Decentralized Self-enforcing Trust Management System for Social Internet of Things

Muhammad Ajmal Azad, Samiran Bag, Feng Hao *Senior Member, IEEE* and Andrii Shalaginov, *Member, IEEE*

Abstract—The Internet of Things, or IoT, is the network of connected computing devices that have the ability to transfer valued data between each other via the Internet without requiring human intervention. In such a connected environment, the Social Internet of Things (SIoT) has become an emerging trend where multiple IoT devices owned by users support communication within a social circle. Trust management in the SIoT network is imperative as trusting the information from compromised devices could lead to serious compromises within the network. It is important to have a mechanism where the devices and their users evaluate the trustworthiness of other devices and users before trusting the information sent by them. The privacy-preservation, decentralization and self-enforcing management without involving trusted third parties are the fundamental challenges in designing a trust management system for SIoT. To fulfill these challenges, this paper presents a novel framework for computing and updating the trustworthiness of participants in the SIoT network in a self-enforcing manner without relying on any trusted third party. The privacy of the participants in the SIoT is protected by using homomorphic encryption in the decentralized setting. To achieve the properties of self-enforcement, the trust score of each device is automatically updated based on its previous trust score and the up-to-date tally of the votes by its peers in the network with zero-knowledge proofs to enforce that every participant follows the protocol honestly. We evaluate the performance of the proposed scheme and present evaluation benchmarks by prototyping the main functionality of the system. The performance results show that the system has a linear increase in computation and communication overheads with more participants in the network. Furthermore, we prove the correctness, privacy, and security of the proposed system under a malicious adversarial model.

Index Terms—self-enforcing trust aggregation, Secure Multiparty Computation, Privacy-preserving aggregation, Social Internet of Things

1 Introduction

The Internet of Things (IoT) is a network of connected smart devices, sensors, actuators, and people that use the Internet for transferring valued information. These smart devices produce a massive amount of data that can be used for meaningful analytics. The number of smart and connected IoT devices has dramatically increased over the last few years. It has been predicted that there would be around more than 29 billion connected devices by the year 2022, of which around 65% (18 billion) will be related to smart IoT [1] devices. The boom in the IoT business has also increased business and financial opportunities. It has been predicted that by 2025, the IoT business could have an annual economic forecast of \$3.9 trillion to \$11.1 trillion worldwide. The success of IoT will critically depend on the security and trust of these devices.

A standard IoT system is similar to the traditional peer-to-peer (P2P) network where there exist two types of parties, one using services, and the other providing services. It is important for the service requester to evaluate the trustworthiness of the IoT devices and their users before acting on the information provided by them. The trust evaluation of the IoT devices and users is important because there exist malicious entities in the network who want to misuse the network resources for malicious purposes such as spreading malware or false information. Furthermore, it is important to identify misbehaving IoT objects before they bring damage to the overall IoT-based smart system. The aggregated trust of devices and their owners in the IoT system could help service requester to make meaningful decisions before acting on the provided information. Trust management systems can provide a way to evaluate the trustworthiness of devices and identify malicious actors. Trust evaluation systems have been widely used in many domains for various purposes: e.g., they have been deployed in the vehicular and ad-hoc wireless network to evaluate the trustworthiness of vehicles and objects [1], [2], utilized in the P2P (peer to peer) networks [3], [4] to assess the behavior of peers before downloading the content, used in online marketplaces to provide opinions to users how retailers are behaving in their past transactions [5]–[7], and used in value-added communication networks (such as email, telephony, social networks) [8] for the identification of untrusted parties by leveraging collaboration among users of the network.

The trust management approaches for the social Internet of Things mainly involve users of the system to provide their views about the behavior of others. In these settings, the data or feedback reported by each user may have some private, sensitive information, hence require full protection from disclosure. Therefore, it is important to consider user privacy while designing a trust management system to

- *Muhammad Ajmal Azad is with the School of Computer Science, University of Derby, UK. Samiran Bag and Feng Hao are with the Department of Computer Science at the University of Warwick, United Kingdom. Andrii Shalaginov is with the Faculty of Information Technology and Electrical Engineering, Norwegian University of Science and Technology, Norway. E-mail:m.azad@derby.ac.uk, {samiran.bag, feng.hao}@warwick.ac.uk, andrii.shalaginov@ntnu.no*

[1] Internet of Things forecast https://www.ericsson.com/en/mobility-report/internet-of-things-forecast

protect the private information of individual users. Existing trust management systems preserve privacy of users by utilizing three major methods: a) anonymization [9]–[11] which replaces the true identity of the participant with a randomized identity before reporting the data for analysis; b) data perturbation [12], [13], which perturbs data with added noise to minimize the information leakage to intruders; and c) cryptography [14]–[20] that computes analytics of the shared data using secure multiparty computation techniques. Although the anonymization systems can ensure the privacy of users, however, these approaches can still be bypassed through de-anonymization methods. The privacy can also be protected through the use of a trusted third-party system [21]–[23], however, in such settings the users have to trust the central system for protecting private information. Furthermore, many of the existing systems ensure privacy and security for the users data only for participants who provide honest feedback but try to infer private information of others (i.e., honest-but-curious) [3], [17], [24], [25]. These systems could be easily misused in practice by malicious actors providing fake out-of-range trust scores.

The trust management system for the decentralized social IoT should ensure the following: 1) the system design should ensure that neither the sensitive information of participants nor their private communication network is exposed to other users of the system; 2) it should not use trusted third party systems for holding user feedback; 3) it should be suitable for resource-constrained devices to maintain small communication bandwidth and computation overheads, and 4) the computation of the final scores should be verifiable by users of the system, 5) the system should be self-enforcing in the sense that it updates the trust score by itself in a verifiable manner without involving any trust third party, and finally, 6) the system should also consider interaction between the IoT devices and social network of people who own those devices.

To address these challenges, in this paper, we describe a new trust evaluation system for the Social Internet of Things (SIoT), which enables participants of the IoT ecosystem to evaluate the social and observable behavior of IoT objects before having any interaction with those objects. The computation of the trust score of IoT devices and users is based on the crowd-sourced information contributed by users of the system in a collaborative way. The system has inherent properties of self-enforcing updates of the trust scores and public verification of the scores contributed by the participants of the system. The aggregated trust score of the IoT object and user is computed in a secure, private and decentralized manner without revealing any individual feedback value used in the computation process.

Our trust management system is self-enforcing, which means that the participants of the system execute the steps of the protocol to compute the trust scores of everyone else in the system in a publicly verifiable manner without involving any trusted third party. The use of zero-knowledge proofs effectively enforces every participant to honestly follow the protocol specification. If anyone misbehaves in the system, the misbehavior will be publicly evident and their trust scores will degrade gradually until an extreme point such that misbehaving parties will be automatically expelled from the network.

In the proposed system, the participants first rank their interacted users and IoT devices based on their past behavior and then report the encrypted feedback values to a public bulletin board, which is readable to all but writable only to authenticated entities. The encrypted trust scores of the objects are then aggregated in a secure multi-party computation setting without letting participants infer the private feedback values of other individuals. This design does not place any trusted system or trusted set up for protecting the private information of individuals. In this paper, we consider the malicious adversarial model, in which malicious entities are not willing to follow the protocol specification and try to provide fake out-of-range feedback scores in order to affect operations of the system. The design also allows verification of trust scores reported by all participants. We prototype the main functionality of the protocol and assess its performance and provide security analysis. The results show that the system has a small overhead in both the computation load and communication bandwidth.

This paper makes the following contributions:

- A new privacy-preserving decentralized system is presented for assessing the trustworthiness of IoT nodes and their users in a Social IoT ecosystem. The system considers the important properties of using the social network and the users interaction behavior with the IoT objects. The proposed system utilizes the semantics of homomorphic cryptographic techniques with efficient zero-knowledge proof methods for protecting the privacy of the user data.
- The proposed system has an important novel feature: namely, the self-enforcing update of the trust score with public verifiability without involving any trusted third party.
- The system ensures privacy and security of the participant's private information under the malicious and honest-but-curious adversary models by using efficient zero-knowledge proof.
- A performance benchmarks are provided by implementing the prototype for the main functions of the system.

The rest of the paper is organized as follows. In Section 2, we review the state of the art in SIoT and other domains specifically focusing on privacy-preservation. In Section 3, we present a discussion on the preliminaries used in the design of the proposed system. Section 4 defines the problem. Section 5 presents the system architecture and discusses important features of the system. Section 6 describes protocol operations and the aggregation process. Section 7 provides a discussion on the privacy and security of the proposed system. Section 8 analyses the complexity of the system. Section 9 empirically evaluates the computation and communication performance of the system. Section 10 concludes the paper.

2 Related Work

This section reviews existing works on trust management in social IoT, intelligent transportation systems and P2P networks.

2.1 Trust Management in Social IoT

Over recent years, trust management in the social IoT ecosystem has received great attention. [26]–[28]. In an IoT environment the trust of devices changes with respect to environment, circumstances, and scenarios. Chen et al. [29] identified a number of environment and social features and suggested a model for adaptive trust estimation in the social IoT system. However, the system has not given importance to the privacy of users and furthermore, the trust scores of users could reveal the communication and movement patterns of users in the network. Nitti et al. [30] proposed schemes for identifying unreliable nodes by accumulating the trust feedback contributed by other users. In these schemes, the direct experience of the IoT nodes could be exposed to other nodes, hence not providing privacy-preservation. Further, it also reveals the communication network of IoT nodes. Chen et al. [31] proposed IoT-Trust, a system for computing the trust of IoT devices by using the semantics of the Software-Defined Network (SDN) and cross-layer communication protocol. However, the solution does not protect the privacy of the IoT devices and their users. Hui et al. [32], [33] proposed a context-aware framework for the computation of the trustworthiness of IoT nodes in the SIoT. The system considers the concepts from social and physiological science for computing the trust between IoT objects and their owners. Anuoluwapo et al. [34] proposed a collaborative approach that reliably estimates the trust between objects in the IoT network. However, the privacy aspects are not considered in this work. In [35] the privacy of the user is protected by using homomorphic encryption techniques that enable a user to provide trust scores in the encrypted form and only the aggregate result is decrypted. Chen et al. [36] computed trust and reputation of IoT objects using collaborative filtering methods. The approach uses similarity measure, social contract and community of interest while computing the aggregated trust. The privacy of nodes is not considered in the design.

Several trust evaluation systems have been proposed to ensure that the privacy of IoT nodes remains preserved during the computation process. A privacy-preserving reputation system is proposed by Yan et al. [17] which uses an additive homomorphic encryption system and an additive Pallier-cryptosystem for the preservation of trust values of IoT nodes. However, the privacy properties are only achieved in an honest-but-curious model, in which nodes correctly follow the protocol steps and always provide honest feedback within the prescribed range but meanwhile trying to learn private information of others. The homomorphic-encryption based scheme achieves optimized computation, whereas the Paillier-based cryptosystem achieves high security but is not computationally efficient. Prem et al. [37] proposed a private data sharing scheme in the IoT network to protect privacy. The scheme uses cloud storage to ensure privacy of private data of users and performs data analytics in a secure way using a homomorphic encryption scheme. For this purpose, the data points from the IoT devices are randomly distributed among the cloud data holders and are then aggregated in a secure way. The scheme provides privacy-preservation and correct computation only for the semi-honest nodes. Jeonggil et al. [38] proposed the MEDiSN framework that accumulates the data from the sensors in the wireless sensor network. The framework specifically uses a centralized data store for the collection and aggregation. The end-users need to trust the centralized data store for privacy-preservation and security.

Recently blockchain technology has also been used to assure the privacy of users in an IoT network [39]. Chen et al. [40] designed a blockchain-based model to protect the privacy of participants in the big data environment. The system is more generally designed for protecting raw data but in our case, we protect the users data while still performing some meaningful analytics over the encrypted data without actually decrypting it. Gan et al. [41] proposed a privacy-preservation model for task allocation in a crowd-sourced environment. The privacy of IoT nodes which allocate jobs to others is protected by the means of task division and hiding social network of IoT nodes. Fortino et al. [42] designed a blockchain-based model to distribute the reputation score among nodes in a distributed IoT network. The proposed approach first computes the reputation of each node in the network and then develops the collaborative network among nodes for the network-wide view about the trustworthiness of nodes in the network as a whole. Tang et al. [43] proposed a protocol named IoT Passport that enables IoT devices from a different platform to collaborate with each other using the blockchain system. In this setup, the interaction between devices is signed with a digital signature and recorded in the temper-proof blockchain. A three-player game model is proposed in [44] that protects private information and friendship network of devices and users in the context of the connected social Internet of Things.

2.2 Trust Management in Transportation Systems

Several trust models have been proposed for evaluating the trustworthiness of vehicles in the Internet of vehicles network [45], [46]. Tong et al. [47] proposed a three-layered distributed model to identify malicious vehicles in the vehicular network. Azad et al. [48] proposed TrustVote that aggregates the trust scores of vehicles in a decentralized way while preserving the privacy of users. The system utilizes a homomorphic system for the privacy-preservation, however, the system does not provide self-enforcement and self-correctness in the process of updating the scores. Guleng et al. [49] compute the trustworthiness of nodes on the Internet of Vehicle network by applying the fuzzy logic theory to the trust scores of vehicles. Yang et al. [50] used blockchain technology to compute the trustworthiness of vehicles by first validating the information provided by the participants. Riahi Sfar et al. [51] presented a context-aware system for the intelligent transportation system that not only assures privacy of users but also considers the environment attributes while evaluating the behavior of the nodes.

2.3 Trust Management in P2P Networks

Several homomorphic systems have been proposed within the context of crowdsourcing and P2P networks. Dongxiao et al. [23] proposed a reputation system that hides the identity of consumers using anonymous identities to ensure the confidentiality and integrity of reviews submitted by the consumers. The system is specifically based on the blockchain technology with efficient proof-of-stake. Chenglin et al. [52] proposed a PPTD (privacy-preserving truth discovery) framework which computes functions over the sensor data reported by users using a homomorphic cryptosystem. Kajino et al. [53] proposed a homomorphic encryption-based scheme to preserve privacy of users in the crowdsourced application. In [54] the aggregated trust of the nodes is computed in the decentralized setting by using an additive cryptographic system. The system ensures privacy of participants but it requires participants to adhere to the protocol operations. In [3], [55] the trust scores are aggregated in the decentralized way which ensures privacy and correctness in the presence of disruptive malicious intruders and semi-honest adversaries. However, the protocol requires a set of trusted users whom the participants need to trust for holding and processing of private information. In [22], the authors proposed a decentralized system for the privacy-preserving exchange of information related to the identities of the users. The system specifically ensures privacy and correctness through the use of a blockchain system and zero-knowledge proof techniques. In [56] the authors proposed a publicly verifiable aggregation scheme that ensures privacy of users by deploying a trusted third party system. However, the system only considers semi-honest users.

2.4 Final Remarks

To the best of our knowledge, the work presented in this paper is the first attempt to assess the trustworthiness of users and their IoT nodes within Social IoT ecosystems while ensuring the following major properties: privacy-preservation, self-enforcing computation of trust scores and public verification of aggregate trust scores. The proposed system achieves privacy-preservation under both malicious and semi-honest threat models. By contrast, existing systems preserve privacy under a semi-honest model and do not provide public verification and self-enforcement. The proposed scheme has a small computation and communication bandwidth overhead which makes the scheme suitable for the resource-constrained IoT network. The proposed system executes all its steps in a decentralized manner and fully ensures confidentiality of information contributed by the participants.

3 Preliminaries

In this section, we provide background on the social internet of things (SIoT) and the cryptographic tools used in the design of the privacy-preserving trust management system.

3.1 Social Internet of Things

In this paper, we evaluate the trustworthiness of IoT devices for the setup shown in Figure 1. We define two types of parties in our SIoT setup: users and devices that are owned by the users. We define IoT nodes in the network as $D = D_1, D_2, \ldots, D_n$, and the users who own these devices as $U = U_1, U_2, \ldots, U_n$. Suppose IoT devices and users communicate with each other using communication technologies like ZigBee, Bluetooth or GSM. A single user may have many IoT devices that perform particular tasks, e.g., monitoring road conditions, temperature, etc. These devices can be any handheld device: for example smartwatch, mobile phone, laptop, connected smart vehicle, etc. The communication between IoT users and devices can be represented as a graph like structure where nodes are the end-users or devices and edges are the communication links between them. The graph network of IoT devices and users is represented as $G = \{U, E\}$, where $E \in \{U \times U\}$ is a set of edges, each denoting the relationship between users/devices in the SIoT. Figure 1 provides a simple example of a generic connected network where $D = D_1, D_2, \ldots, D_n$, is the set of nodes capable of providing services to other IoT devices, and $U = U_1, U_2, \ldots, U_n$ represents a set of users who own these devices. Furthermore, devices provide services to other devices as well, for example, D_1 of user U_1 provides service to D_2 of user U_2. We assume each device in the network provides a particular service to other devices and users. For example, a smart vehicle can assist other road users by providing value-added information. The user in this setting would also provide feedback about their interaction so that other nodes become aware of the past behavior of certain nodes. This can help other users of the system to know the trustworthiness of devices or users before making any decision about the information provided from them. The aim of this paper is to evaluate the trustworthiness of users and devices simultaneously in a privacy-preserving and decentralized way.

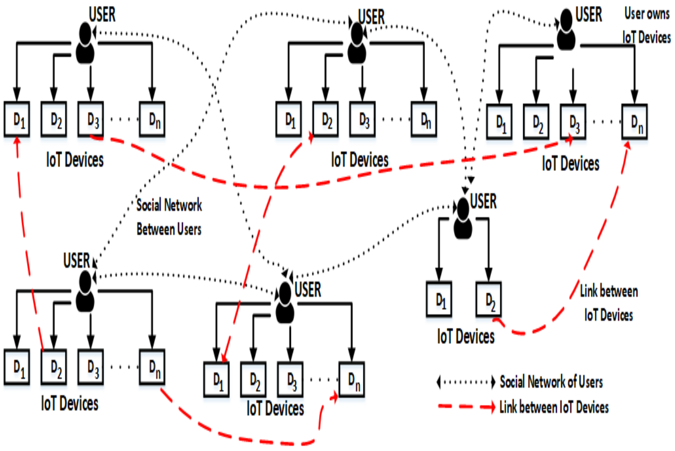


Fig. 1: The Social Internet of Things System.

Each IoT device and each user in the network has a unique identity that is being used for evaluating the trustworthiness of the users and IoT devices in the network. Inspired from the social relationships between people, there can be defined as a general set of relationships between IoT objects in SIoT such as friendship, colleagues, parental, social or co-ownership [57], [58]. Social relationships between owners of the IoT objects can influence the social relationships between IoT devices in the sense that two IoT devices from different users develop a social relationship if devices communicate with others. However, malicious attacks against the IoT ecosystem can have a considerable impact on the trustworthiness in both scenarios of SIoT:

human and objects-wise. Here our objective is to evaluate the trustworthiness in two aspects: *trustworthiness of IoT device* and *trustworthiness of the owner*. The former would provide a clear picture about the behavior of a specific IoT device which also characterizes the behavior of the owner for the particular service, i.e., a personalized trust of the owner, and the latter presents how the owner behaves as an aggregate for his provided services using all owned devices. In our trust model, we assume that at the end of the transaction, the user assigns a trustworthiness score to the IoT device which is then combined in a decentralized way for computing the trustworthiness in an aggregate manner.

3.2 Secure Multiparty Computation

Secure Multiparty Computation (SMC) enables parties to securely compute mathematical functions without inferring the values of the input data points provided by the participating parties. SMC methods have been used in several domains: for example, online e-voting [59], [60], privacy-preserving statistical analysis and data aggregation [18], [61], [61], [62], and privacy-preserving aggregation of user feedback [63] in online social networks. In SMC settings, the system has several collaborating parties, say $p_1, p_2, \ldots, p_n$, each with a private input $x_1, x_2, \ldots, x_n$. The involved parties perform mathematical operations (e.g., addition, mean, median, etc.) on the encrypted data points provided by the participating parties say $f(x_1, x_2, \ldots, x_n)$ without being to infer the values of other individuals' inputs. To ensure privacy, SMC techniques operate over the encrypted data points and the result of the computation is the same as the operations performed over the corresponding non-encrypted values, i.e., $Encryption(a) * Encryption(b) = Encryption(a \oplus b)$.

The SMC system consists of the following steps: generation of public and private keys, encrypting the user's submitted values using the generated keys, performing mathematical computation over the corresponding encrypted text or data, and finally revealing the final results. In this paper, an additive homomorphic cryptosystem is adopted for accumulating the data provided by users without using any trusted third party.

3.3 Homomorphic Encryption Method

The Homomorphic cryptographic primitives that are used in the design of self-enforcing trust management are based on the homomorphic encryption method proposed for the self-enforcing verifiable electronic voting [60]. The system computes the final result without relying on any trusted tallying authorities. Let $U = \{1, 2, \ldots, N\}$ be the set of users in the IoT ecosystem having the private trust ratings (0,1) for their interacted IoT objects and owners of the IoT devices. We assume a multiplicative cyclic group of $\mathbb{Z}_q^*$, e.g., the same as used in the Digital Signature Algorithm (we can also use the additive cyclic group as used in the Elliptic Curve Digital Signature Algorithm, but the protocol works basically the same). q is a large prime. Let there be another large prime p such that $p \mid q-1$. Further, let there be a subgroup G and g is the generator of G. G is a subgroup of $\mathbb{Z}_q^*$ with prime order p. All modular operations in G are performed with respect to the modulus q. However, we omit $\bmod p$ for simplicity. In order to participate in the secure feedback aggregation, the owner or user of the IoT object first generates a private key, i.e., $Sk_i \in \mathbb{Z}_p$ for $i \in N$. The user also generates the corresponding public key Pk_i and publishes it at the public bulletin board (PBB). The Pk_i is computed as follows.

$$Pk_i = g^{Sk_i}$$

When all participating users have published their respective Pk_i at the PBB, the user generates a key used for encrypting the direct feedback values. The encryption key (restructured key) is generated 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

$$\prod_{i \in N} Y_i^{Sk_i} = 1 \tag{1}$$

Equation 1 ensures that $Y_i^{Sk_i}$ can be used as a randomizer for computing the secret feedback. Any participant of the system or anyone having access to PBB can compute the final score of the IoT device or user using the encrypted values from the published cryptograms. In our design, we do not require a trusted third party system for the decryption of the final results.

3.4 Non-Interactive Zero-Knowledge Proof

A zero-knowledge proof (ZKP) is the system that consists of two entities, a prover, and a verifier. The verifier can establish the truthfulness of the information provided by the prover (participants in our case) without learning any information other than the truth of the statement. The use of ZKP ensures that the malicious Prover could not convince the Verifier of false (out-of-range) feedback values, and a malicious verifier would not infer anything from the values other than that correctness that the values fall within the prescribed range. Non-interactive zero-knowledge proofs (NIZK) is the class of ZKPs that do not require the explicit interaction between the verifier and the prover. In our trust management system, we adopted proof of knowledge to establish two statements: firstly, the secret parameters chosen by participants are truly generated, and secondly, the encrypted feedbacks are correctly constructed i.e., are within the prescribed range of value. In this work, Fiat-Shamir heuristics are used to transform interactive zero-knowledge proofs as non-interactive proof [64] (also see [65]).

3.5 Public Bulletin Board

The public bullet board (PBB) is a publicly readable append-only database for publishing crypto parameters for the participants. It is commonly used in verifiable e-voting systems [59], [65] to realize a public authenticated channel, so everyone can freely read the information published by authorized entities and verify it accordingly. Without the PBB, participants will need to send data directly to each other, which requires $O(n^2)$ channels. With the help of a PBB, participants only need to interact with a central bulletin board for publishing and receiving data. The PBB does not hold any secret information and does not need to be trusted since all operations are publicly verifiable. In our proposed scheme, the PBB holds the following information: the public keys of users participating in the collaborative process, the identity of the IoT object (which can be a unique hash or the IP address), and the encrypted trust feedback scores submitted by the users about the trustworthiness of others (IoT objects or users) based on their recent interactions. The published data also include NIZK proofs provided by the participants in order to prove that their submitted trust values are well-formed and are within the prescribed range. The PBB is available publicly and anyone can read data from the PBB and can compute the trust score of any object, however, only the authenticated participants who wish to participate in the collaborative process are allowed to write information on the PBB. The authenticity of the data can be checked through digital signatures. The malicious parties having access to PBB would not be able to infer the individual values of the feedback reported by the participating users, however, they can compute the trustworthiness of objects. In our settings, the PBB can be provided by the entity providing the reputation services. The PBB is not the central point of failure as it can be implemented in a distributed way, e.g., as a mirrored website or a blockchain system [66], [67].

4 Problem Definition

The challenge we are considering is the computation of the trustworthiness of users and IoT devices in the SIoT. Our main objective is to carry out trust analysis and computation in a decentralized privacy-preserving way. The performance and robustness of any IoT network depend upon their ability to assess how objects behave in a decentralized manner. A steadfast method of achieving this is to invite the users of the system to provide feedback vis-à-vis on a particular thing, event or object. This feedback is then agglomerated to compute the aggregate trust-score of an object in question.

However, two important requirements need to be kept in mind while performing this computation – 1) the feedback supplied by different users must remain confidential, and 2) the feedback of different users in the network should have different weights. These weights should depend upon the expected accuracy of the feedback measured on the basis of the performance of the same feedback provider in the recent past. In other words, the weight associated with a feedback provider should be continuously updated based on how accurate she has been in providing ratings in her last few attempts. This ensures that if a user has provided accurate feedback in most of her recent attempts, then higher weights should be assigned to her latest feedback than others who have a less impeccable track record. When a user joins a network, her weight should be initialized to a low value. If the feedback provided by a user in an iteration is in line with the overall reputation computed in that iteration, then her weight should be upgraded. On the other hand, if there is any disparity between the feedback provided by a user, and the overall reputation computed at any stage, then the weight of the concerned user should be downgraded. Such an actualization of the weights of the user should be done seamlessly. The former goal can be achieved by employing encryption techniques, whereas the latter goal can be achieved by making efficient use of Non-interactive Zero-Knowledge (NIZK) proofs. Here, we must keep in mind that the weight associated with a user should be private to the user herself and is not to be made public. Nonetheless, the NIZK proofs should be crafted in a way to establish that the update of the weights associated with a user has been done correctly, and thus the encrypted feedback provided by the user is within the permissible limits as specified by her secret weight at that point in time. Every user should dynamically update her weight after the end of the evaluation cycle and, when she provides her next feedback, she should provide NIZK proof of correct updating of her weight without revealing the weight itself.

In this paper, we provide a scheme that allows the users of the IoT system to provide feedback corresponding to an object or event in the IoT network. The feedback represents the weighted rating provided by a user. The feedback is encrypted using a public-key encryption scheme in a way that allows public computation of the aggregated feedback without reverting to any trusted third party for decryption. The aggregated feedback or the overall reputation is the weighted sum of all the ratings given by the users. Thus, the aggregate feedback represents the weighted average of all ratings provided by all the users. Each of the participants also provides NIZK proofs to prove that the feedback represents the correct encryption of her input with the correct weight assigned to her. The assigned weight is based on her previous weight and her accuracy in that iteration. The user dynamically updates her weights at the end of each iteration depending upon how accurate she has been in her opinion about the object/event in question. Once all the trust scores are published, anyone can compute the overall trust score of any IoT object in the system. The NIZK proofs also enable public verification of the correctness of the protocol.

5 Proposed System Design

In this section, we discuss the mechanism for trust aggregation. Before we get into the steps used in computing the aggregated trust, we will first discuss the system architecture, the assumptions used in this paper and provide the threat model.

5.1 System Components

The system setup of the proposed trust management system is presented in Figure 2. The system consists of three functional entities: users, IoT objects and a public bulletin board. *The users* receive information from the IoT objects and provide a rating based on the positive or negative response from the IoT objects. *The IoT objects* provide service to the users, for example, providing road conditions, etc. Users rate IoT objects at the scores of 0 to 1. The bulletin board holds the information reported by the users and is publicly readable. The PBB is responsible for two types of operations: 1) providing the facility to allow authenticated users to post the crypto parameters, the encrypted feedback, and NIZK proof of feedback scores 2) making data publicly available to everyone for computing aggregated trustworthiness of the objects. The functionality of PBB can be either distributed or

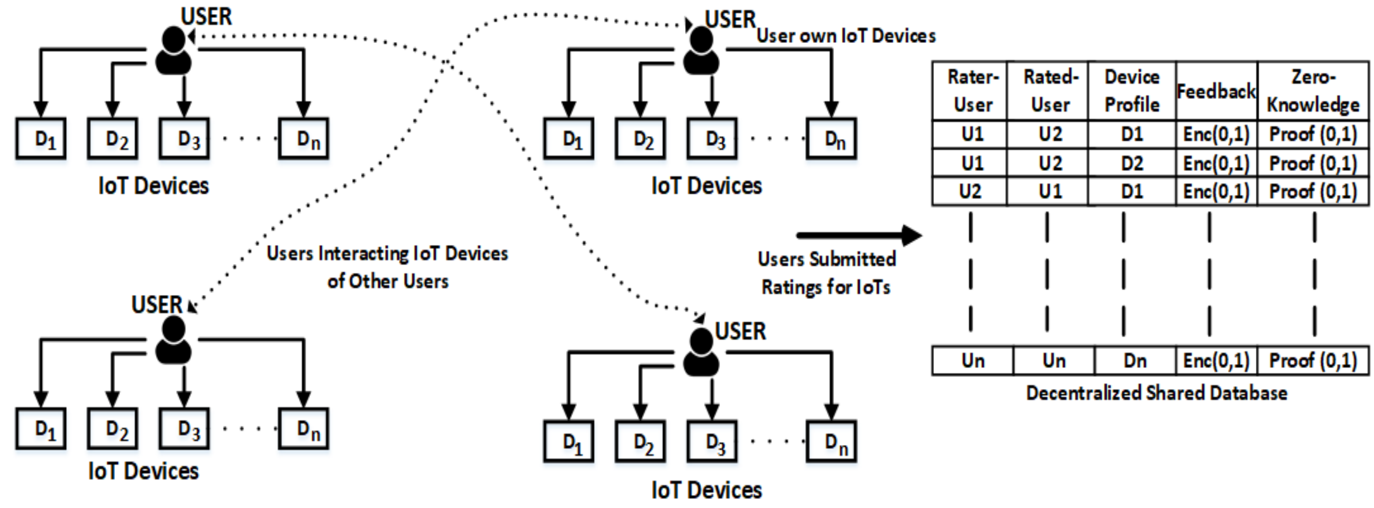

USER
User own IoT Devices
D1
D2
D3
Dn
IoT Devices
Users Interacting IoT Devices of Other Users
Users Submitted Ratings for IoTs
Rater-User
Rated-User
Device Profile
Feedback
Zero-Knowledge
U1 U2 D1 Enc(0,1) Proof (0,1)
U1 U2 D2 Enc(0,1) Proof (0,1)
U2 U1 D1 Enc(0,1) Proof (0,1)
Un Un Dn Enc(0,1) Proof (0,1)
Decentralized Shared Database

Fig. 2: The System Architecture of the Proposed System

centralized and is managed by any entity in the social IoT ecosystem.

5.2 Adversarial Model

The objective of a crowdsourced trust management system is to compute the trustworthiness of IoT objects from the data or feedback trust values provided by the participants. Another major goal of such a system is to carry out trust computation with the inherent property of privacy-preservation. The computation system broadly has two types of participants: an honest participant who always correctly follows the protocol specification, and a malicious participant who has motives of disrupting the functionality of the system. The honest and malicious participants also have a common motive: i.e., inferring the trust values of participants and their communication behavior.

In our privacy-preserving trust system, we consider the following adversarial models to ensure the correctness of computation and privacy-preservation. The first is a semi-honest model also known as an honest-but-curious adversarial model. In this model, the participants adhere to the protocol specification; however, they also attempt to interpret the meaning of information contributed by other participants and try to infer their communication network. The second model is a malicious model. Participants in this model not only try to interpret the contributed information but also try to disrupt the functionality of the protocol by not following the protocol specification. In our settings, the malicious participants may try to submit feedback values that are not within the prescribed range in order to artificially increase the trust of some specified users.

5.3 Assumptions of the Protocol

The participants of the system have to agree on the cyclic group G with p elements, p being a prime number. The cyclic group G is publicly known to everyone. We assume that the Decisional Diffie-Hellman (DDH) problem is computationally hard in the group G. Let g be a generator of G which is known publicly. All modular operations in G are performed with reference to a prime modulus q. We have omitted "$\mod q$" throughout the paper for simplicity. For the information holding and processing, we assume that there exists a public bulletin board which is readable to all, but writable only to authenticated participants. The authenticity of the data sent by the participants can be checked by using digital signatures. We assume an 'append-only' PBB: i.e., participants are not allowed to overwrite their already submitted data. In the setup phase of the trust management system, the public keys of authorized users are published on the PBB. Any subsequent posting by the same users can be ensured by verifying the digital signature against the initially committed public keys. The same append-only bulletin board is commonly used in e-voting, e.g., see [60], [65], [68]. A blockchain is essentially a public bulletin board with distributed data storage and computing power, and hence can be used in our system to realize the PBB. An alternative way to implement the PBB is to use a mirrored public website [65].

6 The Scheme

Notation	Description
G	Algebraic group
p	order of G
$*$	Group operation in G
g	Random generator of G
n	no. of users
U_i	i'th user
W	weight vector
s_{ij}	secret rating of U_i in iteration j
w_{ij}	weight of U_i in iteration j
τ_j	weighted sum of all ratings in iteration j
T_j	Overall reputation computed in iteration j
x_{ij}	Secret key of U_i in iteration j
X_{ij}	Public key of U_i in iteration j
Y_{ij}	Restructured key of U_i in iteration j

TABLE 1: Table of Notations

Initialization. We consider a network of n IoT users where the value of n is a variable. The value of n changes as new devices join or leave the network. There is a weight vector $W = (w_1, w_2, \ldots, w_n)$ associated with the network, where $w_i \in \{1, 2, 3, 4, 5\}$. When a user (device) U_i is added to the IoT network, her weight w_i is initialized to a low value such as 1, and the value of n is incremented by one.

The Goal. The network continuously computes the reputation of events or objects concerning the functioning of the IoT devices. In iteration j, the network computes the reputation of an object and depending upon the decision updates the weights of all the IoT devices participating in the process. In each iteration j the network computes the weighted sum of all ratings provided by the n devices as follows: $\tau_j = \sum_{i=1}^{n} w_{ij} s_{ij}$, where $s_{ij} \in \{-1, 1\}$ is the secret rating of U_i in iteration j and $w_{ij} \in \{1, 2, 3, 4, 5\}$ is the weight of U_i in iteration j. Then everyone computes:

$$T_j = \begin{cases} +1 : \text{ if } \tau_j > 0 \\ -1 : \text{ otherwise} \end{cases} \tag{2}$$

Hence the overall reputation of the subject is $+1$ if the weighted sum of the reputation scores of all participating users is positive, and is -1 otherwise. Note, that the value of τ_j is between $-\theta$ and $+\theta$, where θ is the sum of weights associated with the users, that is, $\theta = \sum_{i=1}^{n} w_{ij}$. Hence, we assign $+1$ value to the reputation of the subject if the value of τ_j falls on the higher side of its range.

Once T_j is computed, each U_i updates its weight according to the following rule:

$$w_{i(j+1)} = \begin{cases} w_{ij} : \text{ if } (w_{ij} = 1 \wedge T_j = -s_{ij}) \vee \\ (w_{ij} = 5 \wedge T_j = s_{ij}) \\ \\ w_{ij} + 1 : \text{ if } w_{ij} < 5 \wedge T_j = s_{ij} \\ \\ w_{ij} - 1 : \text{ if } w_{ij} > 1 \wedge T_j = -s_{ij} \end{cases} \tag{3}$$

In the above equation, the weight associated with a user is incremented by 1 if she had given a correct rating in the previous iteration inconsistent with the aggregate result. That is, the weight of a user is upgraded if her rating in the previous iteration was the same as the overall rating computed in that iteration. For example, if a user gives $+1$ rating to a subject in an iteration i, and the overall rating of that subject computed in iteration i is also $+1$, then the weight associated with the user will be upgraded by 1. Similarly, the weight associated with a user is decreased by 1 if the rating provided by her in an iteration is contrary to the overall rating computed in that iteration. We also emphasize that the weight associated with a user is between 1 and 5. That is the weight of a user cannot go below 1 or higher than 5. If the weight of a user is 5 in an iteration, and she does give the correct rating in that iteration, her weight will remain so in the next iteration. Here, 1 and 5 are just examples for the lower and higher bounds, but they can be other values as well.

The Protocol

In each iteration j, each user $U_i : i \in [1, n]$ needs to execute two rounds. These are explained below.

Round I (Key Generation). In this round each device $U_i : i \in [1, n]$ selects a random $x_{ij} \in \mathbb{Z}_p$ and posts on the bulletin board a public key $X_{ij} = g^{x_{ij}}$. U_i also posts a NIZK proof π_1 on the bulletin board. This NIZK proof proves that given X_{ij}, the user U_i has knowledge of $x_{ij} = \log_g X_{ij}$. In this paper, we use Schnorr signature protocol to form the NIZK [69].

Round II (Feedback Generation). In this round, every user $U_i : i \in [1, n]$ downloads all public keys X_{ij} uploaded by every user $U_k : k \in [1, n] \setminus \{k\}$, in the first round. Then user U_i computes the restructured key $Y_{ij} = \prod_{k=1}^{i-1} X_{kj} / \prod_{k=i+1}^{n} X_{kj} = g^{\sum_{k=1}^{i-1} x_{kj} - \prod_{k=i+1}^{n} x_{kj}}$. Thereafter, U_i computes a ballot $B_{ij} = Y_{ij}^{x_{ij}} g^{s_{ij} w_{ij}}$. U_i posts B_{ij} on the bulletin board. B_i also posts a NIZK proof Π_i on the bulletin board. B_{ij} proves the well-formedness of B_{ij} given X_{ij}, B_{ij} and the fact that $s_{ij} \in \{-1, 1\}$ and w_{ij} is computed from $w_{i(j-1)}$ in accordance with equation 3.

Computation of Reputation. Once every user has posted B_{ij} on the bulletin board, the users can compute $D_{ij} = \prod_{k=1}^{n} B_{ij} = \prod_{i=1}^{n} Y_{ij}^{x_{ij}} g^{s_{ij} w_{ij}} = \left(\prod_{i=1}^{n} Y_{ij}^{x_{ij}}\right) * g^{\sum_{i=1}^{n} s_{ij} w_{ij}}$. According to the proposition in Section 3.3, $\prod_{i=1}^{n} Y_{ij}^{x_{ij}} = 1$. Hence, $D_{ij} = g^{\sum_{i=1}^{n} s_{ij} w_{ij}} = g^{\tau_j}$. A brute force search on D_{ij} will yield the value of $\tau_j = \sum_{i=1}^{n} s_{ij} w_{ij}$. Brute force search will be feasible since the value τ_j is a small number within the range $[-n, n]$. Once τ_j is computed, T_j can be found using Equation 2. Similarly, the weight w_is can be updated using Equation 3.

7 Security and Privacy Analysis

In this section, we address how the proposed mechanism ensures privacy, confidentiality, integrity, and correctness of computation.

7.1 Defence against malicious adversary

The intention of a polynomial-time adversary is to disrupt the functioning of the trust management system by providing out-of-range inputs. Our trust management system makes use of non-interactive zero-knowledge proofs, so the malicious adversary will not be able to churn out feedback corresponding to out-of-range inputs unless she can break the security of the NIZK proof system which only occurs with a negligible probability.

7.2 Defence against honest-but-curious adversary

Now, we consider an honest-but-curious adversary who follows the protocol correctly but tries to gain information about the rating of one or more specific users' inputs. We recognize that all cryptograms generated by the users are well-formed, as ensured by the NIZK proofs. The adversary colludes with some users to find information about the secret ratings provided by non-colluding users. Our main result is Theorem 1. In order to prove Theorem 1, we need the help of Assumption 2. In order to prove Assumption 2, we need the Assumption 1. We show that the proposed approach only allows the adversary to infer the final trustworthiness of users and devices ($\sum_i w_{ij} \cdot s_{ij}$) which is the public information. The transcript of the protocol does not allow the adversary to deduce any valuable information that cannot be inferred from the final trustworthiness scores. Note that, if the adversary colludes with some of the users, she will know their input ratings and the associated weights. So, the adversary can subtract the weighted ratings of the colluding users from the overall weighted sum. In this case, the attacker can find out the value of the partial weighted sum of the ratings contributed by honest users. The proposed protocol under this setup would not allow the adversary to infer any private information other than the publicly available tallying information, and the inputs of the colluding compromised users. Hence, our scheme offers the best possible security protection even in the presence of colluding users.

Assumption 1. Given g, g^a, g^b and a challenge $\Omega \in \{g^{ab}, R\}$, it is difficult to compute whether value of Ω equals g^{ab} or R.

Assumption 2. Let us assume, $x = (g^{a_1}, g^{a_2}, \ldots, g^{a_{k-1}}), y = (g^{b_1}, g^{b_2}, \ldots, g^{b_{k-1}})$ and, $X = (g^{a_1 b_1}, g^{a_2 b_2}, \ldots, g^{a_{k-1} b_{k-1}})$. Also let us assume, $\mathcal{X} = \mathcal{X}_1 * \mathcal{X}_2 * \ldots * \mathcal{X}_k$ and $\mathcal{Y} = \mathcal{Y}_1 * \mathcal{Y}_2 * \ldots * \mathcal{Y}_k$. If $\mathcal{X} \overset{c}{\approx} \mathcal{Y}$, then $(g^{a_1 b_1} * \mathcal{X}_1, g^{a_2 b_2} * \mathcal{X}_2, \ldots, g^{a_{k-1} b_{k-1}} * \mathcal{X}_{k-1}, \frac{\mathcal{X}_k}{\prod_{i=1}^{k-1} g^{a_i b_i}}) \overset{c}{\approx} (g^{a_1 b_1} * \mathcal{Y}_1, g^{a_2 b_2} * \mathcal{Y}_2, \ldots, g^{a_{k-1} b_{k-1}} * \mathcal{Y}_{k-1}, \frac{\mathcal{Y}_k}{\prod_{i=1}^{k-1} g^{a_i b_i}})$.

Lemma 1. Assumption 1 implies assumption 2.

Proof 1. Let $A_i = g^{a_i b_i}$ for $i \in [1, k-1]$. $(A_1 * \mathcal{X}_1, A_2 * \mathcal{X}_2, \ldots, A_{k-1} * \mathcal{X}_{k-1}, \frac{\mathcal{X}_k}{\prod_{i=1}^{k-1} A_i}) = (A_1 * \mathcal{X}_1, A_2 * \mathcal{X}_2, \ldots, A_{k-1} * \mathcal{X}_{k-1}, \frac{\mathcal{X}}{\prod_{i=1}^{k-1} A_i * \mathcal{X}_i})$. Since, according to assumption 1; $A_i = g^{a_i b_i} \overset{c}{\approx} R, \forall i \in [1, k-1]$, $A_i * \mathcal{X}_i \overset{c}{\approx} A_i * \mathcal{Y}_i, \forall i \in [1, k-1]$. $(A_1 * \mathcal{X}_1, A_2 * \mathcal{X}_2, \ldots, A_{k-1} * \mathcal{X}_{k-1}, \frac{\mathcal{X}}{\prod_{i=1}^{k-1} A_i * \mathcal{X}_i}) \overset{c}{\approx} (A_1 * \mathcal{Y}_1, A_2 * \mathcal{Y}_2, \ldots, A_{k-1} * \mathcal{Y}_{k-1}, \frac{\mathcal{X}}{\prod_{i=1}^{k-1} A_i * \mathcal{Y}_i})$. Now, we claim that: $(A_1 * \mathcal{Y}_1, A_2 * \mathcal{Y}_2, \ldots, A_{k-1} * \mathcal{Y}_{k-1}, \frac{\mathcal{X}}{\prod_{i=1}^{k-1} A_i * \mathcal{Y}_i}) \overset{c}{\approx} (A_1 * \mathcal{Y}_1, A_2 * \mathcal{Y}_2, \ldots, A_{k-1} * \mathcal{Y}_{k-1}, \frac{\mathcal{Y}}{\prod_{i=1}^{k-1} A_i * \mathcal{Y}_i})$, otherwise anyone can distinguish between $\mathcal{X}$ and $\mathcal{Y}$ by choosing random A_is and random $\mathcal{X}_i$'s and thus computing a challenge $(A_1 * \mathcal{Y}_1, A_2 * \mathcal{Y}_2, \ldots, A_{k-1} * \mathcal{Y}_{k-1}, \frac{\mathcal{Q}}{\prod_{i=1}^{k-1} A_i * \mathcal{Y}_i})$, where $Q \in \{\mathcal{X}, \mathcal{Y}\}$. If the challenge $(A_1 * \mathcal{Y}_1, A_2 * \mathcal{Y}_2, \ldots, A_{k-1} * \mathcal{Y}_{k-1}, \frac{\mathcal{Q}}{\prod_{i=1}^{k-1} A_i * \mathcal{Y}_i})$ is correctly identified then so will be Q. Hence, $(A_1 * \mathcal{Y}_1, A_2 * \mathcal{Y}_2, \ldots, A_{k-1} * \mathcal{Y}_{k-1}, \frac{\mathcal{X}}{\prod_{i=1}^{k-1} A_i * \mathcal{Y}_i}) \overset{c}{\approx} (A_1 * \mathcal{Y}_1, A_2 * \mathcal{Y}_2, \ldots, A_{k-1} * \mathcal{Y}_{k-1}, \frac{\mathcal{Y}}{\prod_{i=1}^{k-1} A_i * \mathcal{Y}_i})$. Thus, the lemma holds.

Theorem 1. Let $\mathcal{A}$ be an adversary that has corrupted up to c users U_i with indices $i \in C$. As such, in an iteration j, $\mathcal{A}$ will learn nothing other than $\sum_{i \in [n] \setminus C} s_{ij} w_{ij}$.

Proof 2. We show that the adversary $\mathcal{A}$ will not be able to distinguish between two bulletin boards $\mathcal{B}$ and $\mathcal{B}'$ in which the honest users have different inputs as well as different weights if the weighted sums of inputs are the same in both the bulletin boards. Let us assume $D = [n] \setminus C$ and $D = \{h_1, h_2, \ldots, h_t\}$. Let $s_{h_k j}$ and $s'_{h_k j}$ be the secret inputs of each honest user U_{h_k} corresponding to bulletin board $\mathcal{B}$ and $\mathcal{B}'$ respectively. Similarly, let $w_{h_k j}$ and $w'_{h_k j}$ be the secret inputs of each honest user U_{h_k} corresponding to bulletin board $\mathcal{B}$ and $\mathcal{B}'$ respectively. We have $\sum_{k=1}^{t} s_{h_k j} w_{h_k j} = \sum_{k=1}^{t} s'_{h_k j} w'_{h_k j}$. Let, the public keys of U_{h_i} be given by $X_{h_i} = g^{x_{h_i}}$ for $i \in [t]$. Let $C = \{c_1, c_2, \ldots, c_{n-t}\}$. The public key of U_{c_i} is given by $X_{c_i} = g^{x_{c_i}}$ for $i \in [n-t]$. The secret input and the weight of each U_{c_i} is $s_{c_i j}$ and $w_{c_i j}$ respectively for both the bulletin boards. The ballots submitted by U_{c_i} is given by $B_{c_i j} = Y_{c_i j}^{x_{c_i j}} g^{s_{c_i j} w_{c_i j}} = g^{x_{c_i j} y_{c_i j}} g^{s_{c_i j} w_{c_i j}} : i \in [1, n-t]$. The ballot submitted by each honest user U_{h_i} to bulletin board $\mathcal{B}$ is given by $B_{h_i j} = Y_{h_i j}^{x_{h_i j} y_{h_i j}} g^{s_{h_i j} w_{h_i j}} = g^{x_{h_i j}} g^{s_{h_i j} w_{h_i j}} : i \in [1, t]$. Similarly, the ballot submitted by each honest user U_{h_i} to bulletin board $\mathcal{B}'$ is given by $B'_{h_i j} = Y_{h_i j}^{x_{h_i j} y_{h_i j}} g^{s'_{h_i j} w'_{h_i j}} = g^{x_{h_i j}} g^{s'_{h_i j} w'_{h_i j}} : i \in [1, t]$. Note that $\sum_{i=1}^{n-t} x_{c_i j} y_{c_i j} + \sum_{i=1}^{t} x_{h_i j} y_{h_i j} = 0$. So, $g^{x_{h_t j} y_{h_t j}} = \frac{1}{\prod_{i=1}^{n-t} g^{x_{c_i j} y_{c_i j}} \prod_{i=1}^{t} g^{x_{h_i j} y_{h_i j}}}$. Now, since the users in C are compromised, $\mathcal{A}$ can find $\prod_{i=1}^{t} g^{x_{h_i j} y_{h_i j}}$. So, given $B_{h_t j}$, the adversary can compute $\tilde{B}_{h_t j} = \frac{g^{s_{h_t j} w_{h_t j}}}{\prod_{i=1}^{t-1} g^{x_{h_i j} y_{h_i j}}} = B_{h_t j} * \prod_{i=1}^{n-t} g^{x_{c_i j} y_{c_i j}}$. Similarly, given $B'_{h_t j}$, the adversary can compute $\tilde{B}'_{h_t j} = \frac{g^{s'_{h_t j} w'_{h_t j}}}{\prod_{i=1}^{t-1} g^{x_{h_i j} y_{h_i j}}} = B'_{h_t j} * \prod_{i=1}^{n-t} g^{x_{c_i j} y_{c_i j}}$. We have assumed that $\sum_{k=1}^{t} s_{h_k j} w_{h_k j} = \sum_{k=1}^{t} s'_{h_k j} w'_{h_k j}$. According to assumption 2,

$$(B_{h_1 j}, B_{h_2 j}, \ldots, \tilde{B}_{h_t j}) \overset{c}{\approx} (B'_{h_1 j}, B'_{h_2 j}, \ldots, \tilde{B}'_{h_t j})$$

Hence, the lemma holds.

7.3 Unlinkability

In the above section, we have shown that no polynomial-time adversary can find any information about the secret input and weight of an honest user, other than what she can learn from the publicly known output of the protocol. If she colludes with some of the participants, she will get to learn their inputs and weights associated with them. Then she can eliminate the colluding users' inputs from the overall weighted sum. As such, she will be left with the partial weighted sum of the ratings of honest users. This is what she can learn in the worst case. Given this partial sum, there could be many possible inputs and associated weights that would lead to the same partial sum. The adversary has no way to distinguish between all these possibilities. In a way, this would cripple the ability of the adversary to link a particular input or weight to a particular user. However, if the adversary corrupts a sufficient number of users she can reduce the number of possibilities. For example, if the adversary corrupts $n-1$ users, then she can compute the weighted value of the sole honest user by subtracting the weighted sum of the corrupt users' inputs from the output of the protocol τ_j. This is unavoidable as the goal of this protocol is to compute τ_j. So, our protocol ensures the unlinkability of inputs to the users as long as the output of the protocol does not negate it.

8 Efficiency

This section evaluates the performance of our proposed system for performance metrics, i.e, computation and bandwidth overheads incurred by the protocol operations. We evaluate the computation and communication overheads required for generating the cryptograms of the trust scores and NIZK proofs. Our protocol consists of performing two rounds in every iteration. In the first round, each of the users chooses a secret key and publishes the corresponding public key. The computation of the public key from the secret key requires exactly one exponentiation. Further, the user needs to submit encrypted feedback in the second round, which requires one exponentiation per user. Thus, for generating the feedback, each user needs to do 2 exponentiations. However, the most expensive operation in our scheme is the generation of the NIZK proofs. The NIZK proof π_i proves that each user knows the secret key corresponding to her public key. This NIZK proof is constructed using the Schnorr protocol [70], which requires one exponentiation for the generation of the arguments and two exponentiations for verification of the arguments. The other NIZK proof is Π_i which proves the well-formedness of the encrypted feedback B_{ij} in iteration j. As shown in the Appendix, the construction of each of these NIZK proofs needs 156 exponentiations per user and verification of the same requires 160 exponentiations for one NIZK proof. Table 2 shows a computation overhead observed while generating the feedback scores at the user side and then verifying it during the verification process.

Now, we discuss the bandwidth cost of our scheme. Each of the public keys and the feedbacks is represented by one element of G. The NIZK proof π_i comprises one commitment, one challenge and a single response, hence, the total size of the proof is 3. Note that the commitment is a group element whereas the challenge and the response belong to $\mathbb{Z}_p$. Also, as discussed in the Appendix, the NIZK proof Π_i comprises 140 elements. Thus the bandwidth overhead on one user is equal to 145. Table 3 shows the communication bandwidth required for submitting cryptograms to the bulletin board.

	Round I		Round II		
Entity	Public Key	NIZKP	Feedback	NIZKP	**Total**
User	1	1	1	156	159
Verifier	-	$2n$	-	$160n$	$162n$

TABLE 2: The exponentiations required in Feedback and Verification Process. Here, n is the total number of users.

	Round I		Round II		
Entity	Public Key	NIZKP	Feedback	NIZKP	**Total**
User	1	3	1	140	145
Verifier	n	$3n$	n	$140n$	$145n$

TABLE 3: The Communication Bandwidth required for committing feedback. Here, n is the total number of users.

9 Evaluation and benchmarks

In this section, we present the prototype implementation of the proposed self-enforcing trust management system and analyze its computational and bandwidth overheads. Finally, we discuss how the system can be deployed in a real-world scenario.

9.1 Prototype Implementation

We have implemented a prototype of the proposed trust management system which consists of two major components: a user module, and the trust aggregation module. For the bulletin board, we used a simple database to hold the user-submitted data. We coded the functionality of the user module and aggregation module in Java using the cryptographic library, i.e. BouncyCastle [2]. The prototype supports all major functions supported by the proposed systems including the construction of NIZK proofs, generation of crypto parameters and checks of the NIZK proof to ensure the well-formedness. For the crypto parameters, we used standard NIST Curve P-256 [71] for 128-bit security. We have performed an evaluation using a system with CPU Intel i-7 and 8GB RAM. For the NIZK proof, we used hashing commitment method as it incurs a small computation and communication overhead.

9.2 Benchmarks

We present experimental results for the following three phases: 1) generating the public, private and encryption keys, 2) generating the encrypted responses for the user trust scores, and 3) performing the aggregation to compute aggregated trustworthiness of users and objects. The time taken to generate the public, private and encryption keys is not high. The generation of a pair of public and private keys would only take around 2 milliseconds, whereas the restructured key from the posted public keys would be computed in less than 10 milliseconds for around 100 users. However, this time increases linearly with the number of users who agree to provide the responses. The computationally most expensive step in our scheme is the calculation of the ciphertexts about the user's trust scores because of the inclusion of NIZK-proof. Table 4 shows the CPU time required for calculating the ciphertexts of responses (trust scores) when the number of responses varies from 1 to 100. The table shows the running time for each protocol step as well as the whole running time. The results show that the computation time is not very large even for a high number of responses: i.e., the user is able to encrypt responses for 6 devices in less than a second. However, with the parallel optimization, this time would further increase to a few milliseconds. Similarly, communication overheads are not very large. A single encrypted response including the NIZK proof would only require 140 bytes which increase linearly with the number of responses.

The results presented in Table 4 achieve comparable performance as compared to [17], however, the schemes proposed in [17] utilize additional trusted centralized servers (two) for the privacy-preservation, trust evaluation and dissemination of final trust score among the collaborating devices. However, in this paper, we consider the scenario of trust aggregation without any trusted party in a decentralized setup, thus the system does not require any trusted system for score dissemination and management.

10 Conclusion

A Social Internet of Things (SIoT) is the network paradigm where the IoT devices interact with each other and develop the social network among the devices and their users. The strength of social relationships among users depends

[2] https://www.bouncycastle.org/java.html

	Round I		Round II		
Entity	Public Key	NIZKP	Feedback	NIZKP	**Total**
User	$6msec$	$6msec$	6msec	$936msec$	$954msec$
Verifier	-	$1.2sec$	-	$32sec$	$34sec$

TABLE 4: The computational overhead for 100 users, each with 6 IoT devices.

upon the services provided by the users and their IoT devices. Within this structure, it is important to evaluate the trustworthiness of IoT devices as well as the users so as to prevent malicious entities from disseminating malicious content or causing network disruption. One way to assess the trustworthiness of entities (users and IoT devices) is to crowdsource entities to provide feedback about their interaction with certain IoT devices and users. However, crowdsourcing has the challenge of privacy-preservation as the contributed responses could disclose the communication behavior of devices and users. To address this challenge, this paper presents a new privacy-preserving system for assessing the trustworthiness of the IoT devices and users simultaneously within the SIoT ecosystem. The proposed system adopts a homomorphic encryption system with the properties of decentralization, self-enforcement, and privacy-preservation. The proposed system assures correct computation, privacy and security of users even in the presence of malicious parties and colluding users. The self-enforcing computation is a unique feature of our system, as the whole computation process does not involve any trusted third party and it allows publication verification on the integrity of the scores in an autonomous way. The privacy and security analysis shows that the contributions of participants would not allow other users, colluders or malicious parties to learn the private information during and after the protocol operations. Through extensive experiments on the prototype implementation, we show that the proposed system has introduced a small communication and computation overhead and has the potential to be used in real-world IoT devices and smartphones.

Acknowledgment

Feng Hao and Samiran Bag are partly funded by the ERC starting grant No. 306994 and the Royal Society grant, ICA/R1/180226.

References

[1] F. Gómez Mármol and G. Martínez Pérez, "Trip, a trust and reputation infrastructure-based proposal for vehicular ad hoc networks," *Network and Computer Applications*, vol. 35, no. 3, pp. 934–941, May 2012.

[2] W. Li and H. Song, "Art: An attack-resistant trust management scheme for securing vehicular ad hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 4, pp. 960–969, 2016.

[3] O. Hasan, L. Brunie, and E. Bertino, "Preserving privacy of feedback providers in decentralized reputation systems," *Computer and Security*, vol. 31, no. 7, pp. 816–826, Oct. 2012.

[4] 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*, ser. WWW '03, 2003, pp. 640–651.

[5] 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.

[6] L. De Alfaro, A. Kulshreshtha, I. Pye, and B. T. Adler, "Reputation systems for open collaboration," *ACM Communications*, vol. 54, no. 8, pp. 81–87, Aug. 2011.

[7] 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.

[8] M. Sirivianos, K. Kim, and X. Yang, "Socialfilter: introducing social trust to collaborative spam mitigation," in *Proceedings of Workshop on Collaborative Methods for Security and Privacy (collSec)*, 2010.

[9] R. Neisse, G. Steri, and G. Baldini, "Enforcement of security policy rules for the internet of things," in *2014 IEEE 10th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Oct 2014, pp. 165–172.

[10] C. Chow, M. F. Mokbel, and T. He, "A privacy-preserving location monitoring system for wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 10, no. 1, pp. 94–107, Jan 2011.

[11] S. Marti and H. Garcia-Molina, "Identity crisis: anonymity vs reputation in p2p systems," in *Proceedings Third International Conference on Peer-to-Peer Computing (P2P2003)*, Sept 2003, pp. 134–141.

[12] J. Liu, C. Zhang, and Y. Fang, "Epic: A differential privacy framework to defend smart homes against internet traffic analysis," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1206–1217, April 2018.

[13] L. Lyu, K. Nandakumar, B. Rubinstein, J. Jin, J. Bedo, and M. Palaniswami, "Ppfa: Privacy preserving fog-enabled aggregation in smart grid," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 8, pp. 3733–3744, Aug 2018.

[14] C. Zhang, L. Zhu, C. Xu, X. Liu, and K. Sharif, "Reliable and privacy-preserving truth discovery for mobile crowdsensing systems," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–1, 2019.

[15] A. Schaub, R. Bazin, O. Hasan, and L. Brunie, "A trustless privacy-preserving reputation system," in *ICT Systems Security and Privacy Protection*, 2016, pp. 398–411.

[16] Z. Li and C. T. Chigan, "On joint privacy and reputation assurance for vehicular ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 13, no. 10, pp. 2334–2344, 2014.

[17] Z. Yan, W. Ding, V. Niemi, and A. V. Vasilakos, "Two schemes of privacy-preserving trust evaluation," *Future Generation Computer Systems*, vol. 62, pp. 175 – 189, 2016.

[18] C. Stergiou, K. E. Psannis, B.-G. Kim, and B. Gupta, "Secure integration of iot and cloud computing," *Future Generation Computer Systems*, vol. 78, pp. 964 – 975, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X1630694X

[19] Z. Guan, Y. Zhang, L. Wu, J. Wu, J. Li, Y. Ma, and J. Hu, "Appa: An anonymous and privacy preserving data aggregation scheme for fog-enhanced iot," *Journal of Network and Computer Applications*, vol. 125, pp. 82 – 92, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804518303060

[20] Z. Erkin, J. R. Troncoso-pastoriza, R. L. Lagendijk, and F. Perez-Gonzalez, "Privacy-preserving data aggregation in smart metering systems: an overview," *IEEE Signal Processing Magazine*, vol. 30, no. 2, pp. 75–86, March 2013.

[21] E. Damiani, S. D. C. D. Vimercati, S. Paraboschi, and P. Samarati, "Managing and sharing servants' reputations in p2p systems," *IEEE Transactions on Knowledge and Data Engineering*, vol. 15, no. 4, pp. 840–854, 2003.

[22] H. Gunasinghe, A. Kundu, E. Bertino, H. Krawczyk, S. Chari, K. Singh, and D. Su, "Prividex: Privacy preserving and secure exchange of digital identity assets." in *The World Wide Web Conference*, ser. WWW '19. New York, NY, USA: ACM, 2019, pp. 594–604. [Online]. Available: http://doi.acm.org/10.1145/3308558.3313574

[23] D. Liu, A. Alahmadi, J. Ni, X. Lin, and X. Shen, "Anonymous reputation system for iiot-enabled retail marketing atop pos blockchain," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3527–3537, June 2019.

[24] E. Androulaki, S. G. Choi, S. M. Bellovin, and T. Malkin, "Reputation systems for anonymous networks," in *Proceedings of the 8th International Symposium on Privacy Enhancing Technologies*, 2008, pp. 202–218.

[25] E. Gudes, N. Gal-Oz, and A. Grubshtein, "Methods for computing trust and reputation while preserving privacy," in *Proceedings of*

23rd Annual IFIP WG 11.3 Working Conference Data and Applications Security, 2009, pp. 291–298.

[26] J. Liu, H. Shen, H. S. Narman, W. Chung, and Z. Lin, "A survey of mobile crowdsensing techniques: A critical component for the internet of things," *ACM Trans. Cyber-Phys. Syst.*, vol. 2, no. 3, pp. 18:1–18:26, Jun. 2018. [Online]. Available: http://doi.acm.org/10.1145/3185504

[27] M. binti Mohamad Noor and W. H. Hassan, "Current research on internet of things (iot) security: A survey," *Computer Networks*, vol. 148, pp. 283 – 294, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128618307035

[28] E. Ahmed, I. Yaqoob, A. Gani, M. Imran, and M. Guizani, "Internet-of-things-based smart environments: state of the art, taxonomy, and open research challenges," *IEEE Wireless Communications*, vol. 23, no. 5, pp. 10–16, October 2016.

[29] I. Chen, F. Bao, and J. Guo, "Trust-based service management for social internet of things systems," *IEEE Transactions on Dependable and Secure Computing*, vol. 13, no. 6, pp. 684–696, Nov 2016.

[30] M. Nitti, R. Girau, and L. Atzori, "Trustworthiness management in the social internet of things," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 5, pp. 1253–1266, May 2014.

[31] J. Chen, Z. Tian, X. Cui, L. Yin, and X. Wang, "Trust architecture and reputation evaluation for internet of things," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 8, pp. 3099–3107, Aug 2019. [Online]. Available: https://doi.org/10.1007/s12652-018-0887-z

[32] H. Xia, F. Xiao, S. Zhang, C. Hu, and X. Cheng, "Trustworthiness inference framework in the social internet of things: A context-aware approach," in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, April 2019, pp. 838–846.

[33] H. Xia, C. qiang Hu, F. Xiao, X. guo Cheng, and Z. kuan Pan, "An efficient social-like semantic-aware service discovery mechanism for large-scale internet of things," *Computer Networks*, vol. 152, pp. 210 – 220, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S138912861930177X

[34] A. A. Adewuyi, H. Cheng, Q. Shi, J. Cao, . MacDermott, and X. Wang, "Ctrust: A dynamic trust model for collaborative applications in the internet of things," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5432–5445, June 2019.

[35] C. Castelluccia, E. Mykletun, and G. Tsudik, "Efficient aggregation of encrypted data in wireless sensor networks," in *Proceedings of the The Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services*, 2005, pp. 109–117.

[36] I. R. Chen, J. Guo, and F. Bao, "Trust management for soa-based iot and its application to service composition," *IEEE Transactions on Services Computing*, vol. 9, no. 3, pp. 482–495, May 2016.

[37] P. P. Jayaraman, X. Yang, A. Yavari, D. Georgakopoulos, and X. Yi, "Privacy preserving internet of things," *Future Gener. Comput. Syst.*, vol. 76, no. C, pp. 540–549, Nov. 2017. [Online]. Available: https://doi.org/10.1016/j.future.2017.03.001

[38] J. Ko, J. H. Lim, Y. Chen, R. Musvaloiu-E, A. Terzis, G. M. Masson, T. Gao, W. Destler, L. Selavo, and R. P. Dutton, "Medisn: Medical emergency detection in sensor networks," *ACM Trans. Embed. Comput. Syst.*, vol. 10, no. 1, pp. 11:1–11:29, Aug. 2010. [Online]. Available: http://doi.acm.org/10.1145/1814539.1814550

[39] S. Moin, A. Karim, Z. Safdar, K. Safdar, E. Ahmed, and M. Imran, "Securing iots in distributed blockchain: Analysis, requirements and open issues," *Future Generation Computer Systems*, vol. 100, pp. 325 – 343, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X18330851

[40] Y. Chen, H. Xie, K. Lv, S. Wei, and C. Hu, "Deplest: A blockchain-based privacy-preserving distributed database toward user behaviors in social networks," *Information Sciences*, vol. 501, pp. 100 – 117, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0020025519305250

[41] X. Gan, Y. Li, Y. Huang, L. Fu, and X. Wang, "When crowdsourcing meets social iot: An efficient privacy-preserving incentive mechanism," *IEEE Internet of Things Journal*, pp. 1–1, 2019.

[42] G. Fortino, F. Messina, D. Rosaci, and G. M. L. Sarne, "Using blockchain in a reputation-based model for grouping agents in the internet of things," *IEEE Transactions on Engineering Management*, pp. 1–13, 2019.

[43] B. Tang, H. Kang, J. Fan, Q. Li, and R. Sandhu, "Iot passport: A blockchain-based trust framework for collaborative internet-of-things," in *Proceedings of the 24th ACM Symposium on Access Control Models and Technologies*, ser. SACMAT '19. New York, NY, USA: ACM, 2019, pp. 83–92. [Online]. Available: http://doi.acm.org/10.1145/3322431.3326327

[44] K. Li, L. Tian, W. Li, G. Luo, and Z. Cai, "Incorporating social interaction into three-party game towards privacy protection in iot," *Computer Networks*, vol. 150, pp. 90 – 101, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128618312945

[45] B. Parno and A. Perrig, "Challenges in securing vehicular networks,," *Proceedings of the Workshop on Hot Topics in Networks (HotNets-IV), 2005.*

[46] X. Wang, Z. Ning, M. Zhou, X. Hu, L. Wang, Y. Zhang, F. R. Yu, and B. Hu, "Privacy-preserving content dissemination for vehicular social networks: Challenges and solutions," *IEEE Communications Surveys Tutorials*, vol. 21, no. 2, pp. 1314–1345, Secondquarter 2019.

[47] T. Cheng, G. Liu, Q. Yang, and J. Sun, "Trust assessment in vehicular social network based on three-valued subjective logic," *IEEE Transactions on Multimedia*, vol. 21, no. 3, pp. 652–663, March 2019.

[48] M. A. Azad, S. Bag, S. Parkinson, and F. Hao, "Trustvote: Privacy-preserving node ranking in vehicular networks," *IEEE Internet of Things Journal*, pp. 1–1, 2019.

[49] S. Guleng, C. Wu, X. Chen, X. Wang, T. Yoshinaga, and Y. Ji, "Decentralized trust evaluation in vehicular internet of things," *IEEE Access*, vol. 7, pp. 15 980–15 988, 2019.

[50] Z. Yang, K. Yang, L. Lei, K. Zheng, and V. C. M. Leung, "Blockchain-based decentralized trust management in vehicular networks," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1495–1505, April 2019.

[51] A. Riahi Sfar, Y. Challal, P. Moyal, and E. Natalizio, "A game theoretic approach for privacy preserving model in iot-based transportation," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, 2019.

[52] C. Miao, W. Jiang, L. Su, Y. Li, S. Guo, Z. Qin, H. Xiao, J. Gao, and K. Ren, "Privacy-preserving truth discovery in crowd sensing systems," *ACM Trans. Sen. Netw.*, vol. 15, no. 1, pp. 9:1–9:32, Jan. 2019. [Online]. Available: http://doi.acm.org/10.1145/3277505

[53] H. Kajino, H. Arai, and H. Kashima, "Preserving worker privacy in crowdsourcing," *Data Min. Knowl. Discov.*, vol. 28, no. 5-6, pp. 1314–1335, Sep. 2014. [Online]. Available: http://dx.doi.org/10.1007/s10618-014-0352-3

[54] E. Pavlov, J. S. Rosenschein, and Z. Topol, "Supporting privacy in decentralized additive reputation systems," in *Proceedings of Second International Conference on Trust Management*, C. Jensen, S. Poslad, and T. Dimitrakos, Eds., 2004, pp. 108–119.

[55] 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.

[56] T. Li, C. Gao, L. Jiang, W. Pedrycz, and J. Shen, "Publicly verifiable privacy-preserving aggregation and its application in iot," *Journal of Network and Computer Applications*, vol. 126, pp. 39 – 44, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804518303059

[57] L. Atzori, A. Iera, and G. Morabito, "Social internet of things: turning smart objects into social objects to boost the iot," *Newsletter*, 2015.

[58] S. Rho and Y. Chen, "Social internet of things: Applications, architectures and protocols," *Future Generation Computer Systems*, vol. 82, pp. 667 – 668, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X18301158

[59] 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.

[60] 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.

[61] J. He, L. Cai, P. Cheng, J. Pan, and L. Shi, "Consensus-based data-privacy preserving data aggregation," *IEEE Transactions on Automatic Control*, pp. 1–1, 2019.

[62] Q. Kong, R. Lu, M. Ma, and H. Bao, "A privacy-preserving sensory data sharing scheme in internet of vehicles," *Future Generation Computer Systems*, vol. 92, pp. 644 – 655, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X17317004

[63] 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.

[64] 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.

[65] 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," *USENIX Journal of Election Technology and Systems (JETS)*, pp. 1–25, 2014.

[66] G. Fortino, F. Messina, D. Rosaci, and G. M. L. Sarne, "Using blockchain in a reputation-based model for grouping agents in the internet of things," *IEEE Transactions on Engineering Management*, pp. 1–13, 2019.

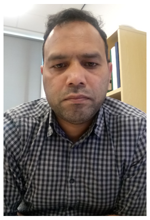
[67] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Cryptography Mailing list at https://metzdowd.com*, 03 2009.

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

[69] F. Hao and P. Zieliński, "A 2-round anonymous veto protocol," in *Security Protocols*, B. Christianson, B. Crispo, J. A. Malcolm, and M. Roe, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 202–211.

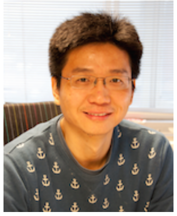
[70] C. P. Schnorr, "Efficient signature generation by smart cards," *Journal of Cryptology*, vol. 4, no. 3, pp. 161–174, 1991.

[71] "Digital signature standard (dss), u.s. department of commerce/national institute of standards and technology." 2017. [Online]. Available: http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf.



Muhammad Ajmal Azad received a Ph.D. (2016) degree in Electrical and Computer Engineering from the University of Porto, Portugal. He is currently a lecturer in cybersecurity in the school of computer science at The University of Derby.

Samiran Bag received the Ph.D. degree from the Indian Statistical Institute. He is currently a Research Fellow with the Department of Computing Science, University of Warwick, UK.



Feng Hao is a Professor of Security Engineering in the Department of Computer Science, University of Warwick. He graduated with a Ph.D. in 2007 from the Computer Laboratory, University of Cambridge. His research interests include applied cryptography, security engineering, and efficient computing. Since 2013, he has been serving as an associate editor for the IEEE Security and Privacy magazine. He is supported by ERC Starting Grant (No. 306994) and ERC Proof of Concept Grant (No. 677124).

Andrii Shalaginov has been awarded a Ph.D. degree in Information Security from NTNU in February 2018. His primary expertise is in static and dynamic malware analysis, development of machine learning-aided intelligent computer viruses detection models and similarity-based categorization of malware types and families.

APPENDIX

Here we show how each user U_i can construct a NIZK proof of well-formedness of B_{ij}. This NIZK proof Π_i proves the following statement

$$\left\{\begin{array}{l}
(w_{ij} = w_{i(j-1)}) \wedge (w_{i(j-1)} = 1) \wedge (T_j = -s_{ij}) \vee \\
(w_{ij} = w_{i(j-1)}) \wedge (w_{i(j-1)} = 5) \wedge (T_j = s_{ij}) \vee \\
(w_{ij} = w_{i(j-1)} + 1) \wedge (w_{i(j-1)} = 4) \wedge (T_j = s_{ij}) \vee \\
(w_{ij} = w_{i(j-1)} + 1) \wedge (w_{i(j-1)} = 3) \wedge (T_j = s_{ij}) \vee \\
(w_{ij} = w_{i(j-1)} + 1) \wedge (w_{i(j-1)} = 2) \wedge (T_j = s_{ij}) \vee \\
(w_{ij} = w_{i(j-1)} + 1) \wedge (w_{i(j-1)} = 1) \wedge (T_j = s_{ij}) \vee \\
(w_{ij} = w_{i(j-1)} - 1) \wedge (w_{i(j-1)} = 2) \wedge (T_j = -s_{ij}) \vee \\
(w_{ij} = w_{i(j-1)} - 1) \wedge (w_{i(j-1)} = 3) \wedge (T_j = -s_{ij}) \vee \\
(w_{ij} = w_{i(j-1)} - 1) \wedge (w_{i(j-1)} = 4) \wedge (T_j = -s_{ij}) \vee \\
(w_{ij} = w_{i(j-1)} - 1) \wedge (w_{i(j-1)} = 5) \wedge (T_j = -s_{ij})
\end{array}\right.$$

Let $T_j = 1$. As such Π_i can be written as

$$\left\{\begin{array}{l}
(w_{ij} = w_{i(j-1)}) \wedge (w_{i(j-1)} = 1) \wedge (s_{ij} = -1) \vee \\
(w_{ij} = w_{i(j-1)}) \wedge (w_{i(j-1)} = 5) \wedge (s_{ij} = 1) \vee \\
(w_{ij} = w_{i(j-1)} + 1) \wedge (w_{i(j-1)} = 4) \wedge (s_{ij} = 1) \vee \\
(w_{ij} = w_{i(j-1)} + 1) \wedge (w_{i(j-1)} = 3) \wedge (s_{ij} = 1) \vee \\
(w_{ij} = w_{i(j-1)} + 1) \wedge (w_{i(j-1)} = 2) \wedge (s_{ij} = 1) \vee \\
(w_{ij} = w_{i(j-1)} + 1) \wedge (w_{i(j-1)} = 1) \wedge (s_{ij} = 1) \vee \\
(w_{ij} = w_{i(j-1)} - 1) \wedge (w_{i(j-1)} = 2) \wedge (s_{ij} = -1) \vee \\
(w_{ij} = w_{i(j-1)} - 1) \wedge (w_{i(j-1)} = 3) \wedge (s_{ij} = -1) \vee \\
(w_{ij} = w_{i(j-1)} - 1) \wedge (w_{i(j-1)} = 4) \wedge (s_{ij} = -1) \vee \\
(w_{ij} = w_{i(j-1)} - 1) \wedge (w_{i(j-1)} = 5) \wedge (s_{ij} = -1)
\end{array}\right.$$

This statement is logically equivalent to the below statement.

$$\left\{\begin{array}{l}
((B_{ij} = g^{x_{ij}y_{ij}}g) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}/g))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}/g) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}/g))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}g^5) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}g^5))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}/g^5) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}g^5))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}g^5) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}g^4))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}/g^5) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}g^4))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}g^4) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}g^3))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}/g^4) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}g^3))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}g^3) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}g^2))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}/g^3) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}g^2))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}g^2) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}g))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}/g^2) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}g))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}g^4) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}/g^5))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}/g^4) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}/g^5))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}g^3) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}/g^4))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}/g^3) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}/g^4))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}g^2) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}/g^3))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}/g^2) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}/g^3))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}g) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}/g^2))\vee \\
((B_{ij} = g^{x_{ij}y_{ij}}/g) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}/g^2))
\end{array}\right.$$

We show how the user U_i can construct a set of NIZK arguments for the above statement that is when $T_j = 1$. The prover can generate NIZK arguments similarly when $T_j = -1$. Now, since, the above statement is an OR statement; exactly one of the sub-statements is correct. We assume the first sub-statement is correct, that is : $((B_{ij} = g^{x_{ij}y_{ij}}g) \wedge (B_{i(j-1)} = g^{x_{i(j-1)}y_{i(j-1)}}/g))$. The prover generates a real proof for this statement and 19 simulated proofs for the other 19 statements. The prover chooses random $r_{11}, r_{12} \in_R \mathbb{Z}_p$ and computes commitments $\epsilon_{11} = g^{r_{11}}, \epsilon_{12} = (g^{x_{ij}})^{r_{11}}, \epsilon_{13} = g^{r_{12}}, \epsilon_{14} = (g^{x_{i(j-1)}})^{r_{12}}$. The prover also chooses random $res_{h,k}, ch_h, h \in [2, 20], k \in [1, 2]$ and computes: $\epsilon_{h,1} = g^{res_{h,1}}(g^{x_{ij}})^{ch_h}, \epsilon_{h,2} = (g^{x_{ij}})^{res_{h,1}}(B^h_{ij})^{ch_h}, \epsilon_{h,3} = g^{res_{h,2}}(g^{x_{ij}})^{ch_h}, \epsilon_{h,4} = (g^{x_{ij}})^{res_{h,2}}(B^h_{i(j-1)})^{ch_h}$ for $h \in [2, 20]$. Here, $B^2_{ij} = B_{ij} * g, B^3_{ij} = B_{ij}/g^5, B^4_{ij} = B_{ij} * g^5, B^5_{ij} = B_{ij}/g^5, B^6_{ij} = B_{ij} * g^5, B^7_{ij} = B_{ij}/g^4, B^8_{ij} = B_{ij}*g^4, B^9_{ij} = B_{ij}/g^3, B^{10}_{ij} = B_{ij}*g^3, B^{11}_{ij} = B_{ij}/g^2, B^{12}_{ij} = B_{ij}*g^2, B^{13}_{ij} = B_{ij}/g^4, B^{14}_{ij} = B_{ij}*g^4, B^{15}_{ij} = B_{ij}/g^3, B^{16}_{ij} = B_{ij} * g^3, B^{17}_{ij} = B_{ij}/g^2, B^{18}_{ij} = B_{ij} * g^2, B^{19}_{ij} = B_{ij}/g, B^{20}_{ij} = B_{ij} * g$. Similarly, $B^2_{i(j-1)} = B_{i(j-1)} * g, B^3_{i(j-1)} = B_{i(j-1)}/g^5, B^4_{i(j-1)} = B_{i(j-1)}/g^5, B^5_{i(j-1)} = B_{i(j-1)}/g^4, B^6_{i(j-1)} = B_{i(j-1)}/g^4, B^7_{i(j-1)} = B_{i(j-1)}/g^3, B^8_{i(j-1)} = B_{i(j-1)}/g^3, B^9_{i(j-1)} = B_{i(j-1)}/g^2, B^{10}_{i(j-1)} = B_{i(j-1)}/g^2, B^{11}_{i(j-1)} = B_{i(j-1)}/g, B^{12}_{i(j-1)} = B_{i(j-1)}/g, B^{13}_{i(j-1)} = B_{i(j-1)} * g^5, B^{14}_{i(j-1)} = B_{i(j-1)} * g^5, B^{15}_{i(j-1)} = B_{i(j-1)} * g^4, B^{16}_{i(j-1)} = B_{i(j-1)} * g^4, B^{17}_{i(j-1)} = B_{i(j-1)} * g^3, B^{18}_{i(j-1)} = B_{i(j-1)} * g^3, B^{19}_{i(j-1)} = B_{i(j-1)} * g^2, B^{20}_{i(j-1)} = B_{i(j-1)} * g^2$.

Let the random challenge of the NIZK proof be ch.The prover computes $ch_1 = ch - \sum_{h=2}^{20} ch$. The prover computes responses $res_{11} = r_{11} - ch_1 * x_{ij}$ and $res_{12} = r_{12} - ch_1 * x_{i(j-1)}$.

The verification equations are as follows

1) $g^{res_{h,1}} \overset{?}{=} \frac{\epsilon_{h,1}}{(g^{x_{ij}})^{ch_h}}, \forall h \in [1, 20]$
2) $g^{res_{h,2}} \overset{?}{=} \frac{\epsilon_{h,3}}{(g^{x_{ij}})^{ch_h}}, \forall h \in [1, 20]$
3) $(g^{y_{ij}})^{res_{1,1}} \overset{?}{=} \frac{\epsilon_{1,2}}{(B_{ij}/g)^{ch_1}}$
4) $(g^{y_{i(j-1)}})^{res_{1,2}} \overset{?}{=} \frac{\epsilon_{1,4}}{(B_{i(j-1)}*g)^{ch_1}}$
5) $(g^{y_{ij}})^{res_{2,1}} \overset{?}{=} \frac{\epsilon_{2,2}}{(B_{ij}*g)^{ch_2}}$
6) $(g^{y_{i(j-1)}})^{res_{2,2}} \overset{?}{=} \frac{\epsilon_{2,4}}{(B_{i(j-1)}*g)^{ch_2}}$
7) $(g^{y_{ij}})^{res_{3,1}} \overset{?}{=} \frac{\epsilon_{3,2}}{(B_{ij}/g^5)^{ch_3}}$
8) $(g^{y_{i(j-1)}})^{res_{3,2}} \overset{?}{=} \frac{\epsilon_{3,4}}{(B_{i(j-1)}/g^5)^{ch_3}}$
9) $(g^{y_{ij}})^{res_{4,1}} \overset{?}{=} \frac{\epsilon_{4,2}}{(B_{ij}*g^5)^{ch_4}}$
10) $(g^{y_{i(j-1)}})^{res_{4,2}} \overset{?}{=} \frac{\epsilon_{4,4}}{(B_{i(j-1)}/g^5)^{ch_4}}$
11) $(g^{y_{ij}})^{res_{5,1}} \overset{?}{=} \frac{\epsilon_{5,2}}{(B_{ij}/g^5)^{ch_5}}$
12) $(g^{y_{i(j-1)}})^{res_{5,2}} \overset{?}{=} \frac{\epsilon_{5,4}}{(B_{i(j-1)}/g^4)^{ch_5}}$
13) $(g^{y_{ij}})^{res_{6,1}} \overset{?}{=} \frac{\epsilon_{6,2}}{(B_{ij}*g^5)^{ch_6}}$
14) $(g^{y_{i(j-1)}})^{res_{6,2}} \overset{?}{=} \frac{\epsilon_{6,4}}{(B_{i(j-1)}/g^4)^{ch_6}}$
15) $(g^{y_{ij}})^{res_{7,1}} \overset{?}{=} \frac{\epsilon_{7,2}}{(B_{ij}/g^4)^{ch_7}}$
16) $(g^{y_{i(j-1)}})^{res_{7,2}} \overset{?}{=} \frac{\epsilon_{7,4}}{(B_{i(j-1)}/g^3)^{ch_7}}$
17) $(g^{y_{ij}})^{res_{8,1}} \overset{?}{=} \frac{\epsilon_{8,2}}{(B_{ij}*g^4)^{ch_8}}$
18) $(g^{y_{i(j-1)}})^{res_{8,2}} \overset{?}{=} \frac{\epsilon_{8,4}}{(B_{i(j-1)}/g^3)^{ch_8}}$
19) $(g^{y_{ij}})^{res_{9,1}} \overset{?}{=} \frac{\epsilon_{9,2}}{(B_{ij}/g^3)^{ch_9}}$
20) $(g^{y_{i(j-1)}})^{res_{9,2}} \overset{?}{=} \frac{\epsilon_{9,4}}{(B_{i(j-1)}/g^2)^{ch_9}}$
21) $(g^{y_{ij}})^{res_{10,1}} \overset{?}{=} \frac{\epsilon_{10,2}}{(B_{ij}*g^3)^{ch_{10}}}$
22) $(g^{y_{i(j-1)}})^{res_{10,2}} \overset{?}{=} \frac{\epsilon_{10,4}}{(B_{i(j-1)}/g^2)^{ch_{10}}}$
23) $(g^{y_{ij}})^{res_{11,1}} \overset{?}{=} \frac{\epsilon_{11,2}}{(B_{ij}/g^2)^{ch_{11}}}$
24) $(g^{y_{i(j-1)}})^{res_{11,2}} \overset{?}{=} \frac{\epsilon_{11,4}}{(B_{i(j-1)}/g)^{ch_{11}}}$
25) $(g^{y_{ij}})^{res_{12,1}} \overset{?}{=} \frac{\epsilon_{12,2}}{(B_{ij}*g^2)^{ch_{12}}}$
26) $(g^{y_{i(j-1)}})^{res_{12,2}} \overset{?}{=} \frac{\epsilon_{12,4}}{(B_{i(j-1)}/g)^{ch_{12}}}$
27) $(g^{y_{ij}})^{res_{13,1}} \overset{?}{=} \frac{\epsilon_{13,2}}{(B_{ij}/g^4)^{ch_{13}}}$
28) $(g^{y_{i(j-1)}})^{res_{13,2}} \overset{?}{=} \frac{\epsilon_{13,4}}{(B_{i(j-1)}*g^5)^{ch_{13}}}$
29) $(g^{y_{ij}})^{res_{14,1}} \overset{?}{=} \frac{\epsilon_{14,2}}{(B_{ij}*g^4)^{ch_{14}}}$
30) $(g^{y_{i(j-1)}})^{res_{14,2}} \overset{?}{=} \frac{\epsilon_{14,4}}{(B_{i(j-1)}*g^5)^{ch_{14}}}$
31) $(g^{y_{ij}})^{res_{15,1}} \overset{?}{=} \frac{\epsilon_{15,2}}{(B_{ij}/g^3)^{ch_{15}}}$
32) $(g^{y_{i(j-1)}})^{res_{15,2}} \overset{?}{=} \frac{\epsilon_{15,4}}{(B_{i(j-1)}*g^4)^{ch_{15}}}$
33) $(g^{y_{ij}})^{res_{16,1}} \overset{?}{=} \frac{\epsilon_{16,2}}{(B_{ij}*g^3)^{ch_{16}}}$
34) $(g^{y_{i(j-1)}})^{res_{16,2}} \overset{?}{=} \frac{\epsilon_{16,4}}{(B_{i(j-1)}*g^4)^{ch_{16}}}$
35) $(g^{y_{ij}})^{res_{17,1}} \overset{?}{=} \frac{\epsilon_{17,2}}{(B_{ij}/g^2)^{ch_{17}}}$
36) $(g^{y_{i(j-1)}})^{res_{17,2}} \overset{?}{=} \frac{\epsilon_{17,4}}{(B_{i(j-1)}*g^3)^{ch_{17}}}$
37) $(g^{y_{ij}})^{res_{18,1}} \overset{?}{=} \frac{\epsilon_{18,2}}{(B_{ij}*g^2)^{ch_{17}}}$
38) $(g^{y_{i(j-1)}})^{res_{18,2}} \overset{?}{=} \frac{\epsilon_{18,4}}{(B_{i(j-1)}*g^3)^{ch_{17}}}$
39) $(g^{y_{ij}})^{res_{19,1}} \overset{?}{=} \frac{\epsilon_{19,2}}{(B_{ij}/g)^{ch_{19}}}$
40) $(g^{y_{i(j-1)}})^{res_{19,2}} \overset{?}{=} \frac{\epsilon_{19,4}}{(B_{i(j-1)}*g^2)^{ch_{19}}}$

41) $(g^{y_{ij}})^{res_{20,1}} \overset{?}{=} \frac{\epsilon_{20,2}}{(B_{ij}*g)^{ch_{20}}}$

42) $(g^{y_{i(j-1)}})^{res_{20,2}} \overset{?}{=} \frac{\epsilon_{20,4}}{(B_{i(j-1)}*g^2)^{ch_{20}}}$

If the above 80 equations are satisfied, the NIZK proof is correct. The prover needs to perform 156 exponentiations for generating all the NIZK arguments. On the other hand, the verifier needs to do 160 exponentiations for verifying all of them. The NIZK proof consists of 80 commitments, 20 challenges and 40 responses, hence, the communication cost of the NIZK proof is 140. In a similar way, the prover can compute NIZK proofs when the other statements are true.