PrivaSense: Privacy-Preserving and Reputation-Aware Mobile Participatory Sensing

Hayam Mousa
INSA Lyon - LIRIS, FCI Menoufia University
Menofia, Egypt
hayam910@gmail.com

Sonia Ben Mokhtar, Omar Hasan, Lionel Brunie
INSA Lyon, LIRIS
Lyon, France
sonia.benmokhtar,omar.hasan, Lionel.Brunie@insa-lyon.fr

Osama Younes, Mohiy Hadhoud
FCI, Menoufia University
Menoufia, Egypt 94035
osama_youness@hotmail.com, mmhadhoud@yahoo.com

ABSTRACT

The integration of privacy into reputation systems is a crucial need for building secure and reliable participatory sensing applications. Participants are given the assurance that their privacy is preserved even if they contribute some personal sensitive data. In addition, reputation systems allow an application server to monitor participants' behaviors and evict those who provide the system with corrupted data. However, this integration requires achieving seemingly conflicting objectives. Reputation systems monitor participants behaviors along subsequent interactions. Whereas, one of the major objectives of privacy preserving systems is to unlink subsequent interactions. In this paper, we define a new attack (RR attack), which exploits this conflict in order to detect the succession of contributions provided by the same participant and to subsequently re-identify his original identity. We show that using this attack, more than 35% of contributions can be associated to their successive contributions in each campaign. We then propose PrivaSense as a new privacy preserving reputation system that integrates both reputation and privacy such that their objectives are simultaneously achieved. Experimental results are conducted using a real data-set. These results show that PrivaSense decreases by up to 80% the number of contributions linked to their original providers.

CCS CONCEPTS

• **Computer systems organization** → ;

KEYWORDS

Participatory sensing, Privacy, Reliability, Reputation, Re-identification attacks

ACM Reference format:
Hayam Mousa, Sonia Ben Mokhtar, Omar Hasan, Lionel Brunie, and Osama Younes, Mohiy Hadhoud. 2017. PrivaSense: Privacy-Preserving and Reputation-Aware Mobile Participatory Sensing.
In *Proceedings of ACM Conference, Washington, DC, USA, July 2017 (Conference'17),* 10 pages.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

The advancement and widespread use of mobile computing smart devices have helped towards the emergence of a new kind of application called *participatory sensing* [2]. These applications exploit both the mobility of participants and the sensing capabilities of their devices to construct mobile sensor networks [12] with much less cost and effort compared with traditional Wireless Sensor Networks (WSNs). During the last decade, several participatory sensing applications have been widely used to serve in different areas including health, commerce, etc [11]. Researchers have studied numerous challenges that should be addressed to build reliable and secure participatory sensing system [3, 10, 14]. These challenges include the assurance of participants' privacy and management of data reliability, which we discuss below.

On the one hand, different applications collect different types of sensed data (e.g. spatial, temporal, images, pollution, sound samples, accelerometer, biometric, barometric, etc) [3]. These data can be exploited to leak participants' privacy through accurately re-identifying their identity, their location at some given time, with whom they were, their movements (e.g. walking, running, sitting down, etc.)[10]. Subsequently, participants can be physically traced and hacked or robbed based on these data [10]. In [1, 7], Montjoye et al. and Antoine et al. approved that 95% of original identities are re-identified through sharing four contributions including time and location data. Different techniques were proposed to assure the privacy of participants [4]. The main objective of those techniques is to detach the link between each contribution and its provider as well as among multiple contributions provided by the same provider (i.e participant).

On the other hand, these applications are vulnerable to malicious participants who disrupt the system by contributing fabricated or corrupted contributions which affect data reliability and accuracy. To enhance data reliability, application severs adopt reputation systems to trace participants' behaviors along subsequent contributions in order to estimate their honesty and to evaluate the quality of their contributions. In [14], we have extensively studied, analyzed and compared reputation systems that were proposed in this context. From

which, it is evident that existing reputation systems manage the link between successive contributions of participants and their real identities to evaluate their behaviors and to evict malicious ones form the campaign. This objective contradicts with the objective of privacy preserving systems mentioned before. That is, managing the linkage among successive contributions leads to privacy leakage. This conflict is referred to as the *linkability* problem. As a simple example of linkability problem, consider a participant p_i who has a reputation score x in some campaign. When a new campaign starts, this participant shares his new contribution tagged with his current reputation x. The application server evaluates the contribution according to a reputation system (e.g. [14]) and assigns a feedback f to p_i's contribution. x is updated to $x + f$. Consequently, in an upcoming campaign, it is evident that the contribution tagged with reputation $x + f$ has been forwarded from the same identity with reputation x in the previous campaign even if the identity and the data are anonymized. That is, both contributions and their associated pseudonyms are linked according to the reputation account. That is why, monitoring reputation scores for a sequence of contributions clearly leads to profile participants' contributions and subsequently re-identifying their identities.

In the context of participatory sensing, both the challenges of privacy preservation and reputation management have been individually studied in literature (e.g. [4] and [14]). However, the integration of both these systems in the context still in its infancy. Existing privacy preserving reputation systems in participatory sensing do not have the ability to fulfill the objectives of both privacy and reputation systems simultaneously (e.g. [13] [18]). Such systems either allow participants to launch other attacks (e.g Sybil, and report flooding) which affect data reliability (e.g. [18]). Other systems allow to profile participants subsequent contributions which leads to participant re-identification (e.g. [13]). In this paper, we mainly define a new attack, *Reputation based Re-identification* (**RR attack**), which exploits the linkability problem to re-identify participants' original identities in a privacy preserving reputation aware participatory sensing system. Next, we propose a *Privacy-Preserving and Reputation-Aware Mobile Participatory Sensing System* that we call **PrivaSense**. PrivaSense system integrates privacy preserving and reputation systems such that their objectives are simultaneously achieved. It defends against the RR attack as well.

Our specific contributions can be summarized as follows:

(1) We define a new attack (RR attack) that aims to link multiple contributions from the same participant, and subsequently re-identify participants' identities.
(2) We present a novel *privacy-preserving and reputation-aware mobile participatory sensing system* **PrivaSense**. In this system, each participant is assigned a new pseudonym for each contribution. The application server evaluates a participant's contribution, assigns it a feedback, forwards this feedback to the reputation server who updates the corresponding reputation account and transfers this account to the next pseudonym of the same participant. Reputation scores are anonymized and transferred in the form of anonymous certificates. This allows participants to conserve their reputation scores across multiple interactions while preventing associations between consecutive contributions.
(3) We undertake an analysis revealing the robustness of our proposal against the attacks considered in the threat model described in Section 3.
(4) We conducted some experiments based on a real-world data-set [16] to measure both the resilience of our system against the RR attack (i.e. privacy issues) added to the effect of the proposed system on the data reliability. The experimental results indicate that our system introduces higher anonymity (i.e. better privacy) with more accurate data aggregation which enhances the system reliability compared with the state-of-the-art.

The rest of this paper is organized as follows: Section 2 states the previous work and its limitations. We then define the considered threat model in Section 3. Next, we present our proposed system and discuss its details in Section 4. In Section 5, we present an analysis of our proposal. Experimental results are presented in Section 6. Finally, this paper is concluded in Section 7.

2 RELATED WORK

Indeed, significant research effort has been directed toward ensuring privacy preservation in participatory sensing applications as described by Christin et al. in [4] . However, such works consider how to anonymize participants' real identities and/or to anonymize their provided data. Some other works consider the problem of reputation management such as [8] and our system presented in [15]. We have surveyed and compared these systems in [14]. Those systems essentially manage the link to participants real identities in order to assess their reputation. However, very few works consider the problem of privacy preserving reputation systems in mobile participatory sensing applications.

In [8], each participant is assigned different pseudonyms for each time intervals and to exchange the assigned reputation between those pseudonyms through a trusted server. Christin et al. in [5] propose a similar scheme which adopts the blind signature scheme to create pseudonyms. Through this system, a malicious participant can create multiple identities (i.e. Sybil attack) such that he can disrupt the system by providing multiple sensing contributions for the same task.

Another privacy preserving reputation system which assures the participant's anonymity through the group signature technique is presented by Michalas et al. [13]. Although, the system assures the anonymity of participants, it allows some entities to record a profile of participants through subsequent interactions. That is participant's privacy is leaked.

The scheme presented by Wang et al. in [18] utilizes the blind signature technique in order to ensure participant's anonymity. In this scheme, malicious participants can create multiple authentic identifiers based on a single blind identity granted for them (i.e. Sybil attack). In addition, they

can launch a report flooding attack. Subsequently, they can submit numerous contributions for the same task while the application server can't detect such behavior.

To sum up, existing works either accurately manage participants' reputation and allow for participants' re-identification and privacy leakage or focus on participants' anonymity and allow participants to launch other attacks that disrupt the system and affect data reliability. In this paper, we attempt to propose a privacy-preserving and reputation-aware system that allows the participatory sensing applications to be more reliable and secure.

3 THREAT MODEL

Commonly, in a sensing campaign we have mainly two parties: a participant p_i is a member in the participants set P, where $|P| = P$ such that $i \in \{1, 2, 3, ..., P\}$, and an application server noted as *App.Server*. We not only consider the attacks that can be launched by a malicious participant that affect the system reliability, but we consider the privacy threats that lead to participants re-identification as well.

Reliability oriented attacks:

- **Sybil Attack:** Malicious participants attempt to generate multiple pseudonyms to increase their reputation through cross-recommendations or providing multiple reports for the same task which subsequently leads to system disruption.
- **Replay Attack:** Malicious participants attempt to replay either old pseudonym which has a good reputation score. Replay attackers artificially increase their own reputation by replaying old reputation messages including good scores.
- **Report Falsified sensor readings:** Adversaries try to report falsified sensor readings on behalf of others to degrade their reputation.

Privacy oriented attacks:

- **Identity and Data Re-identification (IDR attack):** It is evident, in [7], that 95% of participants' identities are re-identified after submitting 4 spatial and temporal observations. Therefore, participants use pseudonyms instead of their original identities to anonymize their contributions. Subsequently, attackers cannot link multiple contributions to the same identity. However, adversaries try to infer the original identities of contributions' providers based on the content of their contributions [10]. In [1], Vincent et al. clarify that 94% of original identities are re-identified since participants share multi sensors data-set.
- **Reputation based Re-identification (RR attack):** We define this attack here for the first time. The challenge that faces both identity and data re-identification (IDR) attacker is it to associate numerous contributions to the same identity while participants share their contributions anonymously (i.e.using pseudonyms). Here, a new attack that enables adversaries to link different contributions to the same identity is defined. This attack has arisen as a result of incorporating reputation systems in a privacy preserving enabled participatory sensing system. RR attacker applies three consecutive phases (i.e. monitoring, uniqueness assessment, and profiling). Firstly, through the monitoring phase, an attacker listens to the network and records the message exchange among the different parties in the sensing campaign. For each task T_j, the attacker keeps the following information for each contribution (1) the pseudonyms $RID^j_{p_i}$ of participant p_i $(\forall i \in 1, 2, 3, ..., P)$, (2) the contents of p_i's contribution including location, time, and sensed data $(x_{T_j}, y_{T_j}), t_{T_j}, data_{T_j})$, (3) and the reputation scores of the same pseudonym $(\hat{R}_{p_i})$, (4) the feedback calculated based on the evaluation of p_i's contribution. The feedback is associated to its corresponding pseudonym $(RID^j_{p_i}, f_{RID^j_{p_i}})$. That is for each pseudonym, an attacker keeps a record for each task containing the following information $(RID^j_{p_i}, (x_{T_j}, y_{T_j}), t_{T_j}, data_{T_j})$ $(\hat{R}_{p_i}, f_{RID^j_{p_i}})$. The attacker updates the monitored reputation score $\hat{R}_{p_i}$ according to the feedback $f_{RID^j_{p_i}}$ for the same task to get the expected reputation score noted as ER_{p_i} and appends it to its corresponding record. Hence, the attacker knows in advance the reputation score (R_{p_i}) that is going to accompany the upcoming contribution of the next task T_{j+1}. However, he does not know the new pseudonym that is going to carry this score. That is why a uniqueness assessment step is required.

Secondly, through uniqueness assessment, unique reputation score R_{p_i} monitored at task T_{j+1} are identified. Intuitively, the pseudonym (e.g. $RID^j_{p_i}$) having unique reputation score R_{p_i} is linked to the pseudonym having the same unique value of expected reputation ER_{p_i} calculated at task T_j. If the reputation R_{p_i} score is not unique, this means the update process functions such that multiple participants are assigned the same reputation. In this case, all the contributions carrying the same reputation at T_{j+1} are considered as potential successors.

While, pseudonym carrying reputation R_{p_i} is linked to the pseudonym with the same expected reputation ER_{p_i}, if they are unique and they both have the same value. The RR attacker not only links pseudonyms but also contributions from both pseudonyms and records them in a profiling table under the same identity. An example of a profiling table for a set of subsequent tasks is depicted in Table 1. The results of RR attack are depicted in Section 6.2. It is evident that, large number of contributions are linked to their successors (e.g. 35%) which leads to more easier identities re-identification.

In this paper, a new privacy preserving reputation system (**PrivaSense**) that incorporates a reputation system into a privacy preserving participatory sensing

	p_1	p_2	p_3	p_4	...	p_n
T_1	$((x_{T_1}, y_{T_1}), t_{T_1}, data_{T_1})RID^1_{p_1}$	...	...	$((x_{T_1}, y_{T_1}), t_{T_1}, data_{T_1})RID^1_{p_4}$	...	$((x_{T_1}, y_{T_1}), t_{T_1}, data_{T_1})RID^1_{p_n}$
T_2	...	$((x_{T_2}, y_{T_2}), t_{T_2}, data_{T_2})RID^2_{p_2}$	$((x_{T_2}, y_{T_2}), t_{T_2}, data_{T_2})RID^2_{p_3}$	...	...	$((x_{T_2}, y_{T_2}), t_{T_2}, data_{T_2})RID^2_{p_n}$
T_3	$((x_{T_3}, y_{T_3}), t_{T_3}, data_{T_3})RID^3_{p_1}$	$((x_{T_3}, y_{T_3}), t_{T_3}, data_{T_3})RID^3_{p_2}$	...	$((x_{T_3}, y_{T_3}), t_{T_3}, data_{T_3})RID^3_{p_4}$	...	...
T_4	$((x_{T_4}, y_{T_4}), t_{T_4}, data_{T_4})RID^4_{p_1}$	...	$((x_{T_4}, y_{T_4}), t_{T_4}, data_{T_4})RID^4_{p_3}$	$((x_{T_4}, y_{T_4}), t_{T_4}, data_{T_4})RID^4_{p_4}$	...	$((x_{T_4}, y_{T_4}), t_{T_4}, data_{T_4})RID^4_{p_n}$
T_5	...	$((x_{T_5}, y_{T_5}), t_{T_5}, data_{T_5})RID^5_{p_2}$	$((x_{T_5}, y_{T_5}), t_{T_5}, data_{T_5})RID^5_{p_3}$	...	...	...
....	...	...	...	...	...	...
T_N	$((x_{T_N}, y_{T_N}), t_{T_N}, data_{T_N})RID^N_{p_1}$	$((x_{T_N}, y_{T_N}), t_{T_N}, data_{T_N})RID^N_{p_2}$	...	$((x_{T_N}, y_{T_N}), t_{T_N}, data_{T_N})RID^N_{p_4}$	...	$((x_{T_N}, y_{T_N}), t_{T_N}, data_{T_N})RID^N_{p_n}$

Table 1: An example of the profiling table

application is proposed. PrivaSense takes into account all the attacks considered in the threat model.

4 PRIVASENSE

The framework of privacy-preserving and reputation-aware mobile participatory sensing system, proposed in this paper, is depicted in Figure 1. It is clear, the proposed framework include four main parties a participant, an application server noted as App.Server, Authentication server noted as Auth.Server, and reputation server referred to as Rep.Server. A participant and App.Server are the common parties in a participatory sensing campaign. However, both Auth.Server and Rep.Server are trusted entities that are involved in order to manage the threat model described earlier. The assumptions related to each entity are defined as follows:

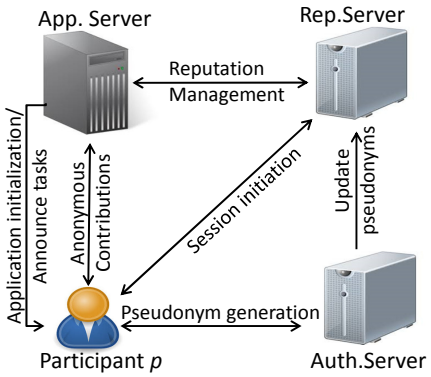


Figure 1: Privasense Architecture

- **Participant** A participant, (1) authenticates himself with Auth.Server, (2) selects a task from the tasks announced by the application server, (2) constructs a sensing report (i.e. Contribution), (3) forwards it to the application server, (4) re-authenticate with Auth.Server for a new task.
- **App.Server** The App.Server, (1) initiates sensing campaign, (2) receives and aggregated sensing reports, (3) adopts a reputation system, (4) assigns a feedback to each contribution, (5) and forwards this feedback to Rep.Server.
- **Auth.Server** is the entity that, (1) generates pseudonyms, (2) Forwards these pseudonyms to Rep.Server, (3) renews these pseudonyms after each campaign, (4) and sends this renew to Rep.server. Auth.Server keeps the succession of participants pseudonyms.
- **Rep.Server** is the entity that, (1) sends anonymous reputation certificates to application server, (2) receives feedback from the application server, (3) uses the feedback to update reputation scores, (4) receives pseudonym update from Auth.Server, (5) and links the reputation score of the old pseudonym to the current one, (6) discards old pseudonyms. Rep.Server has no information about original identities or contributions' contents.

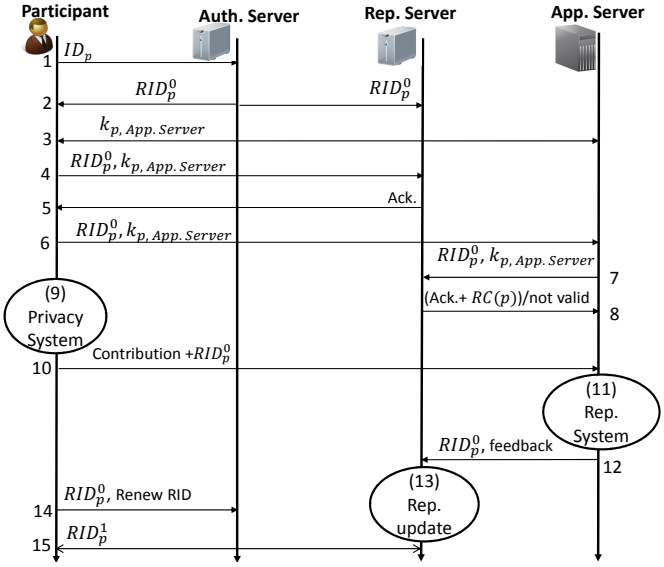


Figure 2: PrivaSense: a detailed scenario

4.1 Overview

The complete scenario of PrivaSense is depicted in Figure 2. First, participant p_i registers with an Auth.Server. The Auth.server creates the first random authentication identifier $RID^0_{p_i}$ as described in [9] (step 1). This $RID^0_{p_i}$ is the first pseudonym granted to participant p_i and it is simultaneously sent to the reputation server (Rep.Server) (step 2). Both p_i and the App.Server communicate to create a session key $(k_{p_i}, k_{App.Server})$ (step 3). The participant sends this key to the Rep.Server to initialize the session (step 4). Rep.Server records this key and acknowledges p_i (step 5), indicating the session has been initiated at the Rep.Server. p_i forwards his first anonymous identity $RID^0_{p_i}$ to App.Server (step 6). The App.Server sends a query about $RID^0_{p_i}$ to the Rep.server (step 7). The Rep.server checks if the received

identifier matches a user identity. So, a valid acknowledgment is sent back to the App.Server in line with an anonymous reputation certificate of the considered participant. This certificate contains both his current identifier and his current reputation score ($RID^0_{p_i}$, $\hat{R}_{p_i}$). Otherwise, a non valid acknowledgment is sent (step 8). If p_i is correctly verified by the Rep.Server, $RID^0_{p_i}$ senses his environment, constructs his first contribution ($C_{p_{i1}}$), applies a privacy preserving mechanism on the report contents (e.g. [4]) (step 9). Then, this contribution is forwarded to App.Server (step 10). The App.Server accepts the contribution, assesses its trust by applying a reputation management system (e.g. [15]) and calculates a feedback (step 11). This feedback is shared with the Rep.Server, (step 12), to update the reputation score of the considered participant (step 13). p_i contacts the Auth.Server to get a new pseudonym $RID^1_{p_i}$ by submitting his last pseudonym $RID^0_{p_i}$ (step 14). The new pseudonym is forwarded for both the participant and for the Rep.Server as well (step 15).

4.2 Protocol Outline

Our privacy preserving reputation system goes through four consecutive phases (1) Participant Registration and Authentication, (2) Issuing a Reputation Certificate, (3) Privacy and Reputation Assessment, (4) Re Authentication. The details of these phases are described as follows:.

4.2.1 Participant Registration and Authentication. Through this phase, participant p_i joins a sensing campaign. First, p_i sends his permanent identifier (e.g. ID_{p_i}) to the Auth.Server. The original identity of the participant is always kept secret along his lifetime. The first pseudonym $RID^0_{p_i}$ is then generated by the Auth.Server. We assume that pseudonyms are calculated as described in [9] using the public key of the Auth.Server (i.e. $k_{Auth.Server.pub}$) as shown in Equation 1. Only the Auth.Server is able to decrypt it and reveal the real identity of the participant. This pseudonym is forwarded for both the owner participant and the Rep.Server.

$$RID^0_p = E(ID_{p_i}, r^0_{p_i})_{k_{Auth.Server.pub}} \tag{1}$$

where $RID^0_{p_i}$ is the pseudonym generated, E is an encryption function of the identity ID_{p_i} and a random variable $r^0_{p_i}$. $k_{Auth.Server.pub}$ is the Auth.Server public key.

A participant then authenticates himself with the App.Server. First, both the participant p_i and the App.Server communicate through some key exchange mechanism, (e.g. Diff Helman key exchange mechanism), and generate a session key ($k_{p_i}, k_{App.Server}$). The participant sends this key to the Rep.Server in line with his pseudonym $RID^0_{p_i}$. Rep.Server records the received data and acknowledges the participant. The participant sends his $RID^0_{p_i}$ and the session key previously generated ($k_{p_i}, k_{App.Server}$) to the App.Server. The App.Server sends to the Rep.Server asking for the validity of $RID^0_{p_i}$ to make sure that the identity is valid and it has already initiated a session through the key exchange mechanism. Rep.Server checks the validity of the received anonymous identity and if the session key matches the one received earlier from this identity. If so, Rep.Server sends a valid acknowledgment and the reputation score of $RID^0_{p_i}$ embedded in an anonymous reputation certificate to the App.Server as shown in the upcoming phase. Otherwise, a non-valid acknowledgment is sent.

4.2.2 Issuing a Reputation Certificate. In order to manage the conflict discussed in Section 1, and to defend against the threat model described in Section 3, specifically RR attack. We adopt a double fold anonymization mechanism. First, we adopt a cloaking mechanism on the reputation scores before their transfer to the App.Server. Then, cloaked reputation scores are embedded within anonymous reputation certificates.

Firstly, to anonymize reputation scores, we cloak them by adding a small random noise. Reputation scores are randomly incremented or decremented by a value noted as *i*ncrement/*d*ecrement *a*mount referred to as *ida*. This change prevents the App.Server to link two consecutive reputation scores assigned to the same participant. For this, a *r*andom *i*ncrement/*d*ecrement variable noted as *rid* is generated such that $rid \in \{0, 1\}$. 0 is used to increment, and 1 is to decrement. Then, *ida* is generated such that it belongs to a specified cloaking interval noted as clk_{intr}, $ida \in [0, clk_{intr}]$. That is $ida \leq clk_{intr}$. Reputation scores are incremented such that they do not surpass 1 or decremented such that they are not less than 0. The maximum change imposed on the reputation score is $\pm clk_{intr}$. The anonymous reputation is referred to as $\hat{AR}_{p_i}$. The anonymization process is formulated according to Equation 2.

$$\hat{AR}_{p_i} = \begin{cases} min\left\{\hat{R}_{p_i} + ida, 1\right\} & \text{if } rid == 0 \\ max\left\{\hat{R}_{p_i} - ida, 0\right\} & \text{if } rid == 1 \end{cases} \tag{2}$$

Where $\hat{R}_{p_i}$ is the reputation score of p_i and $\hat{AR}_{p_i}$ is the output anonymized reputation score.

Using large values for clk_{intr} adds more noise to reputation scores. Intuitively, large values of cloaking intervals have a better performance from the anonymization point of view. However, this leaves a negative impact on the accuracy of the aggregated data. That is clk_{intr} is a system parameter that manages the trade-off between anonymity and accuracy. Subsequently, we test this parameter to see its effect on the system performance (e.g. anonymity and reliability) in Section 6.2.1.

Secondly, the cloaked reputation score of a participant is transferred to the App.Server in the form of an anonymous **r**eputation **c**ertificate noted as RC_{p_i}. This certificate is generated such that, it contains both the current pseudonym of the participant, $RID^0_{p_i}$, and its corresponding cloaked reputation score $\hat{AR}_{p_i}$. That is, RR attacker becomes puzzled to detect whom participant has which score and which pseudonym previously. That is, it ensures better anonymity and solves the problem of linkability described earlier. Subsequently, PrivaSense defends against RR attack that mainly depends on the linkability. Anonymous reputation certificate RC_{p_i} is

signed by the Rep.Server's private key. Therefore, it cannot be fabricated and Replay attack mentioned before cannot be launched. Adversaries has to access the private key of Rep.Server in order to fabricate a reputation certificate and to have the ability to deceive the App.Server.

$$RC_{p_i} = \left[RID_p^0 | \hat{AR}_{p_i}\right]_{k_{Rep.Server.priv}} \quad (3)$$

4.2.3 Privacy and reputation Assessment. Participant senses the required observation and constructs a contribution $C_{p_{ij}}$. We propose that participants apply one of the existing privacy preserving mechanisms summarized by Christin et.al. in [4]. Subsequently, the data provided by a participant are anonymized or cloaked such that they mitigate the effect of re-identification attacks [3, 10]. The anonymous contribution is subsequently forwarded to App.Server.

In PrivaSense, App.Sever adopts one of the existing reputation system such our system presented in [15]. According to this system, each contribution is evaluated and assigned a feedback which reflects the participant's honesty. Feedback is noted as $f_{RID_{p_i}^0}$. This feedback is forwarded to the Rep.Server in the following form.

$$f_{p_i} = \left[RID_{p_i}^0 | \hat{RC}_{p_i} | f_{RID_p^0}\right]_{k_{App.Server.priv}} \quad (4)$$

Rep.Server depends on this feedback to update the reputation score of the considered participant such that participant's reputation is increased if the feedback exceeds some threshold ϵ. Otherwise, participant's reputation is decreased as follows:

$$R_{p_i} = \begin{cases} min\left\{\hat{R}_{p_i} + f_{RID_p^0}, 1\right\} & \text{if } f_{RID_{p_i}^0} \geq \epsilon \\ max\left\{\hat{R}_{p_i} - f_{RID_p^0}, 0\right\} & \text{if } f_{RID_{p_i}^0} < \epsilon \end{cases} \quad (5)$$

4.2.4 Re Authentication. A participant asks for a new pseudonym to join an upcoming campaign with a different identifier. Thus, he sends his current identifier $RID_{p_i}^0$ to the Auth.Server. Now, Auth.Server knew earlier that this identity is valid. Thus, this time and in each subsequent renew, the Auth.Server generates a new identifier (e.g. $RID_{p_i}^1$) using a new random value $r_{p_i}^1$ as described in the following equation.

$$RID_{p_i}^1 = E(ID_{p_i}, r_{p_i}^1)_{k_{Aut.Server.pub}} \quad (6)$$

The new identifier is forwarded to both the participant and the Rep.Server in the form of a new identifier as follows.

$$New_{ID} = \left[RID_{p_i}^0 | RID_{p_i}^1\right]_{k_{Aut.Server.priv}} \quad (7)$$

The objective of sending both the old and new identifier simultaneously to the Rep.Server is apparent in different reasons. First, the Rep.Server links the reputation score of $RID_{p_i}^0$ to $RID_{p_i}^1$. Secondly, recording this new identity allows the participant to use this new identifier to contact the App.Server for a new task such that he is correctly validated by the Rep.Server. In addition, the Rep.Server discards the old pseudonym $RID_{p_i}^0$ such that it cannot be used again by an adversary.

5 SECURITY ANALYSIS

The goal of our evaluations is threefold: (1) analyze the robustness of our proposal against the threats identified in Section 3, (2) empirically evaluate the performance of PrivaSense, (3) compare PrivaSense with the existing systems. The first goal is discussed in this section whereas the second and the third goals are discussed in Section 6.

Starting by analyzing the Resilience of PrivaSense against the reliability oriented attacks:

- ***Sybil Attack***: PrivaSense system is protected against this attack. If the attacker tries to re-randomize the received RID using a new random number according to Equations 6 and 1, the resulting pseudonym RID will not match any valid authentication RID stored in the Rep.Server. This is because the attacker does not know the randomization seed used by the Auth.Server, and hence the attacker will not be able to generate the same series of randomized RIDs that match the real ones.
- ***Replay Attack***: The attacker cannot directly use the pseudonym RID twice since each RID is allowed to be used only once and is discarded after the use by the Rep.Server. Replay attacker also attempts to demonstrate a recent pseudonym and an old reputation score to the App.Server. Attackers are prevented from launching replay attack. In PrivaSense, Rep.Server maintains an up-to-date list of both valid pseudonyms and their associated reputation scores. Rep.Server is responsible for creating reputation certificate, signing it and forwarding it to the App.Server. Original reputation certificates include the most recent pseudonyms and their corresponding reputation scores and they are signed by the Rep.Server's private key $k_{Rep.Server.priv}$ which is not accessible to attackers. Private keys are kept secret all the time. Subsequently, attackers cannot replay reputation certificates.
- ***Reporting falsified sensor readings***: Malicious participants try to report falsified sensor readings on behalf of other participants to degrade their reputation. PrivaSense protects honest participants against this attack by requesting participants to authenticate with the Auth.Server and Rep.Server. These entities verify the validity of the pseudonyms before considering their contributions. Such an attack would only be successful if attackers access the original identities of the targeted participants and those of their respective pseudonyms. However, participants' original identities are kept secret during their lifetime.

Resilience against privacy oriented attacks:

- ***Identity and Data Re-identification***: Participants authenticate themselves using their anonymous identities RID. RID is encrypted by Auth.Server. Thus, it does not reveal any information concerning the participant's real identity. Therefore, an adversary cannot reveal the real identity of the sensing report provider unless he has access to the private key of the Auth.Server.

In addition, participants adopt one of the existing privacy preserving cloaking mechanisms to anonymize their sensed data. Therefore, our system ensures participants anonymity from both the identity and data point of views.

- ***Reputation based Re-identification (RR attack)***: Given that the anonymity of identity and the sensed data are ensured, the supplementary objective of designing a privacy preserving reputation protocol is to simultaneously ensure the un-linkability based on the reputation scores assigned to participants. PrivaSense adopts a double fold anonymization for reputation accounts. First, Rep.Server forwards an anonymous reputation certificate to the App.Server as described in Equation 4. That is, the App.Server cannot link the assigned reputation score to the original identity. In addition to that, reputation scores are cloaked based on Equation 2. So, App.Server knows the anonymized reputation score related to its pseudonym RID. That is, adversaries cannot link two consecutive reputation scores assigned to the same participant based on neither the identity nor the value of the reputation.

6 EXPERIMENTAL RESULTS

6.1 Evaluation Setup

6.1.1 Simulation Model. Let us now describe the simulation model of a noise monitoring paticipatory sensing application similar to [17]. Each simulation involves running the example application in the sequence described in Section 4. Each participant is assigned GPS timestamps and coordinates taken from the taxi mobile traces real dataset [16]. The data-set contains the GPS timestamps and coordinates of approximately 500 taxis collected in May, 2008 in the San Francisco Bay Area. For the first interaction, the Rep.Server simply assigns participants with some initial reputations while subsequent reputation values are calculated as described in Section 4.2.3.

We synthesize the noise distribution in an urban environment by assuming the data agree with the real noise levels described in [6]. We consider a quiet sensing area where the mean μ and standard deviation of the correct noise data is 60 db and 5 respectively. An honest participant sends correct sensing data. We also include malicious participants in the simulation to reflect a more realistic usage scenario. However, malicious participant sends false sensing data. We set the false data to contradict with the correct data. Therefore, the mean of false data is $\mu + \mu/3$ (i.e. 80 db). This means that malicious participant contributes data which correspond to a completely different level of noise. Thus, even one false report has a significant impact on the measurements. In addition, all false reports support each other. Thus, the standard deviation of false data is set to 0. Hence, we look for the worst case when all malicious participants collude to cause the biggest possible disturbance to the system. Table 2 lists our default parameter settings.

Parameter	Value
Number of participant for each task P	150
Number of adversaries for each task A	40
The mean value of correct data μ	60
The standard deviation of correct data	5
The mean value of adversary data ($\mu + \mu/3$)	80
The standard deviation of adversary data	0

Table 2: Default Parameter Settings

6.1.2 Evaluation Metrics. We evaluate our proposal according to three metrics to measure the levels of anonymity and reliability as follows:

Links. First, RR Links metric measures the number of contributions linked to their successors in each campaign based on the RR attack defined in Section 3. Whereas, PrivaSense Links are the links detected based on RR attack even with the adoption of PrivaSense.

Linkability. PrivaSense makes the Links metric does not make sense, because not only a participant's identity and data are anonymized but also due to the reputation scores. Thus, the links that can be detected due to the reputation scores are mostly removed. To measure the effect of PrivaSense on participants' anonymity, we use another metric called linkability. For this metric, a set of potential successors of each pseudonym is defined such that it contains a list of pseudonyms that can be the successor of the target participant. A large potential successors set ensures better anonymity. Note that the following description is applied to one reputation update. First, the Euclidean Distances between the location of $RID^0_{p_i}$ and the location of all the pseudonyms in the subsequent contribution are calculated. Then, the pseudonyms which ensure a distance less than λ are considered as the potential successors for $RID^0_{p_i}$. Next, an adversary selects a subset of the potential successors whose reputation scores are closer to the reputation of $RID^0_{p_i}$ noted as (β_s), the linkability metric is defined as $\frac{1}{\beta_s}$. Small values of this metric indicate much better anonymity and vice Versa.

RMSE. To measure the PrivaSense reliability, the accuracy of the aggregated data is evaluated. Application server calculates a weighted sum average of the aggregated data using reputation scores as weights. To measure the disruption incurred due to the anonymization of reputation scores, we compare the RMSE of the average noise levels calculated based on the anonymized reputation scores against the RMSE calculated using the original reputation scores without anoymization. The RMSE between two vectors of values is defined as follows:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{NC} (v_{1,i} - v_{2,i})^2}{NC}} \tag{8}$$

6.2 Results

6.2.1 Privacy.

Links. In this experiment, we measure the effect of RR attack. We observe the number of contributions that can be linked to their successive contributions in each two consecutive campaigns. clk_{intr} is set to 0.5. The results are depicted in Figure 3. It is evident that a large number of participants are linked to their contributions. In sub-figure (a), where P=100, RR attacker detects around 40 up to 60 RR links out of 100 contributions received from 100 participant. Which means, on average, half of the contributions are linked to their successors. Whereas in sub-figure (b), where P=300, the number of RR links detected vary from 30 to 90 out of 300 contributions (i.e. on average 20% of RR links detected). That is, the number of RR links detected depends on the number of participants P involved in the campaign. In our experiment, we have an average number between 20% and 50%. This means 35% of RR links are detected. Subsequently, a while after the system initiation, adversaries construct a profile of each participant containing their pseudonyms, contributions' contents, and reputation scores. That is privacy preserving attacks (e.g. [7],[1]) are easily adopted to re-identify the original identities. In the same figure, we can notice the links detected while PrivaSense is adopted. In both sub-figures (a) and (b), the PrivaSense links detected are so small. This is because the anonymization of reputation scores added to using anonymous certificate described in Section 4.2.2. PrivaSense links detected refer to the ones when the noise added is zero (i.e. $ida = 0$), since $ida \in [0, clk_{intr}]$. That is, the links detected can be removed if we do not include 0 as a member of the cloaking interval. That is $ida \in (0, clk_{intr}]$ since the zero as a noise conserves the original values and subsequently keeps the links.

Linkability. In this experiment, we measure the linkability metric as depicted in Figure 4. As the dataset includes a large number of participants, we only show the linkability for a subset of them, selected randomly, as well as the average values of this metric over all participants. We set clk_{intr} to 0.5 and 0.3 and show the results in Figure 4 a and b respectively.

From the results of RR attack, we concluded that adversaries construct a profile for each participant. Then, we attempt to measure the effect of PrivaSense to defend against this attack. In the results of our system depicted in Figure 4 (a), the average probability of a successful linkage is reduced to around 30% at the beginning of the sensing campaign with the first reputation update. As time progresses, the average probability continues to decrease and it reaches 6% at the end of the campaign. This is equivalent to a 94% average improvement. To explain the declining trend in Figure 4, we recall that the linkability technique works on location coordinates in successive time intervals. That is, if the adversary made a false link between contributions in Task t and $t+1$, the error in the spatial information would propagate and compound to that at $t+2$, which makes it increasingly difficult to track participants.

In Figure 4 (b), we have used a much lower value of cloaking range *clkintr*. We can see that the average linkability starts at 50% at the beginning of the sensing campaign and decreases to around 20% at the end of the campaign. This is equivalent to 80% average improvement. Comparing the results in sub-figure (b) with the one in (a), we can conclude that, using higher values for clk_{intr} in Sub-figure (a) allows for much lower values of the average linkability 30% at the beginning and it reaches 6% at the end. This leads to better anonymity.

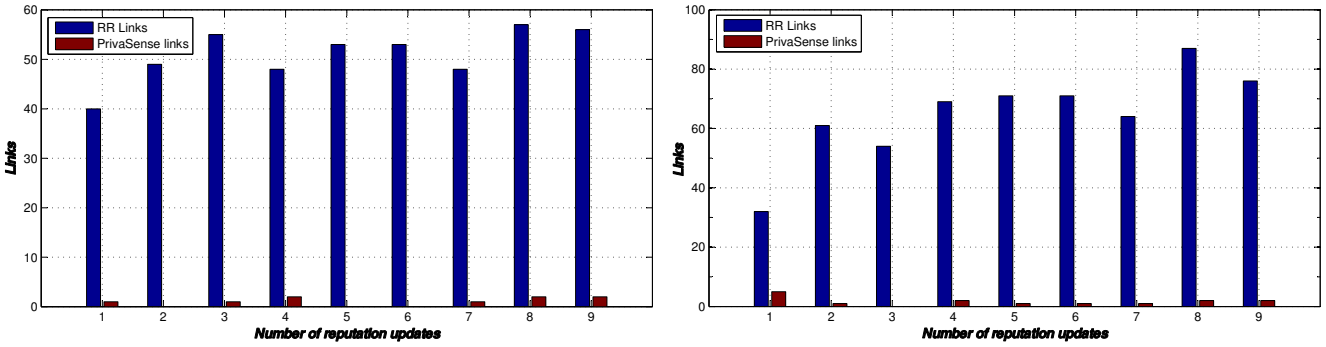
According to this experiment, it is evident that our reputation cloaking mechanism introduces better anonymity and unlinkability for the participants engaged in the sensing campaign with higher levels of cloaking. However, we should evaluate the effect of using such cloaked reputation scores on aggregating the collected data to define how much they deviate from aggregation based on the original reputation scores. That is we observe the accuracy of PrivaSense.

Accuracy. In the second experiment, we have used different values of clk_{intr} to cloak the reputation scores. Although, using large values for this parameter allows for better anonymity and unlinkability, as demonstrated in the previous experiment, it affects the accuracy of the aggregated data. Therefore, we consider a real world participatory sensing application as the model discussed above. Then, Figure 5 (a) depicts the weighted sum average of the aggregated contributions. Where $R - avg$ is the weighted sum average of the contributions using the original reputation scores as weights. $clk - 0.1$, $clk - 0.3$, $clk - 0.5$, and $clk - 0.7$ are the weighted sum average of the contributions based on the cloaked reputation scores using different values for the cloaking interval (e.g. $clk_{intr} \in \{0.1, 0.3, 0.5, 0.7\}$). The figure also includes the average of the aggregated data calculated without incorporating any additional weights noted as $N - avg$.

It is clear from figure 5 (a) that, normal average $N - avg$ calculated without adding any weights is far from the ground truth while the reputation based average $R - avg$ is much closer to the ground truth. It is evident that incorporating reputation scores in data aggregation gives better insights about the ground truth. In addition, it is clear from the figure that each of the averages calculated based on the cloaking intervals of $clk - 0.1$, $clk - 0.3$, $clk - 0.5$, and $clk - 0.7$ do not deviate significantly from average calculated based on the original reputation values $R - avg$. That is our cloaking does not disrupt the aggregated data significantly.

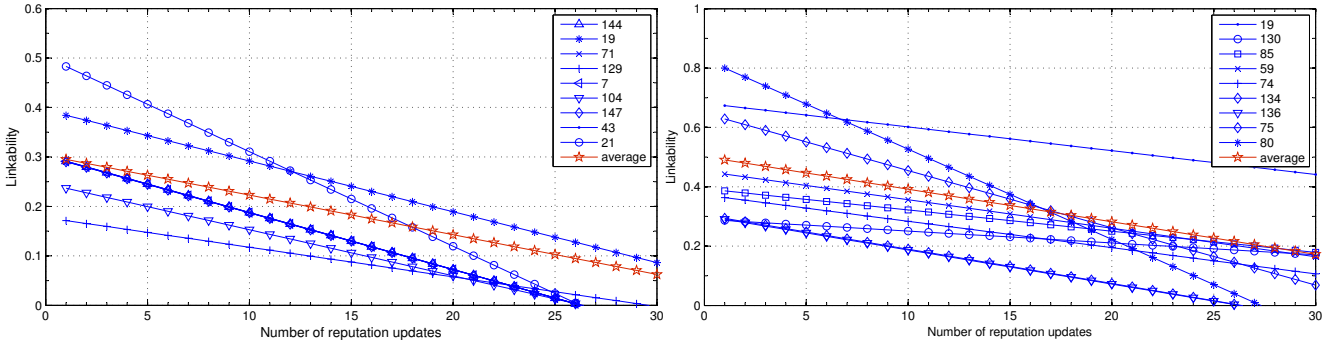
To better quantify the distortion occurred due to the incorporation of our reputation cloaking mechanism, we measure the RMSE for the same experiment and depict it in Figure 5 (b). It is evident that the RMSE calculated according to the actual values of reputation scores, $R - avg$, has usually less values of RMSE compared with those calculated based on cloaked reputation. In addition, using less values of cloaking intervals clk_{intr} ensures RMSE values which are closer to that is calculated according to the actual reputation scores (i.e. $RMSE(R-avg) < RMSE(clk-0.1) < RMSE(clk-0.3) < RMSE(clk-0.5) < RMSE(clk-0.7)$). That is cloaking based on less values of cloaking interval clk_{intr} allows for aggregating data closer to the ground truth with less RMSE.

6.2.2 Comparisons. We compare our proposal against previous work which intended to ensure the same objectives as



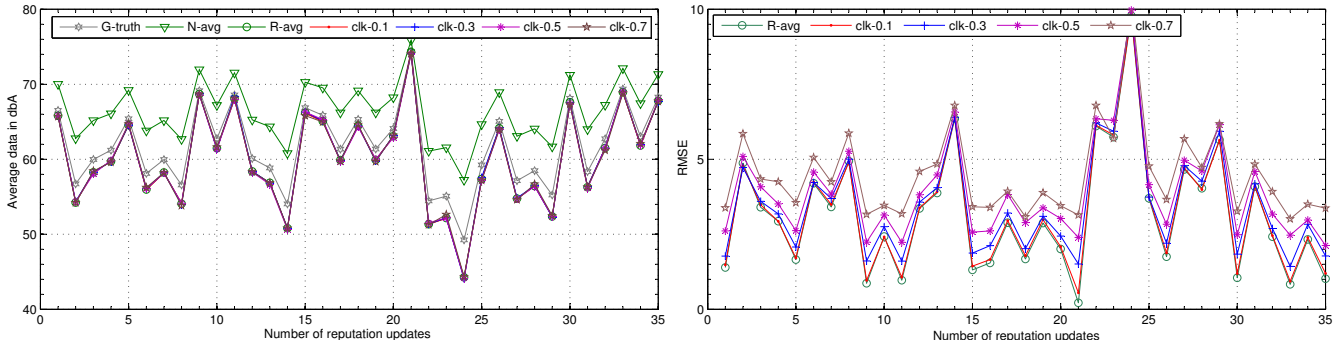
(a) $P = 100$ **(b)** $P = 300$

Figure 3: The number of links detected based on RR attack and PrivaSense



(a) $clk_{intr} = 0.5$ **(b)** $clk_{intr} = 0.3$

Figure 4: The linkabiliy for randomly selected participants



(a) Aggregated Data **(b) RMSE**

Figure 5: The effect of cloaking interval's size on the aggregated data and RMSE

the system presented in [8]. We compare the RMSE of the aggregated data based on Huang et al. in [8] with RMSE of PrivaSense. In Figure 6, the RMSE of PrivaSense has less values compared with the previous work by Huang et al. This ensures that the aggregated data according to our proposal are more accurate.

To summarize PrivaSense introduces better anonymity through better un-linkability. In addition, the aggregated data are more accurate compared with the previous work. Moreover, all the overhead for generating pseudonyms and cloaking reputation scores has been transferred to the Auth.Server and Rep.Server. Whereas in the literature, participants are

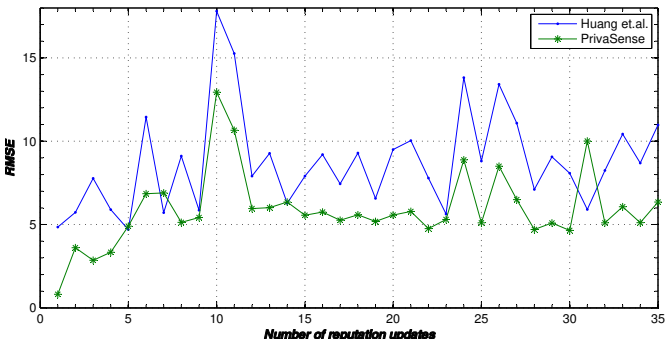


Figure 6: The effect of cloaked reputation scores in different systems

usually engaged with other entities in order to generate their pseudonyms. This ensures that in PrivaSense, participants' devices are much less overloaded with cryptographic and computational tasks which may lead to battery drain. We consider implementing a real world application and measuring the battery drain as a future work.

7 CONCLUSION

In this paper, We define a new attack in participatory sensing environments. This attack enables an attacker to link multiple contributions to the same identity. The results obtained clarify that 35% of contributions are linked to their successors in each campaign. Then, we propose a privacy preserving reputation system PrivaSense for participatory sensing applications. Our system adopts a registration and an authentication phases that ensure participants' anonymity and improve the system resilience against the Sybil and replay attacks. In addition, a privacy preserving mechanism is adopted for the contents of the participants' contributions which prevents adversaries from using the data to infer the identity of participants. Moreover, data reliability is ensured due to the incorporation of a reputation system. Finally, PrivaSense adopts a mechanism to cloak the reputation scores of participants. That is, the participants can not be linked to their contributions according to their assigned reputation scores. The discussion and the results obtained based on a real dataset demonstrate that the PrivaSense system ensures better anonymity and un-linkability with a ratio that reaches about 80%, with much low mean square error introduced to the aggregated data. We consider ensuring the same objectives within a trustless system model, (i.e. avoid relying on the trusted entities such Rep.Server and Auth.Server), as a future work.

REFERENCES

[1] Antoine Boutet, Sonia Ben Mokhtar, and Vincent Primault. 2016. *Uniqueness Assessment of Human Mobility on Multi-Sensor Datasets.* Research Report. LIRIS UMR CNRS 5205. https://hal.archives-ouvertes.fr/hal-01381986

[2] J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M. B. Srivastava. 2006. Participatory sensing. In *Workshop on World-Sensor-Web (WSW 06): Mobile Device Centric Sensor Networks and Applications*. 117–134.

[3] Delphine Christin. 2015. Privacy in mobile participatory sensing: Current trends and future challenges. *Journal of Systems and Software* (2015).

[4] Delphine Christin, Andreas Reinhardt, Salil S. Kanhere, and Matthias Hollick. 2011. A survey on privacy in mobile participatory sensing applications. *Journal of Systems and Software* 84, 11 (2011), 1928–1946.

[5] D. Christin, C. Rosskopf, M. Hollick, L.A Martucci, and S.S. Kanhere. 2012. IncogniSense: An anonymity-preserving reputation framework for participatory sensing applications. In *IEEE International Conference on Pervasive Computing and Communications (PerCom)*. 135–143.

[6] Marisol Concha-Barrientos, Diarmid Campbell-Lendrum, Kyle Steenland, Annette Prüss-Üstün, Carlos Corvalán, and Alistair Woodward. 2004. Occupational noise. *Assessing the* 7 (2004).

[7] Yves-Alexandre De Montjoye, César A Hidalgo, Michel Verleysen, and Vincent D Blondel. 2013. Unique in the crowd: The privacy bounds of human mobility. *Scientific reports* 3 (2013), 1376.

[8] Kuan Lun Huang, Salil S. Kanhere, and Wen Hu. 2012. A Privacy-preserving Reputation System for Participatory Sensing. In *Proceedings of the 2012 IEEE 37th Conference on Local Computer Networks (LCN 2012) (LCN '12)*. IEEE Computer Society, Washington, DC, USA, 10–18.

[9] Wei Jiang, Feng Li, Dan Lin, and Elisa Bertino. 2017. No one can track you: Randomized authentication in Vehicular Ad-hoc Networks. In *Pervasive Computing and Communications (PerCom), 2017 IEEE International Conference on*. IEEE, 197–206.

[10] Apu Kapadia, David Kotz, and Nikos Triandopoulos. 2009. Opportunistic Sensing: Security Challenges for the New Paradigm. In *Proceedings of the First International Conference on COMmunication Systems And NETworks (COMSNETS'09)*. IEEE Press, Piscataway, NJ, USA, 127–136.

[11] W.Z. Khan, Yang Xiang, M.Y. Aalsalem, and Q. Arshad. 2013. Mobile Phone Sensing Systems: A Survey. *Communications Surveys Tutorials, IEEE* 15, 1 (First 2013), 402–427.

[12] Nicholas D. Lane, Emiliano Miluzzo, Hong Lu, Daniel Peebles, Tanzeem Choudhury, and Andrew T. Campbell. 2010. A Survey of Mobile Phone Sensing. *IEEE Communications Magazine* 48, 9 (Sept. 2010), 140–150.

[13] Antonis Michalas and Nikos Komninos. 2014. The lord of the sense: A privacy preserving reputation system for participatory sensing applications. In *Computers and Communication (ISCC), 2014 IEEE Symposium on*. IEEE, 1–6.

[14] Hayam Mousa, Sonia Ben Mokhtar, Omar Hasan, Osama Younes, Mohiy Hadhoud, and Lionel Brunie. 2015. Trust Management and Reputation Systems in Mobile Participatory Sensing Applications. *Computer Networks* 90, C (Oct. 2015), 49–73. https://doi.org/10.1016/j.comnet.2015.07.011

[15] Hayam Mousa, Sonia Ben Mokhtar, Omar Hasan, Osama Younes, Mohiy Hadhoud, and Lionel Brunie. 2017. A reputation system Resilient against Malicious and Colluding adversaries in particiatory sensing applications. In *Proceeding of CCNC 2017*. IEEE CCNC.

[16] Michal Piorkowski, Natasa Sarafijanovic-Djukic, and Matthias Grossglauser. 2009. CRAWDAD dataset epfl/mobility (v. 2009-02-24). Downloaded from http://crawdad.org/epfl/mobility/20090224. (Feb. 2009). https://doi.org/10.15783/C7J010

[17] Rajib Kumar Rana, Chun Tung Chou, Salil S Kanhere, Nirupama Bulusu, and Wen Hu. 2010. Ear-Phone: An end-to-end participatory urban noise mapping system. In *Proceeding of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN10)*. Stockholm, Sweden, 105–116.

[18] Xinlei Oscar Wang, Wei Cheng, Prasant Mohapatra, and Tarek Abdelzaher. 2014. Enabling Reputation and Trust in Privacy-Preserving Mobile Sensing. *IEEE Transactions on Mobile Computing* 99, PrePrints (2014), 1.