exists $\omega \in \mathbb{Z}_p^n$, such that $\omega^T M = (1,1,\ldots,1) \mod p$. If $\omega = (\omega_1, \omega_2, \ldots, \omega_n)$ and $\sum_{i=1}^{n} \omega_i r_{ij} = 1 \mod p, \forall j \in$

$\{1, 2, \ldots, n\}$ then ω can be uniquely computed because M is a full rank matrix. The setup computes the power matrix $\mathcal{G}_M = g^M = [g^{r_{ij}}]$, and $\mathcal{G}_M \in G^{n \times n}$, and made it public, and secretly send vector ω secretly to the protocol initiator. Further, the setup also publishes a vector $g^\omega = \{g_1^{\omega_1}, g_2^{\omega_2}, \ldots, g_n^{\omega_n}\}$ on the public BB, where $g_1, g_2, \ldots, g_n$ are n random generators of G.

2) **Initialization:** The protocol initiator/aggregator selects a random $K \in \mathbb{Z}_p$, publishes the matrix $N = [g^{\eta_{ij}}] = [g^{Kw_i r_{ij}}]$, g^K and a NIZK of well-formedness of N. The NIZK proof shows that $N = g^{Kw_i r_{ij}}$, given $g^K, g^{r_{ij}}, \forall i, j \in \{1, 2, \ldots, n\}$. We assume that the weights w_i are less than some constant τ. The NIZK proof should be constructed to validate the fact that $N_{ij} = g^{\eta_{ij}} \in \{(g^{Kr_{ij}})^{w_i} : 1 \leq w_i \leq \tau\}$. The protocol initiator also publishes a NIZK proof of the fact that $g^{\sum_{i=1}^n \eta_{ij}\omega_i} \in \{g^K, g^{2K}, \ldots, g^{\tau K}\}, \forall j \in \{1, 2, \ldots, n\}$.
3) **Phase I:** In the first step, each service provider $TSP_i, i \in \{1, 2, \ldots, n\}$ publishes a public key $Pub_i = (g^{x_{1i}}, g^{x_{2i}}, \ldots, g^{x_{ni}})$ on the BB.
4) **Phase II:** Each collaborating TSP_i computes a vector $\mathbf{c}_i$ such that, $\mathbf{c}_i^T = (c_{1i}, c_{2i}, \ldots, c_{ni})$, where $c_{ji} = g^{x_{ji}y_{ji}} g^{\eta_{ji}(s_i+\alpha_i)}$, and α_i is a random nonce. TSP_i publishes $\{\mathbf{c}_i, g^{\alpha_i}\}$ and a NIZK of well formedness of $c_{ji} = g^{x_{ji}y_{ji}} g^{\eta_{ji}(s_i+\alpha_i)}$ for each $j \in \{1, 2, \ldots, n\}$, where s_i is the secret score of TSP_i and $y_{ji} = \sum_{k=1}^{i-1} x_{jk} - \sum_{k=i+1}^{n} x_{jk}$ is the restructured key.The NIZK proof should provide prove for the statement $c_{ji} \in \{g^{x_{ji}y_{ji}} g^{\eta_{ji}\alpha_i}, g^{x_{ji}y_{ji}} g^{\eta_{ji}(1+\alpha_i)}\}$. As the values of $g^{x_{ji}}, i \in \{1, 2, \ldots, n\}, j \in \{1, 2, \ldots, n\}$ are public, anyone can compute $g^{y_{ji}}$, for any values of i and j without having y_{ji}. This vector $\mathbf{c}_i$ is uploaded to the public BB along with g^{α_i}. Once all operator contributed and aggregation cycle terminated, any TSP_a can calculate $\mathbf{C} = (\prod_{j=1}^n c_{1j}, \prod_{j=1}^n c_{2j}, \ldots, \prod_{j=1}^n c_{nj})$.
5) **Computing Weighted Average:** Protocol initiator computes $L = \prod_{k=1}^n (g^{\sum_{i=1}^n \eta_{ki}(s_i+\alpha_i)})^{\omega_k} = (g^S \prod_{i=1}^n (g^{\alpha_i})^{w_i})^K$. From this it calculates $g^S = L^{1/K} / \prod_{i=1}^n (g^{\alpha_i})^{w_i}$. Then it finds S using brute force search and compute the weighted average: $\frac{S}{\sum_{i=1}^n w_i}$.

6 Security and Privacy Analysis of privy

The collaborating TSP wishes that no entity taking part in the collaboration would learn the information provided by them. The privy system ensures privacy of collaborators and their users with the use of encrypted scores published on the BB. The aggregator would only learn the aggregate statistics without learning any individual feedback score. As long as the trusted setup does not collude with protocol initiator, the privy system ensures the following security properties that we proved in lemma 1 to 4:

- The setup alone is not able to compromise the privacy of TSPs. It is proved in a Lemma 1.
- The TSPs would not learn the values of the trust weights assigned by the protocol initiator, hence privacy of the protocol initiator remains preserved. This property is proved in a Lemma 2.
- If two TSPs have the same trust weights and they assign different scores to a user (one assign 1 and other assign 0) then the protocol initiator would not learn any scores. It is proved in a Lemma 3
- The weighted sum $S = \sum_{i=1}^n w_i s_i$ does not allow protocol initiator to learn the scores of particular TSP even after colluding with others. This is the most crucial security property of the system and is proved in a Lemma 4.

Lemma 1. **The setup alone cannot compromise the privacy of a TSP.**

Proof 1. We prove that an adversary, who plays as the setup, cannot distinguish between two BBs where the same TSP (say TSP_n) has provided two different scores if the DDH assumption holds true. Let us assume that the adversary has received the input g^α, g^K and a challenge $\Omega \in \{g^{\alpha K}, g^{\alpha K} g^K\}$. The adversary has to find whether $\Omega = g^{\alpha K}$ or $\Omega = g^{\alpha K} g^K$. The adversary performs the following operations: she computes the matrix $M \in \mathbb{Z}_p^{n \times n}$, $\mathcal{G}_M = [g^{r_{ij}}]$, and the vector ω satisfying $\omega^T M = (1, 1, \ldots, 1)$. Then, she publishes the $n \times n$ matrix $N = [(g^K)^{r_{ij}}], i, j \in [n]$. For all service provider $TSP_i, i \in [n]$, the adversary publishes $Pub_i = (g^{x_{1i}}, g^{x_{2i}}, \ldots, g^{x_{ni}})$, and keeps the corresponding secret key $(g^{x_{1i}}, g^{x_{2i}}, \ldots, g^{x_{ni}})$. The adversary chooses suitable scores for all TSPs except TSP_n. For all TSP_i, the adversary generates the scores as she knows all secret keys and scores. Now, the adversary chooses $\mathbf{c}_n = (c_{1n}, c_{2n}, \ldots, c_{nn})$, where $c_{jn} = g^{x_{jn}y_{jn}} \Omega^{r_{jn}}, \forall j \in [n]$. The adversary knows r_{jn}, $x_{ji} \forall j, i \in [n]$ and $y_{ji} = \sum_{k=1}^{i-1} x_{jk} - \sum_{k=i+1}^{n} x_{jk}$. Note that if $\Omega = g^{\alpha K}$, $\mathbf{c}_n$ corresponds to the case: $s_n = 0$ and if $\Omega = g^{\alpha K} g^K$, $\mathbf{c}_n$ corresponds to the case: $s_n = 1$. Now, if the setup can identify the score s_n from the bulletin board, the adversary can identify the correct value of Ω.

Lemma 2. **The service providers would not learn the secret weights.**

Proof 2. Let us assume there are two sets $W = \{w_i : i \in \{1, 2, \ldots, n\}\}$ and $W' = \{w_i' : i \in \{1, 2, \ldots, n\}\}$. Let, $N = [g^{r_{ij}Kw_j}]$ and $N' = [g^{r_{ij}Kw_j'}]$, where $1 \leq w_i, w_i' \leq 3, i, j \in \{1, 2, \ldots, n\}$. Now, from Assumption 2, we can say $(g, g^{r_{ij}}, g^K, g^{r_{ij}Kw_j}) \overset{c}{\approx} (g, g^{r_{ij}}, g^K, R) \overset{c}{\approx} (g, g^{r_{ij}}, g^K, g^{r_{ij}Kw_j'})$. Thus, no Probabilistic Polynomial Time (PPT) adversary can distinguish between N and N'.

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

Lemma 3. **The protocol initiator TSP_a cannot distinguish between two bulletin boards where two service providers having equal weights interchange their scores.**

Proof 3. Let us assume that the two honest service providers are TSP_i and $TSP_j, i < j$. We assume that the weights assigned to both of them is w. The NIZK proofs do not provide any useful information that can help TSP_a to

A	B
$g^{-x_{1i}x_{1j}}(g^{\eta_{1i}})^{1+\alpha_i}, g^{x_{1i}x_{1j}}(g^{\eta_{1j}})^{\alpha_j}$	$g^{-x_{1i}x_{1j}}(g^{\eta_{1i}})^{\alpha_i}, g^{x_{1i}x_{1j}}(g^{\eta_{1j}})^{1+\alpha_j}$
$g^{-x_{2i}x_{2j}}(g^{\eta_{2i}})^{1+\alpha_i}, g^{x_{2i}x_{2j}}(g^{\eta_{2j}})^{\alpha_j}$	$g^{-x_{2i}x_{2j}}(g^{\eta_{2i}})^{\alpha_i}, g^{x_{2i}x_{2j}}(g^{\eta_{2j}})^{1+\alpha_j}$
$g^{-x_{3i}x_{3j}}(g^{\eta_{3i}})^{1+\alpha_i}, g^{x_{3i}x_{3j}}(g^{\eta_{3j}})^{\alpha_j}$	$g^{-x_{3i}x_{3j}}(g^{\eta_{3i}})^{\alpha_i}, g^{x_{3i}x_{3j}}(g^{\eta_{3j}})^{1+\alpha_j}$
...	...
$g^{-x_{ni}x_{nj}}(g^{\eta_{ni}})^{1+\alpha_i}, g^{x_{ni}x_{nj}}(g^{\eta_{nj}})^{\alpha_j}$	$g^{-x_{ni}x_{nj}}(g^{\eta_{ni}})^{\alpha_i}, g^{x_{ni}x_{nj}}(g^{\eta_{nj}})^{1+\alpha_j}$

TABLE 2: Protocol Initiator's Two Possible Views of the BB as Mentioned in Lemma 3

.

distinguish between the two bulletin boards mentioned above. Therefore, for the time being, we may assume that the NIZK proofs need not be published to the bulletin board. This assumption will not affect the adversary's ability to breach the privacy of TSP_i and TSP_j. We assume that the public key vectors of these two TSPs are given by $Pub_i = (g^{x_{1i}}, g^{x_{2i}}, \ldots, g^{x_{ni}}), Pub_j = (g^{x_{1j}}, g^{x_{2j}}, \ldots, g^{x_{nj}})$. Also, let s be the score of TSP_i and the score of TSP_j will be $1-s$. Hence, the public bulletin board will contain a vector of cryptograms $(\mathbf{c}_i, g^{\alpha_i})$ such that $\mathbf{c}_i^T = (c_{1i}, c_{2i}, \ldots, c_{ni})$, $c_{ki} = g^{x_{ki}y_{ki}}g^{\eta_{ki}s}g^{\eta_{ki}\alpha_i}, \forall k \in \{1,2,\ldots,n\}$. Similarly, the vector of cryptograms from TSP_j will be $(\mathbf{c}_j, g^{\alpha_j})$ such that $\mathbf{c}_j^T = (c_{1j}, c_{2j}, \ldots, c_{nj})$, $c_{kj} = g^{x_{kj}y_{kj}}g^{\eta_{kj}(1-s)}g^{\eta_{kj}\alpha_j}, \forall k \in \{1,2,\ldots,n\}$.

Now, we show that if there exists an adversary $\mathcal{A}$ that can distinguish between two bulletin boards where the scores of two honest TSPs TSP_i and TSP_j having the same weight are interchanged. We assume all all TSPs other than TSP_i and TSP_j reveal their secrets to the protocol initiator TSP_a. It is easy to see that if the honest TSPs TSP_i and TSP_j have the same score, then TSP_a can trivially find out their scores. So, we assume that their scores are different, that is, one of them is s and the other one is $1-s$, $s \in \{0,1\}$. Now the vector of cryptograms published by TSP_i will be $(\mathbf{c}_i, g^{\alpha_i})$, where $\mathbf{c}_i = (c_{1i}, c_{2i}, \ldots, c_{ni}), c_{ki} = g^{x_{ki}y_{ki}}(g^{\eta_{ki}})^{s+\alpha_i}, \forall k \in \{1,2,\ldots,n\}$. Now, $y_{ki} = \sum_{t=1}^{i-1} x_{kt} - \sum_{t=i+1}^{n} x_{kt} = \sum_{t=1}^{i-1} x_{kt} - \sum_{t=i+1}^{j-1} x_{kt} - x_{kj} - \sum_{t=j+1}^{n} x_{kt} = \lambda_1 - x_{kj}$, where $\lambda_1 = \sum_{t=1}^{i-1} x_{kt} - \sum_{t=i+1}^{j-1} x_{kt} - \sum_{t=j+1}^{n} x_{kt}$. Since, all the TSPs excepting TSP_i and TSP_j have colluded, we may assume λ_1 is known to TSP_a. Now the vector of cryptograms published by TSP_j will be $(\mathbf{c}_j, g^{\alpha_j})$, where $\mathbf{c}_j = (c_{1j}, c_{2j}, \ldots, c_{nj}), c_{kj} = g^{x_{kj}y_{kj}}(g^{\eta_{kj}})^{1-s+\alpha_j}$. Now, $y_{kj} = \sum_{t=1}^{j-1} x_{kt} - \sum_{t=j+1}^{n} x_{kt} = \sum_{t=1}^{i-1} x_{kt} + x_{ki} + \sum_{t=i+1}^{j-1} x_{kt} - \sum_{t=j+1}^{n} x_{kt} = \lambda_2 + x_{ki}$, where $\lambda_2 = \sum_{t=1}^{i-1} x_{kt} + \sum_{t=i+1}^{j-1} x_{kt} - \sum_{t=j+1}^{n} x_{kt}$. Here too, we can assume λ_2 is known to TSP_a.

With this we can rewrite each $c_{ki} = (g^{x_{ki}})^{\lambda_1}g^{-x_{ki}x_{kj}}(g^{\eta_{ki}})^{s+\alpha_i}$ and $c_{kj} = (g^{x_{kj}})^{\lambda_2}g^{x_{ki}x_{kj}}(g^{\eta_{kj}})^{1-s+\alpha_j}, \forall k \in \{1,2,\ldots,n\}$. All other cryptograms uploaded by other dishonest TSPs are chosen by TSP_a itself. Now, TSP_a can distinguish between the two cases: $s=0$ and $s=1$ only if it can distinguish between the following two distribution of the table 2. It can be observed that A corresponds to the case $s=1$ and B corresponds to the case $s=0$. So, if a distinguisher can distinguish between the two cases, it can find whether $s=1$ or $s=0$.

Lemma 4. **If there exist $e+1$ honest TSPs, $TSP_{k_1}, TSP_{k_2}, \ldots, TSP_{k_e}, TSP_{k_{e+1}}, \{k_j : j \in [e]\} \subset [n]$, who do not reveal their scores and if there exist $(s'_{k_1}, s'_{k_2}, \ldots, s'_{k_e})$ and $(s''_{k_1}, s''_{k_2}, \ldots, s''_{k_e}) \in \{0,1\}^e, s'_{k_j}, s''_{k_j} \in \{0,1\}, \forall j \in [e]$ such that $(s'_{k_1}, s'_{k_2}, \ldots, s'_{k_e}) \neq (s''_{k_1}, s''_{k_2}, \ldots, s''_{k_e})$ and $\sum_{j=1}^{e} s'_{k_j} w_{k_j} = w_{k_{e+1}} + \sum_{j=1}^{e} s''_{k_j} w_{k_j}$, then the score of $TSP_{k_{e+1}}$ cannot be compromised.**

Proof 4. For ease of notation, let us assume that the first $e+1$ TSPs are honest, i.e $k_i = i, \forall i \in [e+1]$. Also assume that $s'_{e+1} = 0$ and $s''_{e+1} = 1$. The $n-e-1$ TSPs, namely $TSP_{e+2}, TSP_{e+3}, \ldots, TSP_n$ have colluded with the P.I. The encrypted score of TSP_i is of the form $C_i = (c_{1i}, c_{2i}, \ldots, c_{ni})$, where $c_{ji} = g^{x_{ji}y_{ji}}g^{\eta_{ji}(s_i+\alpha_i)}, j \in [n]$. The P.I. knows all the secrets of the colluding TSPs. Since, $g^{\sum_{i=1}^{n} x_{ji}y_{ji}} = 0, g^{\sum_{i=1}^{e+1} x_{ji}y_{ji}} = 1/\prod_{i=e+2}^{n} g^{x_{ji}y_{ji}}$. The P.I. can compute $\prod_{i=e+1}^{n} g^{x_{ji}y_{ji}}$ and hence can compute $g^{\sum_{i=1}^{e+1} x_{ji}y_{ji}}$. Hence, the P.I. can compute $\bar{X}_{j(e+1)} = c_{j(e+1)}/g^{\sum_{i=1}^{e+1} x_{ji}y_{ji}} = g^{\eta_{j(e+1)}(s_{e+1}+\alpha_{e+1})}/\prod_{i=1}^{e} g^{x_{ji}y_{ji}}, \forall j \in [n]$. We assume that $c_{ji} = c'_{ji}, \forall j \in [n], \forall i \in [e+1]$ if $s_i = s'_i$ and $c_{ji} = c''_{ji}, \forall j \in [n], \forall i \in [e+1]$ if $s_i = s''_i$. Let us assume $\bar{X}_{ji} = c'_{ji} = g^{x_{ji}y_{ji}}g^{\eta_{ji}(s'_i+\alpha_i)}, \forall j \in [n], \forall i \in [e+1]$ and $\bar{X}'_{ji} = c''_{ji} = g^{x_{ji}y_{ji}}g^{\eta_{ji}(s''_i+\alpha_i)}, \forall j \in [n], \forall i \in [e+1]$. Again, $\phi_i = (\bar{X}_{i1}, \bar{X}_{i2}, \ldots, \bar{X}_{i(e+1)}), \forall i \in [n]$ and $\phi'_i = (\bar{X}'_{i1}, \bar{X}'_{i2}, \ldots, \bar{X}'_{i(e+1)}), \forall i \in [n]$. Using Lemma **??** and Lemma **??**, we can claim that $(\phi_1, \phi_2, \ldots, \phi_n) \overset{c}{\approx} (\phi'_1, \phi'_2, \ldots, \phi'_n)$. Hence, prove the result.

Note that, Lemma 4 proves that the adversary will not be able to compromise the score of a TSP if the partial tally of all the honest TSPs does not allow her to do so. The protocol outputs the overall tally of all the TSPs and the adversary knows the scores of all colluding TSPs. Hence, she can compute the partial tally of all honest TSPs through subtracting the former by the latter. Thus, we can say that our protocol is secure in the sense that it does not allow the adversary to know anything in addition to what she can learn trivially from the intended output of the protocol, that is, the weighted sum of scores $S = \sum_{i=1}^{n} w_i s_i$.

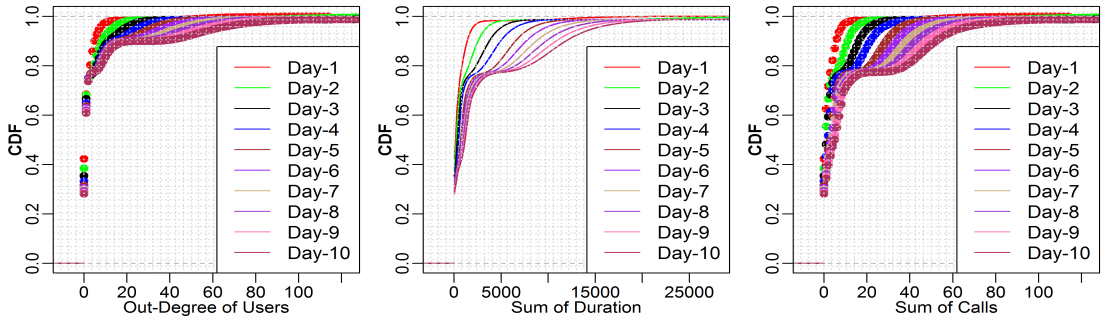
7 IMPLEMENTATION

In this section, we discuss the evaluation methodology and implementation of prototype.

7.1 Synthetic Data-Set

It is hard to obtain the real call records from multiple service providers. Therefore, we verify the performance of the privy system using the anonymized CDRs from one anonymous service provider.

The anonymized dataset consists of around 1 million unique users and more than 50 million call records. As



(a) (b) (c)

Fig. 3: Statistics for Data-Set A) Cumulative distribution for Edge Degree of Users with average out-degree distribution of 15; B) Cumulative distribution of total Call Duration for ten days with the average out-degree of 15, average call rate of 3 calls and average call duration of 200 seconds per day; C) Cumulative distribution of Call-Rate for 10 days with the average out-degree of 15, and call rate of 3 per day.

the data provided by the TSP is not labeled, therefore we considered all users in the data set as the legitimate. Analyzing CDRs, we observed that legitimate users follow the power law distribution for their in-degree and out-degree distributions similar to what has been proved in previous works [45]–[48]. Using these observations, we generated the synthetic CDR's for 6 service providers.

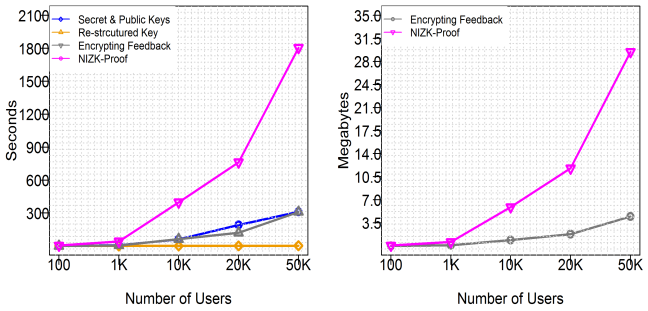
Specifically, first a Barabàsi-Albert (BA) graph model $G(V, E)$ has been generated and then its edges are labeled with weights. We used the following settings for the degree distribution, the call duration and the call rate. The average degree distribution of the legitimate users follow power law distribution with the average out-degree of 15 unique callees, the exponential distribution is used for the call duration with the average call duration of 200 seconds, and the call rate of the user is modeled using Poisson distribution with the mean value of 3.5 calls per day. For generating spammers, we used the following configurations. The call duration of the spammer follows an exponential distribution but with the average duration of 90 seconds towards few callees, and the average duration of 40 seconds with a large number of callees [49], [50], the out-degree of the spammer is randomly chosen between 500 and 2000 unique callees and the average call rate is 1.5. We do not have any information regarding how calls are distributed among operators in anonymized CDRs. Therefore, calls are distributed among all 6-service providers (70% calls made by legitimate users to users registered on the same network while remaining 30% are equally distributed among other service providers). The number of legitimate users are fixed (50K) in each TSP with 20% spammers. Each collaborating TSP computes reputation score and classifies the caller as a spammer (0) and non-spammer (1) using the approach presented in [9]. We aggregated the reputation scores over the period of one day and repeated the experiments for 10 times. The statistics for the different call features extracted from the synthetic data of one service provider are presented in a figure 3.

7.2 Crypto Implementation

To evaluate the computational and bandwidth overheads of our scheme, we developed and tested the prototype implementation in the Java programming language using the bouncy castle cryptographic library [51]. We implemented only the cryptographic parts of the protocol with the NIST elliptic curve P-256 standard. We ran all experiments on a single core of an Intel Core i7 3.40 GHz with 8 GB memory on a Windows 10 operating system. The implementation consists of two major modules. The first module implements the functionality of collaborating TSPs: i.e. generating the secret, the public and the restructured key, generating the cryptograms of the feedback scores, and generating the associated zero knowledge proof. The second module implements the functionality of the aggregator: i.e. checking the NIZK proof of the provided feedback and then aggregate them. The code can be optimize further and is easily parallelizable. We tested the performance of the collaborator and the aggregator using the same machine.

TABLE 3: Collaborator and Aggregator microbenchmark of timing and space.

Operation	Time (msec)	Space (Bytes)
Encryption (per Cryptogram)	10.1	87
NIZK-Proof (per NIZK proof)	59.17	589
Aggregation (100 Feedback)	57.32	-



(a) (b)

Fig. 4: Microbenchmark (Computation and Bandwidth Overheard) evaluation of cryptographic operations for different number of users.

7.3 Microbenchmark

The timings of two important operations such as encryption and NIZK proof of the protocol at the collaborator side, and

the aggregation operation for 100 feedbacks are summarized in a table 3. The computation time for both operations at the collaborator is important for the scalability as cryptograms and NIZK are generated for each feedback value. For a large data size such 50K users, the collaborator could perform encryption and NIZK proof in around 1800 seconds.

We now turn to the timings of basic operations of the privy protocol for different number of users. We conducted our experiments on a single thread of the processor. All experiments were performed 10 times, and the results are reported in the average of measurements. Figure 4.A shows the computation time required for the four major operations: generating the public and the secret keys, computing the restructure key, the encrypted response, and the NIZK proof. The key generation and encryption operations are very efficient, since it only involves simple exponentiations and multiplication. On the contrary, the NIZK proof generation is more expensive, since it needs to generate a NIZK proof non-interactively. As shown in the Figure 4.A the computation time increases linearly with the number of users. Specifically, for 50000 users, the collaborator needs about 1800 seconds to generate the encrypted feedback and its associated NIZK proof. The aggregator module can aggregate the 500K responses from the bulletin board in about 210 seconds using one core of the machine. The aggregator computation overhead increases linearly with the number of responses.

The privy protocol scales well with the number of cores in the CPU because all computation that grows linearly with the number of users is parallelizable. On our eight core machines, the computation time for each operation considerably decreased to acceptable time: i.e. 270 seconds for TSP and 30 seconds for the aggregator.

Figure 4.B shows the collaborators bandwidth for the encrypted response and NIZK proof for the different number of users, respectively. The collaborator bandwidth overhead for one user is less than a kilobyte for the encrypted response and NIZK proof. Specifically, for 50K users, the collaborator would consume about 29.5mb for the NIZK proof and about 4.5mb for the encrypted responses. Most bandwidth consumption is due to the non-interactive NIZK proof; however, the overhead is still acceptable. We also measured the storage requirement for the bulletin board. For holding 500K responses from the collaborators, the bulletin board requires the storage of about 0.361 GB.

8 Performance Evaluation

In this section, we evaluate the performance of privy for two parameters: The true positive rate (TPR) and the false positive rate (FPR). TPR is the ratio of correctly identified spammers to the total number of spammers in the network, and FPR is a total number of legitimate users misclassified as the spammers to the total number of legitimate users in the network. We investigate how TPR increases and FPR decreases with the number of collaborators. For the classification at the TSP level, we used the approach mentioned in [9] and compared performance of the collaborative system with standalone systems.

TABLE 4: True Positive Rate of privy System for different number of equally trusted collaborators.Percentage of Spammers is 20%.

System	Day-1	Day-2	Day-3	Day-4	Day-5
3 Collaborators	58.47%	71.54%	92.74%	96.5%	99.72%
4 Collaborators	64.76%	78.26%	99.52%	99.85%	100.0%
5 Collaborators	73.17%	88.97%	99.92%	99.97%	100.0%
CallerREP [9].	18.33%	25.34%	37.01%	43.19%	52.04%
CallRank [7].	0.00%	0.00%	0.00%	14.14%	27.54%

TABLE 5: False Positive Rate of privy System for different number of equally trusted collaborators.Percentage of Spammers is 20%.

System	Day-1	Day-2	Day-3	Day-4	Day-5
3 Collaborators	7.67%	7.18%	6.70%	6.31%	5.51%
4 Collaborators	7.29%	6.29%	6.18%	5.73%	4.41%
5 Collaborators	6.07%	3.59%	0.36%	0.00%	0.00%
CallerREP [9]	10.28%	8.68%	8.07%	7.91%	7.75%
CallRank [7]	22.52%	20.48%	20.19%	18.86%	14.24%

8.1 True Positive Rate

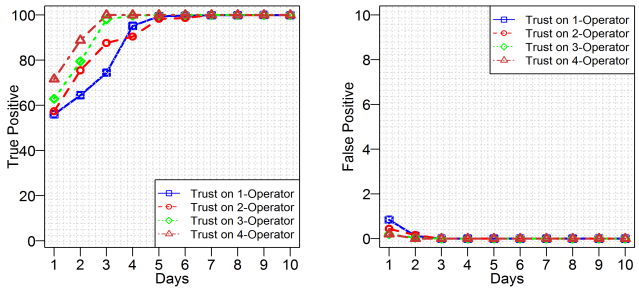
We evaluated the performance of privy system for two features: 1) systems performance over the time, and 2) effect of the number of collaborators on the detection performance. The detection rate of system depends on the number of collaborators, the more the number of collaborators higher would be the detection rate. Table 4 demonstrates the results for both parameters. It is clear from the table 4 that privy system out-performs the non-collaborative systems in terms of detection rate and is able to block all spammers within 3 days much earlier than the non-collaborative systems. Specifically, with the 5 collaborators, the system achieves a TPR of more than 80% on the first day, and further reaches to 99% TPR within 3 days, and 100% in 5 days. On the other hand, we observed that standalone system [9] is able to detect the spammers only if spammers target a large number of recipients of the same network.

8.2 False Positive Rate

Although TPR is important feature for evaluating the performance of any detection system, however, the detection system requires to have a zero FPR. A high FPR would not only irritate legitimate callers and callees but would also result in a revenue loss for the TSP because of blocking legitimate callers. The privy system achieves 0% FPR over the time and outperforms the non-collaborative systems as shown in table 5. Specifically, privy achieves FPR of 0% in 3 days much earlier than the non-collaborative system which suffers from a high FPR even after 5 days. The FPR also decreases with the number of collaborators and with the 5 collaborators, it achieves a FPR of 0%, thus allowing all non-spammers to use the network. The FPR of non-collaborative systems is not acceptable they have FP rate of more than 5% even after 5 days.

8.3 Effect of Trust Weights

In Sections 8.1, 8.2, we provided TPR and FPR for the condition when collaborating TSP are equally trusted i.e. they are not supporting any spam campaign and honest in providing the feedback. Further, TSPs shows different



(a) True Positive Rate (b) False Positive Rate

Fig. 5: privy performance when trust weights are assigned to collaborators.(a) Trust Positive Rate, and (b) False Positive Rate. The number spammers have been set to 20%.

trust level on other TSPs. They assign high trust score to the TSP they have the bilateral agreement and assign small trust score to TSP that are not directly connected with them and send spam calls. TSPs require that trust weights on other TSP should be consider while computing aggregate reputation of the user. Figures 5 show the TPR and FPR of the privy when different weights are assigned to the collaborating TSPs. In a setup, we assigned a fixed weight of 0.1 to non-trusted TSP and 1 to the trusted TSP. The true positive rate decreases with the number of non-trusted TSPs as shown in a Figure 5. The more the number of non-trusted TSPs the more the spammers passed through the network. The privy system with consideration of trust weights would manage to have 100% TPR in 7 days, which is very late than the fully trusted environment. The FPR is 10% on the first day but reduces to zero within three days.

9 Practical Deployment and Limitations

In real practice, a telecommunication service provider may have multiple call-handling servers but logs users transaction on a single billing or call record database. The privacy preservation property of privy system could convince a TSP to participate in the collaboration process, and convincing only a small number of TSP's for collaboration would greatly minimize detection time and increase detection rate. A TSP computes local feedback score of the user explicitly using recorded logs or implicitly asking their customers for the feedback about others. The challenge in deploying the privy is setting the aggregation time. We suggest aggregation cycle to be one day because it would have small overheads on TSP side. Smaller or larger aggregation cycles are also possible but it would either increase the computation time and the communication load, or delay the detection process. For the detection, the TSP either can accept the recommendation of collaborators for blocking the user or uses recommendation along with the local behavior of users to him in the white, black or under observation list.

The proposed privy system assumes that the collaborating TSP would perform two activities: 1) publish keys to the bulletin board, and 2) report encrypted feedback to the bulletin board. A malicious TSP could distort operations of system by simply reporting public keys and then withholding the feedback scores. However, the protocol initiator knows that a particular TSP have not provided the scores. This limitation can be overcome in two ways: 1) imposing some penalty on the TSP not providing scores, and 2) disgracefully removing such TSP by removing his public keys with the help of trusted setup. In the privy system, whenever a new TSP wishes to join or leave the collaboration system, the TSP needs to compute the restructured key again. This makes our system more appropriate for the conditions where the number of collaborators remains static over the time. However, as a part of future work we intend to solve this limitation.

10 Conclusion

We believe that convincing only a few TSPs for the collaboration by ensuring privacy of their customers would greatly minimize the frauds and spams over the telecommunication networks. In this paper, we have described the privacy preserving decentralized collaboration system for the effective spam detection without incurring high overheads and trusted third party. The privy system is based on the concept of decentralization and homomorphic cryptography that securely aggregates the feedback scores provided by the collaborators without learning value of the feedback. Our extensive evaluation and analysis shows that privy not only improves the detection rate but also has a small communication and computational overhead. Further, privacy and security analysis shows that privy strongly protects private information of collaborators and their customers under malicious and honest but curious models. The system is easily scalable to handle large number of users. Finally, the system can also be applied in other domains such as private reputation aggregation on the on-line marketplaces, network intrusion detection, and the fraud and spam detection over social networks with minor modifications.

Acknowledgment

We thank the anonymous reviewers for their helpful comments. The research in this paper was supported by the ERC starting grant No. 306994.

References

[1] (2015, 04, April) Communications fraud control association (cfca), 2015 global fraud loss surveys, 2015.

[2] Spam Phone Calls Cost U.S. Small Businesses Half-Billion Dollars in Lost Productivity, Marchex Study Finds. [Online]. Available: http://goo.gl/jTrgp3

[3] H. Tu, A. Doupè, Z. Zhao, and G. J. Ahn, "Sok: Everyone hates robocalls: A survey of techniques against telephone spam," in *Proc. 2016 IEEE Symposium on Security and Privacy (SP)*, May 2016, pp. 320–338.

[4] M. Hansen, M. Hansen, J. Möller, T. Rohwer, C. Tolkmit, and H. Waack, "Developing a Legally Compliant Reachability Management System as a Countermeasure against SPIT," in *Proc. Third Annual VoIP Security Workshop, Berlin, Germany*, 2006.

[5] Y. Hong, S. Kunwadee, Z. Hui, S. ZonYin, and S. Debanjan, "Incorporating Active Fingerprinting into SPIT Prevention Systems," in *Proc. Third Annual VoIP Security Workshop, Berlin, Germany*, 2006.

[6] D. Lentzen, G. Grutzek, H. Knospe, and C. Porschmann, "Content-Based Detection and Prevention of Spam over IP Telephony - System Design, Prototype and First Results," in *Proc. IEEE International Conference on Communications, ICC 2011*, June,2011, pp. 1–5.

[7] V. Balasubramaniyan, M. Ahamad, and H. Park, "CallRank: Combating SPIT Using Call Duration, Social Networks and Global Reputation," in *Proc. Fourth CONFERENCE ON EMAIL AND ANTI-SPAM, CEAS2007*, Aug,2007.

[8] P. Kolan and R. Dantu, "Socio-Technical Defense Against Voice Spamming," *ACM Trans. Auton. Adapt. Syst.*, vol. 2, no. 1, Mar,2007.

[9] M. A. Azad and R. Morla, "Caller-Rep: Detecting unwanted calls with caller social strength," *Computers & Security*, vol. 39, Part B, pp. 219–236, Nov,2013.

[10] R. Jabeur Ben Chikha, T. Abbes, W. Ben Chikha, and A. Bouhoula, "Behavior-based approach to detect spam over ip telephony attacks," *International Journal of Information Security*, vol. 15, no. 2, pp. 131–143, Apr 2016.

[11] J. Quittek, S. Niccolini, S. Tartarelli, M. Stiemerling, M. Brunner, and T. Ewald, "Detecting SPIT Calls by Checking Human Communication Patterns," in *Proc. IEEE International Conference on Communications, ICC,Scotland*, Jun,2007, pp. 1979–1984.

[12] M. Azad and R. Morla, "Multistage SPIT Detection in Transit VoIP," in *Proc. 19th International Conference on Software, Telecommunications and Computer Networks IEEE SoftCOM*, Sep,2011, pp. 1–9.

[13] M. Burkhart, M. Strasser, D. Many, and X. Dimitropoulos, "Sepia: Privacy-preserving aggregation of multi-domain network events and statistics," in *Proc. 19th USENIX Conference on Security*, 2010, pp. 15–32.

[14] Y. Chen, K. Hwang, and W.-S. Ku., "Collaborative detection of ddos attacks over multiple network domains," *IEEE Trans. Parallel Distrib. Syst.*, vol. 18, no. 12, pp. 1649–1662, Dec. 2007.

[15] T. van de Kamp, A. Peter, M. H. Everts, and W. Jonker, "Private sharing of IOCs and sightings," in *Proc. of the 2016 ACM on Workshop on Information Sharing and Collaborative Security*, ser. WISCS '16. ACM, 2016, pp. 35–38.

[16] J. Freudiger, E. Cristofaro, and A. E. Brito, "Controlled data sharing for collaborative predictive blacklisting," in *Proc. of the 12th DIMVA*. Springer-Verlag New York, Inc., Jul,2015, pp. 327–349.

[17] M. Sirivianos, K. Kim, and X. Yang, "Social filter: Introducing social trust to collaborative spam mitigation," in *Proc. 2011 IEEE INFOCOM*, Apr,2011, pp. 2300–2308.

[18] Distributed checksum clearinghouses. [Online]. Available: http://www.rhyolite.com/dcc/

[19] G. Singaraju and B. B. Kang, "Repuscore: Collaborative reputation management framework for email infrastructure," in *Proc. 21st conference on Large Installation System Administration*, Nov 2008, pp. 1–9.

[20] J. Kong, B. Rezaei, N. Sarshar, V. Roychowdhury, and P. Boykin, "Collaborative spam filtering using e-mail networks," *Computer*, vol. 39, no. 8, pp. 67–73, 2006.

[21] The spamhaus project. [Online]. Available: www.spamhaus.org/

[22] J. K. Baker, Dixie B. and S. F. Terry, "Governance through privacy, fairness, and respect for individuals," *eGEMs*, vol. 4, 2016.

[23] B. Applebaum, H. Ringberg, M. J. Freedman, M. Caesar, and J. Rexford, "Collaborative, privacy-preserving data aggregation at scale," in *Proc. 10th Privacy Enhancing Technologies Symposium, PETS*. Springer-Verlag, July 2010, pp. 56–74.

[24] E. Zhai, D. I. Wolinsky, R. Chen, E. Syta, C. Teng, and B. Ford, "Anonrep: Towards tracking-resistant anonymous reputation," in *Proc. 13th USENIX Symposium on Networked Systems Design and Implementation, Usenix NSDI*, Mar,2016, pp. 583–596.

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

[26] Y.-S. Wu, S. Bagchi, N. Singh, and R. Wita, "Spam Detection in Voice- Over-IP Calls through Semi-Supervised Clustering," in *Proc.39th Annual Conference on Dependable Systems and Networks IEEE/IFIP DSN*, Jun 2009, pp. 307–316.

[27] H. Bokharaei, A. Sahraei, Y. Ganjali, R. Keralapura, and A. Nucci, "You can SPIT, but You can't hide: Spammer Identification in Telephony Networks," in *Proc. 2011 IEEE INFOCOM*, Apr 2011, pp. 41–45.

[28] R. Zhang and A. Gurtov, "Collaborative reputation-based voice spam filtering," in *Proc. International Conference on Database and Expert Systems Applications DEXA '09.*, Aug 2009, pp. 33–37.

[29] K. Toyoda, M. Park, N. Okazaki, and T. Ohtsuki, "Novel unsupervised spitters detection scheme by automatically solving unbalanced situation," *IEEE Access*, vol. 5, pp. 6746–6756, 2017.

[30] P. Gupta, B. V. Srinivasan, B, and M. Ahamad, "Phoneypot: Data-driven Understanding of Telephony Threats," in *Proc. 20th Network and Distributed System Security Symposium NDSS*, Feb 2015.

[31] M. Balduzzi, P. Gupta, L. Gu, Gao.D, and M. Ahamad, "MobiPot: Understanding Mobile Telephony Threats with Honeycards," in *Proc. 11th ACM Asia Conference on Computer and Communications Security ACM ASIACCS*, Jun 2016.

[32] B. Mathieu, S. Niccolini, and D. Sisalem, "SDRS: A Voice-over-IP Spam Detection and Reaction System," *IEEE Security and Privacy*, vol. 6, pp. 52–59, 2008.

[33] C. Sorge and J. Seedorf, "A Provider-Level Reputation System for Assessing the Quality of SPIT Mitigation Algorithms," in *Proc. International Conference on Communications IEEE ICC*, Jun 2009, pp. 1 –6.

[34] Y.-S. Wu, V. Apte, S. Bagchi, S. Garg, and N. Singh, "Intrusion detection in voice over IP Environments," *Int. J. Inf. Secur.*, vol. 8, no. 3, pp. 153–172, 2009.

[35] A. Gazdar, Z. Langar, and A. Belghith, "A distributed cooperative detection scheme for SPIT attacks in SIP based systems," in *Proc. Third Network of the Future (NOF)*, Nov 2012, pp. 1–5.

[36] E. Damiani, S. De Capitani di Vimercati, S. Paraboschi, and P. Samarati, "P2p-based collaborative spam detection and filtering," in *Proc.Fourth International Conference on P2P Computing*, Aug 2004, pp. 176–183.

[37] K. Li, Z. Zhong, and L. Ramaswamy, "Privacy-aware collaborative spam filtering," *IEEE Trans. Parallel Distrib. Syst.*, vol. 20, no. 5, pp. 725–739, 2009.

[38] J. Camenisch, A. Lysyanskaya, and M. Meyerovich, "Endorsed e-cash," in *Proc. IEEE Symposium on Security and Privacy, SP '07*, May 2007, pp. 101–115.

[39] E. Androulaki, S. G. Choi, S. M. Bellovin, and T. Malkin, "Reputation systems for anonymous networks," in *Proc. 8th International Symposium Privacy Enhancing Technologies, PETS 2008*, Jul 2008.

[40] S. Schiffner, S. Clauß, and S. Steinbrecher, "Privacy and liveliness for reputation systems," in *Proc. 6th European Workshop Public Key Infrastructures, Services and Applications, EuroPKI 2009*, Sep 2009, pp. 209–224.

[41] E. B. Sasson, A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, and M. Virza, "Zerocash: Decentralized anonymous payments from bitcoin," in *Proc. 2014 IEEE Symposium on Security and Privacy*, May 2014, pp. 459–474.

[42] B. Huang, M. T. Kechadi, and B. Buckley, "Customer churn prediction in telecommunications," *Expert Syst. Appl.*, vol. 39, no. 1, pp. 1414–1425, Jan. 2012.

[43] H. Fingera, T. Genoleta, L. Marib, G. Constantin de Magnyc, and N. M. Mangad, "Mobile phone data highlights the role of mass gatherings in the spreading of cholera outbreaks," *Proc. of the National Academy of Sciences of the United States of America PNAS.*, vol. 113, Jun. 2016.

[44] 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, June 2010.

[45] A. A. Nanavati, S. Gurumurthy, G. Das, D. Chakraborty, K. Dasgupta, S. Mukherjea, and A. Joshi, "On the Structural Properties of Massive Telecom Call Graphs: Findings and Implications," in *Proc. ACM international conference on Information and knowledge management CIKM*, Nov 2005, pp. 435–444.

[46] W. Henecka and M. Roughan, "Privacy-preserving fraud detection across multiple phone record databases," *IEEE Transactions on Dependable and Secure Computing*, vol. 12, no. 6, pp. 640–651, Nov 2015.

[47] R. Sarmento, M. Oliveira, M. Cordeiro, S. Tabassum, and J. Gama, "Social network analysis in streaming call graphs," in *Big Data Analysis: New Algorithms for a New Society*. Springer, 2016, pp. 239–261.

[48] S. Tabassum and J. Gama, "Sampling massive streaming call graphs," in *Proc. 31st Annual ACM Symposium on Applied Computing*. ACM, 2016, pp. 923–928.

[49] S. Chiappetta, C. Mazzariello, R. Presta, and S. Romano, "An anomaly-based approach to the analysis of the social behavior of voip users," *Computer Networks*, vol. 57, no. 6, pp. 1545 – 1559, 2013.

[50] N. d'Heureuse, S. Tartarelli, and S. Niccolini, "Analyzing Telemarketer Behavior in Massive Telecom Data Records," in *Springer Trustworthy Internet*, 2011, pp. 261–271.

[51] Bouncy castle. [Online]. Available: http://www.bouncycastle.org/.