Privacy-Respecting Auctions as Incentive Mechanisms in Mobile Crowd Sensing

Tassos Dimitriou, Ioannis Krontiris

Privacy-respecting Auctions as Incentive Mechanisms in Mobile Crowd Sensing

Tassos Dimitriou[1,2] and Ioannis Krontiris[3]

[1] Computer Engineering Dept., Kuwait University, Kuwait
[2] Research Academic Computer Technology Institute, Patras, Greece
[3] Huawei Technologies Duesseldorf GmbH, Munich, Germany
tassos.dimitriou@ieee.org, ioannis.krontiris@huawei.com

Abstract. In many mobile crowdsensing scenarios it is desirable to give micro-payments to contributors as an incentive for their participation. However, to further encourage participants to use the system, one important requirement is protection of user privacy. In this work we present a reverse auction mechanism as an efficient way to offer incentives to users by allowing them to determine their own price for the data they provide, but also as a way to motivate them to submit better quality data. At the same time our auction protocol guarantees bidders' anonymity and suggests a new rewarding mechanism that enables winners to claim their reward without being linked to the data they contributed. Our protocol is scalable, can be applied to a large class of auctions and remains both computation- and communication-efficient so that it can be run to the mobile devices of users.

Keywords. Mobile Crowd Sensing, Multi-attribute auctions, Incentive Mechanisms, Security and Privacy

1 Introduction

The availability of sensors in today's smartphones, carried by millions of people, has led to a new sensing paradigm, where people provide sensing capabilities to applications in order to map the environment and offer a better understanding of people's activities and their surroundings. This trend is often referred to as Mobile Crowdsensing (MCS) [1] or, using the more general term, as Mobile Crowdsourcing [2].

In this paradigm there is a platform provider who publicizes multiple sensing tasks from which people can choose and execute those that match their location and sensing capabilities. However, there are two factors that hinder the large-scale deployment of such applications. First, lack of proper incentives does not motivate users to participate, and second, in many cases data coming from users' smartphones can have a large impact on user privacy.

These two issues have been studied separately in existing research. For example, a number of MCS systems started incorporating different incentive features, including various forms of rewards based on monetary [3], social or gaming-related mechanisms [4]. The work from Zaman et al. present an overview of many available incentive mechanisms in MCS till today [5]. In particular, micro-payments have been shown to be effective in encouraging participation [6] and recently Rula et al. [7] presented additional experimental evidence that such a mechanism can increase the productivity of the participants.

One of the challenges in offering micro-payments to contributors is to determine the right amount they expect to receive as a payment for their effort in reporting sensing data. This amount may depend on personal preferences and the perceived cost of their participation, but also on the context and situation users are currently involved which can be different among individuals. One attractive solution to this problem is the use of *reverse* auctions, where the auction takes place among data providers (sellers) and data requester (buyers) of sensing data [8, 3, 9]. This mechanism is more attractive as it eliminates the need for the requester to set or guess the price which users consider reasonable for their data; instead it is the data provider who sets the price for the data it is willing to provide to the requester.

However, as mentioned above, privacy is an important factor that hinders user participation. Indeed, collecting data from users' devices has many privacy implications since user-sensitive information such as daily patterns, location and social relationships can easily be deduced from provided data [10, 11]. It is thus imperative to address privacy in mobile crowdsensing systems. While several efforts already exist that suggest anonymizing users' contributions to protect user privacy (see for example [12]), it still remains an open problem on how to provide privacy protection when incentive mechanisms are also incorporated in the system.

Our Contribution: In this work, we suggest a privacy-respecting protocol that allows anonymous users to participate in reverse auctions employed by an MCS system. Our protocol consists of two main parts. The first part provides bidders' anonymity for the auction while it offers guarantees in terms of correctness and fairness of the auction process. The second part explores different options of rewarding users and suggests a new mechanism that enables winners of the auction to claim their rewards without being linked to their contributed data. Thus participants can have the highest privacy assurance, while the MCS platform operator can maintain the flexibility of offering incentives to users and encouraging participation. More specifically, our protocol (i) offers strong privacy protection by guaranteeing user anonymity and unlinkability of transactions, (ii) it is scalable and applicable to typical MCS applications, (iii) it offers resilience to compromised or colluding MCS entities, and (iv) it can support any type of reverse auction.

Organization: The rest of the paper is organised as follows. In Section 2, we overview work related to privacy and incentives for MCS systems, while in Section 3, we describe the system and adversarial models for our protocol. In this section we also present a *generic* auction mechanism that does not take privacy into account. Then, in Section 4, we add privacy by describing a scheme that provides for bidder anonymity in MCS auctions as well as different mechanisms that can be used to reward participating users. In this section we also specify the security properties expected by both the auction and rewarding schemes. The protocols' security guarantees and performance are analyzed in Section 5, while Section 6 concludes the paper.

2 Related Work

One of the earliest works that addresses the use of incentives for participatory sensing using auctions is [9]. Since then several other incentive models based on reverse auctions have been proposed [3, 13, 14]. At the same time, auction theory for electronic commerce continues to advance and multi-attributive auctions have gradually become a research hot spot, incorporating qualitative attributes to decide the winner [15]. This was shown to have many advantages for the MCS case, too [8].

Privacy is an important requirement in auctions which are used to facilitate the trade of goods. For example, Shi [16] proposes a sealed bid multi-attribute contract auction protocol that pays special attention on bid privacy and bidder anonymity. However, this and previous work [17–20] on conducting secure auctions has emphasized on attaining full privacy in which case bids remain secure even after the auction is over. This is typically achieved by distributing trust among bidders or by using multiple auctioneers. As a result, these works rely on heavier cryptographic operations and primitives (e.g. secret sharing techniques, multi-party computations, etc.) and as such they are not considered suitable for the MCS model described here. To this end, we have chosen to protect bids only during the bidding phase. Once this phase is over, all bids are revealed as they don't affect the correctness and fairness of the process or the privacy of users.

Some generic privacy-respective architectures for MCS exist that could be of interest in our discussion. For example, Gisdakis et al. [21] recently proposed the SPPEAR architecture, which supports anonymous users to contribute to sensing tasks and receive credits, as long as they submit at least n reports. In that sense it supports incentive mechanisms, but it concentrates mainly on the rewarding process.

Another recent work that places emphasis on rewards is given by Li and Cao [22], who propose two privacy-aware schemes for mobile sensing, where each data provider gets some credit for each contribution they make. The use of these credits/tokens may incentivize users to participate, however no auction mechanism is presented to help improve the quality of data provided.

Finally, Krontiris and Dimitriou [23] have proposed a solution to protect the privacy not only of data providers but also of data requesters. However, to the best of our knowledge, this is the first work that shows how to integrate more advanced incentive mechanisms, like auctions, in mobile sensing frameworks, while offering strong privacy protection guarantees.

3 Preliminaries

3.1 System Model

We consider a generic Mobile Crowdsensing (MCS) system that consists of the following three actors.

Service Providers: These are the requesters of sensing data. We assume that a requester has a specific budget and wants to collect real-time sensing data from a specific area of interest. To ensure data is real-time, the requester defines short time periods T_i, within which data are to be collected from a given area.

Users: They participate in the sensing process using various types of mobile devices such as smartphones or wearable devices. These devices come equipped with different types of sensors such as cameras, microphones, GPS, etc.

Auction Infrastructure: For the sake of modularity we separate this into three different servers, even though they can belong to the same entity: The Task Server, which is responsible of publishing the sensing tasks, the Auction Server, which is responsible for running the auction process, and the Report Server, which collects the reports from the auction winners and forwards them to the Service Provider.

Participating users first contact the task server to see if there are any tasks that match their preferences and context. Then, they decide which ones to download and execute. The advantage of this approach is that users do not reveal private information, like context or location, to the task server in order to execute the sensing task. At the network level, we assume the existence of an *anonymizing* network which can be used to protect the network identities of the communicating devices as in [12].

3.2 Threat Model

We assume that both internal and external adversaries could try to compromise the system. External adversaries can monitor communications, in order to extract information about user activities. They can also manipulate the collection of information by submitting unauthorized data or replaying data of benign users. Typically, these attacks can be mitigated using traditional cryptographic mechanisms to provide confidentiality and integrity guarantees. External attackers can also target system availability by launching jamming and DoS attacks, but here we assume that these are handled by the network operators and so they fall outside the scope of this work.

Internal adversaries, on the other hand, can be malicious users or MCS system entities that target the auction and/or rewarding processes. For example, adversarial users could try to obtain rewards without offering contributions or could try to double-spend already redeemed quotas. Internal adversaries can also target the privacy of participating users, by trying to profile them and reveal their identities by colluding with other entities in the system. Thus, with respect to user privacy, our goal would be to ensure that bids, reports and rewards cannot be linked to a particular user even if that user has submitted multiple bids and has accepted multiple rewards for the data it has provided.

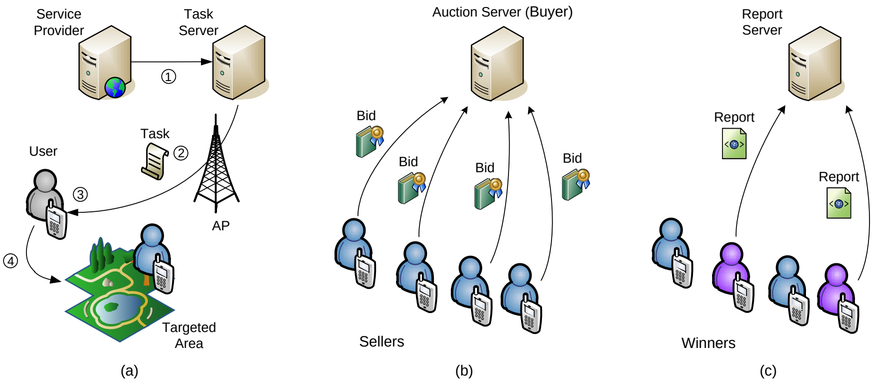


Fig. 1: Acquisition of sensing data.

Attacks where malicious users submit false sensing data are outside the scope of this work, as these can be addressed by different methods, such as anonymous reputation schemes [24].

3.3 A Generic Auction Mechanism

The goal of this section is to give a description of a generic MCS system, integrating an auction mechanism, without, however, taking privacy concerns into consideration. Then, in the next section, we will pose our security and privacy requirements and add all the mechanisms required.

As we mentioned in Section 2, many auction mechanisms proposed for MCS systems are mainly based on reverse auctions [9, 3, 13, 14]. However, reverse auctions constitute a sub-optimal solution because they incorporate only the expected price into the user's auction bid and they do not allow participants to negotiate on *data quality* as well. In MCS applications, sensing data may be of different qualities and this has to be considered in the auction mechanism in order for better data to reach the service providers.

A more general form of auction is the *multi-attributive* auction, which enables service providers also pose quality criteria on the sensed data they are looking to buy, in addition to the price. This, however, does not affect our privacy solution, which is generic and can work with any reverse auction variant. In Figure 1, we highlight the three main phases of the auction process. The specific steps involved are described below, however for a more detailed discussion the reader is referred to [8].

Step 1: The Task Server (TS) publishes the tasks received by the Service Providers. Once a task is published, the bidding phase for this task begins and lasts for a fixed amount of time T_i. This deadline is announced in the description of the task, which also contains other details like the *acceptance conditions* C_i

which define the required sensors, termination conditions, etc., and a *utility function* $S(x)$ for this task, setting the ground for which mobile devices qualify for executing it. The role of the utility function is to allow the service provider to announce its budget and quality requirements in order to be addressed by the mobile user (seller) when bidding for this task.

In summary, each task contains the following information: (i) Geographic area of interest, (ii) Acceptance Conditions C_i, (iii) Utility Function $S(x)$, and (iv) Bid duration T_i.

Step 2: The mobile devices periodically check with the Task Server to see if there are any tasks available for them, filtered based on their acceptance criteria which may also include other local, user-defined conditions like remaining battery level of the mobile device, and so on.

Step 3: If the user/mobile device decides to execute a task, then it bids for it by calculating and sending the value of the utility function $S(x)$ to the Auction Server (AS), during the bidding phase of this task. The value of $S(x)$ is calculated *locally* at the mobile device, as all necessary information is already available to the device.

One important attribute that affects the calculation of $S(x)$ is the price that the user expects for this task. However, besides the price, additional attributes can be integrated into the bid using multi-attribute auctions. More precisely, a bid can be expressed as a n-dimensional tuple of attributes x_i, represented as $x = (x_1, \ldots, x_n)$, which can be weighted together to compute the overall utility of a bid in terms of a *utility score.*

Typical examples of attributes x_i that can be incorporated in the utility function include the distance from the desired location, the location accuracy, the sampling frequency, etc.

Step 4: Before submitting their bids, the users can see their utility score, and if not satisfied, they can choose to improve it by adjusting the various attributes. For example, a user could set a lower price, move closer to the sensing area in order to provide more accurate data (see Figure 1(a)), and so on. When the utility score can no longer be improved, the bidder submits her bid to the AS (Figure 1(b)). We stress again that the user's bid is *not* comprised of the actual sensed data but is equal to the computed utility score.

Step 5: Once the bidding phase for a task is over, the *opening phase* begins where the winning bidders will be determined. More specifically, the Auction Server determines the winners to be those with the n highest utility scores (n does not have to be equal to 1). These winners are publicly announced by the Auction Server.

Step 6: The actual sensing data of the winning bidders are submitted to the Report Server (Figure 1(c)). Once the RS verifies the provided data match the utility scores promised in the bid, the winners collect their rewards for this task, typically the price that they have asked for in their bids.

4 Privacy-respecting Auction and Rewarding Protocols

So far we have not considered security and privacy in the auction protocol. Here we pose such requirements and we demonstrate how they can be satisfied in an efficient way.

The participants of the protocol include the m bidders (potential sellers of sensing data) and the Auction Server (buyer) who will get the data from the bidder(s) with the maximum utility scores. The auction consists of two main phases: the *bidding* and the *opening* phase. During the bidding phase, each bidder commits to a bid that is kept secret from the other participants. When the bidding phase is over, the Auction Server (AS) opens the bids and determines the winners with the highest utility scores. These bidders are the winners of the auction and they will be rewarded for their participation once they upload their sensing data.

4.1 Security and Privacy Requirements

The following properties are expected from our protocol.

- *Correctness and Fairness*: The result of the auction is determined according to the rules described in the previous section. In our case the first $n < m$ bidders (e.g. $n = 2$) that made the highest utility bid win the auction and get rewarded. Additionally, no bidder can obtain an unfair advantage over the rest of the bidders by determining or changing its own bid based on information revealed about other bids.
- *Bidders' privacy*: Bidders remain anonymous throughout the whole process of the auction. This means that the identity of the bidder cannot be linked in any way to the bids they submit. Moreover two different bids from the same user cannot be linked with each other, thus nobody can build a profile and reduce users' anonymity. Finally, the claim of a reward by a winning bidder cannot be linked with either a bid or a user ID.
- *Confidentiality of bids*: All bids remain secret until the opening phase. If the Auction Server (AS) or any other party can recover some of the bids before the opening, he can inform a colluding bidder in order to cheat and win the auction. Contrary to prior work [16]-[20] that requires distribution of shares among participants or the use of heavy zero-knowledge schemes which are not easy to apply in the participatory sensing paradigm, we will develop a lightweight, yet secure protocol, that guarantees bid secrecy up until the opening phase.
- *Public verifiability*: The correctness of the auction process should be easy to verify by any interested party. This includes assurance about the validity of the bids, as well as winner selection.
- *Non-repudiation*: No bidder should be able to change its mind (e.g. deny or modify its bid) once the bid is submitted. Our protocol will ensure this property by requiring the bidders to commit to their bids prior to the opening phase.

4.2 Auction Protocol

In what follows we assume that bidders are aware of the public key K_{AS} of the auction server. They can use this to send confidential information to AS and authenticate messages signed by the AS. We denote by $H()$ a secure cryptographic hash function with at least 256 bits of output. In this context 'secure' means that $H()$ is one-way and collision resistant. Thus inverting the hash function or finding x and y satisfying $H(x) = H(y)$ is computationally infeasible.

We also assume the existence of a "bulletin board" that is used to communicate messages between the bidders and the Auction Server (AS). Once a message is posted to the bulletin board, anybody (even third-parties) can read it. However, erasing from the bulletin board is not possible. Thus, the bulletin board is nothing more than a public channel where broadcasted messages are received by anybody and can be verified by any third party [25].

The two main phases of the auction are bidding and opening, however, there is also an implicit, *registration* phase in which the AS sets up the bulletin board, publishes its public key and announces various parameters of the auction like the auction ID, starting/ending time, duration of each phase, and so on. Upon registration, each bidder i sends to the AS a pseudonymous ID ($BidderID$) to represent its identity during the auction along with a *one-time* public key K_i. AS publishes this information to the bulletin board and every bidder can verify it has been properly registered for the auction.

Bidding: During the bidding phase, each bidder i computes its utility score S_i, masks it with a random number r_i and sends a commitment C_i of the form

$$C_i = Sig_i(AuctionID||BidderID||h_i), \tag{1}$$

where $h_i = H(S_i||r_i)$. Thus the auction server receives a bid, however it cannot read this bid before the opening phase. Additionally, these values are published in the bulletin board so that anybody can verify that its bid has been correctly accounted for.

Opening: When the auction server marks the end of the bidding phase, each bidder reveals both S_i and r_i that have been used in the computation of the commitment C_i. The server goes through the values C_i and recovers the n highest utility scores as the winners of the auction. Then it sends a signed message

$$\begin{aligned} Sig_{AS}(AuctionID,\ &\langle BidderID_{i_1}, K_{i_1}, S_{i_1}, r_{i_1}\rangle, \\ &\langle BidderID_{i_2}, K_{i_2}, S_{i_2}, r_{i_2}\rangle, \ldots) \end{aligned} \tag{2}$$

which contains the pseudonymous IDs of the winners along with their public keys and the committed values that have been opened in the beginning of the phase. Thus, any participant can verify correctness by computing $H(S_i||r_i)$ and comparing with the signature C_i received during the bidding phase. In the next section we discuss how the winning bidders can be rewarded for the data they provide.

4.3 Rewarding mechanisms

Once the winning bidders are selected, they contact the Report Server (RS) in order to transmit their sensed data (recall Figure 1(c)). Each winning bidder i provides RS with the winning notification shown in (2) (alternatively the auction server can forward this directly to the RS) and submits its sensed data as follows:

$$Bidder_i \rightarrow RS : \ \langle BidderID, AuctionID, D, \sigma \rangle, \tag{3}$$

where $\sigma = Sig_i(BidderID, AuctionID, H(D))$ and D is the sensed data for the relevant task. The Report Server goes on to verify if the signature comes from a winning bidder whose public key is listed in the winning notification shown in (2) and evaluates the utility function on the received data D. If the utility score matches the one shown in the winning notification, it proceeds to reward the bidder for the data provided[4]. In the following, we demonstrate how this can be achieved using (i) an existing payment service (e.g. a Bank) along with an e-cash scheme, and (ii) a decentralized token-based scheme.

While the e-cash scheme may be easier conceptually, it suffers from a potential loss of privacy if the report server and the Bank collude together to reveal the bidder's identity. To mitigate this possible loss of trust, we have developed a decentralized variant where the RS is the sole issuer of rewards that can be redeemed by the bidder.

Using e-cash: An e-cash scheme in its general form is a set of cryptographic operations that allows a party S (in our case the report server) to withdraw electronic money from a bank in order to purchase something from a second party B (the bidder), and B to deposit the money in its bank account. E-cash schemes are distinguished between "on-line" and "off-line" ones depending on whether the bank has to be actively involved in the purchase protocol. The auction protocol that we present here works with both schemes, however an off-line scheme is more preferable as the bidder does not have to query the bank for the validity of the payment it will receive from the report server. We therefore abstract away the actual implementation details of the digital cash protocol used and describe a coin as tuple $\langle c, \sigma_{Bank}(c) \rangle$, where c basically denotes the value of the coin and $\sigma_{Bank}(c)$ is the signature of the bank. Other information such as expiration day, or details that might help in extracting the ID of the owner in case of double-spending are omitted here [27].

Once the Report Server receives the data sent by the winning bidder in message (3), it sends back to the bidder a coin $\langle c, \sigma_{Bank}(c) \rangle$ *encrypted* with

[4] An issue may arise if the provider refuses to reward the bidder after obtaining the sensed data. Although there are cryptographic protocols to mitigate this type of behavior [26], we chose to keep the protocol as simple as possible since (i) the damage to the reputation of the provider will be much higher than any gains for data received but not paid, and (ii) the provider runs the risk of losing potential bidders which goes against the idea of introducing payments in the first place as a means to enhance user participation in crowdsensing applications.

the bidder's public key. The value of the coin matches the price agreed in the specification of the utility function. The RS does not need to know who the bidder is, only that it is one of the winners of the auction. The encryption of the coin is required so that only the bidder can recover (and use) the coin. Thus, anyone else who eavesdrops on the communication line cannot steal and spend the coin. The bidder now can either deposit the coin to the Bank or spend it if the coin is transferable. This depends on the underlying digital cash protocol used.

Another alternative, that avoids the use of digital cash but still uses a centralized payment service, is for the Report Server to authorize a payment *directly* with the bank. In this scheme, the winning bidder computes

$$\mu = F(H(D), N),$$

where F is a secure one-way function, D is the data submitted and N is a new unpredictable number. Then it includes μ in the message and the signature σ shown in (3). Once the Report Server receives and validates the signature, it produces a payment $p_{RS} = Sig_{RS}(\langle H(D), \mu, Amount \rangle)$, where *Amount* corresponds the price for the data received, and forwards p_{RS} to both the bidder and the payment service. To claim the money from the bank, the bidder has to reveal N used in the computation of μ. Once the bank verifies the signature of RS on p_{RS} and validates μ, it credits the bidder with the specified amount.

Both schemes presented here protect the bidder's anonymity as long as the Report Server and the payment service/Bank do not collude to reveal the bidder's identity. If the payment service is not trusted, we can use a decentralized variant where the bidder itself constructs the coins to be redeemed by the RS. This is explained below.

Using a decentralized scheme: To eliminate the need for a centralized payment service, we can use the Report Server as an *issuer* of reward tokens that can be redeemed by the bidder. However, since it is the RS who issues these tokens we must be sure that tokens cannot be used to track bidders. A similar token service was developed in [28], however for use in a different setting. There, a querier Q wishing to access an MCS network for sensor data could use tokens issued by the application owner to pay for data received by some producer P.

In our setting there is no querier for data, however we can leverage this technology to allow a winning bidder to obtain rewards for the data provided once the auction is over. This approach can still be thought as a lightweight e-cash scheme, yet without the requirement of a trusted payment service. However, in this case, double-spending detection becomes an important property as bidders may be tempted to redeem these tokens more than once. In what follows, we explain how to adapt these ideas to build a rewarding mechanism for auctions. In particular, we will explain (i) how tokens can be constructed during submission of sensing data without compromising bidder privacy, and (ii) how tokens can be redeemed.

Token construction: To create such a token, the collaboration of both the bidder and the Report Server is needed. The bidder has to introduce some *private* piece of information (a unique ID) to the token T so that upon redemption, the RS can tell if T is already spent. The token will also contain a *public* part, introduced by the RS, minimally containing the value of the coin but perhaps an expiration date, etc. To make these tokens untraceable, blind signatures will be used to blind the private information introduced to the token by the bidder before it is signed by the RS. When the blinding factor is later removed, the token will bear the signature of the RS but the only identifiable information on the coin will be its public part.

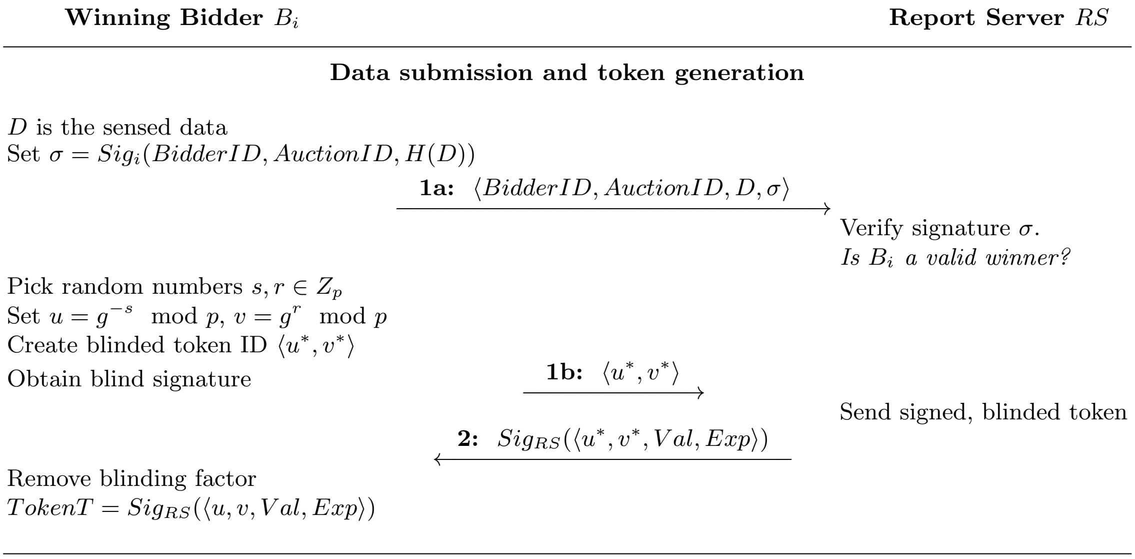
To introduce this unique ID to the token and make double-spending possible, we leverage the identification scheme of Schnorr [29]. The bidder first selects two primes p and q, where $q|p-1$. Then it chooses at random two numbers $s, r \in Z_p$ and computes $u = g^{-s} \mod p$ and $v = g^r \mod p$, where g is a generator of order q in Z_p^*. The token ID now is comprised of the two values u and v, which has to be further blinded by the bidder and signed by the RS as mentioned above (details omitted due to space restrictions, however the interested reader is referred to [30]). After the signature by the RS, the bidder will have in its possession a token T of the form

$$T = Sig_{RS}(\langle u, v, Value, Expiration \rangle). \tag{4}$$

Token spending: When, at some future time t, the bidder wants to spend T, it has to prove it knows s, r. This is possible using a non-interactive zero-knowledge proof. In particular, the bidder sends along with the token T, the pair $\langle y, t \rangle$, where t is the date/time of the transaction, $e = H(T, t)$ and $y = r + es \mod q$. The RS verifies the authenticity of the token by first checking its signature on T and then wether $v = g^y u^e \mod p$. If both tests succeed, the token is considered *valid.* However, RS still needs to check that T has not been used before.

So, RS looks in its database of spent tokens for a token with the same ID $\langle u, v \rangle$. If no such token exists, T is accepted and RS records the tuple $\langle T, y, t \rangle$. If, however, a token T with the same ID already exists, there will be two tuples $\langle y, t \rangle$ and $\langle y', t' \rangle$ of token T such that $v = g^y u^e \mod p$, $v = g^{y'} u^{e'} \mod p$, $y = r + es$ and $y' = r + e's$. From these two last values, RS can obtain the secret value $s = (y - y')/(e - e')$ and subsequently r. Thus, two submissions of the same token will result in evidence that the coin has already been spent. However, as these values are not tied to the bidder's identity, its privacy is maintained even in the case of double-spending.

In summary, the protocol ensures that i) tokens are not tied to bidder identities, and ii) the RS is protected by malicious bidders who try to double-spend tokens. A snapshot of both data submission and token construction phases is shown in Figure 2 (although Steps 1a and 1b are shown separate for presentation clarity, they can be merged into one.)



Winning Bidder B_i **Report Server** RS

Data submission and token generation

D is the sensed data

Set $\sigma = Sig_i(BidderID, AuctionID, H(D))$

$\xrightarrow{\textbf{1a:}\ \langle BidderID, AuctionID, D, \sigma\rangle}$

Verify signature σ.

Is B_i *a valid winner?*

Pick random numbers $s, r \in Z_p$

Set $u = g^{-s} \mod p$, $v = g^{r} \mod p$

Create blinded token ID $\langle u^*, v^* \rangle$

Obtain blind signature

$\xrightarrow{\textbf{1b:}\ \langle u^*, v^*\rangle}$

Send signed, blinded token

$\xleftarrow{\textbf{2:}\ Sig_{RS}(\langle u^*, v^*, Val, Exp\rangle)}$

Remove blinding factor

$TokenT = Sig_{RS}(\langle u, v, Val, Exp\rangle)$

Fig. 2: Data Reporting and Token Generation.

5 Evaluation

In this section we first emphasize on how our solution satisfies the security and privacy requirements posed in Section 4.1. Then, we also discuss performance issues.

- *Confidentiality of bids.* Since bids are opened only after the bidding phase, nobody can compute the bids before they are opened. This is because the bids at this time consist of only a commitment of the form $h_i = H(S_i||r_i)$. The one-wayness of the hash function H ensures that bid values remain hidden, eliminating the possibility of collusion since no bidder or the Auction server can leak any information about anybody else's bidding. It is only after the opening phase that bids are revealed to all.
- *Correctness & Verifiability.* Since all values are published in the bulletin board anybody can verify the correctness of the auction. This is possible as all bidders reveal their utility scores S_i and the random numbers r_i used in the computation of the signed bid commitment h_i. Hence no new values can be introduced at this point (all values must already exist in the bulletin board) or changed (due to the collision resistance of the hash function H). A value that is not available in the bulletin board at the end of the opening phase automatically excludes the bidder for the remaining of the auction. Additionally, anybody can compare and verify if the winning bidders published in message (2) by the AS are indeed the ones with the highest utility scores. Thus, correctness of the auction is assured.
- *Non-repudiation.* Since each bid carries the bidder's signature, nobody can deny its bid. The collision resistance of the hash function also ensures that

it is not possible to find a different set of (S_i', r_i') such that $H(S_i'||r_i') = H(S_i||r_i)$. Hence nobody can deny its bidding price once the bids are opened. Furthermore, if a dispute arises over the winning bids, the bid commitment can be used to resolve the dispute: the values (S_i, r_i) and the bidder's signature can be used to prove authenticity of a bid.

- *Unlinkability between bids.* This property is related to the privacy of the bidder. In particular, we would like to be sure that it is not possible to relate two bids submitted at *different* auctions by the same bidder. It should be clear that this property holds as bidders participate in auctions using different pseudonyms and public keys. Hence it is not possible to relate the bids.[5]
- *Unforgeability/Unreusability of tokens.* The zero knowledge proofs used during token spending ensure that only a bidder who knows the representation of u and v in the token ID can supply these proofs. Furthermore if the bidder supplies two different proofs for the same token, the secret values r, s used in the construction u and v can be extracted, thus providing a proof of double-spending. Thus, a token can be used only once, satisfying the unreusability property.
- *Bidder privacy/Unlinkability of tokens.* When a user tries to redeem a token and provides the server (directly or indirectly through a proxy) the zero knowledge proof, the server cannot tell which bidder created the token as the only visible part during the token construction is the public part $\langle Val, Exp \rangle$ of the token.

  There are, however, other *side channels* that can be used to infer bidder information. Consider, for example, the case where the IP address of a bidder is visible when the user submits sensed data/retrieves a token to/from the server and then tries to spend this token. Obviously, in such a case additional mechanisms are required to ensure that a connection cannot be made with the reporting bidder. However this can be avoided by using an anonymizing network at the network layer, as we mentioned in our system model.

  Another side channel is the structure of the token's public part $\langle Val, Exp \rangle$. If the value of the token or its expiration date is an unusual quantity, both can be used to associate the data with the bidder upon redeeming the token. Hence these values must be drawn from a universe that does not allow for this kind of discrimination. For example, expiration dates can be set to the end of the current year and token values can be *coarsely* defined. This would exclude tokens with unusually precise values, e.g. \$1.236743. A simpler alternative, however, is to use a trusted proxy or representative that can redeem these rewards on behalf of a user.

[5] This, however, necessitates the use of an anonymity service so that bid submissions cannot be linked to an internet identifier such as the IP address of the bidder. Hence the use of services like TOR mentioned in the system model.

Performance

The bid submission protocol is very simple, requiring the submission of just a single message (recall the bidding message shown in (1)). The bidder has to compute a hash value $H(S_i||r_i)$ on the utility score S_i and random r_i, along with a signature C_i on this data. A typical signature using a 1024-bit signing key on a 450MHz processor takes approximately 30ms as shown in [31], which is well within the capabilities of modern-day phones incorporating much faster CPUs. Similarly, the opening phase requires one more message in which each bidder reveals the committed values S_i and r_i.

Perhaps it is more instructive to consider the token rewarding protocol we developed in Section 4.3, as this actively involves the bidder in the token generation process. Here we argue that the most expensive operation is the actual transmission of the sensed data submitted by the user (Step 1a in Figure 2). The creation of the token requires two modular exponentiations for u and v, and two modular multiplications for the blinding and unblinding of the u^* and v^*. However, these operations are well within the capabilities of modern phones as mentioned above. Token redemption requires the user to prove knowledge of the values u and v, however this requires only one extra addition and multiplication to compute $\langle T, y, t\rangle$. The burden is on the side of the server who has to verify the corresponding signature, but this overhead is negligible given the capabilities of the RS.

Finally, from a storage point of view, the server has to maintain only the collection of tokens that have not expired yet. As the sensor data collected along with these tokens are perhaps orders of magnitude larger, the overhead for the Report Server is again minimal.

6 Conclusions

In this paper we have presented a protocol for privacy-protecting auctions in mobile crowdsensing systems. Users of mobile devices can participate anonymously in the auctions and define the price they expect for contributing sensing data. On the other side, the buyer of the data can select the winners based not only on the price, but also on the quality of the offered data. The winners of the auction can then collect their price without linking their real identity to the data they contributed. Our solution uses a lightweight and decentralized rewarding scheme eliminating the need for a single trusted payment system.

As future work, we plan to extend our protocol to address some research questions that remain open. In particular, we plan to incorporate a mechanism for encouraging users who lose the auction, to return, so that the system maintains its base of participants. We also think it is important to include user credibility as one of the attributes that determine the winners of the auction. In order to do that, we plan to show how to integrate an anonymous reputation mechanism in our auction protocol so that winners can collect reputation points based on the quality of their submitted data.

7 Acknowledgments

The first author would like to acknowledge support of this work by Kuwait University, Research Grant No. QE 01/13.

References

1. Guo, B., Yu, Z., Zhou, X., Zhang, D.: From participatory sensing to mobile crowd sensing. In: Proceedings of the IEEE PERCOM Workshops. (March 2014) 593–598
2. Chatzimilioudis, G., Konstantinidis, A., Laoudias, C., Zeinalipour-Yazti, D.: Crowdsourcing with smartphones. IEEE Internet Computing **16**(5) (2012) 36–44
3. Yang, D., Xue, G., Fang, X., Tang, J.: Crowdsourcing to smartphones: Incentive mechanism design for mobile phone sensing. In: Mobicom '12. Istanbul, Turkey (2012) 173–184
4. Di, B., Wang, T., Song, L., Han, Z.: Incentive mechanism for collaborative smartphone sensing using overlapping coalition formation games. In: IEEE Globe Communication Conference (Globecom), Atlanta, USA (2013) 1705–1710
5. Zaman S. and Abrar, N. and Iqbal, A.: Incentive model design for participatory sensing: Technologies and challenges. In: International Conference on Networking Systems and Security (NSysS) (2015) 1–6
6. Reddy, S., Estrin, D., Hansen, M., Srivastava, M.: Examining micro-payments for participatory sensing data collections. In: Proceedings of the 12th ACM international Conference on Ubiquitous Computing (UbiComp), (2010) 33–36
7. Rula, J.P., Navda, V., Bustamante, F.E., Bhagwan, R., Guha, S.: No "one-size fits all": Towards a principled approach for incentives in mobile crowdsourcing. In: Proceedings of the 15th Workshop on Mobile Computing Systems and Applications (HotMobile) (2014) 3:1–3:5
8. Krontiris, I., Albers, A.: Monetary incentives in participatory sensing using multi-attributive auctions. International Journal of Parallel, Emergent and Distributed Systems **27**(4) (2012)
9. Lee, J.S., Hoh, B.: Sell your experiences: a market mechanism based incentive for participatory sensing. In: Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PerCom) (2010) 60–68
10. Christin, D., Reinhardt, A., Kanhere, S.S., Hollick, M.: A survey on privacy in mobile participatory sensing applications. Journal of Systems and Software 84(11) (November 2011)
11. Wang, Y., Huang, Y., Louis, C.: Respecting user privacy in mobile crowdsourcing. ASE Science Journal **2**(2) (2013)
12. Shin, M., Cornelius, C., Peebles, D., Kapadia, A., Kotz, D., Triandopoulos, N.: AnonySense: A system for anonymous opportunistic sensing. Journal of Pervasive and Mobile Computing 7(1) (2010) 16–30
13. Zhang, X., Yang, Z., Zhou, Z., Cai, H., Chen, L., Li, X.: Free market of crowdsourcing: Incentive mechanism design for mobile sensing. IEEE Transactions on Parallel and Distributed Systems **25**(12) (Dec 2014) 3190–3200
14. Koutsopoulos, I.: Optimal incentive-driven design of participatory sensing systems. In: Proceedings of IEEE INFOCOM (2013) 14–19
15. Pham, L., Teich, J., Wallenius, H., Wallenius, J.: Multi-attribute online reverse auctions: Recent research trends. European J. of Oper. Research **242**(1) (2015) 1–9

16. Shi, W.: A sealed-bid multi-attribute auction protocol with strong bid privacy and bidder privacy. Security and Communication Networks **6**(10) (2013) 1281–1289
17. Peng, K., Boyd, C., Dawson, E.: Optimization of electronic first-bid sealed-bid auction based on homomorphic secret sharing. In: Progress in Cryptology - Mycrypt (2005) 84–98
18. Brandt, F.: How to obtain full privacy in auctions. Intern. Journal of Information Security **5**(4) (2006) 201–216
19. Zheng, S., McAven, L., Mu, Y.: First price sealed bid auction without auctioneers. In: Proceedings of the International Conference on Wireless Communications and Mobile Computing (IWCMC)(2007) 127–131
20. Nojoumian, M., Stinson, D.: Efficient sealed-bid auction protocols using verifiable secret sharing. In: Information Security Practice and Experience. V. 8434. (2014) 302–317
21. Gisdakis, S., Giannetsos, T., Papadimitratos, P.: SPPEAR: security & privacy-preserving architecture for participatory-sensing applications. In: Proc. of the 7th ACM Conf. on Security and Privacy in Wireless and Mobile Networks (WiSec) (2014) 39–50
22. Li, Q., Cao, G.: Providing efficient privacy-aware incentives for mobile sensing. In: Proceedings of the 34th IEEE International Conference on Distributed Computing Systems (ICDCS) (2014) 208–217
23. Krontiris, I., Dimitriou T. A platform for privacy protection of data requesters and data providers in mobile sensing. Computer Communications **11** (2015) 43–54
24. Christin, D., Rosskopf, C., Hollick, M., Martucci, L.A., Kanhere, S.S.: Incognisense: An anonymity-preserving reputation framework for participatory sensing applications. In: Proceedings of the IEEE PerCom (2012) 135–143
25. Cohen, J.D., Fischer, M.J.: A robust and verifiable cryptographically secure election scheme. In: Proceedings of the 26th Annual Symposium on Foundations of Computer Science (SFCS) (1985) 372–382
26. Rial, A., Preneel, B.: Optimistic fair priced oblivious transfer. In: Proceedings of the Third International Conference on Cryptology in Africa (AFRICACRYPT) (2010) 131–147
27. David Chaum, Amos Fiat, Moni Naor: Untraceable electronic cash. In: Proceedings of the 8th Annual International Cryptology Conference on Advances in Cryptology (CRYPTO) (1988) 319–327
28. Dimitriou, T., Krontiris, I., Sabouri, A.: PEPPeR: a querier's privacy enhancing protocol for participatory sensing. In: Proceedings of the 4th International Conference on Security and Privacy in Mobile Information and Communication Systems (MobiSec) (2012) 93–106
29. C.P. Schnorr: Efficient signature generation by smart cards. In: Journal of Cryptology, Vo1.4, No.3, (1991) 161–174
30. Chaum, D.: Blind signatures for untraceable payments. In: Advances in Cryptology Proceedings of Crypto 1982 (3): 199203
31. Lauter, K. The advantages of elliptic curve cryptography for wireless security. IEEE Wireless Communications **11**(1) (2004) 62–67