Pseudonym-based Privacy Protection Scheme for Participatory Sensing with Incentives

Junsong Zhang*, Lei He, Qikun Zhang and Yong Gan
School of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou, 450002, China
[e-mail: zhangjs2002@gmail.com]
*Corresponding author: Junsong Zhang

Received December 6, 2015; revised March 20, 2016; revised September 20, 2016; accepted October 15, 2016; published November 30, 2016

Abstract

Participatory sensing applications rely on recruiting appropriate participants to share their surrounding conditions with others, and have been widely used in many areas like environmental monitoring, health care, and traffic congestion monitoring, etc. In such applications, how to ensure the privacy of a participant is important, since incentive mechanisms are used to maintain their enthusiasm for sustainable participation by offering certain amount of reward. In this paper, we propose a pseudonym-based privacy protection scheme, that takes both privacy protection and user incentives into consideration. The proposed scheme uses the pseudonym mechanism and one-way hash function to achieve user incentives, while protecting their identity. We also show extensive analysis of the proposed scheme to demonstrate that it can meet the security and performance the requirement of a participatory sensing application.

Keywords: Participatory Sensing, Privacy Protection, Elliptic Curve Cryptography

The authors are grateful to the editor and anonymous reviewers for their valuable suggestions which improved this paper. This research was supported by the National Natural Science Foundation of China (Grant no. 61272038, 61572445), Henan scientific and technological research projects (no. 162102210217), key scientific research projects of colleges and universities in Henan province, (no. 16A520075).

http://dx.doi.org/10.3837/tiis.2016.11.025 **ISSN : 1976-7277**

1. Introduction

In recent years, smartphones have gained rapid development with more processing power, bigger screens and higher resolutions, more accurate sensors, higher-resolution cameras, huge storage capacities and higher network data rates. The embedded sensors can provide advanced sensing capabilities that make it possible to capture real-time information of the user. These technological features have contributed to the emergence of a new paradigm, known as "participatory sensing" [1], whose essential idea is to collect and share the sensory data from a large population of such privately owned smartphones, and then provide valuable services to the end user and/or social groups.

A large amount of applications based on participatory sensing paradigm has emerged [2-8]. For example, noise signals taken by the acoustic sensor together with the GPS data can be used to locating noise pollution in a city [6]. CarTel [9] is a software platform that uses smartphones mounted on vehicles to collect information about traffic, and the quality and prevalence of Wi-Fi access points on drive routes. In addition, high level user activities, such as cyclist experiences, eating, jogging, etc., can be learned from accelerometer sensors and camera [4, 11-12].

The general system architecture and information flow supporting a participatory sensing application is shown in **Fig. 1**, including a number of participants, a registration center (RC), and a number of application servers. Participants are responsible for collecting the sensory data from their surrounding areas, by using the smartphones, and then submitting the required data to the corresponding application server. The major functionality of an application server is to receive the data from the participants, and generate corresponding application for the end user/consumers. The RC is responsible for registration and certification about the whole participatory sensing system. In it, every entity (including user and application server) who wants to access the system must be first registered at the RC. When a sensing task, or simply task, needs to be published, the RC is responsible for verifying the legitimacy of both application servers and the participants.

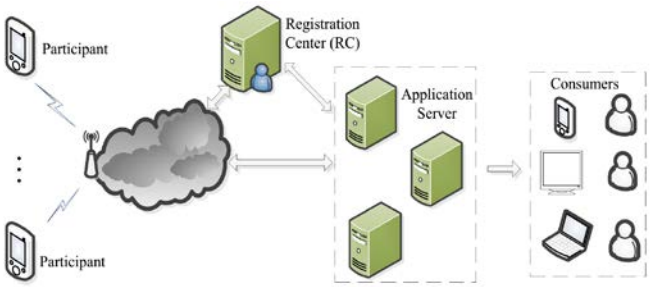

Fig. 1. The system architecture and system flow considered in this paper

From the system perspective, a number of challenges exist that should be addressed properly. First is the privacy protection of the participants constantly moving in a region. Due to its inherent open nature, messages submitted by participants are easy to be intercepted by an adversary. Since the sensory data may contain the user's location and other sensitive information, users are reluctant to deliver these data to the untrusted entities, e.g., service provider, individuals, etc. That is, users must encrypt their sensory data first, so that only the correct application server can decrypt it. Besides, in some situations, the service provider

needs to authenticate the participants. Take the long-term chronical disease monitoring like high blood pressure application for example, if the medical sensor data is forged or tampered by an adversary or malicious users, the diagnosis based on these forged or tampered data will cause serious damage to patients' health. Therefore, we need to carefully ensure the legitimacy of the participants.

In order to maintain the participants' enthusiasm and sustainability, there should be a relevant incentive or reward mechanism in a participatory sensing system [13, 14]. To implement such mechanisms, the system needs the user's identity information to pay the reward. However, as mentioned earlier, users are reluctant to divulge their identity information. Therefore, one major challenge is how to protect the privacy of the user while meeting the demand by incentive schemes.

To address these challenges, in this paper, we propose a pseudonym-based privacy protection scheme that takes both privacy protection and user incentive into consideration. Our contributions can be summarized as follows:

- We propose a privacy protection scheme using elliptic curve cryptography (ECC) to encrypt the sensory data generated by users' smart phones.
- We adopt a pseudonym and one-way hash function based approach which allows the participants to get reward certificate without leaking their identity.
- We propose a privacy protection scheme with the following features. First, it can verify the legitimacy of participants, thus guarantee the legality of the sensory data generated by the users' smartphones. Second, it can break the linkage between the identity of the participants and the sensing data sent from these participants.
- We extensively analyze the security and performance of the proposed scheme, and demonstrate that the proposed scheme can meet the security and performance requirements of participatory sensing applications.

The rest of the paper is organized as follows. In Section 2, we describe the related research activities on privacy protection mechanisms for participatory sensing applications. In Section 3, we describe the system architecture and the attack model used in this paper. Then, we give the basic assumptions and preliminaries in Section 4. In Section 5, we propose a pseudonym-based privacy protection scheme for participatory sensing application with incentives. In Section 6, we discuss the security and performance of our scheme. Finally, our conclusion and future work are given in Section 7.

2. Related Work

Many research activities have been done [2, 3, 4, 5] to propose emerging participatory sensing applications. However, since the sensory data usually contain the time and location information, it may inadvertently reveal the user's private information at a particular time. This would lead to the fact that users are reluctant to contribute the data to the system once they are aware of leaking their privacy. Depending on different application scenarios, several aspects, including location or personal identity information, need to be protected in a proper way.

Several privacy protection schemes for participatory sensing have been proposed [10,15-18] . Christin *et al*. in [20] surveyed the state-of-the-art privacy and security measures in participatory sensing applications and identified the future research directions that need to be address. Kapadia *et al*. in [21] proposed a privacy-aware architecture, named "AnonySense", to realize the anonymous task allocation and data reporting. By replacing the precise position with the identifier of an area, the adversary cannot distinguish and localize the specific user

who sends the sensing data. Gao *et al*. in [16] proposed a trajectory privacy-preserving framework for participatory sensing. Cristofaro *et al*. in [17] analyzed the privacy requirement for both data producers and consumers, and then proposed a privacy-preserving participatory sensing infrastructure and introduce an efficient cryptographic solution to protect users' privacy. Xing *et al*. in [18] proposed a series of privacy preserving, data gathering, and model fitting designs for mutual privacy preservation participatory sensing. Cristofaro and Soriente in [19] proposed a privacy-enhanced participatory sensing infrastructure and two instantiations, that attain privacy guarantees with provable security.

Meanwhile, there are several research activities focusing on the location privacy-preserving for participatory sensing [15, 22-25]. K-Anonymity, proposed by Sweeney in [26], is a popular technique for this purpose, where the user's location is blurred in a cloaked area that contains at least $K-1$ other users. The frameworks used in [15] and [16] are based on Kanonymity technique. Nevertheless, since the application server cannot distinguish the user from $K-1$ other users, which leads to difficulties in the implementation of incentive mechanisms. Kazemi and Shahabi in [23] proposed a privacy-aware framework for participatory sensing system that enables participation of the users without compromising their privacy. Boutsis and Kalogeraki in [27] proposed an efficient approach for privacy preservation that enables the participants to disclose their trajectory without compromising their privacy. Vu *et al*. in [24] proposed a privacy protection mechanism based on locality-sensitive hashing, to prevent the disclosure of personal data. Oscar *et al*. in [25] proposed a framework to solve the problem of "trust without identity" in participatory sensing networks.

Pseudonym method, which is widely used for breaking the linkage between a user's identity and his/her behavior, is another category of approaches to protect location privacy in mobile applications. Christin *et al*. in [28] proposed an anonymity-preserving framework based on periodic pseudonyms to protect users' privacy. Li *et al*. in [29] proposed a pseudonym based authenticated key establishment scheme with privacy preservation to secure the communications between mobile vehicles and roadside infrastructure in vehicular ad hoc networks.

There are also some other research activities [13-15] contributed to the incentive schemes of a participatory sensing application. Among many others, two representative are as follows. Duan *et al*. in [13] analyzed and compares different incentive mechanisms to motivate the collaboration of smartphone users on both data acquisition and distributed computing applications. Lee *et al*. in [14] designed a reverse auction based dynamic price incentive mechanism, where participants can sell their sensing reports to a service provider with participants' claimed bid prices. Nevertheless, neither of them consider the protection of user privacy during the incentive negotiation process.

Therefore, although plenty of research efforts have been paid to investigate privacy protection or incentive allocation for participatory sensing, only a few of them explicitly consider both the privacy protection and user incentives at the same time. Li and Cao in [15] proposed two privacy-aware incentive schemes for mobile sensing to promote user participation, but their schemes still do not consider the legitimacy of the participants and the application servers. Towards this end, in this paper we propose a novel pseudonym-based privacy protection scheme to fill this research gap.

3. System Model and Assumptions

In this section, we first describe our formal system model and the threats model. Then, we make some basic assumptions.

3.1 System Model

This section presents a formal system model for describing our scheme based on **Fig. 1**. The system mainly consists of a dynamic set of M participants, denoted as $U = \{u_i \mid i = 1, 2, \ldots, M\}$, a set of N application servers as $S = \{s_j \mid j = 1, 2, \ldots, N\}$ and a registration center (RC). The basic workflow of our scheme is as follows. When the application server s_j, $\forall j \in N$ needs some type of sensory data, it submits a task, denoted as τ, to the RC. After verifying the legitimacy of the application server s_j, RC publishes the task τ to all participants to collect the required data. The task needs to explicitly specify what sensory readings to report, and when and where to sense. After receiving a task, the participant u_i, $\forall i \in M$ decides whether to accept the task or not, based on certain criteria. If he/she accepts the task, he/she will collect his/her terminal's sensor data at the time and location specified by the task τ, and generates one sensing report and then submits it to the application server s_j based on a pseudonym. Then, the application server s_j issues a pseudonym-based reward certificate to the participant. When the participant receives a reward certificate, he/she transforms them into a credit token (as a 5-tuple, details in Section 5). Later the participant could redeem each credit token at RC.

The amount of reward paid for different sensing reports may be different. It depends on the type, accuracy and location of the provided sensory readings. Let $G = \{G_1, G_2, \ldots, G_n\}$ denote different reward "grades" of the sensory data. The number of grades n is set by the RC. Here the reward grade is determined by the type of sensing data and the amount of effort needed to collect the sensing data. For example, task TA_1 needs the participants to submit a high-definition photograph as sensory data which requires user intervention and much network traffic; task TA_2 only needs the participants to submit an accelerator reading that can be obtained without human intervention, and thus does not have much communication cost. In this example, the reward grade of task TA_1 is higher than that of TA_2. Definitely, the higher reward grade means more incentive requirement as a return. The application server calculates the reward certificate using a message authentication code (MAC) algorithm. The details of the proposed scheme is described in Section 5. **Table 1** shows the list of important notations used in this paper.

Table 1. List of important notations used in this paper

Notation	Description
u_i	The ith user
pid_i	The identity of u_i
pw_i	The password of u_i
s_j	The jth application server
sid_j	The identity of the jth application server
$H(\cdot)$	A hash function, where $H : E_p(a, b) \Rightarrow \{0, 1\}^l$, l is the length of the string
$h(\cdot)$	A secure hash function, where $h : \{0, 1\}^* \Rightarrow Z_q^*$
$H_1(\cdot)$	A map-to-point hash function, where $H_1 : \{0, 1\}^* \Rightarrow E_p(a, b)$
G_x	The grades of reward
$\hat{e}(P, Q)$	A bilinear map $G \times G \rightarrow G_T$
$f_k(\cdot)$	A message authentication code algorithm, where k is an encryption key

c_{ij}	The sensing data collected by the participant u_i and will be sent to the application server s_j
ρ_i	The pseudonym generated by u_i
σ_{ji}	The participant u_i's reward certificate generated by s_j

3.2 Threats Model

Due to the error-prone and open nature of wireless channel, a participatory sensing system is vulnerable to various attacks such as interception, eavesdropping, and alteration of interchanged messages. The primary goal of our proposed scheme in this paper is to provide two key security properties, namely: (a) to protect the user's anonymity, and (b) to support the successful and easy implementation of applied incentive mechanisms. In this section, we consider the possible threats related to our goals.

3.2.1 Threats to Anonymity

An adversary seeks to breach the anonymity of the participants. The adversary *A* may eavesdrop on the communication links among the users, *RC* and application servers. *A* may also attempt to establish the relationships between successive pseudonyms and link these pseudonyms to a unique real entity.

3.2.2 Threats to Implement the Incentive Mechanisms

The adversary may attempt to impersonate a legal participant to deliver bogus sensing reports to the application server. He/she may also intercept and replay old sensing reports to obtain illegal benefits. We assume that the adversary may tamper or forge the sensing reports sent from legal participants. In addition, the adversaries mentioned in this paper also include malicious users and servers trying to obtain illegal benefits.

3.3 Assumptions

Without loss of generality, we assume that the application server and RC are protected against fraudulent access by well-established security mechanisms (including system and physical securities). Therefore, the adversaries are not able to access these servers, or change the behavior of such applications. Besides, we assume that users are able to keep their smartphones very well, so that these devices will not be lost or stolen. In other words, such situation that user's smartphone is lost or stolen is not in the scope of our considerations.

Using pseudonyms technology is useless if a participant can be tracked by using his/her smartphone's MAC or IP address. Thus, we assume that the device has the ability to change its MAC and IP addresses through the techniques proposed in [30, 31]. Therefore, the adversary cannot track the participant by using a participant's MAC or IP address.

4. Preliminaries

In the proposed scheme, we use some cryptographic solutions to protect the privacy of users and guarantee the successful implementation of incentives. Such techniques we used are: one-way hash function, message authentication code (MAC) algorithm, and elliptic curve cryptography (ECC), that will be described in detail in the following sections.

4.1 Hash Function

One-way hash function [32] is an algorithm that takes an arbitrary block of data and returns a fixed-size bit string, and it is easy to compute on every input but hard to compute the input from a given output. The one-way hash function has the following properties.

- The hash function can be applied to a data block of all sizes.
- For any given input x, it is easy to compute the output.
- It is infeasible to deriving x from the given value $y = h(x)$.
- It is infeasible to find two different inputs with the same output.
- It is infeasible to modify an input without changing the output.

In the proposed scheme, two types of secure hash function are used: $h(\cdot)$ and $H(\cdot)$, where $h(\cdot) : \{0, 1\}^* \rightarrow Z_q^*$, and $H(\cdot) : E_P\ (a, b) \rightarrow \{0, 1\}^l$, l denotes the length of the string.

4.2 Elliptic Curve Cryptosystem (ECC)

ECC was first proposed by Miller and Koblitz in 1985 [33]. The security of ECC is based on the difficulty of solving the elliptic curve discrete logarithm problem (ECDLP). Compared with RSA, it is able to achieve the same security level with shorter key length [34].

Let p be a large prime number, and let $GF(p)$ be the field of integers modulo p. An elliptic curve E over $GF(p)$ is defined by an equation of the following form:

$$y^2 = x^3 + ax + b, \tag{1}$$

where $a, b \in GF(p)$, and they satisfy $4a^3 + 27b^2 \neq 0 \pmod p$. A pair $P(x, y)$ is a point on the curve, if $P(x, y)$ satisfies (1), where $x, y \in GF(p)$. The point at infinity, denoted as ∞, is also on the curve. The set of all points on the curve E is denoted by $E_p(a, b)$, or simply: E. The point multiplication over E can be computed by repeated addition as,

$$k \cdot P = P + P + \cdots + P\ (k \text{ times}), \tag{2}$$

where k is a constant integer and P is a point on the curve E. For more information about ECC, please refer to [33, 34] for details.

In order to prove the security of our proposed protocol, here we present two important mathematical problems on elliptic curves as follows.

- Elliptic Curve Discrete Logarithm Problem (ECDLP): given an elliptic curve E defined over a finite field $GF(p)$, and two points $Q, P \in E$ of order q, it is hard to find an integer $k \in Z_q^*$ such that $Q = k \cdot P$.
- Computational Diffie-Hellman Problem (CDHP): given an elliptic curve E defined over a finite field $GF(p)$, $P, aP, bP \in E$, it is hard to compute abP.

4.4 Bilinear Pairings

Let $\mathbb{G}$ be a cyclic additive group generated by a point P, and $\mathbb{G}_T$ be a cyclic multiplicative group. $\mathbb{G}$ and $\mathbb{G}_T$ have the same primer order q. Also, let $\hat{e} : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ be a computable bilinear map, which satisfies the following properties.

- Bilinearity: For any $P, Q \in \mathbb{G}$ and $a, b \in Z_q^*$, $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$, here $Z_q^* = \{j | 1 \leqslant j \leqslant q - 1\}$. This can be restated in the following way, for any $P, Q \in \mathbb{G}$ and $a \in Z_q^*$, $\hat{e}(aP, Q) = \hat{e}(P, aQ) = \hat{e}(P, Q)^a$.
- Non-degenerate: For any $P \in \mathbb{G}$, $\hat{e}(P, P) \neq e$, where e is the identity element of the group $\mathbb{G}_T$.

- Computability: There exists an efficient algorithm to compute $\hat{e}(P, Q)$ for any $P, Q \in \mathbb{G}$.

5. Proposed Scheme

In this section, we describe the details of our proposed pseudonym based privacy protection scheme. It contains six phases: (a) system initialization phase, (b) registration phase, (c) task issuing and sensing report update phase, (d) the authentication and reward certificate generation phase, (e) reward redemption phase, and (f) the password change phase. The detailed descriptions of these six phases are explained as follows.

5.1 Phase 1: System Initialization

As mentioned earlier, in the proposed scheme, *RC* is considered as a trusted third party and a reward redeeming institution for application servers and participants. In this phase, *RC* sets the required parameters used for all participatory sensing applications.

- **Step 1-1**: *RC* selects an appropriate elliptic curve E over finite field $GF(p)$. Then, *RC* chooses a base point P based on the elliptic curve E with the order q.
- **Step 1-2**: *RC* chooses a random number $r_s \in Z_q^*$ as the private key, and computes the corresponding public key $P_{pub} = r_s \cdot P$. Then, *RC* selects two random numbers $r_1, r_2 \in Z_q^*$ as its secret values.
- **Step 1-3**: *RC* chooses a MAC algorithm $f_k(\cdot)$, where k is the encryption key.
- **Step 1-4**: *RC* selects two secure hash functions $h(\cdot)$ and $H(\cdot)$, where $h(\cdot) : \{0, 1\}^* \rightarrow Z_q^*$, $H(\cdot) : E_p(a, b) \rightarrow \{0, 1\}^l$, l is the length of the string.
- **Step 1-5**: *RC* determines the reward grade of the sensing data $G_x \in \{G_1, G_2, \ldots, G_n\}$. The reward grade is determined by both the type of sensing data, and the amount of effort required to collect and submit to the system.
- **Step 1-6**: *RC* publishes the public system parameters $\{E, p, P, P_{pub}, h(\cdot), H(\cdot)\}$ and keeps the secret values r_1, r_2 and r_s secretly.

5.2 Phase 2: Registration

The registration phase can be divided into two parts: (a) the application server registration, and (b) the user registration.

5.2.1 Application Server Registration (SR)

When an application server s_j wants to provide participatory sensing applications, it must be registered by RC.

- **Step SR1**: s_j sends its identity sid_j to RC.
- **Step SR2**: RC computes $h(sid_j \| r_2)$, $h(sidj \oplus r_2)$, and sends the message $\{h(sid_j \| r_2), h(sid_j \oplus r_2), G_x, f_k(\cdot)\}$ to the application server s_j via a secure channel.
- **Step SR3**: When receiving the message $\{h(sid_j \| r_2), h(sid_j \oplus r_2), G_x, f_k(\cdot)\}$, s_j selects its private key r_j and computes its public key $P_j = r_j \cdot P$. Then, s_j keeps the parameters $\{h(sid_j \| r_2), h(sid_j \oplus r_2), G_x, f_k(\cdot), r_i\}$ secretly.

After the application server s_j is registered, it has the unique authentication certificates $h(sid_j \| r_2)$ and $h(sid_j \oplus r_2)$. Furthermore, it obtains the set of reward grades G and the MAC

algorithm $f_k(\cdot)$. Then, s_j is able to take part in the participatory sensing system and provide relevant services.

5.2.2 User Registration (UR)

Before a user u_i participates in a participatory sensing application, he/she must be registered at the RC as the following steps.

- **Step UR-1**: The participant u_i freely chooses his/her identity pidi and password pw_i, and selects a random number r_u. Then, u_i's smartphone computes $h(r_u \oplus pw_i)$ and sends the registration request $\{pid_i, h(r_u \oplus pw_i)\}$ to RC via a secure channel.
- **Step UR-2**: Upon receiving the message $\{pid_i, h(r_u \oplus pw_i)\}$, RC computes $A_i = h(pid_i \| r_1)$, $H_i = h(A_i)$, $V_i = A_i \oplus h(pid_i \| h(r_u \oplus pw_i))$, $B_i = h(pid_i \| h(r_u \oplus pw_i) \| r_1)$, (3)
  where H_i, V_i are used to verify the legitimacy of the participant by the smart card, and B_i is used to verify the legitimacy of the participant by RC.
- **Step UR-3**: RC issues a smart card, that contains the authentication information $\{H_i, V_i, B_i\}$. Then, RC sends the smart card to the participant u_i.
- **Step UR-4**: Upon receiving the smart card, the participant u_i inputs the random number r_u to the smart card. Then, the smart card contains the values $\{H_i, V_i, B_i, r_u\}$.

Upon finishing the user registration, the participant u_i obtains the authentication certificate B_i that is used for being verified in the sensing report update phase. Then, u_i becomes a qualified participant to collect and upload the sensing data according to the task requirements.

5.3 Phase 3: Task Issuing and Sensing Report Update

When an application server s_j plans to carry out a sensing task, it must apply to the RC for distributing the corresponding sensing task to all participants. The detailed work flow of this phase is described as follows.

- **Step 3-1**: The application server s_j generates a timestamp t_s, and then computes: $H(r_j \cdot P_{pub})$, $Q_j = h(h(sid_j \| r_2) \| t_s) \oplus H(r_j \cdot P_{pub})$, $\forall j \in S$, (4)
  where Q_j is used to verify the legitimacy of the application server by RC, τ_s is the type of task, sid_j is the identity of s_j and P_j is s_j's public key. Next, s_j sends the task request message $\{\tau_s, sid_j, P_j, Q_j, t_s\}$ to RC for distributing the sensing task.
- **Step 3-2**: Upon receiving the message $\{\tau_s, sid_j, P_j, Q_j, t_s\}$, RC computes: $Q_j^* = h(h(sid_j \| r_2) \| t_s) \oplus H(r_s \cdot P_j)$ and checks whether $Q_j^* = Q_j$ satisfies. If the condition holds, RC believes that the task request is valid and computes the signature θ_j as: $\theta_j = r_s \cdot H_1(\tau_s \| sid_j \| t_s)$, and then it broadcasts the message $\{\tau_s, sid_j, P_j, \theta_j, t_s\}$ to the users in a specific area for sensing data collection.
- **Step 3-3**: Upon receiving the message $\{\tau_s, sid_j, P_j, \theta_j, t_s\}$, the participant u_i checks the validity of t_s, compute computes $P_{ch} = H_1(\tau s \| sid_j \| t_s)$ and then checks whether $\hat{e}(\theta_j, P) = \hat{e}(P_{ch}, P_{pub})$ satisfies. If positive, u_i believes that the sensing task is valid, and he/she opens corresponding sensing units, and start to collect the data according to the task requirements.
- **Step 3-4**: After finishing the data collection, u_i's smartphone generates the sensing data m_{ij} and the timestamp t_i. Then, u_i uses his/her identity pid_i and the timestamp t_i as the input of a random function to generate a random number ρ_i and computes the pseudonym $h(\rho_i)$. After, u_i computes:
  $H(\rho_i \cdot P_{pub})$, $P_\rho = \rho_i \cdot P$, $\forall i \in U$, and, $R_i = W_i \oplus H(\rho_i \cdot P_{pub})$, $\forall i \in U$, (5)

where $C_i = B_i \oplus h(sid_j \| t_i \| t_s)$ and $W_i = pid_i \| h(r_u \oplus pw_i) \| C_i$, and both C_i and R_i are used to verify the legitimacy of the participant by RC. Also, the user u_i computes:

$$c_{ij} = m_{ij} \oplus H(\rho_i \cdot P), \forall i \in U, \ j \in S, \tag{6}$$

where c_{ij} denotes the encrypted sensing data submitted from the participant u_i to the application server s_j.

- **Step 3-5**: u_i sends the message $\{sid_j, h(\rho_i), P_\rho, R_i, c_{ij}, t_i, t_s\}$ to the application server s_j via a common channel. Meanwhile, u_i keeps the pseudonym ρ_i secretly to prevent others from obtaining it.

As a summary, in this phase, the application server s_j issues the sensing task through RC when it is verified as legitimate. Upon receiving the sensing task, the participants collect the sensing data m_{ij} according to the requirements of the task τ_s. Then, the participants encrypt the sensing data mij and submit the sensing data with encrypted mode c_{ij}. Therefore, the sensing data cannot be obtained by other entities. In addition, the participant u_i used a pseudonym $h(\rho_i)$ as his/her temporary identity. Thus, others cannot obtain his/her real identity.

5.4 Phase 4: Authentication and Reward Certificate Generation

- **Step 4-1**: Upon receiving the message $\{sid_j, h(\rho_i), P_\rho, R_i, c_{ij}, t_i, t_s\}$, the application server s_j checks whether $t_{c1} - t_i \leqslant \Delta t$ satisfies, where t_{c1} is the current time and Δt is the expected time interval for the transmission delay. If the condition holds, s_j extracts the encrypted sensing data c_{ij}, the participant's pseudonym $h(\rho_i)$ and the corresponding public key P_ρ, and then sends the message $\{sid_j, h(\rho_i), P_\rho, R_i, t_i, t_s\}$ to RC via a common channel.
- **Step 4-2**: When RC receives the message $\{sid_j, h(\rho_i), P_\rho, R_i, t_i, t_s\}$, it checks whether $t_{c2} - t_i \leqslant \Delta t$ holds. If positive, RC computes $W_i^* = R_i \oplus H(r_s \cdot P_\rho)$ from (5), and then extracts pid_i, $h(r_u \oplus pw_i)$ and C_i from W_i^*.
- **Step 4-3**: RC computes

$$C_i^* = h(pid_i \| h(r_u \oplus pw_i) \| r_1) \oplus h(sid_j \| t_i \| t_s), \tag{7}$$

and checks whether $C_i^* = C_i$ holds. If they are equal, RC considers $h(\rho_i)$ as a legal user who has the right to take part in the considered participatory sensing application. Then, RC calculates

$$\delta = h(h(sid_j \oplus r_2) \| h(\rho_i) \| t_i \| t_s) \oplus H(r_s \cdot P_j), \tag{8}$$

and sends the message $\{h(\rho_i), \delta, t_i, t_s\}$ to s_j via a common channel. Otherwise, RC sends a message to s_j to indicate that $h(\rho_i)$ is an illegal user.

- **Step 4-4**: Upon receiving the message $\{h(\rho_i), \delta, t_i, t_s\}$, where δ is computed by equation (8), s_j calculates:

$$\delta^* = h(h(sid_j \oplus r_2) \| h(\rho_i) \| t_i \| t_s) \oplus H(r_j \cdot P_{pub}), \tag{9}$$

and it checks whether $\delta^* = \delta$. If they are equal, s_j believes $h(\rho_i)$ as a legal user.

- **Step 4-5**: s_j computes $m_{ij}^* = c_{ij} \oplus H(r_j \cdot P_\rho)$ and extracts the sensing data, where c_{ij} is calculated via equation (6). Then, the application server sets the grade of the sensing data according to its quality, accuracy, and type, etc. Utilizing the grade of sensing data, the application server is able to calculate a certificate based on the user's pseudonym $h(\rho_i)$ via equation (10) as:

$$\sigma_{ji} = f_{km}(G_i, h(\rho_i), h(sid_j \| r_2), t_i), \tag{10}$$

where $km = H(r_j \cdot P_{pub})$, $h(\rho_i)$ is u_i's pseudonym, $G_i \in G$ is the grade of sensing data and t_i is the timestamp. Then, s_j sends the message $\{h(\rho_i), \sigma_{ji}, P_j, t_i\}$ to u_i via a common channel.

- **Step 4-6**: Upon receiving the message $\{h(\rho_i), \sigma_{ji}, P_j, t_i\}$, u_i extracts the certificate σ_{ji} and stores this certificate with the pseudonym ρ_i, s_j's identity sid_j, s_j's public key P_j and time stamp t_i as a 5-tuple $<\rho_i, \sigma_{ji}, sid_j, P_j, t_i>$ into its memory.

The sequence diagram of the above steps is described in **Fig. 2**. It is worth noting that users must make sure of their smartphone safety. That is, the 5-tuple stored in users' smartphone cannot be extracted by others. In addition, the user is able to offer the sensing reports to the application server for many times. Therefore, a user may receive one or more reward certificates and generates the corresponding 5-tuple.

As a summary, the main objective of this phase is to verify the legitimacy of participants and generate the reward certificate for the participants. Since the reward certificate σ_{ji} is calculated by the pseudonym $h(\rho_i)$, the others cannot obtain the participants' identity information. Therefore, this scheme can protect participants' identity privacy.

5.5 Phase 5: Reward Redemption

When a user u_i wants to exchange the reward certificate stored in his/her smartphone, he/she can redeem the reward at the RC.

- **Step 5-1**: u_i sends the credit token, as a 5-tuple $<\rho_i, \sigma_{ji}, sid_j, P_j, t_i>$ to the RC via a secure channel.
- **Step 5-2**: After receiving credit token, RC computes the values $km = H(s \cdot P_j)$, $h(\rho_i)$.
- **Step 5-3**: RC computes $\sigma_x = f_{km}(G_x, h(\rho_i), h(sid_j \| r_2), t_i)$ by using each $G_x \in \{G_1, G_2, \ldots, G_n\}$. Next, RC compares σ_x with σ_{ji} extracted from the 5-tuple $<\rho_i, \sigma_{ji}, sid_j, P_j, t_i>$. If there is one σ_x that matches σ_{ji}, RC confirms that the certificate σ_{ji} is a valid certificate and the user who owns σ_{ji} can obtain a reward. Otherwise, RC confirms that the certificate σ_{ji} is invalid and does not give any incentive to the user who owns σ_{ji}.

As a summary, if a user u_i has multiple certificates that have not been cashed, he/she is able to exchange certificates repeatedly by using the above method. In order to avoid the problem that the malicious user exchanges the certificate repeatedly, RC must maintain a data table CTable that contains the information of cashed certificates. When receiving a 5-tuple $<\rho_i, \sigma_{ji}, sid_j, P_j, t_i>$ from a user u_i for exchanging rewards, RC uses database queries to determine whether the record $<\rho_i, \sigma_{ji}, sid_j, P_j, t_i>$ in the database CTable. If there is no record matching with $<\rho_i, \sigma_{ji}, sid_j, P_j, t_i>$, RC believes that the 5-tuple $<\rho_i, \sigma_{ji}, sid_j, P_j, t_i>$ has not been cashed. Then, RC redeems this certificate and then inserts the tuple $<\rho_i, \sigma_{ji}, sid_j, P_j, t_i>$ into CTable.

5.6 Phase 6: Password Change

In our proposed scheme, a user u_i can change his/her password any time when he/she wants, by using the following steps.

- **Step 6-1**: participant u_i inputs his/her pidi and pw_i in his/her smartphone.
- **Step 6-2**: the smartphone computes $A_i^* = V_i \oplus h(pid_i \| h(r_u \oplus pw_i))$, $H_i^* = h(A_i^*)$ and checks if H_i^* is the identical to H_i that is stored in the smartphone's memory. If the condition holds, the smart card enables u_i to choose a new password pw_{new} and a new random number r_u^*.

- **Step 6-3**: the smartphone computes $V_{new} = V_i \oplus h(pid_i \| h(r_u \oplus pw_i)) \oplus h(r_u^* \oplus pw_{new})$ and sends the message $\{pid_i, h(r_u^* \oplus pw_{new})\}$ to RC via a secure channel.
- **Step 6-4**: Upon receiving the message $\{pid_i, h(r_u^* \oplus pw_{new})\}$, RC computes $B_{new} = h(pid_i \| h(r_u^* \oplus pw_{new}) \| r_1)$ and sends back B_{new} to u_i via a secure channel.
- **Step 6-5**: participant u_i replaces V_i, B_i and r_u with V_{new}, B_{new} and r_u^*, respectively.

Discussion: our scheme's security mainly depends on the difficulty of solving the elliptic curve discrete logarithm problem (ECDLP). In the proposed scheme, we use the difficulty of solving Diffie-Hallman problem and the feature of hash function (i.e., collision resistance) to achieve the secure authentication between RC, the application server and the participants. In addition, the proposed scheme cleverly combined the pseudonym method and the MAC

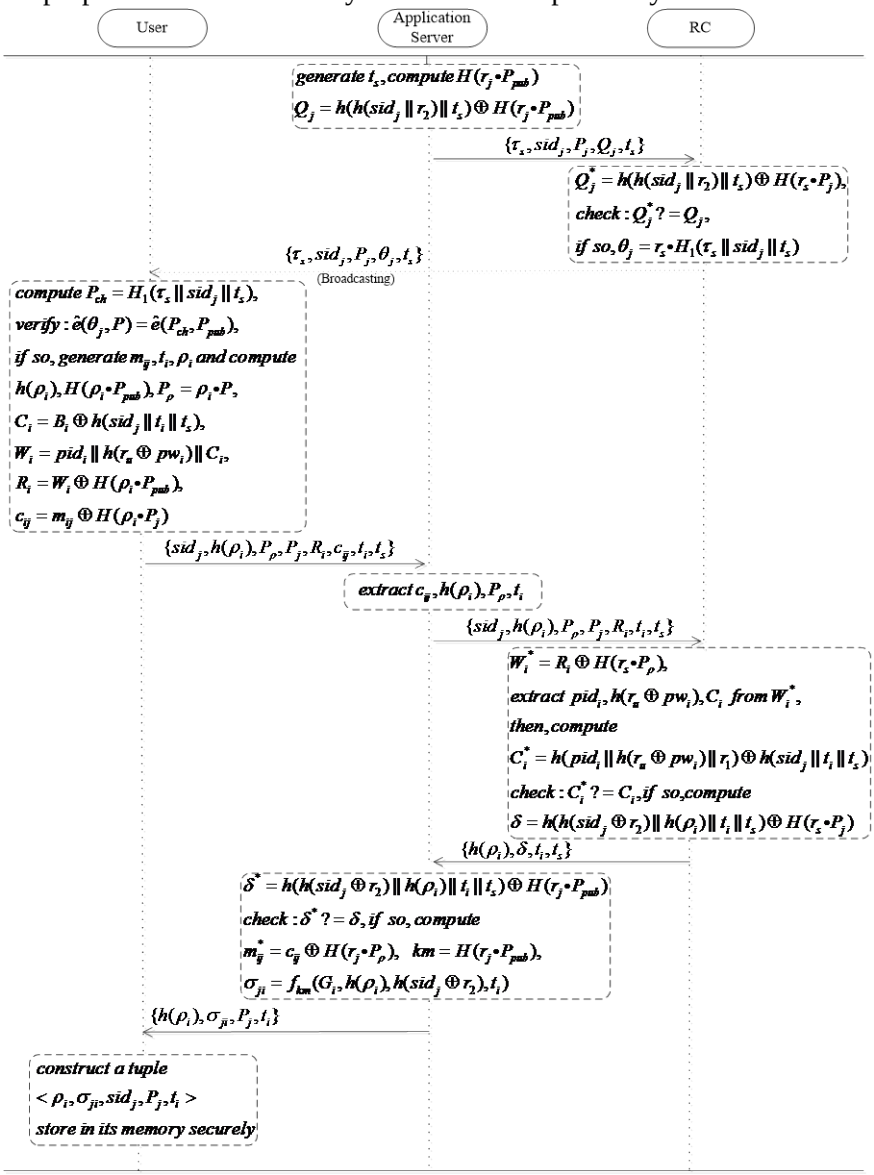

Fig. 2. The sequence diagram of the proposed scheme

algorithm to achieve the goals about privacy protection and incentive. It should be emphasized that other types of public-key encryption can also be applied in our scheme. However, compared with other public-key encryption schemes, ECC has significant advantages like smaller key sizes, and faster computations [34]. Therefore, ECC is chosen as the cryptographic solution in our proposed scheme.

6. Security Analysis and Performance Evaluation

6.1 Security Analysis

In this section, based on the assumptions and threat model presented in Section 3, we prove that the proposed scheme is resilient against the privacy, incentive and other types of attacks.

6.1.1 Attacks on Privacy

Lemma 1. The proposed scheme is able to achieve users' privacy protection.

Proof. In the proposed scheme, the encrypted sensing data c_{ij} sent from user u_i is done by xor the hash function $H(\rho_i \cdot P_j)$ based on (6). Except the application server s_j, other entities (including application servers, RC or users) cannot decrypt and identify the true value $H(r_j \cdot P_\rho)$ based on **Step 4-5**, and then extracts the realistic sensing data m_{ij}. Therefore, the proposed scheme is able to protect the users' sensing data privacy. On the other hand, each time a participant u_i sends a sensing data message $\{h(\rho_i), P_\rho, R_i, c_{ij}, t_i\}$, the pseudonym $h(\rho_i)$ and the value R_i are used to replace the real user identity.According to the characteristics of Computational Diffie-Hellman Problem [36], no information about pid_i can be extracted from R_i by the application server s_j or other illegal entities. Therefore, except RC, other entities are not able to obtain the participant's identity information. □

Lemma 2. The application server cannot link multiple sensing reports as originating from the same user.

Proof. In the proposed scheme, when a user u_i aims to deliver the sensing data to the application server, u_i's smartphone generates a random number ρ_i and computes the pseudonym $h(\rho_i)$. For every sensing report delivered to the application server, it contains a unique pseudonym $h(\rho_i)$, without direct relationship with the u_i's identity. Therefore, the application server or other entities cannot link multiple sensing reports as originating from the same user.

6.1.2 Attacks on Incentives

Lemma 3. The proposed scheme has the ability to resist against replay attack.

Proof. Malicious users may attempt to replay the old sensing reports to obtain the certificate. In our proposed scheme, the timestamp t_i is used to keep the freshness of the messages and resist the replay attack. Upon receiving the message $\{sid_j, h(\rho_i), P_\rho, P_j, R_i, t_i\}$, RC first checks whether $t_{c1} - t_i \leqslant \Delta t$ holds or not. If the timestamps t_i is stale, the request will be rejected. On the other hand, if the adversary modifies t_i as t_i^* to satisfy the condition above, i.e., $t_{c1} - t_i^* \leqslant \Delta t$, he/she cannot compute the corresponding R_i based on (5) without the knowledge of $h(\rho_i \cdot P_{pub})$. Therefore, the malicious user cannot structure a valid message $\{sid_j^*, h(\rho_i)^*, P_\rho^*, P_j^*, R_i^*, t_i^*\}$ for authentication.

Similarly, the adversary cannot compute the corresponding δ or σ_{ji}, calculated as in equation (11) and equation (14), by using a modified timestamp t_i^* when he/she wants to replay the message $\{h(\rho_i), \delta, t_i\}$ or $\{h(\rho_i), \sigma_{ji}, t_i\}$ in the rewards certificate generation phase (see in Section 5.5). Based on the above analysis, we can see that the proposed scheme is able to resist the replay attack.

Lemma 4. The proposed scheme is able to resist the impersonation attack.

Proof. In this type of attack, the adversary forges a valid message to impersonate as a legal entity using the information eavesdropped from previous message or obtained from captured nodes. However, in our proposed scheme, the adversary cannot forge a valid message to carry out such attacks. In the task issuing and sensing report update phase (see in Section 5.3), if the adversary wants to forge a message $\{\tau_s', sid_j', P_j', Q_j', t_s'\}$ to impersonate as a legal application server, he/she cannot compute the correct Q_j' based on (4), since he/she is not able to obtain the legal server's secret value $h(sid_j' \| r_2)$.

In addition, if the adversary intends to forge a message $\{\tau_s^*, sid_j^*, P_j^*, \theta_j^*, t_s^*\}$ to impersonate as the RC, he/she cannot compute the correct signature θ_j^* by (6), because he/she cannot obtain the RC's master key r_s. Based on the Discrete Logarithm Problem (DLP, [33]), it is difficult to derive the secret value r_s by way of P and P_{pub}. From the above analysis, we can see that our proposed scheme can resist the impersonation attack.

Lemma 5. A participant cannot redeem a reward certificate multiple times.

Proof. In the proposed scheme, RC maintains a data table CTable that contains the information of cashed certificates. When receiving a 5-tuple $< \rho_i, \sigma_{ji}, sid_j, P_j, t_i >$ from a user u_i for cashing rewards, RC matches the 5-tuple $< \rho_i, \sigma_{ji}, sid_j, P_j, t_i >$ with CTable. If there is no record matching $< \rho_i, \sigma_{ji}, sid_j, P_j, t_i >$, RC believes that the 5-tuple $< \rho_i, \sigma_{ji}, sid_j, P_j, t_i >$ is a new record. Otherwise, RC considers it as a cashed record.

On the other hand, the probability of generating a same random number ρ_i from different users at the same time is negligibly small. Furthermore, the probability of generating a same certificate σ_{ji} by (14) by using different sensing reports that contain the same pseudonym $h(\rho_i)$ is also negligibly small. As a result, we believe that it is not feasible to generate a same 5-tuple $< \rho_i, \sigma_{ji}, sid_j, P_j, t_i >$ from different reports. Therefore, if there is a record contained in CTable matching $< \rho_i, \sigma_{ji}, sid_j, P_j, t_i >$, RC considers the 5-tuple as a stale record. Through the above analysis, we can see that the users cannot redeem a reward certificate multiple times.

6.1.3. Other Security Features

Proposition 1. RC has the ability to verify the legitimacy of the participant.

Discussion: In our scheme, only the registered user will be eligible for a reward. When receiving the message $\{sid_j, h(\rho_i), P_\rho, P_j, R_i, c_{ij}, t_i\}$, the application server sj extracts cij, h(ρi), Pρ, ti and then sends the message $\{sid_j, h(\rho_i), P_\rho, P_j, R_i, t_i\}$ to RC. Upon receiving the message $\{sid_j, h(\rho_i), P_\rho, P_j, R_i, t_i\}$, RC extracts the values pid_i, $h(b \oplus pw_i)$, C_i by computing $R_i \oplus H(r_s \cdot P_\rho)$. Next, RC computes $C_i^* = h(pid_i \| h(r_u \oplus pw_i) \| r_1) \oplus h(sid_j \| t_i)$ based on (7) and checks whether $C_i^* = C_i$ holds. If they are equal, RC believes that user pid_i is a legitimate user. Since the value r_1 is a secret number maintained by the RC, only the legitimate user has the correct result $h(pid_i \| h(r_u \oplus pw_i) \| r_1)$, received from RC via a secure channel in the registration phase (see Section 5.2) . Therefore, RC has the ability to verify the legitimacy of the user by checking whether $C_i^* = C_i$ holds.

Proposition 2. Our scheme can successfully achieve the employed incentive mechanism.

Discussion: To implement the incentive mechanism successfully, the proposed scheme utilizes a secure one-way hash function and a MAC algorithm in the phases of rewards certificate generation and reward redemption. That is, when a participant u_i sends a sensing report to the application server, the sensing report contains the pseudonym $h(\rho_i)$. Once the application server accepts the sensing data sent from u_i, it will generate a certificate σ_{ji} by utilizing the parameter $h(\rho_i)$ (see Section 5.3). Then, the certificate σ_{ji} is delivered to u_i together with the pseudonym $h(\rho_i)$. However, in the reward redemption phase, a man who wants to cash

the certificate σ_{ji} from RC must issue the random number ρ_i rather than the pseudonym $h(\rho_i)$. Based on Section 4.5, only the participant u_i keeps the random number ρ_i. Therefore, even if an adversary could obtain the message $\{h(\rho_i), \sigma_{ji}, t_i\}$ sent from the application server, according to the properties of the one-way hash function, he/she cannot calculate ρ_i via the pseudonym h(ρi), thus he/she cannot generate the 5-tuple $< \rho_i, \sigma_{ji}, sid_j, P_j, t_i >$ to redeem the rewards.

Besides, the adversary cannot forge a certificate to defraud the reward. When generating the reward certificate, the adversary must compute $\sigma_{ji} = f_{km}(G_i, h(\rho_i), h(sid_j \| r_2), t_i)$ based on (10). Since the value $h(sid_j \| r_2)$ is a secret value that is sent by RC in registration phase, the adversary cannot obtain the secret value r_2 to compute the correct $h(sid_j \| r_2)$ to compute the certificate. Based on the above analysis, we can conclude that the proposed scheme is able to implement incentive mechanism successfully.

6.2 Performance Evaluation

We implemented a proof-of-concept of our proposed scheme to demonstrate its performance in a realistic participatory sensing application. In our implementation, we use an Android smartphone, equipped with 1.2 GHz ARM processor and 1 GB RAM. The application server and the RC are implemented on an Intel E8400 processor with 3.0 GHz CPU and 4.0 GB RAM. In our experiment, the point multiplication operations of ECC are based on a 160-bit private key. We choose Poly1305-AES [33] as the MAC algorithm used in the proposed scheme, and SHA-256 is used as the elementary hash function to structure the hash functions used in this paper (i.e., $h(\cdot)$, $H(\cdot)$, $H_1(\cdot)$). In our implementation, all encryption operations (e.g., 160-bit point multiplication operations of ECC, Poly1305-AES) are built with MIRACLE [35]. For convenience of evaluating the computational complexity, we define some metrics as follows.

T_h: the time of performing a one-way hash function $H(\cdot)$ or $h(\cdot)$;

T_{Gh}: the time of performing a map-to-point hash function $H_1(\cdot)$;

T_{pair}: the time of performing a bilinear pairings computation;

T_{add}: the time of performing an addition operation of points;

T_{mec}: the time of performing a scalar multiplication of elliptic curve.

Table 2. Computational cost on the user side and the server side

	T_{pair}	T_{mec}	T_h	T_{Gh}	T_{add}	T_{en}
Client (Smart phone)	0.015s	0.01s	< 0.001s	< 0.01s	< 0.001s	0.01s
Server (Intel E8400)	3.58ms	1.71ms	< 0.01ms	< 1ms	< 1ms	0.1ms

Table 2 illustrates the experimental results for related pairing-based operations on the Android smartphone and the Intel E8400 processor, respectively. For simplicity reasons, we use a 500 Bytes string to simulate the sensing data. In particular, the most expensive operation in our scheme is bilinear pairings and scalar multiplication of elliptic curve. From the results, we observe that the bilinear pairings operation takes 3.58 milliseconds at the application server, and takes 0.015 seconds at an Android smartphone, when averaging over 10 experiments to run the pairing based operation. **Fig. 3** further shows the results on Intel E8400 processor for above metrics. Furthermore, if the proposed scheme is implemented on more powerful high-end servers, the running time will be reduced as shown in **Table 2**.

To simulate the performance of our scheme in different system settings, we use the other two types of Android smartphones (one equipped with 800 MHz ARM processor and 512 MB RAM, and the other one equipped with a 1.7 GHz quad core ARM processor and 2 GB RAM, respectively), to run these encryption operations. Figure 4 shows the results. We observe that a high-end smartphone achieves better results in terms of running time, as approximately 55% less than the smartphone equipped with 1.2 GHz ARM processor and 1 GB RAM. However, even the low-end Android smartphone is still able to run the pairingbased operations with satisfactory performance. This means that our scheme is suitable for the environment of participatory sensing in terms of time complexity.

Table 3 illustrates the running time of our scheme in different phases at both the application server and the participant's smartphone. Note that the system initialization phase can be computed at offline, and thus we omit the computational overhead of this phase in **Table 3**. We observe that every phase can be completed within the scale of only tens of milliseconds. That is, as shown in Table 3, the participant only needs 92 millisecond for task issuing and sensing report phase, the application server needs 2.72 millisecond in task issuing and sensing report phase, and 0.51 millisecond in authentication and certificate generation phase. RC needs 5.44 millisecond in task issuing and sensing report phase, and 5.46 millisecond in authentication and certificate generation phase. This indicates a negligible time complexity of our scheme.

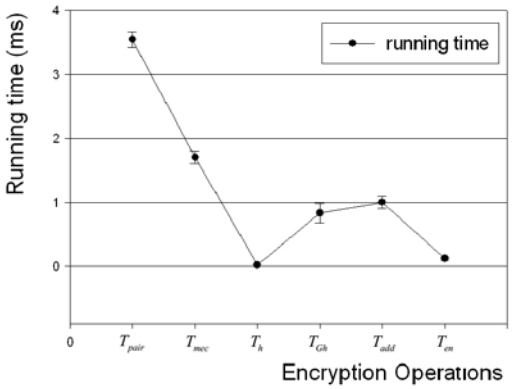

Fig. 3. Computational cost on Intel E84

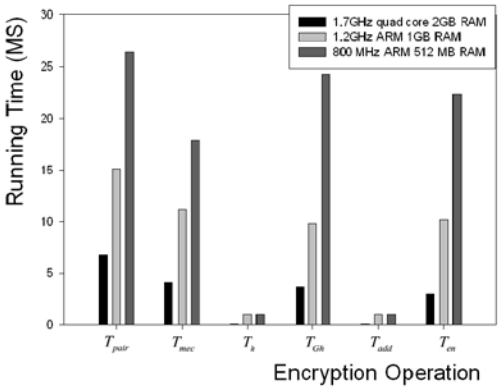

Fig. 4. Computational cost on different Android phones

Table 3. Execution time of the proposed scheme in different phases

	User	Application Server	RC
Registration	Th = 0.001s	$2Th + Tmul$ = 1.72ms	$6Th$ = 0.06ms
Task issuing and sensing report	$3TGh + 2Tpair + 3Tmul + 2Th$ = 0.092s	$Th + Tmul + TGh$ = 2.72ms	$2Th + 2TGh + 2Tmul$ = 5.44ms
Authentication and certificate generation	N.A.	$Th+Tmac$ = 0.51ms	$4Th + 2TGh + 2Tmul$ = 5.46ms

We also measure the energy consumption of the smartphone. In the experiment, the smartphone communicates with the server via Wi-Fi. We use PowerTutor [36], an application for Android smart phone that displays the power consumed by different applications, to show the energy consumption of our scheme in six different phases. **Table 4** shows the results. We observe that the execution of task issuing and sensing report update phase consumes only slightly more energy than other phases, as account for nearly 44.2% of total consumption. The energy consumption in registration phase accounts for nearly 39.6% of total consumption.

This is because that the device requires more energy to submit the sensing data in sensing report update phase. However, the entire power consumption of our scheme still maintains at a very low level.

We also monitor the remaining battery level when the smartphone (of 1.2 GHz ARM) runs our scheme continuously for 12 hours. **Fig. 4** illustrates the results under different smartphone operational status, i.e., only the phone is only running our scheme, normal operation, normal operation and our scheme, respectively. As seen from **Fig. 4**, the smartphone still exceeds 60% remaining capacity when continuously running the proposed scheme for 12 hours. Even though it is used by users in normal operations (i.e., phone calls, web surfing and text messages), while running our scheme simultaneously, the remaining capacity is still over 35%. This means that our scheme is effective minimizing the energy consumption to be used in a participatory sensing environment.

Table 4. Energy consumption of the proposed scheme in different phases

Energy cost in different phases	Energy	Fraction
Registration phase	38.4 mJ	39.6 %
Task issued and sensing report	42.8 mJ	44.2 %
authentication and certificate generation	15.7 mJ	16.2 %

Table 5. Communication overhead of our scheme

Date streams in different directions	Message Length (Bytes)
Server → RC	130
RC → User	118
User → RC	628
RC → Server	128
Server → User	112

Communication overhead is closely related to the size of sensing report and the certificate message, where it contains authentication information and the encrypted sensing data c_{ij}. In our experiment, the length of sid_j is set to 160 bits, the length of the hash function values $h(\cdot)$ and $H(\cdot)$ are 256 bits, the length of the employed MAC algorithm $f_k(\cdot)$ output is 256 bits, the length of the sensing data m_{ij} is set to 500 bytes to express rich contents, respectively. Due to the property of xor operation, the length of c_{ij} is the same as m_{ij}'s. **Table 5** shows the detailed data transfer overhead for each direction between participants, application server and RC. As shown in the table, the size of message transmitted from application server to RC, from RC to User, and from user to server, from RC to server and from server to user are: 130, 118, 628, 128 and 112 bytes, respectively. From this, we can see that the communication overhead of our scheme is very low.

7. Conclusions and Future Work

In this paper, we discussed the challenges between privacy protection and the implementation of incentive mechanisms in participatory sensing applications. Then, we proposed a novel pseudonym-based privacy protection scheme that takes both privacy protection and user incentives into considerations. The proposed scheme used pseudonym method and one-way hash function to achieve user incentive allocation, while successfully protecting their privacy. Furthermore, it is able to verify the legitimacy of the user and encrypt the sensing data sent to

the application server. Finally, extensive security and performance analysis are given to demonstrate that the proposed scheme meets the security and performance requirements of participatory sensing applications.

In the future, we plan to investigate the location and the user's trajectory/mobility privacy protection issues. Also, we are interested in studying the relationship between the privacy protection, incentive mechanism and trust mechanism from theoretical perspectively.

References

[1] Burke, J., Estrin, D., Hansen, M., Parker, A., Ramanathan, N., & Reddy, S., et al., "Participatory sensing," in *Proc. of Workshop on World-Sensor-Web (WSW'06)*: *Mobile Device Centric Sensor Networks and Applications*, 117-134. 2006. Article (CrossRef Link).

[2] B. Predic, Z. Yan, J. Eberle, D. Stojanovic, K. Aberer, "Exposuresense: Integrating daily activities with air quality using mobile participatory sensing," in *Proc. of IEEE Percom'13 Workshops*, pp. 303–305, 2013. Article (CrossRef Link).

[3] A. Waqar, A. Raza, H. Abbas, Khan MK, "A Framework for Preservation of Cloud Users' Data Privacy using Dynamic Reconstruction of Metadata," *Journal of Network & Computer Applications*, vol. 36, no. 1, pp. 235-248, 2013. Article (CrossRef Link).

[4] D. Mendez, A. J. Perez, M. A. Labrador, J. J. Marron, "P-sense: A participatory sensing system for air pollution monitoring and control," in *Proc. of IEEE Percom'11 Workshops*, pp. 344–347, 2011. Article (CrossRef Link).

[5] D. Yang, G. Xue, X. Fang, J. Tang, "Crowdsourcing to smartphones: incentive mechanism design for mobile phone sensing," in *Proc. of ACM MobiCom*'12, pp. 173–184, 2012. Article (CrossRef Link).

[6] D. Mishra, S. Mukhopadhyay, A. Chaturvedi, S. Kumari, Khan MK, "Cryptanalysis and Improvement of Yan et al.'s Biometric-based Authentication Scheme for Telecare Medicine Information Systems," *Journal of Medical Systems*, vol. 38, no. 24, June 2014. Article (CrossRef Link).

[7] H.-L. Fu, H.-C. Chen, P. Lin, "Aps: Distributed air pollution sensing system on wireless sensor and robot networks," *Computer Communications*, vol. 35, no. 9, pp. 1141–1150, 2012. Article (CrossRef Link).

[8] J. Chen, X. Cao, P. Cheng, Y. Xiao, Y. Sun, "Distributed collaborative control for industrial automation with wireless sensor and actuator networks," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 12, pp. 4219–4230, 2010. Article (CrossRef Link).

[9] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, S. Madden, "Cartel: a distributed mobile sensor computing system," in *Proc. of SenSys '06*, *ACM*, pp. 125–138, 2006. Article (CrossRef Link).

[10] Jeong-Hyo Park, Yong-Hoon Jung, Kwang-Hyung Lee and Moon-Seog Jun, "A New Privacy Scheme for Providing Anonymity Techniques on Sensor Network," in *Proc. of UCMA '11*, pp. 10-14, 2011. Article (CrossRef Link).

[11] Qiu F, Wu F, Chen G., "Privacy and Quality Preserving Multimedia Data Aggregation for Participatory Sensing Systems," *IEEE Transactions on Mobile Computing*, vol. 14, pp. 1287-1300, 2015. Article (CrossRef Link).

[12] D. Christin, C. Roßkopf, M. Hollick, "usafe: A privacy-aware and participative mobile application for citizen safety in urban environments," *Pervasive and Mobile Computing*, vol. 9, no. 5, pp. 695–707, 2013. Article (CrossRef Link).

[13] L. Duan, T. Kubo, K. Sugiyama, J. Huang, T. Hasegawa, J. Walrand, "Incentive mechanisms for smartphone collaboration in data acquisition and distributed computing," in *Proc. of IEEE INFOCOM'12*, pp. 1701–1709, 2012. Article (CrossRef Link).

[14] J.-S. Lee, B. Hoh, "Sell your experiences: a market mechanism based incentive for participatory sensing," in *Proc. of IEEE PerCom'10*, pp. 60–68, 2010. Article (CrossRef Link).

[15] Q. Li, G. Cao, "Providing privacy-aware incentives for mobile sensing," in *Proc. of ICDCS '14*, pp. 208–217, 2014. Article (CrossRef Link).
[16] S. Gao, J. Ma, W. Shi, G. Zhan, C. Sun, "Trpf: A trajectory privacy-preserving framework for participatory sensing," *IEEE Transaction on Information Forensics and Security*, vol. 8, no. 6, pp. 874 – 887, 2013. Article (CrossRef Link).
[17] E. D. Cristofaro, C. Soriente, "Participatory privacy: Enabling privacy in participatory sensing," *IEEE Network*, vol. 27, no. 1, pp. 32–36, 2013. Article (CrossRef Link).
[18] K. Xing, Z. Wan, P. Hu, H. Zhu, Y. Wang, X. Chen, Y. Wang, L. Huang, "Mutual privacy-preserving regression modeling in participatory sensing," in *Proc. of IEEE INFOCOM'13*, pp. 3039–3047, 2013. Article (CrossRef Link).
[19] E. De Cristofaro, C. Soriente, "Extended capabilities for a privacy-enhanced participatory sensing infrastructure (pepsi)," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 12, pp. 2021–2033, 2014. Article (CrossRef Link).
[20] D. Christin, A. Reinhardt, S. S. Kanhere, M. Hollick, "A survey on privacy in mobile participatory sensing applications," *Journal of Systems and Software*, vol. 84, no. 11, pp. 1928–1946, 2011. Article (CrossRef Link).
[21] A. Kapadia, N. Triandopoulos, C. Cornelius, D. Peebles, D. Kotz, "Anonysense: Opportunistic and privacy-preserving context collection," *Pervasive Computing*, pp. 280–297, 2008. Article (CrossRef Link).
[22] Khan MK, JS Zhang, L. Tian, "Protecting Biometric Data for Personal Identification," *Sinobiometrics'04, Lecture Notes in Computer Science*, no. 3383, pp. 629-638, 2004. Article (CrossRef Link).
[23] L. Kazemi, C. Shahabi, "Towards preserving privacy in participatory sensing," in *Proc. of IEEE Percom'11 Workshops*, pp. 328–331, 2011. Article (CrossRef Link).
[24] K. Vu, R. Zheng, J. Gao, "Efficient algorithms for k-anonymous location privacy in participatory sensing," in *Proc. of IEEE InfoCom'12*, pp. 2399–2407, 2012. Article (CrossRef Link).
[25] X. Oscar Wang, W. Cheng, P. Mohapatra, T. Abdelzaher, "Artsense: anonymous reputation and trust in participatory sensing," in *Proc. of IEEE InfoCom'13*, pp. 2517–2525, 2013. Article (CrossRef Link).
[26] L. Sweeney, "k-anonymity: A model for protecting privacy, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems," *World Scientific*, vol. 10, no. 05, pp. 557–570, 2002. Article (CrossRef Link).
[27] I. Boutsis, V. Kalogeraki, "Privacy preservation for participatory sensing data," in *Proc. of IEEE PerCom'13*, pp. 103–113, 2013. Article (CrossRef Link).
[28] D. Christin, C. Roßkopf, M. Hollick, L. A. Martucci, S. S. Kanhere, "Incognisense: An anonymity preserving reputation framework for participatory sensing applications," *Pervasive and mobile Computing*, vol. 9, no. 3, pp. 353–371, 2013. Article (CrossRef Link).
[29] C.-T. Li, M.-S. Hwang, Y.-P. Chu, "A secure and efficient communication scheme with authenticated key establishment and privacy preserving for vehicular ad hoc networks," *Computer Communications*, vol. 31, no. 12, pp. 2803–2814, 2008. Article (CrossRef Link).
[30] T. Jiang, H. J. Wang, Y.-C. Hu, "Preserving location privacy in wireless LANs," in *Proc. of ACM Mobisys'07*, pp. 246–257, 2007. Article (CrossRef Link).
[31] M. Gruteser, D. Grunwald, "Enhancing location privacy in wireless lan through disposable interface identifiers: a quantitative analysis," *ACM/Springer MONET*, vol. 10, no. 3, pp. 315–325, 2003. Article (CrossRef Link).
[32] R. Watro, D. Kong, S.-f. Cuti, C. Gardiner, C. Lynn, P. Kruus, "Tinypk: securing sensor networks with public key technology," in *Proc. of the 2nd ACM workshop on Security of ad hoc and sensor networks*, pp. 59–64, 2004. Article (CrossRef Link).
[33] N. Koblitz, A. Menezes, S. Vanstone, "The state of elliptic curve cryptography," in *Proc. of Towards a Quarter-Century of Public Key Cryptography*, Springer, pp. 103–123, 2000. Article (CrossRef Link).
[34] L. C. Washington, *Elliptic curves: number theory and cryptography*, CRC press, 2008. Article (CrossRef Link).

[35] S. S. Ltd., Miracl Library, http://www.compapp.dcu.ie/ mike/shamus.html (2015).
[36] M. Gordon, L. Zhang, B. Tiwana, Z. M. Mao, L. Yang, Powertutor, http://ziyang.eecs.umich.edu/ projects/powertutor/ (2015).

Junsong Zhang received his master's degree in computer software and theory from Zhengzhou University (ZZU) in 2008 and Ph.D. degree in computer science and technology from Beijing University of Posts and Telecommunications (BUPT) in 2014. Dr. Zhang is a lecturer of Zhengzhou University of Light Industry (ZZULI). His research interests include information security and mobile network, etc.

Lei He received his Master Degree in Cryptography from Southwest Jiaotong University in 2006. He is now an associate professor in the School of Computer and Communication Engineering, Zhengzhou University of Light Industry. His research interest mainly focuses on wireless network security and cryptography.

Qikun Zhang, Ph.D. Zhengzhou University of Light Industry , Zhengzhou. China. Zhang Qikun received his Bachelor of Engineering degrees from Xidian University in 2004. He received his Master of Engineering degrees from LanZhou University of Technology in 2008. He received his Doctor of Engineering from Beijing Institute of Technology in 2013.His research interests include information security and cryptography.

Yong Gan, Ph.D. Professor, School of Computer and Communication Engineering, Zhengzhou University of Light Industry. His research interests include multimedia communications, image processing, coding and network engineering.