An Efficient Anonymous Reputation System for Crowd Sensing

Shahidatul Sadiah
School of Electrical Engineering
Universiti Teknologi Malaysia
Skudai, Johor, Malaysia
e-mail: shahidatulsadiah@utm.my

Toru Nakanishi
Dept. of Information Eng.
Hiroshima University
Higashi-Hiroshima, Japan
e-mail: t-nakanishi@hiroshima-u.ac.jp

***Abstract*—Recently, crowd sensing has been intensively researched, due to the rapid growth of sensor-integrated mobile devices. Crowd sensing is a participatory sensing service where a server gathers and analyzes sensing data submitted from mobile devices of lots of users. In crowd sensing, the user's anonymity is desired, since the server gathers sensitive data from the participants including their GPS locations and moving path. However, the anonymous data submission may compromise the trust of the sensing data, because anonymous users may submit inappropriate sensing data, but they cannot be traced. Therefore, as the system to achieve both anonymity and trust in crowd sensing, ARTSense has been proposed. In the system, the trust of the sensing data is assessed on the sensed environment, other users' sensing, and the reputation of the user, and furthermore the reputation of the user is anonymously managed on the feedback from the trust assessment for the data. However, the anonymous reputation system of ARTSense has the efficiency problem, i.e., the user needs to wait a random time after the data submission phase before requesting the reputation update, which causes the communication delay.**

In this paper, we propose an efficient anonymous reputation system for crowd sensing, which can be integrated to the trust assessment in ARTSense. In the proposed system, during the data submission, the reputation update is anonymously completed. This is because the server does not manage the reputation of each user, but each user manages his/her reputation in the user side, where the the validity of the reputation is ensured by a certificate and anonymously checked by zero-knowledge proofs. Therefore, the proposed system achieves the better efficiency with no delay.

***Keywords*-crowd sensing, anonymity, trust assessment, reputation, zero-knowledge proofs, pairings**

I. Introduction

In recent years, *crowd sensing* [1] has been paid attention and researched, due to the spread of sensor-integrated mobile smart devices such as smartphones, wearable devices, and in-vehicle devices. In crowd sensing (or known as participatory sensing), a server gathers and analyzes sensing data from lots of mobile devices. The example applications include monitoring real-time traffic patterns and pollution in the city level. The flow of crowd sensing model starts with the user's registration to the server in the service provider. Then, a user voluntarily moves with a mobile device while sensing, and submits the sensing data to the server together with the GPS location. The server gathers the sensing data from lots of users to mine meaningful results for the applications.

In crowd sensing, the user's GPS locations are frequently submitted to the server. This concerns the user's privacy, since the user's movement is tracked and recorded by the server. Therefore, in crowd sensing, the anonymity of users is desired to preserve the user's privacy, as in [2], [3], [4]. However, if the user could submit the sensing data anonymously, the service are vulnerable to a malicious user that gives inappropriate sensing data. Hence, it is needed that both anonymity and trust are satisfied.

Thus, as the system to achieve both anonymity and trust for crowd sensing, ARTSense [5] was proposed. ARTSense consists of two components: Trust assessment for sensing data and anonymous reputation system. The former provides the trust of sensing data, and the latter manages the reputation of users. In the trust assessment, the trust of submitted sensing data is evaluated by the server, based on the sensed location, time, and environment together with the user's reputation level and the similarity to the other users' sensed data for the same sensing task. In the reputation system, the reputation value is anonymously manged by the server, and it is given a feedback based on the trust of the sensing data.

In the reputation system of ARTSense, to achieve the anonymity and unlinkability (i.e., infeasibility to decide the sameness of users in any two data submissions), a blind signature is used, as follows. Before the data submission, the user obtains a certificate certifying the reputation level (i.e, a rough estimate of the user's reputation value). In the sensing data submission, a blinded certificate without revealing the user's ID is also sent to show the reputation level. The server calculates the feedback value based on the trust assessment, and returns the feedback certificate to the user. After that, the user re-sends the server an unblinded reputation certificate with the user's ID and the feedback certificate, and the server updates the user's reputation in the reputation database.

However, we can observe that the reputation system in ARTSense has an efficiency problem as follows. After the data submission, the user must wait a random period to re-send the unblinded reputation certificate and the feedback certificate. If the user quickly re-sends them, the server can link the data submission to the same user's re-sending. This implies linking the data submission to the user's ID, which compromises the

978-1-7281-5268-4/19/$31.00 ©2019 IEEE
DOI 10.1109/CANDARW.2019.00071

anonymity. However, the waiting causes the communication delay. Another problem is that the server may link the two rounds of the same user by the value of the feedback.

Therefore, in this paper, we propose an efficient anonymous reputation system for crowd sensing, which can be integrated to the trust assessment in ARTSense. The proposed system is based on the anonymous reputation system in [6] for P2P services such as marketplaces and adjusted to the crowd sensing. In the P2P system, the user's reputations are not kept in the server's database, and thus the user's ID is not needed in the protocols between the server and the user. The user's reputation is signed by the server as a certificate and issued to the user, where the integer range including the reputation value, which corresponds to the reputation level, is anonymously verified through a zero-knowledge proof of knowledge. Then, the reputation certificate can be updated to reflect the feedback value without revealing the reputation value. The P2P system has a complex model and mechanism to address the P2P environment. Thus, in this paper, the model and the construction of the previous P2P system are simplified to adjust the crowd sensing environment. In the proposed system, during the sensing data submission phase, the user's reputation and certificate can be updated. This means that the user does not need to wait to complete the whole process, and thus the proposed system is more efficient than the reputation system in ARTSense.

II. Preliminaries

A. Bilinear Maps

In this paper, we utilize the bilinear groups with a bilinear map.

1) $\mathbb{G}_1, \mathbb{G}_2$ and $\mathbb{G}_T$ are multiplicative cyclic groups of prime order p. Here, we adopt the asymmetric setting where $\mathbb{G}_1 \neq \mathbb{G}_2$.
2) $g \in \mathbb{G}_1$ and $h \in \mathbb{G}_2$ are randomly chosen generators.
3) e is a computable bilinear map, $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$ with the following properties:
   - Bilinearity: for all $u \in \mathbb{G}_1$ and $v \in \mathbb{G}_2$, and $a, b \in \mathbb{Z}, e(u^a, v^b) = e(u, v)^{ab}$.
   - Non-degeneracy: $e(g, h) \neq 1_{\mathbb{G}_T}$ where $1_{\mathbb{G}_T}$ is an identity element of $\mathbb{G}_T$.

B. Assumptions

The security of our system is based on the q-SDH assumption [7] for BB signatures [7] and BBS+ signatures [8].

Definition 1 (q-SDH assumption)**.** *For all PPT algorithm $\mathcal{A}$, the probability*

$$Pr[\mathcal{A}(u, v, v^a, \dots, v^{(a^q)}) = (b, v^{1/(a+b)}) \wedge b \in \mathbb{Z}_p]$$

is negligible, where $u \in_R \mathbb{G}_1$, $v \in_R \mathbb{G}_2$ and $a \in_R \mathbb{Z}_p$.

C. BB signatures

We use the BB signature scheme proposed in [7]. In this scheme, a message and the signature can be proved with the zero-knowledge by the following PK. The existential unforgeability of BB signatures against the weakly chosen message attack is proved in [7] under the q-SDH assumption.

The algorithms are described as follows.

- **BB-Setup:** Select bilinear groups $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ with a prime order p and a bilinear map e. Then, select $g \xleftarrow{R} \mathbb{G}_1$ and $h \xleftarrow{R} \mathbb{G}_2$.
- **BB-KeyGen:** Choose $\gamma \xleftarrow{R} \mathbb{Z}_p^*$ and let $w = h^{\gamma}$. The public key is $pk = w$ and the secret key is $sk = \gamma$.
- **BB-Sign:** Given a message $m \in \mathbb{Z}_p$, compute $A = g^{1/(m+\gamma)}$.
- **BB-Verify:** Given a message m and a signature A, check if $e(A, wh^m) = e(g, h)$.

D. BBS+ signatures

The BBS+ signature is an extension from the BB signature to sign a block of multiple messages, which is informally introduced in [8], and the concrete construction is shown in [9], [10]. The existential unforgeability of BBS+ signatures against adaptively chosen message attack is proved in [10] under the q-SDH assumption.

The algorithms of the BBS+ signature on a block of L messages are as follows.

- **BBS+-Setup:** Select bilinear groups $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ with a prime order p and a bilinear map e. Then, select $g, g_1, \dots, g_{L+1} \xleftarrow{R} \mathbb{G}_1$ and $h \xleftarrow{R} \mathbb{G}_2$.
- **BBS+-KeyGen:** Choose $\gamma \xleftarrow{R} \mathbb{Z}_p^*$ and let $w = h^{\gamma}$. The public key is $pk = w$ and the secret key is $sk = \gamma$.
- **BBS+-Sign:** Given a vector of messages $(m_1, \dots, m_L) \in \mathbb{Z}_p^L$, choose $\eta, \zeta, \xleftarrow{R} \mathbb{Z}_p$, and compute $A = (gg_1^{\zeta}g_2^{m_1} \dots g_{L+1}^{m_L})^{1/(\eta+\gamma)}$. Let the signature $\sigma = (A, \eta, \zeta)$.
- **BBS+-Verify:** For the signature $\sigma = (A, \eta, \zeta)$ and $(m_1, \dots, m_L)$, check if $e(A, wh^{\eta}) = e(gg_1^{\zeta}g_2^{m_1} \cdots g_L^{m_{L+1}}, h_{)}$.

E. Proof of Knowledge (PK)

We adopt zero-knowledge proofs of knowledge (PK) on representations, which is also known as Sigma protocols [11]. Those are the generalization of the Schnorr identification protocol [12]. Concretely, we utilize PK proving the knowledge of a representation of $C \in \mathbb{G}_1$, i.e., $x_1, \dots, x_t$ s.t. $C = g_1^{x_1} \cdots g_t^{x_t}$. This can also be constructed on groups $\mathbb{G}_2$ and $\mathbb{G}_T$. The PK can be extended to proving multiple representations with equal parts.

III. Previous System

This section reviews the previous system, ARTSense [5]. In the crowd sensing, users with mobile devices and a server participate, where each mobile user submits sensing data to the server. To make the crowd sensing service reliable, ARTSense mainly consists of two components: Trust assessment for sensing data and anonymous reputation system. The former provides the trust of sensing data, and the latter manages the reputation of users.

A. Trust Assessment

The trust assessment of ARTSense is as follows. When the user submits the sensing data, the server also obtains the reputation level ℓ for the user's latest reputation rep_{t-1}, where the reputation level shows a rough value of how much the user is trusted. Then, from the location, time, and environment information included in the sensing data submission, and the reputation level, the server calculates the base trust T_b. In addition, based on the similarity between the submitted data and the other users' sensing reports in the same sensing task, the server calculates the similarity factor sim. Thus, the server can calculate the final trust of the submitted data as $T_f = T_b(1 + sim)$. Furthermore, from the trust T_f and the reputation level ℓ, the server calculates the feedback Δrep_t to the reputation rep_{t-1}, where the feedback is used in the following reputation system.

B. Anonymous Reputation System

For the anonymous reputation system, the paper [5] considers the privacy and soundness as the security. The privacy means that the sensing report does not contain any information on user's ID, and multiple sensing reports from the same user are not linkable. The soundness means that the user cannot control the reputation of the user (only the server can determine the reputation based on the past behaviors), and the user cannot lie the reputation level of a rough reputation value.

The construction of the anonymous reputation system in ARTSense is as follows. In this system, the server maintains the reputation database of each user's reputation which is linked to the user ID.

1) **Issue of Reputation Certificate**: This phase is executed before submitting the sensing data. In this phase, the user sends his/her ID U_i and the task ID TID for this sensing task to the server. Using U_i, the server obtains the reputation level ℓ of the user's reputation rep_{t-1} from the reputation database. Then, the server generates two certificates $Sig(U_i|\ell|TID)$ and $Sig(\ell|TID)$, where Sig is the digital signature function by the server's secret key. The server sends the certificates to the user.
2) **Construction of Blind ID**: In this phase, for $Sig(U_i|\ell|TID)$, the user executes blinding in a blind signature scheme (In [5], the blind RSA signature is used) to obtain the blind ID BID. After this phase, the user submits the sensing data together with BID and $Sig(\ell|TID)$ to the server.
3) **Generation of Reputation Feedback Coupon**: After the trust assessment for the submitted data, the server generates the feedback Δrep_t to the reputation. Then, the server generates the reputation feedback coupon as $Sig(BID)|Sig(Enc(\Delta rep_t)|Sig(\ell|TID))$, where Enc the encryption with the server's public key, and sends the coupon to the user.
4) **Unblinding Coupon**: The user removes the blinding factor from the sent $Sig(BID)$ to obtain $Sig(Sig(U_i|\ell|TID))$ by the unblinding process of the blind signature. After the user waits a random period, the user sends $Sig(Sig(U_i|\ell|TID))|Sig(Enc(\Delta rep_t)|Sig(\ell|TID))$ to the server.
5) **Redemption of Coupon**: The server checks the validity of signatures and the encryption. If these are valid, in the entry of U_i in the reputation database, the server updates the user's reputation rep_{t-1} to rep_t based on feedback Δrep_t.

C. Problems in Anonymous Reputation System

In the previous anonymous reputation system of ARTSense [5], the server manages the reputation of each user in the server's database. To realize the anonymity of the sensing data submission, a blind signature is used, as follows. The user sends a blinded signature BID which does not reveal U_i and TID. Then, the server sends the reputation feedback coupon to ensure the correspondence between blinded $Sig(U_i|\ell|TID)$ and Δrep_t. Finally, the server can correctly update the reputation of U_i by Δrep_t. Due to the blinding process, the communication round of the data submission and feedback coupon response is unlinkable to the communication round of sending unblinded coupon (with the user ID) and the redemption, which leads the anonymity.

To achieve the sufficient unlinkability between the two rounds, the user must wait a random period to send the unblinded coupon. If the user quickly sends the unblinded coupon, the server can link the two rounds by the same user, which weakens the anonymity, since the number of submission are insufficient. However, the waiting causes the communication delay. Another problem is that the server may link the two rounds of the same user by the value of the feedback Δrep_t (In the ARTSense paper [5], the authors suggest the variation of the feedback values is very small such as 5 to avoid this linking. But this may reduce the flexibility of the feedback).

IV. Our Approach to Efficient Anonymous Reputation System

In this paper, we propose an efficient anonymous reputation system for crowd sensing, to which the trust assessment of ARTSense is combined. Our approach is to extend the model of the anonymous reputation system in [6] for P2P services such as marketplaces and adjust it to the crowd sensing. In the P2P anonymous reputation system, a user (ratee) is rated by another user (rater), and additionally a semi-honest TTP server participates. In this system, using **Register** protocol, a user who will be a ratee registers with the server in advance, and the user is issued a certificate. The certificate ensures the user's reputation that is accumulated from past ratings. Using **Show** protocol, a user can anonymously prove his/her reputation to other users, where only the integer range including the reputation value is revealed to show the trust of the user. After a P2P interaction between the ratee user and a rater user, the server is given a rating from the rater. Finally, using **Update** protocol, the server issues the ratee an updated certificate of the reputation summed up by the new rating. The characteristic of this system is that the server does not manage the database of the reputation of each user. Instead, the reputation is managed in each user side. This is why the server does not need the user's ID. To prevent the ratee from maliciously modify the reputation, the reputation is certified by the certificate issued from the server. Furthermore, to achieve the anonymity, the update process of the certificate becomes blind, i.e., the reputation value is kept secret for the server in the certificate generation. The advantage of this system is that after the ratee is rated, the certificate is updated with no delay. By bringing this approach to crowd sensing, we can achieve the efficient reputation-update process with no delay.

However, this previous anonymous reputation system targets P2P services. In such services, before the P2P interaction, a ratee shows his reputation (range). Then, after the P2P interaction, the ratee is rated, and the user's certificate is updated based on the new rating. But, before the rating, the ratee wants to show his/her reputation for another interaction. On the other hand, after the rating, the certificate should be updated to reflect the new rating even if it is a negative rating. But, a malicious user may try to show the previous reputation to discard the current negative rating. Thus, this reputation system has a mechanism to prevent the user from discarding the negative rating, as follow. **Show** protocol correspondent to a P2P interaction is indexed by integer i, which is included in the certificate. In **Show** protocol, it is checked whether the interactions for all indexes i are not rated in the anonymous way. After the i-th interaction is rated, the index i is removed from the certificate. This is why the ratee cannot discard any negative rating.

We adapt this previous P2P system to the crowd sensing environment. The crowd sensing is a simple client-server model, i.e., a central crowd sensing server communicates to each mobile user. In addition, in the model of ARTSense, the server can decide the rating (feedback) during the phase where the sensing data is submitted. This is why we can combine **Update** and **Show** protocol into a single protocol called **Show**. In **Show** protocol of our system, the user shows the reputation range (level), and the certificate is updated by the feedback based on the trust assessment in ARTSense.

In this model, since the certificate is compulsorily updated by the server, we do not need to counter user's discarding the negative rating. Thus, the mechanism to counter it can be removed, and thus the reputation system can be simplified and efficient.

V. Model of Proposed System

A. Syntax

The proposed anonymous reputation system consists of the following algorithm and protocols. The participants of this system are the server and the users with mobile devices for the crowd sensing.

- **Setup:** This is an algorithm for the server. The inputs are security parameter λ and the number of the reputation levels L. The algorithm generates the server's public key spk and secret key ssk, and initializes set $\mathcal{S}$ that keeps the tags for used one-time reputation certificates.
- **Register:** This is an interactive protocol between a user and the server, where the user is registered with the server. The common input is spk and the server's input is ssk. The user's output of this protocol is $cert_0$ that is the user's initial one-time reputation certificate certifying the initial reputation $rep_0 = 0$.
- **Show:** This is an interactive protocol between the user and the server, where the user convinces the server of his/her reputation level (the integer range in which the reputation is included) and the reputation is updated. The common input are spk, and the reputation level ℓ. The user's input is his latest $cert_{t-1}$ certifying the reputation rep_{t-1}. The server's input is $\mathcal{S}$. If the server judges that rep_{t-1} is not included in the integer range of ℓ, the user is rejected. Otherwise, the user's output is an one-time fresh reputation certificate $cert_t$ certifying the updated reputation rep_t added by the feedback Δrep_t, which is derived from the assessment for the sensing report and the reputation level. The server's output is the updated $\mathcal{S}$. Set $\mathcal{S}$ consists of tags included in the past used certificates to detect the double use of a certificate. If the double use is detected, this protocol is aborted.

In this model, for each sensing data submission, only **Show** protocol is executed, where any delay is not needed.

B. Security Requirements

For the security, we consider the reputation unforgeability and anonymity. These requirements are derived from ones for the underlying P2P reputation system [6].

The *reputation unforgeability* means that the user cannot modify the reputation rep_{t-1} in the certificate $cert_{t-1}$ to prove the inappropriate reputation level ℓ in **Show**. Namely, for $\tilde{rep}_{t-1}$ that is the reputation value correctly calculated from the sequence of the past feedback values $\Delta rep_1, \ldots, \Delta rep_{t-1}$, the user cannot prove any inappropriate reputation level ℓ such that $\tilde{rep}_{t-1}$ is not included in the integer range of the level ℓ.

The *anonymity* means that any adversary can obtain no information on the user beyond the reputation level in **Show** protocol, even if the adversary corrupts the server. This also means that the adversary cannot determine whether the user of a **Show** protocol is the same as the user of another **Show** protocol.

The formal definitions will be shown in the journal version of this paper.

VI. Proposed Reputation System for Crowd Sensing

A. Outline of Proposed System

Before describing the construction of the proposed system, we show the outline, and mention the difference from the underlying system.

- **Setup:** In this algorithm, the server generates key pairs of BB signatures and BBS+ signatures. Then, the server computes the BB signature on every value in the integer range of reputation level $1 \leq \ell \leq L$ as the reputation level certificate.

- **Register:** The server issues a registering user an initial reputation certificate $cert_0$, which is a BBS+ signature on the user's secret x, a tag S_0, and the initial reputation $rep_0 = 0$.
- **Show:** The user's input is his/her latest certificate $cert_{t-1}$. At first, the user proves the reputation level ℓ. This is performed by the PK proving the BBS+ signature in $cert_{t-1}$ for rep_{t-1} and proving the BB signature for the reputation level ℓ and the value rep_{t-1}. In addition, by the user's sending tag S_{t-1}, the server checks if the tag has been used in the past. Next, for the feedback Δrep_t, the server blindly updates the user reputation as $rep_t = rep_{t-1} + \Delta rep_t$ via the commitment of rep_{t-1}. Finally, the server generates a new BBS+ signature as the updated certificate $cert_t$ for rep_t to send the user.

The difference from the P2P anonymous reputation system [6] is as follows. As mentioned in Section IV, since **Update** is integrated to **Show**, the mechanism to avoid the user's discarding negative feedbacks is removed and simplified. As the mechanism, an accumulator was used, and a structure-preserving signature was used to sign the accumulator of a group element in the certificate and to prove the knowledge. However, in the proposed system, only $\mathbb{Z}_p$ elements are signed, and thus the more efficient BBS+ signature is used. Because of this, commitments used to blindly sign messages are modified and simplified to a vector-type commitment used in BBS+ signatures. In addition, in [6], the PK for the BB signature needs three proved relations. On the other hand, in [13], the PK using only one relation is shown. Thus, in this paper, using this technique in [13], the PK is optimized.

B. Proposed Construction

Setup: In this algorithm, the server generates key pairs of public and private keys for BB signatures and BBS+ signatures, and issues the certificates (BB signatures) for all integer ranges of reputation level ℓ for $1 \leq \ell \leq L$, where L is the maximum number of the reputation levels.

1) The server selects bilinear groups $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$, and a bilinear map e with a prime order $p > 2^\lambda$, where λ is the given security parameter. Then, the server selects $g_0, g_1, g_2, g_3, g_4, f_0, f_1 \xleftarrow{R} \mathbb{G}_1$, $h_0 \xleftarrow{R} \mathbb{G}_2$. For all $1 \leq \ell \leq L$, the server chooses $\gamma_{0,\ell} \xleftarrow{R} \mathbb{Z}_p^*$, and computes $w_{0,\ell} = h_0^{\gamma_{0,\ell}}$, where $\gamma_{0,\ell}$ is the secret key for the BB signature proving the the reputation level ℓ. Then, as the key pairs of BBS+ signatures, the server chooses $\gamma_1 \xleftarrow{R} \mathbb{Z}_p^*$, and computes $w_1 = h_0^{\gamma_1}$, where γ_1 is the secret key for the user's reputation certificate.
2) For all $1 \leq \ell \leq L$, the server generates the reputation level certificate $A_{\ell,R_{\ell,k}} = {f_0}^{1/(\gamma_{0,\ell}+R_{\ell,k})}$ (BB signature) for every value $R_{\ell,k}$ in the ℓ-th range, where K_ℓ is the number of the values.
3) The server initializes set $\mathcal{S}$ as empty, and outputs the public key
$$spk = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, \{w_{0,\ell}\}_{\ell=1}^{L}, w_1, g_0, g_1, g_2, g_3, g_4, f_0, f_1, h_0, \{\{A_{\ell,k}\}_{k=1}^{K_\ell}\}_{\ell=1}^{L}),$$
and the secret key $ssk = \gamma_1$.

Register: This is a protocol between the user **U** and the server **S**. In this protocol, the server issues an initial reputation certificate $cert_0$ for the user. The common input is spk, and the server's input is ssk.

1) **[U]:** Select secret $x \xleftarrow{R} \mathbb{Z}_p^*$, a reputation certificate's tag $S_0 \xleftarrow{R} \mathbb{Z}_p^*$, and a random factor $\zeta_0' \xleftarrow{R} \mathbb{Z}_p^*$, and compute the commitment to the vector of messages (x, S_0) to be signed by $C'_{m,0} = g_1^{\zeta_0'} g_2^x g_3^{S_0}$. Then, prove to the server that $C'_{m,0}$ is correctly formed by the following PK:
$$PK\{(\zeta_0', x, S_0) : C'_{m,0} = g_1^{\zeta_0'} g_2^x g_3^{S_0}\}.$$
2) **[S]:** Set the initial reputation as $rep_0 = 0$, and choose random factors $\zeta_0'', \eta_0 \xleftarrow{R} \mathbb{Z}_p^*$. Then, using the secret key γ_1 of BBS+ signatures, sign the vector of messages (x, S_0, rep_0) as $B_0 = (g_0 g_1^{\zeta_0''} C'_{m,0} g_4^{rep_0})^{1/(\gamma_1+\eta_0)}$, and send back $\tilde{\sigma}_0' = (B_0, \eta_0, \zeta_0'')$ to the user.
3) **[U]:** Set $C_{m,0} = C'_{m,0} g_4^{rep_0}$ for $rep_0 = 0$, compute $\zeta_0 = \zeta_0' + \zeta_0''$, and set the BBS+ signature on the messages (x, S_0, rep_0) as $\tilde{\sigma}_0 = (B_0, \eta_0, \zeta_0)$, where $B_0 = (g_0 g_1^{\zeta_0} g_2^x g_3^{S_0} g_4^{rep_0})^{1/(\gamma_1+\eta_0)}$. Output $cert_0 = (x, rep_0, \tilde{\sigma}_0, S_0, C_{m,0})$.

Show: In this protocol, the user's reputation level ℓ is proved on the certificate $cert_{t-1}$, the certificate is updated by adding the feedback Δrep_t to the previous reputation rep_{t-1}, and then the updated reputation certificate $cert_t$ is issued. The user's inputs are $cert_{t-1} = (x, rep_{t-1}, \tilde{\sigma}_{t-1}, S_{t-1}, C_{m,t-1})$, where $\tilde{\sigma}_{t-1} = (B_{t-1}, \eta_{t-1}, \zeta_{t-1})$. Here, t indicates the number of updates in the reputation certificates for the user.

1) **[U]:** From spk, retrieve a reputation level certificate $A_{\ell,rep_{t-1}}$ such that his current reputation rep_{t-1} is in ℓ-th range. Choose $r_{A_\ell} \xleftarrow{R} \mathbb{Z}_p$ and compute the commitment $C_{A_\ell} = A_{\ell,rep_{t-1}} f_1^{r_{A_\ell}}$ and $\rho = r_{A_\ell} \cdot rep_{t-1}$. Then, choose $\hat{\zeta} \xleftarrow{R} \mathbb{Z}_p$, compute the commitment $C_{B_{t-1}} = B_{t-1} g_1^{\hat{\zeta}}$, and set $\theta = \zeta_{t-1} + \hat{\zeta}\eta_{t-1}$. Choose $\zeta_t' \xleftarrow{R} \mathbb{Z}_p^*$ and $S_t \xleftarrow{R} \mathbb{Z}_p^*$, and compute $C'_{m,t} = g_1^{\zeta_t'} g_2^x g_3^{S_t} g_4^{rep_{t-1}}$ as the commitment to the vector of (x, S_t, rep_{t-1}). Send $C_{A_\ell}, C_{B_{t-1}}, C'_{m,t}, S_{t-1}$ to the server, and prove that the reputation rep_{t-1} is in the ℓ-th range, $cert_{t-1}$ is valid,

and $C'_{m,t}$ is correct, by showing the following PK.

$$\begin{aligned}PK\{(r_{A_\ell}, rep_{t-1}, \rho, \theta, x, \hat{\zeta}, \eta_{t-1}, \zeta'_t, S_t) :\\
e(C_{A_\ell}, w_{0,\ell}) \cdot e(f_0, h_0)^{-1} &= e(f_1, w_{0,\ell})^{r_{A_\ell}}\\
&\quad \cdot e(C_{A_\ell}, h_0)^{-rep_{t-1}} \cdot e(f_1, h_0)^{\rho}\\
\wedge\, e(C_{B_{t-1}}, w_1) \cdot e(g_0, h_0)^{-1} \cdot e(g_3, h_0)^{-S_{t-1}}\\
&= e(g_1, h_0)^{\theta} \cdot e(g_2, h_0)^{x} \cdot e(g_4, h_0)^{rep_{t-1}}\\
&\quad \cdot e(g_1, w_1)^{\hat{\zeta}} \cdot e(C_{B_{t-1}}, h_0)^{-\eta_{t-1}}\\
\wedge\, C'_{m,t} = g_1^{\zeta'_t} g_2^{x} g_3^{S_t} g_4^{rep_{t-1}}\}.\end{aligned}$$

2) **[S]:** To check the freshness of the proved certificate, check if $S_{t-1} \in \mathcal{S}$. If it is true, abort. Otherwise, add tag S_{t-1} in set $\mathcal{S}$. Next, update the user's reputation certificate to $cert_t$, where Δrep_t is added to commitment as $C_{m,t} = C'_{m,t} g_4^{\Delta rep_t}$ and it is signed as $B_t = (g_0 g_1^{\zeta''_t} C_{m,t})^{1/(\gamma_1+\eta_t)} = (g_0 g_1^{\zeta''_t} g_1^{\zeta'_t} g_2^{x} g_3^{S_t} g_4^{rep_{t-1}} g_4^{\Delta rep_t})^{1/(\gamma_1+\eta_t)}$ for $\zeta''_t, \eta_t \xleftarrow{R} \mathbb{Z}_p^*$. Then, send back $\tilde{\sigma}'_t = (B_t, \eta_t, \zeta''_t)$ to the user. Output the updated $\mathcal{S}$.
3) **[U]:** Compute $\zeta_t = \zeta'_t + \zeta''_t$, $rep_t = rep_{t-1} + \Delta rep_t$ and set the signature on the vector of messages (x, S_t, rep_t) as $\tilde{\sigma}_t = (B_t, \eta_t, \zeta_t)$, where $B_t = (g_0 g_1^{\zeta_t} g_2^{x} g_3^{S_t} g_4^{rep_t})^{1/(\gamma_1+\eta_t)}$. Output $cert_t = (x, rep_t, \tilde{\sigma}_t, S_t, C_{m,t})$.

VII. Security

Before considering the security of the proposed scheme, we show the following lemma.

Lemma 1. *The PK in* **Show** *proves the knowledge of $A'_{\ell,R_{\ell,k}}$, ξ, $R_{\ell,k}$, $rep_{t-1}, B_{t-1}, \zeta_{t-1}, \eta_{t-1}, x$ such that*

$$A'_{\ell,R_{\ell,k}} = (f_0 f_1^{\xi})^{1/(\gamma_{0,\ell}+rep_{t-1})},$$
$$B_{t-1} = (g_0 g_1^{\zeta_{t-1}} g_2^{x} g_3^{S_{t-1}} g_4^{rep_{t-1}})^{1/(\gamma_1+\eta_{t-1})}.$$

Proof. From the PK, we can extract r_{A_ℓ}, rep_{t-1}, ρ, θ, x, $\hat{\zeta}$, and η_{t-1} such that

$$\begin{aligned}e(C_{A_\ell}, w_{0,\ell}) \cdot e(f_0, h_0)^{-1} &= e(f_1, w_{0,\ell})^{r_{A_\ell}}\\
&\quad \cdot e(C_{A_\ell}, h_0)^{-rep_{t-1}} \cdot e(f_1, h_0)^{\rho}, \quad (1)\end{aligned}$$

$$\begin{aligned}&e(C_{B_{t-1}}, w_1) \cdot e(g_0, h_0)^{-1} \cdot e(g_3, h_0)^{-S_{t-1}}\\
&= e(g_1, h_0)^{\theta} \cdot e(g_2, h_0)^{x} \cdot e(g_4, h_0)^{rep_{t-1}}\\
&\quad \cdot e(g_1, w_1)^{\hat{\zeta}} \cdot e(C_{B_{t-1}}, h_0)^{-\eta_{t-1}}. \quad (2)\end{aligned}$$

Then, from Eq. (1), we have the following transformations.

$$\begin{aligned}e(C_{A_\ell}, w_{0,\ell}) \cdot e(C_{A_\ell}, h_0)^{rep_{t-1}} \cdot e(f_1, w_{0,\ell})^{-r_{A_\ell}}\\
= e(f_0, h_0) e(f_1, h_0)^{\rho}\end{aligned}$$

$$e(C_{A_\ell}, w_{0,\ell} h_0^{rep_{t-1}}) \cdot e(f_1, w_{0,\ell})^{-r_{A_\ell}} = e(f_0 f_1^{\rho}, h_0)$$

$$\begin{aligned}e(C_{A_\ell}, w_{0,\ell} h_0^{rep_{t-1}}) \cdot e(f_1, w_{0,\ell})^{-r_{A_\ell}} e(f_1, h_0)^{-r_{A_\ell} rep_{t-1}}\\
= e(f_0 f_1^{\rho}, h_0) e(f_1, h_0)^{-r_{A_\ell} rep_{t-1}}\end{aligned}$$

$$\begin{aligned}e(C_{A_\ell}, w_{0,\ell} h_0^{rep_{t-1}}) \cdot e(f_1^{-r_{A_\ell}}, w_{0,\ell} h_0^{rep_{t-1}})\\
= e(f_0 f_1^{\rho}, h_0) e(f_1^{-r_{A_\ell} rep_{t-1}}, h_0)\end{aligned}$$

$$e(C_{A_\ell} f_1^{-r_{A_\ell}}, w_{0,\ell} h_0^{rep_{t-1}}) = e(f_0 f_1^{\rho - r_{A_\ell} rep_{t-1}}, h_0)$$

Thus, by computing $A'_{\ell,rep_{t-1}} = C_{A_\ell} f_1^{-r_{A_\ell}}$ and $\xi = \rho - r_{A_\ell} rep_{t-1}$, we obtain $e(A'_{\ell,rep_{t-1}}, w_{0,\ell} h_0^{rep_{t-1}}) = e(f_0 f_1^{\xi}, h_0)$, which implies $A'_{\ell,rep_{t-1}} = (f_0 f_1^{\xi})^{1/(\gamma_{0,\ell}+rep_{t-1})}$.

Next, from Eq. (2),

$$\begin{aligned}&e(C_{B_{t-1}}, w_1) \cdot e(g_1, w_1)^{-\hat{\zeta}}\\
&\cdot e(C_{B_{t-1}}, h_0)^{\eta_{t-1}} = e(g_0, h_0) \cdot e(g_1, h_0)^{\theta}\\
&\qquad \cdot e(g_2, h_0)^{x} \cdot e(g_3, h_0)^{S_{t-1}} \cdot e(g_4, h_0)^{rep_{t-1}}\end{aligned}$$

$$\begin{aligned}e(C_{B_{t-1}}, w_1 h_0^{\eta_{t-1}}) \cdot e(g_1, w_1)^{-\hat{\zeta}}\\
= e(g_0 g_1^{\theta} g_2^{x} g_3^{S_{t-1}} g_4^{rep_{t-1}}, h_0)\end{aligned}$$

$$\begin{aligned}e(C_{B_{t-1}}, w_1 h_0^{\eta_{t-1}}) \cdot e(g_1, w_1)^{-\hat{\zeta}} \cdot e(g_1, h_0)^{-\hat{\zeta}\eta_{t-1}}\\
= e(g_0 g_1^{\theta} g_2^{x} g_3^{S_{t-1}} g_4^{rep_{t-1}}, h_0) \cdot e(g_1, h_0)^{-\hat{\zeta}\eta_{t-1}}\end{aligned}$$

$$\begin{aligned}e(C_{B_{t-1}}, w_1 h_0^{\eta_{t-1}}) \cdot e(g_1^{-\hat{\zeta}}, w_1 h_0^{\eta_{t-1}})\\
= e(g_0 g_1^{\theta} g_2^{x} g_3^{S_{t-1}} g_4^{rep_{t-1}}, h_0) \cdot e(g_1^{-\hat{\zeta}\eta_{t-1}}, h_0)\end{aligned}$$

$$\begin{aligned}e(C_{B_{t-1}} g_1^{-\hat{\zeta}}, w_1 h_0^{\eta_{t-1}})\\
= e(g_0 g_1^{\theta-\hat{\zeta}\eta_{t-1}} g_2^{x} g_3^{S_{t-1}} g_4^{rep_{t-1}}, h_0)\end{aligned}$$

Thus, by computing $B_{t-1} = C_{B_{t-1}} g_1^{-\hat{\zeta}}$ and $\zeta_{t-1} = \theta - \hat{\zeta}\eta_{t-1}$, we obtain $e(B_{t-1}, w_1 h_0^{\eta_{t-1}}) = e(g_0 g_1^{\zeta_{t-1}} g_2^{x} g_3^{S_{t-1}} g_4^{rep_{t-1}}, h_0)$, which implies $B_{t-1} = (g_0 g_1^{\zeta_{t-1}} g_2^{x} g_3^{S_{t-1}} g_4^{rep_{t-1}})^{1/(\gamma_1+\eta_{t-1})}$. □

This lemma shows that the user proves the knowledge of $A'_{\ell,R_{\ell,k}}$ s.t. $A'_{\ell,R_{\ell,k}} = (f_0 f_1^{\xi})^{1/(\gamma_{0,\ell}+rep_{t-1})}$. This $A'_{\ell,R_{\ell,k}}$ is a variant of BB signature, and not the same as a BB signature on rep_{t-1}, due to the part f_1^{ξ}. However, as proved in [13], forging the variant can be reduced to forging the BB signature.

In this paper, we discuss the security of the proposed system informally. The formal security proofs will be shown in the journal version. The proofs are derived from the proofs in the original P2P anonymous reputation system [6].

Reputation Unforgeability. In the proposed system, the user's reputation value cannot be modified by anyone except the server. This is because the reputation value rep_{t-1} is certified by the BBS+ signature $\tilde{\sigma}_t = (B_{t-1}, \eta_{t-1}, \zeta_{t-1})$ issued by the server, where anyone except the server cannot compute $\tilde{\sigma}_{t-1}$. In **Show**, the user has to conduct the PK, where, as shown in Lemma 1, it proves the knowledge $(B_{t-1}, \eta_{t-1}, \zeta_{t-1})$ satisfying $B_{t-1} = (g_0 g_1^{\zeta_{t-1}} g_2^{x} g_3^{S_{t-1}} g_4^{rep_{t-1}})^{1/(\gamma_1+\eta_{t-1})}$ that is, the BBS+ signature on (x, S_{t-1}, rep_{t-1}). Furthermore, for the proved rep_{t-1}, the user also proves the knowledge of a BB signature on rep_{t-1} (As above-mentioned, strictly a variant of BB signature) from Lemma 1. Thus, the correct range of rep_{t-1}

is ensured. In addition, the PK shows $C'_{m,t} = g_1^{\zeta'_t} g_2^x g_3^{S_t} g_4^{rep_{t-1}}$ for the rep_{t-1}, and $C'_{m,t} g_4^{\Delta rep_t} = g_1^{\zeta'_t} g_2^x g_3^{S_t} g_4^{rep_{t-1}+\Delta rep_t}$ for feedback Δrep_t is signed by the BBS+ signature as the next certificate. By checking tag S_{t-1}, a past used certificate cannot be used. Since the PK proves the sameness of x in B_{t-1} and $C'_{m,t}$, the certificate of a different user cannot be used. Therefore, the user can show only appropriate reputation level ℓ of the range including the certified rep_{t-1} which is correctly computed from the past feedback values.

Anonymity. In **Show**, the data sent from the user are $C_{A_\ell}, C_{B_{t-1}}, C'_{m,t}, S_{t-1}$ and the communication in PK. The commitments $C_{A_\ell}, C_{B_{t-1}}, C'_{m,t}$ hide any information. Since the PK communication is zero-knowledge, it has no information. Tag S_{t-1} is one-time random. Therefore, even the server cannot obtain any information beyond the reputation level ℓ.

VIII. Efficiency Considerations

In this section, we compare the efficiency between our proposed system and the previous reputation system in ARTSense [5].

As mentioned in Section III-B, in the previous system, after the data submission phase using the blind ID, "the redemption of coupon" phase is needed to update the user's reputation value in the database of the server side, where the user has to wait a random period for the request. If the period is short, the server can link the user's ID to the data submission. Thus, a relatively long delay is needed in a single cycle of a user's data submission and reputation management. Instead, in the proposed system, since the user's reputation is managed in each user side, the reputation management (i.e., **Show** protocol) completes within the data submission phase, which means that any delay is not needed. This is why we conclude that the communication cost in the proposed system is more efficient than the previous system.

On the other hand, our system needs pairing-related computations in **Show**, although the previous system needs only blind RSA signatures, and any ordinary digital signature and encryption. The pairing computation for the bilinear map e is a relatively heavy, compared to RSA computations. But, note that the computations in the user side can be pre-computed (the on-line computations are only response computations in the PK, which are only light multiplications). The implementation-based evaluations to clarify the practicality in crowd sensing is one of our future works.

IX. Conclusions

In this paper, the efficient anonymous reputation system for crowd sensing is proposed. The proposed system achieves the reputation update within the data submission, by adapting a P2P anonymous reputation system from [6] to crowd sensing. As a result, we solved the efficiency problem of the communication delay caused in ARTSense.

Our future works include the implementation of the proposed system, and the efficient user revocations.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number JP19K11964 and UTM GUP Grant Number 16J83.

References

[1] B. Guo, Z. Wang, Z. Yu, Y. Wang, N. Y. Yen, R. Huang, and X. Zhou, "Mobile Crowd Sensing and Computing: The Review of an Emerging Human-powered Sensing Paradigm," ACM Comput. Surv., Vol.48, No.1, pp.7:1–7:31, 2015.

[2] M. Shin, C. Cornelius, D. Peebles, A. Kapadia, D. Kotz, and N. Triandopoulos, "AnonySense: A System for Anonymous Opportunistic Sensing," Pervasive and Mobile Computing, Vol.7, No.1, pp.16–30, 2011.

[3] E. D. Cristofaro, C. Soriente, "Extended Capabilities for a Privacy-Enhanced Participatory Sensing Infrastructure (PEPSI)," IEEE Trans. Information Forensics and Security Vol.8, No.12, pp.2021–2033, 2013.

[4] S. Rahaman, L. Cheng, D. D. Yao, H. Li, J. J. Park, "Provably Secure Anonymous-yet-Accountable Crowdsensing with Scalable Sublinear Revocation," PoPETs, Vol.2017, No.4, pp.384–403, 2017.

[5] X. Oscar, W. Cheng, P. Mohapatra and T. Abdelzaher, "ARTSense: Anonymous Reputation and Trust in Participatory Sensing," 2013 Proceedings IEEE INFOCOM, pp.2517–2525, 2013.

[6] T. Nakanishi and N. Funabiki, "An Anonymous Reputation System with Reputation Secrecy for Manager," IEICE Trans. Fundamentals, Vol.E97-A, No.12, pp 2325–2335, 2014.

[7] D. Boneh and X. Boyen, "Short Signatures Without Random Oracles," Advance in Cryptology – EUROCRYPT 2004, LNCS 3072, pp. 56-73, Springer-Verlag, 2004.

[8] D. Boneh, X. Boyen, and H. Shacham, "Short Group Signatures," Advance in Cryptology – CRYPTO 2004, LNCS 3152, pp. 41–55, Springer-Verlag, 2004.

[9] M.H. Au, W. Susilo, and Y. Mu, "Constant-size dynamic k-TAA," SCN 2006, LNCS 4116, pp.111–125, Springer-Verlag, 2006.

[10] M.H. Au, W. Susilo, and Y. Mu, "Constant-size dynamic k-TAA," Cryptology ePrint Archive: Report 2008/136, 2008. This is the extended version of [9].

[11] I. Damgård, "On Σ-Protocols," `http://www.daimi.au.dk/~ivan/Sigma.pdf`.

[12] C. P. Schnorr, "Efficient signature generation for smart cards," Journal of Cryptology, Vol.4, No.3, pp.239–252, 1991.

[13] T. Nakanishi, H. Fujii, Y. Hira, and N. Funabiki, "Revocable Group Signature Schemes with Constant Costs for Signing and Verifying," IEICE Transactions Fundamentals, Vol.E93-A, No.1, pp.50–62, 2010.