This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2549046, IEEE Internet of Things Journal

Privacy-Preserving Spatiotemporal Matching for Secure Device-to-Device Communications

Jingchao Sun, *Student Member, IEEE,* Rui Zhang, *Member, IEEE,* and Yanchao Zhang, *Senior Member, IEEE*

***Abstract*—Device-to-device (D2D) communications are emerging due to the explosive growth of smartphones and tablets. Given the possible presence of attackers, a fundamental challenge in secure D2D communications is to develop sound mobile authentication techniques whereby mobile users can select the most trustworthy D2D communication partners from possibly many candidates. This paper tackles this open challenge and proposes spatiotemporal matching as a promising enabler for secure D2D communications. Spatiotemporal matching is built upon the location-aware capability of D2D devices. In particular, a mobile user could very easily maintain his spatiotemporal profile recording his continuous whereabouts in time, and the level of his spatiotemporal profile matching that of the other user can be translated into the level of trust they two can have in each other. Since spatiotemporal profiles contain very sensitive personal information, privacy-preserving spatiotemporal matching is needed to ensure that as little information as possible about the spatiotemporal profile of either matching participant is disclosed beyond the matching result. Towards this end, we propose two novel privacy-preserving spatiotemporal matching protocols, which are thoroughly analyzed and evaluated through detailed simulation studies driven by experimental data.**

***Index Terms*—Device-to-Device (D2D) communications, spatiotemporal matching, privacy.**

I. Introduction

DEVICE-TO-DEVICE (D2D) communications are emerging due to the explosive growth of smartphones and tablets. In a typical D2D communication session, two physically proximate mobile devices can directly communicate without involving the base station. D2D communications are widely expected to enhance spectrum efficiency and system throughput, enable efficient cellular traffic offloading, improve energy efficiency and network coverage, and stimulate excitingly new services [2], [3].

Sound mobile authentication techniques are needed for secure and effective D2D communications. In particular, a mobile user interested in initiating a D2D communication session in crowded places may have many candidate D2D partners to choose from, consisting of normal users and possibly attackers. It is thus crucial for the initiating user to select the most trustworthy candidate(s) to ensure effective and secure D2D communications. For example, if an attacker is chosen by mistake, the attacker can obtain sensitive information from the initiating user and also refuse to collaborate in the way he initially agreed to. Such pitfalls can be largely avoided if the initiating user only considers the candidate D2D partners who can be reliably authenticated.

Traditional mobile authentication techniques are insufficient for D2D communications. Specifically, one may think about letting the initiating user seek help from the trusted base station to select trustworthy D2D partners. This approach would place too much burden on base stations and largely offset the benefits of conducting D2D communications. Another plausible approach is to equip every D2D user with a public-key certificate and let the initiating user choose the neighbors with valid public-key certificates. This approach, however, does not permit the initiating user to further distinguish potentially many candidates having a valid certificate.

We propose *spatiotemporal matching* as a promising enabler for secure D2D communications. This technique is motivated by the fact that almost all target D2D devices are location-aware through cellular, WiFi, or GPS technology. A mobile user thus can conveniently maintain his *spatiotemporal profile* recording his continuous whereabouts in time, and the level of his spatiotemporal profile matching that of another mobile user can be translated into the level of trust they two can have in each other. For example, if Alice and Bob discover via spatiotemporal matching that they often go to the same coffee shop or take the same train in the same period, it is natural for Alice to trust Bob over another person whom she only met once before. Spatiotemporal matching is naturally well suited for D2D communications. In particular, if two mobile users have very similar spatiotemporal profiles, it is much more likely that they will stay in each other's communication range for longer time, leading to a longer-live D2D communication session.

There are two critical requirements for releasing the full potential of spatiotemporal matching. In particular, spatiotemporal profiles contain very sensitive personal information, and incautiously disclosing them to the public may cause severe consequences. For example, if an employer surreptitiously discovers an employee's frequent patronage of night clubs, the employee may get unfair treatment at the workplace; if a thief knows the routine of a target victim, he could break in when the victim will be away for a long time. It is thus crucial to have *privacy-preserving* spatiotemporal matching, which ensures that as little information as possible about the spatiotemporal profile of either participant is disclosed beyond the matching result. In addition, spatiotemporal matching is directly performed on mobile devices and thus needs to be very *efficient* in both communication and computation.

J. Sun and Y. Zhang are with the School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ, 85287.
E-mail: {jcsun,yczhang}@asu.edu
R. Zhang is with the Department of Electrical Engineering, University of Hawaii, Honolulu, HI 96822.
E-mail: ruizhang@hawaii.edu
The preliminary version of this paper was published in INFOCOM'13 [1].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2549046, IEEE Internet of Things Journal

We make three main contributions in this paper. First, we coin privacy-preserving spatiotemporal matching as a fundamental primitive for secure D2D communications. Second, we present two solutions towards efficient privacy-preserving spatiotemporal matching. The first solution is a passive approach, in which every mobile user periodically records his locations, and a user's spatiotemporal profile is defined as a set of (time, location) pairs. The second solution is an active approach, where every mobile user continuously broadcasts cryptographic tokens and also records every token he overhears. The tokens a user broadcasts and receives form his spatiotemporal profile. Third, we propose two protocols for the privacy-preserving comparison of two arbitrary active/passive spatiotemporal profiles. The first protocol is based on a novel use of the Bloom filter [4] to enable either user to estimate with tunable accuracy the number of common elements in their spatiotemporal profiles without disclosing too much private information to each other. The second protocol generalizes the first protocol and enables weighted spatiotemporal matching by allowing each user to assign different weights to different elements in his/her profile to obtain the weighted matching result. In addition, we thoroughly analyze both protocols and also evaluate them via detailed simulations driven by experimental data.

The rest of this paper is organized as follows. Section II presents the problem formulation. Section III introduces two approaches for creating spatiotemporal profiles. Section IV presents two protocols for privacy-preserving spatiotemporal matching. Section V theoretically analyzes the proposed protocols. Section VI evaluates the proposed protocols by detailed numerical and experimental results. Section VII surveys the related work. Section VIII concludes this paper.

II. Problem Formulation

A. Problem Statement

We consider a large geographic region such as the NYC metropolitan area with system users as either permanent residents or temporary visitors. Each user carries at least one mobile device which has a WiFi/Bluetooth interface and can acquire his realtime position via on-device positioning software. Such assumptions on device capabilities are fairly justifiable on most current and future mobile devices for D2D communications. Besides, unlike traditional communications between mobile users and the service provider [5], mobile users want to performance secure D2D communications via the WiFi/Bluetooth interfaces on their mobile devices. In addition, time is divided into equal-length *epoch*s, each represented by a globally unique epoch index of l_{epoch} bits. We also postulate that each mobile device, which may traverse different time zones, can always convert its local time into the corresponding epoch index.

Each user u's *spatiotemporal profile* is defined as a set of 2-tuples $(i, loc_{u,i})$, where i and $loc_{u,i}$ denote the epoch index and the corresponding location index, respectively. In our protocol, $loc_{u,i}$ comprises some physical locations closely approximating the user's whereabouts in epoch i. The detailed construction of spatiotemporal profiles is postponed to Section IV.

We use Alice and Bob as two exemplary mobile users throughout the paper. Let $\mathcal{P}_A = \{(i, loc_{A,i})\}_{i>0}^{\infty}$ and $\mathcal{P}_B = \{(i, loc_{B,i})\}_{i>0}^{\infty}$ denote the spatiotemporal profiles of Alice and Bob, respectively. We also let $\mathcal{P}_{A,\alpha\to\beta}$ and $\mathcal{P}_{B,\alpha\to\beta}$ denote their respective spatiotemporal profiles from epochs α to β. Assume that Alice is the initiator of a D2D communication session and that Bob is one of the candidate D2D partners in Alice's proximity. Alice wants to select a trustworthy D2D partner and needs to conduct spatiotemporal matching with every candidate partner. Consider Bob as an example. Alice and Bob need to compare their spatiotemporal profiles from epochs α to β, where α and β are chosen by Alice herself. A complete matching process involves each of them initiating an independent protocol instance. The number of encounters with Bob in Alice's eye in any epoch $i \in [\alpha, \beta]$ equals the number of common locations in their location indexes in epoch i, and the number of encounters with Bob from epochs α to β in her eye equals the sum of total encounters in every epoch from α to β. In the similar fashion, we can define the total number of encounters with Alice from Bob's viewpoint from epochs α to β. We proceed to introduce the following definition.

***Definition** 1:* (**Spatiotemporal Match**) After protocol execution, a *spatiotemporal match* between Alice and Bob from epochs α to β is said to occur if the total number of encounters with Bob exceeds τ_A from Alice's viewpoint, and the total number of encounters with Alice exceeds τ_B from Bob's viewpoint, where τ_A and τ_B are personal thresholds independently chosen by Alice and Bob, respectively.

We assume that Alice and Bob both desire strong spatiotemporal privacy and collaborate only when a spatiotemporal match occurs between them. Our focus is to devise an efficient protocol ensuring that as little information as possible about the spatiotemporal profile of either Alice or Bob is disclosed beyond the matching result. One may think about letting them directly exchange and compare their spatiotemporal profiles under pseudonyms instead of real names so that a known spatiotemporal profile cannot be directly linked to a real identity. Unfortunately, the knowledge of a pseudo-identity's spatiotemporal profile may be disastrous enough, e.g., leading to physical chasing to unveil the corresponding real identity. We thus need a sound solution regardless of pseudonyms.

B. Adversary Model

We assume a honest-but-curious adversary model commonly adopted to study privacy-preserving profile matching [6], [7], [8] or proximity test [9], [10], [11]. With Alice and Bob as an example, they both honestly follow the spatiotemporal matching protocol while having great curiosity about the other's spatiotemporal profile.

We do not consider *continuous fake-profile* attacks and *denial-of-service* (DoS) attacks in this paper. In the former, either matching participant keeps using fake spatiotemporal profiles possibly under different pseudonyms in order to accumulate more information about the other party's spatiotemporal profile as time goes by, while in the latter, an attacker aims at depleting the resources of the other party in the same way. The only feasible countermeasure against both attacks in

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2549046, IEEE Internet of Things Journal

our opinion is for every party to rate-limit the total number of matching requests he/she will accept. Further investigation on these attacks is beyond the scope of this paper.

There might also be external eavesdroppers or physical chasers. The former overhear the messages incurred by a spatiotemporal matching instance and can be easily thwarted by letting the matching participants encrypt the protocol messages. The latter tail a victim user and thus can always have a spatiotemporal profile resembling that of the victim user. There is no sound technical solution to such chasing attacks.

III. Spatiotemporal Profile Construction

In this section, we introduce two approaches for constructing spatiotemporal profile, including a *passive* approach and an *active* approach. In the passive approach, each user records his own spatiotemporal information periodically whereby to construct his spatiotemporal profile. In the active approach, each mobile user continuously broadcasts epoch-specific cryptographic token at an adaptive frequency and also records every token he overhears via WiFi/Bluetooth interface. The spatiotemporal profile of each mobile user is then constructed from the sent and received tokens.

A. A Passive Approach

The passive approach explores the prevalent capability of mobile devices obtaining their physical locations via hybrid GPS, WiFi, and cellular positioning techniques. Assume that each epoch is evenly divided into λ intervals, where $\lambda \geq 1$ is a global parameter. In general, each user passively records his location in the middle of each interval to tolerate synchronization errors among mobile devices. Recall that any user u's spatiotemporal profile is defined in Section II-A as a set of 2-tuples like $(i, loc_{u,i})$. We have $loc_{u,i} = \{p_{u,i}[j]\}_{j=1}^{\lambda}$, where $p_{u,i}[j]$ denotes user u's jth location in epoch i. Consider the exemplary users Alice and Bob with profiles $\mathcal{P}_A = \{i, \{p_{A,i}[j]\}_{j=1}^{\lambda}\}_{i=1}^{\infty}$ and $\mathcal{P}_B = \{i, \{p_{B,i}[j]\}_{j=1}^{\lambda}\}_{i=1}^{\infty}$, respectively. Now they attempt to compare their profiles from epochs α to β, i.e., $\{i, \{p_{A,i}[j]\}_{j=1}^{\lambda}\}_{i=\alpha}^{\beta}$ and $\{i, \{p_{B,i}[j]\}_{j=1}^{\lambda}\}_{i=\alpha}^{\beta}$, equivalent to the comparison of $\lambda(\beta - \alpha + 1)$ location pairs.

We further assume that each physical region of interest (like a metropolitan area) can be approximated by a square called a *level-1* cell. Then we divide the level-1 cell into four equally-sized squares called *level-2* cells, each of which is further divided into four equally-sized squares named as level-3 cells. This process continues until reaching level-θ cells, each having a side length no larger than a desired threshold, and how to determine the cell-division threshold will be discussed later. Note that there are totally 4^{j-1} level-j cells for $\forall j \in [1, \theta]$. Then we assign a unique cell index to the cell(s) on every level. In particular, the index of the level-1 cell is 0, and the indexes of the upper-left, lower-left, upper-right, and lower-right level-2 cells are 00, 01, 02, and 03, respectively. The same indexing rule can be applied to the cells on all levels. The region-division rules are public information and can be downloaded as needed. In practice, each user just needs to have the rules related to the regions he commonly stays in or travel to, so the related storage overhead is negligible.

To facilitate customized spatiotemporal matching, we propose an *adaptive quantization* technique which works by letting each user convert his locations into cell indexes. In particular, assume that Alice and Bob negotiate a common region of interest on which to conduct spatiotemporal matching. Since each region corresponds to a large geographic area, disclosing the regions of interest to each other may not be a serious concern in practice; otherwise, Alice and Bob can apply Private Set Intersection (PSI) [12] to negotiate the common region, which will be very efficient given the limited possible regions. In addition, they agree on a cell level $\xi \in [1, \theta]$ on which the quantization takes place, and the impact of ξ will be discussed shortly. Then Alice converts $\{i, \{p_{A,i}[j]\}_{j=1}^{\lambda}\}_{i=\alpha}^{\beta}$ into $\overline{\mathcal{P}}_{A,\alpha\to\beta} = \{\{\langle i, j, \bar{p}_{A,i}[j]\rangle\}_{j=1}^{\lambda}\}_{i=\alpha}^{\beta}$, where $\bar{p}_{A,i}$ denotes the index of the level-ξ cell that contains $p_{A,i}$. If a certain location is not in the negotiated region, the corresponding cell index is set to some randomly chosen unlikely cell index indicating this abnormality. Similarly, Bob can convert his profile $\{i, \{p_{B,i}[j]\}_{j=1}^{\lambda}\}_{i=\alpha}^{\beta}$ into $\overline{\mathcal{P}}_{B,\alpha\to\beta} = \{\{\langle i, j, \bar{p}_{B,i}[j]\rangle\}_{j=1}^{\lambda}\}_{i=\alpha}^{\beta}$. With adaptive quantization in place, the number of encounters between Alice and Bob equals the number of level-ξ cells they both came across in the same epoch interval, or equivalently the intersection cardinality $|\overline{\mathcal{P}}_{A,\alpha\to\beta} \bigcap \overline{\mathcal{P}}_{B,\alpha\to\beta}|$.

B. An Active Approach

In the active approach, each mobile user continuously broadcasts an epoch-specific cryptographic token at an adaptive frequency and also records every token he overhears via WiFi-direct, Bluetooth, Frequency Hopping, or other available Device-to-Device (D2D) technologies widely used in many applications [13], [11], [14], [15]. For example, the tokens can be exchanged via WiFi/Bluetooth interfaces without requiring the involved parties to explicitly establish any WiFi/Bluetooth connection [14].

Assume that every user u has a unique identifier ID_u and also a secret key k_u. Let $H(\cdot)$ denote any good cryptographic hash function. The token he broadcasts in epoch i is computed as $t_{u,i} = H(k_u, i, ID_u)$ truncated to a given length. User u needs to broadcast token$_{u,i}$ at a personally-chosen frequency to make sure that it can be overheard by sufficient users he encounters, and how to determine this token frequency will be discussed shortly. In addition, user u should use a different pseudonym in every epoch for broadcasting tokens; otherwise, a powerful adversary would be able to associate the tokens he sends in different epochs with him, thus breaching his location privacy.

User u also receives tokens from other users through his WiFi and/or Bluetooth interfaces and only records any token once that he may receive multiple times. Let $\mathcal{R}_{u,i} = \{r_{u,i,j}\}_{j=1}^{n_{u,i}}$ denote the set of $n_{u,i}$ tokens user u receives from others he encounters in epoch i. Any token in $\mathcal{R}_{u,i}$ can serve as the proof that user u was in the WiFi or Bluetooth transmission range of the token sender. User u's whereabouts in epoch i

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2549046, IEEE Internet of Things Journal

can thus be implicitly determined by his physical proximity to other mobile users from which he has received tokens.

We define two types of spatiotemporal profiles for the active approach, including *initiator profile* and *receiver profile*. Recall that user u's spatiotemporal profile is defined in Section II-A as a set of 2-tuples $(i, loc_{u,i})$. The initiator and receiver profiles of user u are defined as $\mathcal{I}_u = \{(i, t_{u,i})\}_{i=1}^{\infty}$ and $\mathcal{R}_u = \{\{(i, r_{u,i,j})\}_{j=1}^{n_{u,i}}\}_{i=1}^{\infty}$.

Continue the example of Alice and Bob. An encounter with Bob (or Alice) occurs in epoch i from Alice's (or Bob's) viewpoint if $t_{A,i} \in \mathcal{R}_{B,i}$ (or $t_{B,i} \in \mathcal{R}_{A,i}$). Suppose they attempt to compare their profiles from epochs α to β to determine the number of their encounters. Let $m_{A,\alpha\to\beta}$ and $m_{B,\alpha\to\beta}$ denote the number of encounters with Bob in Alice's view and with Alice in Bob's view, respectively. We have $m_{A,\alpha\to\beta} = |\mathcal{I}_{A,\alpha\to\beta} \bigcap \mathcal{R}_{B,\alpha\to\beta}|$ and $m_{B,\alpha\to\beta} = |\mathcal{I}_{B,\alpha\to\beta} \bigcap \mathcal{R}_{A,\alpha\to\beta}|$, where $\mathcal{I}_{u,\alpha\to\beta} = \{(i, t_{u,i})\}_{i=\alpha}^{\beta}$ and $\mathcal{R}_{u,\alpha\to\beta} = \{\{(i, r_{u,i,j})\}_{j=1}^{n_{u,i}}\}_{i=\alpha}^{\beta}$ for $u = A$ or B.

C. Discussion

We now discuss some factors that may affect the spatiotemporal profile construction and thus the spatiotemporal matching result. In particular, the passive approach may be affected by the following three factors.

- *Recording frequency*: Each user records his location in the middle of each interval in each epoch of fixed length. The fewer intervals in each epoch, the lower the recording frequency, and the more likely for *false negatives* to occur, in which case a protocol initiator considers the responder a mismatch who actually encountered him multiple times and just did not record the encounter locations due to the low recording frequency. In contrast, the higher the recording frequency, the less likely for false negatives to occur, and the longer every location index in every epoch which will lead to larger computation and communication overhead.
- *Quantization granularity*: The granularity of spatiotemporal matching can be controlled by choosing a proper quantization level $\xi \in [1, \theta]$. A larger ξ can lead to finer-grained matching at the sacrifice of spatiotemporal privacy and matching efficiency, while a smaller ξ can lead to better spatiotemporal privacy at the cost of coarser-grained matching and longer spatiotemporal matching time.
- *Imperfect quantization*: Our quantization process may cause some ambiguity. For example, if the recorded locations of Alice and Bob in the same interval are near the upper-left and lower-right corners of the same level-ξ cell, they will be quantized to the same level-ξ index and thus translated into one encounter. In contrast, if the two locations are in adjacent level-ξ cells and close to each other along the cell boundary, they, however, will be quantized to different level-ξ indexes and translated into a non-encounter.

Similarly, the active approach may be affected by the following two factors.

- *Token broadcasting frequency*: The more frequently a user broadcasts an epoch-specific token, the more users he encounters can receive the token, and the less likely for *false negative*s to occur, in which case a protocol initiator deems the responder a mismatch who actually encountered him many times and just did not receive sufficient tokens from him in the matching epochs due to channel errors, missing the time points for token transmissions, etc. In contrast, the less frequently a token is broadcasted in one epoch, the less energy the user consumes at the cost of higher false-negative rates. The user can adopt an adaptive method by letting a user dynamically adjust his broadcasting frequency proportional to his moving speed which can be readily inferred based on the accelerometer increasingly available on mobile devices. The intuition is that the users encountered by a high-speed (or low-speed) user may quickly (slowly) move out of his WiFi/Bluetooth transmission range, so he can increase (or decrease) the token frequency accordingly.
- *Uniqueness of each user's broadcasted tokens*: The correctness of our protocols depends on $\{t_{A,i}\}_{i=\alpha}^{\beta}$ (or $\{t_{B,i}\}_{i=\alpha}^{\beta}$) being all unique in our previous example. Recall that the token any user u (i.e., $t_{u,i}$) broadcasts in epoch i equals $H(k_u, i, ID_u)$ truncated to a given length. Due to the randomness of the hash output, it is likely that the tokens user u sent in adjacent epochs might be the same. A simple remedy is to let user u keep a FIFO queue of size equal to the longest matching epoch-interval he may be interested in. The queue records all the recently used tokens. Consider epoch i as an example. If the truncated $H(k_u, i, ID_u)$ is in the queue, user u tries $H(k_u, i, ID_u, 1)$, $H(k_u, i, ID_u, 2)$, $\ldots$, until finding a token not in the queue, which will be used as $t_{u,i}$ and inserted into the queue.

IV. Privacy-Preserving Spatiotemporal Matching

In this section, we present two novel privacy-preserving spatiotemporal matching protocols.

From the discussion of Section III, we can see that the problem of privacy-preserving spatiotemporal matching boils down to the problem of enabling two users (e.g., Alice and Bob) to learn the cardinality of the intersection of their spatiotemporal profiles represented by two sets Ψ_A and Ψ_B, respectively, while disclosing as little additional information as possible beyond the matching result. In particular, if the passive approach is adopted to construct the spatiotemporal profile, we have $\Psi_A = \overline{\mathcal{P}}_{A,\alpha\to\beta}$ and $\Psi_B = \overline{\mathcal{P}}_{B,\alpha\to\beta}$. Similarly, under the active approach, we have $\Psi_A = \mathcal{I}_{A,\alpha\to\beta}$ and $\Psi_B = \mathcal{R}_{A,\alpha\to\beta}$ if Alice's point of view is considered, and $\Psi_B = \mathcal{I}_{B,\alpha\to\beta}$ and $\Psi_B = \mathcal{R}_{B,\alpha\to\beta}$ if Alice's point of view is considered.

A. A Bloom-filter-based Privacy-Preserving Spatiotemporal Matching Protocol

Our first spatiotemporal matching protocol is motivated by the observation that an accurate estimation of the number of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2549046, IEEE Internet of Things Journal

encounters may suffice in practice and involves a novel use of the Bloom filter [4].

A Bloom filter [4] is a space-efficient probabilistic data structure [16], [17], [11] for set-membership testing. Assume that a w-bit Bloom filter is used for a data set $\{s_i\}_{i=1}^{d}$, which has every bit initialized to 0. Let $\{h_a(\cdot)\}_{a=1}^{k}$ denote k different hash functions, each with output in $[1, w]$. Every element s_i is added into the Bloom filter by setting all bits at positions $\{h_a(s_i)\}_{a=1}^{k}$ to 1. To check the membership of an arbitrary element e in the given data set, we can simply verify whether all the bits at positions $\{h_a(e)\}_{a=1}^{k}$ have been set. If not, e is certainly not in the data set; otherwise, it is in the data set with some probability jointly determined by d, w, and k.

Our protocol involves Alice and Bob each using a different set of hash functions to construct a Bloom filter based on his/her spatiotemporal profile. In particular, let $\mathcal{H}$ denote a large and public pool of hash functions with each indexed by a unique identifier. Assume that Alice and Bob are to find out $|\Psi_A \cap \Psi_B|$. Without loss of generality, let $\Psi_A = \{a_1, \ldots, a_{n_A}\}$ and $\Psi_B = \{b_1, \ldots, b_{n_B}\}$, where $n_A = n_B$ if the passive approach is adopted and $n_A \neq n_B$ otherwise. The following operations are done in sequence for Alice to obtain an estimated $\hat{m}_A$ about $m_A = |\Psi_A \cap \Psi_B|$, where m_A represents the number of encounters with Bob in Alice's view.

1. Alice sends a spatiotemporal matching request with n_A to Bob.
2. If $n_A > n_B$, Bob adds $n_A - n_B$ dummy elements that are definitely not in Ψ_A to obtain his new spatiotemporal profile Ψ'_B. Bob then randomly chooses k hash functions from $\mathcal{H}$ with indexes denoted by $\mathcal{H}_B$ and then inserts each element in his profile $\widetilde{\Psi}_B$ into a w-bit Bloom filter (denoted by BF_B) with different $l < k$ functions randomly selected from $\mathcal{H}_B$ and $k - l$ random hash functions outside $\mathcal{H}$. Finally, Bob returns n_B, $\mathcal{H}_B$, and BF_B to Alice.
3. If $n_B > n_A$, Alice adds $n_B - n_A$ dummy elements that are definitely not in Ψ_B to obtain his new spatiotemporal profile Ψ'_A.
4. Alice constructs a w-bit Bloom filter (denoted by BF_A) based on the hash functions specified in $\mathcal{H}_B$ and her profile Ψ'_A. Then she counts the number of common bit-0 positions in BF_A and BF_B (denoted by n_0) whereby to compute

$$\hat{m}_A = \frac{2kn - w(\ln w - \ln n_0)}{l}, \tag{1}$$

where $n = \max(n_A, n_B)$. The correctness and accuracy of this estimation will be analyzed in Section V-B.

Likewise, Bob can initiate a spatiotemporal matching process to estimate the number of encounters with Alice $\hat{m}_B$ from his point of view. Finally, they can jointly determine whether there is a successful spatiotemporal matching after independently comparing $\hat{m}_A$ (or $\hat{m}_B$) with the personal threshold τ_A (or τ_B).

We have some important remarks to make. First, since Alice and Bob use some common hash functions in $\mathcal{H}_B$ to construct their respective Bloom filter, the same elements in their spatiotemporal profiles (if any) are likely to set the same bit positions. So we can estimate the number of common elements via the number of common bit-0 and/or bit-1 positions. Second, the reason for Bob using $k - l$ random hash functions unknown to Alice for each element is to prevent Alice from estimating Bob's spatiotemporal presence by simple Bloom set-membership tests. In particular, if Bob uses the same k hash functions in $\mathcal{H}_B$ to generate BF_B, Alice can easily test whether some possible element is in BF_B, which is equivalent to breaching Bob's spatiotemporal privacy. This set-membership test is less critical to the active approach because the adversary does not know the user's secret keys and can only randomly guess the broadcasted tokens. However, it is critical to the passive approach in which all the possible pairs of epoch and cell indexes are known to the adversary as well. The choice of k and l will be detailed in Section V-B. Finally, the construction of many different hash functions for implementing the Bloom filter is also very important. One common method is to seed a cryptographic hash function such as SHA-2 with the indexes of hash functions we want. There are also some more efficient realizations of many hash functions specifically for the Bloom filter [18], [19].

B. A Weighted Privacy-Preserving Spatiotemporal Matching Protocol

We now generalize the above protocol to support weighted privacy-preserving spatiotemporal matching, which is defined as follows.

Definition 2: (**Weighted Spatiotemporal Match**) Assume that Alice and Bob each assign different weights for encounter at different locations and times. A weighted spatiotemporal match between Alice and Bob is said to occur if the weighted sum of encounters with Bob exceeds τ_A from Alice's viewpoint, and the weighted sum of encounters with Alice exceeds τ_B from Bob's viewpoint, where τ_A and τ_B are personal thresholds independently chosen by Alice and Bob, respectively.

More specifically, consider Alice and Bob with spatiotemporal profiles $\Psi_A = \{a_1, \ldots, a_{n_A}\}$ and $\Psi_B = \{b_1, \ldots, b_{n_B}\}$, respectively. Assume that Alice assigns a weight $w_{A,i}$ for possible encounter corresponding to element a_i in Ψ_A for each $i \in [1, n_A]$, and that Bob assigns a weight $w_{B,j}$ for possible encounter corresponding to element b_j in Ψ_B for each $j \in [1, n_B]$. The weighted count of encounters with Bob from Alice's viewpoint is computed as

$$m_A = \sum_{i=1}^{n_A} c_i \tag{2}$$

where

$$c_i = \begin{cases} w_{A,i} & \text{if } a_i \in \Psi_B, \\ 0 & \text{otherwise,} \end{cases}$$

and the weighted count of encounters with Alice from Bob's viewpoint can be computed accordingly.

We observe that weighted spatiotemporal matching can be converted into spatiotemporal matching between two spatiotemporal profiles constructed from weight sets. Specifically, assume that that $w_{A,i} \in \{1, \ldots, \mathsf{w}\}$ for all $i \in [1, n_A]$,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2549046, IEEE Internet of Things Journal

where w is a publicly known parameter. Alice can construct a new spatiotemporal profile Ψ'_A from her original profiles Ψ_A and weight set $\mathcal{W}_A = \{w_{a,i}\}_{i=1}^{n_A}$ as follow. For each element $a_i \in \Psi_A$ with weight assignment $w_{A,i}$, Alice converts a_i into $w_{A,i}$ different elements $a_i||1$, $a_i||2$, ..., $a_i||w_{A,i}$. As a result, Alice obtains her new spatiotemporal profile $\Psi'_A = \{\{a_i||j\}_{j=1}^{w_{A,i}}\}_{i=1}^{n_A}$. On the other hand, Bob can construct a new spatiotemporal profile Ψ_B in a different way. For each element $b_i \in \Psi_B$ with weight $w_{B,i} \in \mathcal{W}_B$, Bob converts b_i into w different elements $b_i||1, b_i||2, \ldots, b_i||\mathsf{w}$ to obtain a new spatiotemporal profile $\Psi'_B = \{\{b_i||j\}_{j=1}^{\mathsf{w}}\}_{i=1}^{n_B}$. It follows that

$$m_A = |\Psi'_A \cap \Psi'_B| \ .$$

Assume Alice and Bob have their respective spatiotemporal profiles Ψ_A and Ψ_B via either the passive or active approaches. The following operations are done in sequence to allow Alice to obtain an estimated $\hat{m}_A$ about $m_A = |\Psi'_A \cap \Psi'_B|$.

1. Alice creates a new spatiotemporal profile $\Psi'_A = \{\{a_i||j\}_{j=1}^{w_{A,i}}\}_{i=1}^{n_A}$.
2. Alice sends a weighted spatiotemporal matching request with $w_A = \sum_{i=1}^{n_A} w_{A,i}$ to Bob.
3. Bob creates a new spatiotemporal profile $\Psi'_B = \{\{b_i||j\}_{j=1}^{\mathsf{w}}\}_{i=1}^{n_B}$, and calculates $w_B = n_B\mathsf{w}$.
4. If $w_A > w_B$, Bob adds $w_A - w_B$ dummy elements that are definitely not in Ψ'_A to Ψ'_B to obtain his new spatiotemporal profile Ψ''_B. Bob then randomly chooses k hash functions from $\mathcal{H}$ with indexes denoted by $\mathcal{H}_B$ and then inserts each element in his profile Ψ''_B into a w-bit Bloom filter BF_B with different $l < k$ functions randomly selected from $\mathcal{H}_B$ and $k - l$ random hash functions outside $\mathcal{H}$. Finally, Bob returns w_B, $\mathcal{H}_B$, and BF_B to Alice.
5. If $w_B > w_A$, Alice adds $w_B - w_A$ dummy elements that are definitely not in Ψ'_B to Ψ'_A to obtain her new spatiotemporal profile Ψ''_A.
6. Alice constructs a w-bit Bloom filter BF_A using the hash functions specified in $\mathcal{H}_B$ and her profile Ψ''_A. Then she counts the number of common bit-0 positions in BF_A and BF_B (denoted by n_0) whereby to compute

$$\hat{m}_A = \frac{2kn - w(\ln w - \ln n_0)}{l} \ , \tag{3}$$

where $n = \max(w_A, w_B)$.

V. Performance Analysis

In this section, we analyze the performance of the proposed protocols.

A. Performance Metrics

We use the following metrics to evaluate our protocols.

1) Accuracy: The following standard (ϵ, δ) guarantee is used to measure the accuracy of the protocol output,

$$Pr[(1-\epsilon)m \leq \hat{m} \leq (1+\epsilon)m] > 1 - \delta, \tag{4}$$

where m is the actual number of common elements (or encounters) in , and $\hat{m}$ is the estimation of m output by a spatiotemporal matching protocol.

2) Privacy: We quantify spatiotemporal privacy by the Shannon entropy, a commonly used measure of uncertainty.

We take Bob as an example to analyze the his spatiotemporal privacy under the passive approach. Recall that Bob's quantized spatiotemporal profile from epochs α to β is $\Psi_B = \overline{\mathcal{P}}_{B,\alpha\to\beta} = \{\{i, j, \bar{p}_{B,i}[j]\}_{j=1}^{\lambda}\}_{i=\alpha}^{\beta}$, where $\bar{p}_{B,i}$ denotes a level-ξ cell index. The only information Alice knows about $\overline{\mathcal{P}}_{B,\alpha\to\beta}$ before protocol execution includes the parameters α, β, and λ. Since there are total $N = 4^{\xi-1}$ level-ξ cell indexes, each of them is equally likely to be $\bar{p}_{B,i}[j]$ from Alice's viewpoint. There are thus total $N^{\lambda(\beta-\alpha+1)}$ candidate quantized profiles for $\overline{\mathcal{P}}_{B,\alpha\to\beta}$ with equal probability from Alice's viewpoint. So the maximum spatiotemporal privacy of Bob with regard to Alice (i.e., the maximum uncertainty of his spatiotemporal profile to Alice) in bits can be computed as

$$\mathbf{E}^* = \log_2 N^{\lambda(\beta-\alpha+1)} = 2\lambda(\beta-\alpha+1)(\xi-1). \tag{5}$$

To make the analysis of the spatiotemporal privacy of Bob under the active approach tractable and comparable with the passive approach, we make the following assumptions. We assume that during each epoch, Alice and Bob each wander in one level-ξ cell as in the passive approach and that Alice keeps broadcasting a unique token at sufficiently high frequency such that Bob always receives Alice's token if they are in the same cell. In addition, we ignore the case in which Bob receives Alice's token while they are in two different cells, e.g., they are close two the boundary of two adjacent cells. Similar to the analysis of the passive approach, since there are total $N = 4^{\xi-1}$ level-ξ cells, each of them is equally likely to be the cell Bob resides from Alice's viewpoint. There are total n_A^N candidate quantized profiles with equal probability from Alice's viewpoint. So the maximum spatiotemporal privacy of Bob with regard to Alice in bits (i.e., the maximum uncertainty of his spatiotemporal profile to Alice) can be computed as

$$\mathbf{E}^* = \log_2 N^{n_B} = 2n_B(\xi-1). \tag{6}$$

After the execution of either protocol, Alice can know more information about the probability of each candidate profile being Bob's profile whereby to reduce the entropy or uncertainty, which we will analyze shortly. The maximum spatiotemporal privacy of Alice with regard to Bob can be analyzed in a similar fashion and thus omitted here.

3) Overhead: We will measure the communication and computation overhead of the spatiotemporal matching protocol using the number of hash computations and the number of bits transferred between two users during protocol execution, respectively.

B. Analysis of the Spatiotemporal Matching Protocol

1) Accuracy Analysis: We have the following theorem regarding the accuracy of the privacy-preserving spatiotemporal matching protocol.

***Theorem** 1:* Given the number of common bit-0 positions n_0 in the w-bit Bloom filters BF_A and BF_B constructed

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2549046, IEEE Internet of Things Journal

in the spatiotemporal matching protocol, Alice can estimate $|\Psi_A \bigcap \Psi_B|$ as

$$\hat{m} = \frac{2nk - w(\ln w - \ln n_0)}{l} , \tag{7}$$

where $n = \max(n_A, n_B)$. Assuming that $\epsilon m \geq 1$, $\hat{m}$ is an (ϵ, δ) estimation of m if

$$\delta \geq \frac{w(e^{\frac{2nk}{w}} - (1 + \frac{2nk}{w}))}{l^2\epsilon^2 m^2}. \tag{8}$$

We give the proof of Theorem 1 in Appendix A.

2) Privacy Analysis: For the passive approach, the privacy analysis of the spatiotemporal matching protocol is given by the following theorem.

***Theorem** 2:* Assuming that Bob constructs a w-bit Bloom filter BF_B from his level-ξ quantized profile $\Psi_B = \{\{i, j, \bar{p}_{B,i}[j]\}_{j=1}^{\lambda}\}_{i=\alpha}^{\beta}$ using l functions from $\mathcal{H}_B$ and $k - l$ functions unknown to Alice. After transmitting BF_B and $\mathcal{H}_B$ to Alice, his remaining privacy of Ψ_B against Alice is given by

$$\mathbf{E} = \lambda(\alpha + \beta - 1)\mathbf{E}[i, j] , \tag{9}$$

where

$$\begin{aligned} \mathbf{E}[i, j] &= \sum_{x=1}^{N} \binom{N}{x} P^x (1 - P)^{N-x} \log_2 x, \\ P &= \sum_{i=l}^{k} \binom{k}{i} p^i (1 - p)^{k-i} , \\ p &= 1 - e^{-\frac{\lambda(\alpha+\beta-1)k}{w}} . \end{aligned} \tag{10}$$

We give the proof of Theorem 2 in Appendix B.

The following theorem is about the privacy of the spatiotemporal matching protocol under the active approach.

***Theorem** 3:* Assuming that Bob constructs a w-bit Bloom filter BF_B from Ψ'_B using l functions from $\mathcal{H}_B$ and $k - l$ functions unknown to Alice. After transmitting BF_B and $\mathcal{H}_B$ to Alice, his remaining privacy of Ψ_B against Alice is

$$\mathbf{E} = \sum_{x=0}^{n_A} \binom{n_A}{x} P^x (1 - P)^{n_A - x} \log_2 N^{n_B - x} , \tag{11}$$

where Ψ_A and Ψ_B are the spatiotemporal profiles of Alice and Bob, respectively,

$$\begin{aligned} P &= \sum_{i=l}^{k} \binom{k}{i} p^i (1 - p)^{k-i} , \\ p &= 1 - e^{-\frac{nk}{w}} . \end{aligned} \tag{12}$$

We give the proof of Theorem 3 in Appendix C.

3) Overhead Analysis: The spatiotemporal matching protocol involves Alice and Bob each performing kn hash operations, where $n = \max(n_A, n_B)$, which is very efficient. The communication overhead mainly comes from the transmission of one Bloom filter and is of w bits.

C. Analysis of Weighted Spatiotemporal Matching Protocol

1) Accuracy Analysis: The accuracy of the weighted spatiotemporal matching protocol is guaranteed by the following theorem.

***Theorem** 4:* Given the number of common bit-0 positions n_0 in the w-bit Bloom filters BF_A and BF_B constructed from Ψ''_A and Ψ''_B, respectively, in the weighted spatiotemporal matching protocol, Alice can estimate the result of the weighted spatiotemporal matching as

$$\hat{m} = \frac{2kn - w(\ln w - \ln n_0)}{l} , \tag{13}$$

where $n = \max(w_A, w_B)$n. Assuming that $\epsilon m \geq 1$, $\hat{m}$ is an (ϵ, δ) estimation of m if

$$\delta \geq \frac{w(e^{\frac{2kn}{w}} - (1 + \frac{2kn}{w}))}{l^2\epsilon^2 m^2}. \tag{14}$$

The proof of Theorem 4 is similar to that of Theorem 1 and is thus omitted here.

2) Privacy Analysis: The privacy guarantee of weighted spatiotemporal matching protocol under the passive approach is given as follows.

***Theorem** 5:* Let BF_B denote a w-bit Bloom filter Bob constructs on his converted spatiotemporal profile Ψ''_B from epoch α to β using l functions from $\mathcal{H}_B$ and $k - l$ functions unknown to Alice. After transmitting BF_B and $\mathcal{H}_B$ to Alice, his remaining privacy of Ψ''_B against Alice is

$$\mathbf{E} = \lambda(\alpha + \beta - 1)\mathbf{E}[i, j] , \tag{15}$$

where

$$\begin{aligned} \mathbf{E}[i, j] &= \sum_{x=1}^{N} \binom{N}{x} P^{x\mathsf{w}} (1 - P^{\mathsf{w}})^{N-x} \log_2 x, \\ P &= \sum_{i=l}^{k} \binom{k}{i} p^i (1 - p)^{k-i} , \\ p &= 1 - e^{-\frac{nk}{w}} , \\ n &= \max(w_A, w_B) . \end{aligned} \tag{16}$$

We give the proof of Theorem 5 in Appendix D.

The privacy guarantee of weighted spatiotemporal matching protocol under the active approach is given by the following theorem.

***Theorem** 6:* Let BF_B denote a w-bit Bloom filter Bob constructs on his converted spatiotemporal profile Ψ''_B using l functions from $\mathcal{H}_B$ and $k - l$ functions unknown to Alice. Assume we adopt level-ξ quantized After transmitting BF_B and $\mathcal{H}_B$ to Alice, his remaining privacy of Ψ''_B against Alice is

$$\mathbf{E} = \sum_{x=0}^{n_A} \binom{n_A}{x} P^{x\mathsf{w}} (1 - P^{\mathsf{w}})^{n_A - x} \log_2 N^{n_B - x} , \tag{17}$$

where n_A and n_B are the sizes of spatiotemporal profiles of Alice and Bob before conversion, respectively,

$$\begin{aligned} P &= \sum_{i=l}^{k} \binom{k}{i} p^i (1 - p)^{k-i} , \\ p &= 1 - e^{-\frac{nk}{w}} , \\ n &= \max(w_A, w_B) . \end{aligned} \tag{18}$$

We give the proof of Theorem 6 in Appendix E.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2549046, IEEE Internet of Things Journal

3) Overhead Analysis: Similar to the spatiotemporal matching protocol, the weighted spatiotemporal matching protocol involves Alice and Bob each performing kn hash operations, where $n = \max(w_A, w_B)$. The communication overhead mainly comes from the transmission of one Bloom filter and is of w bits.

VI. Performance Evaluation

In this section, we evaluate the two proposed protocols using simulations.

A. Simulation Settings

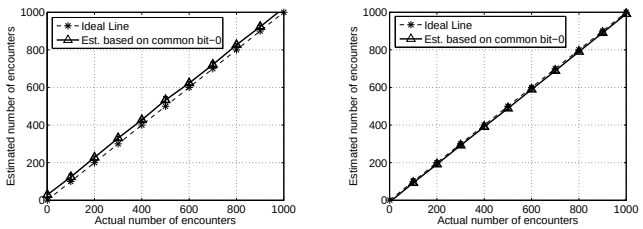
In the preliminary version of this paper [1], we have shown that our protocol incurs significantly lower computation and communication overhead than traditional PSI-CA protocols [20], [12] based on computationally expensive public-key operations. Our simulation studies here will focus on the impact of various parameters on the accuracy and privacy of the spatiotemporal matching protocols.

We assume that the quantization is done on the level $\xi = 6$, i.e., $N = 4^{\xi-1} = 1024$. In addition, our experiments are on a Dell desktop with 2.67 GHz CPU, 9 GB RAM, and Windows 7 64-bit Professional, the evaluation program is written in Java, and every data point represents the average of 1000 runs. As discussed, a complete spatiotemporal matching involves Alice and Bob each initiating one protocol execution, but we only show the results for one protocol execution for simplicity. In addition, we set δ to 0.02, and ϵ is the relative error.

B. Simulation Results

Fig. 1(a) compares the estimated number of encounters $\hat{m}$ with the actual number of encounters m, when $k = 20$, $l = 16$, $n = 1000$, and $w = 40000$. We can see that the estimator in Eq. (3) is always biased. The reason is that traditional analysis about the w-bit Bloom filter assumes that every bit position is set to bit-1 for any of n elements with equal probability $1/w$. In practice, however, the probability that one position is set to bit-1 is not independent of other positions: when one position is set to bit-1, it slightly reduces the probability that other positions are set to bit-1 [16], [21], [22]. Therefore, the actual number of bit-1 positions n_1 in the Bloom filter is a little smaller than that obtained via theoretical analysis, and the actual number of bit-0 positions n_0 in the Bloom filter is a little larger than that obtained via theoretical analysis. Since $\hat{m} = \frac{2kn - w(\ln w - \ln n_0)}{l}$, we can expect $\hat{m}$ to be larger than the true value m.

We resolve the biased estimation by letting $\hat{m} = \frac{2k\hat{n} - w(\ln w - \ln n_0)}{l}$, where $\hat{n} = \frac{\ln(n_{A0}/w)}{k \ln(1-1/w)}$, n_{A0} is the number of bit-0 positions in BF_A. Fig. 1(b) shows that this new estimator is almost unbiased and matches well with m. The reason is that using estimated number of elements $\hat{n}$ instead of the real number of elements $n = \lambda(\beta - \alpha + 1)$ takes into account the above difference between observed and theoretical numbers of bit-0 and bit-1 positions. So we will use this modified estimator hereafter whose effectiveness will be further evidenced.



(a) Impact of n (b) Impact of $\hat{n}$

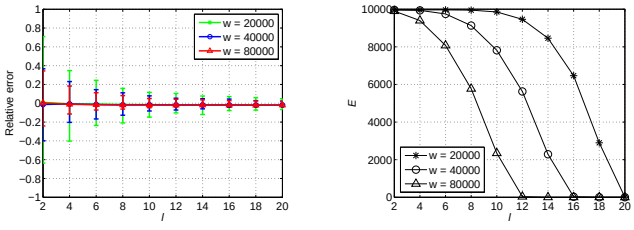
Fig. 1. The estimation accuracy of the advanced protocol.

Fig. 2 shows the impact of l (the number of common hash functions Bob chooses to insert each of his elements) on the performance of advanced protocol, when $n = 1000$, $m = 500$, and $k = 20$. We can see from Fig. 2(a) that the more common hash functions (i.e., larger l), the smaller the variance of the relative error $|\hat{m}_A - m|/m$ (i.e., the more accurate the estimation). The reason is that the more common hash functions, the more common bit-0 positions in BF_A and BF_B, leading to fewer possible Bloom filters for Alice and Bob, and the smaller estimation error variance, because the estimation error mainly comes from the uncertainty of BF_A and BF_B. In addition, the more common hash functions Alice and Bob share, the lower the probability that a random location index having corresponding bits set to bit-1 by at least l out of k hash functions, and thus the lower remaining entropy left for Bob's location profile after Alice testing all possible location indexes. It is thus of no surprise to see that Bob's remaining privacy against Alice decreases with both l and w.

Fig. 3 shows the impact of n (the number of location indexes of each user) on the performance of advanced protocol, when $k = 20$, $w = 40000$, and $m = n/2$. We can see that as n increases, the relative error becomes larger. The reason is that when the Bloom-filter length w is fixed, the more elements inserted, the fewer common bit-0 positions in BF_A and BF_B, the more possible Bloom filters for Alice and Bob, which leads to higher estimation variance. In contrast, Bob's remaining privacy increases as n increases because the fewer bits-0 positions in BF_A, the higher the probability of a random location index having corresponding bits set to bit-1 by at least l out of k known hash functions, and the higher remaining entropy for Bob's location profile from Alice's point of view after testing all possible location indexes.

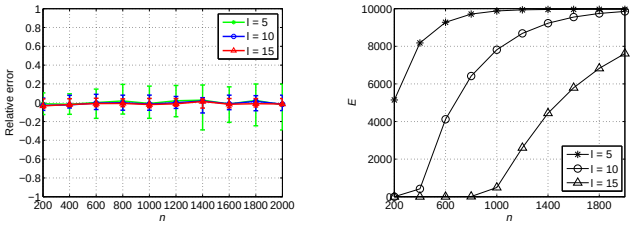
Fig. 4 shows the impact of w (the Bloom-filter length) on the performance of advanced protocol, when $k = 20$, $n = 1000$, and $m = 500$. We can see that the relative error decreases as w increases. This is because when the number of elements n is fixed, increase in w leads to more common bit-0 positions. The more common bit-0 positions, the fewer possible Bloom filters for Alice and Bob, and thus the smaller estimation error variance. In addition, Bob's remaining privacy against Alice decreases as w increases. The reason is that the longer the Bloom filter, the lower the probability that a random location index having corresponding bits set to bit-1 by at least l out of k known hash functions, and thus the lower remaining entropy left for Bob's location profile after Alice testing all possible

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2549046, IEEE Internet of Things Journal



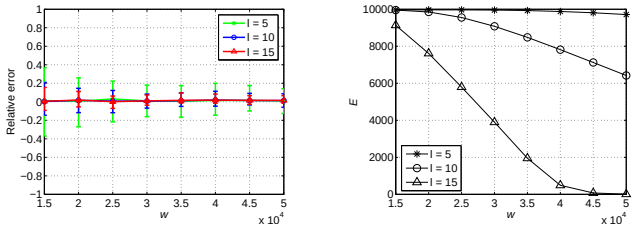
(a) Relative error (b) Privacy

Fig. 2. The impact of l, the number of common hash functions.



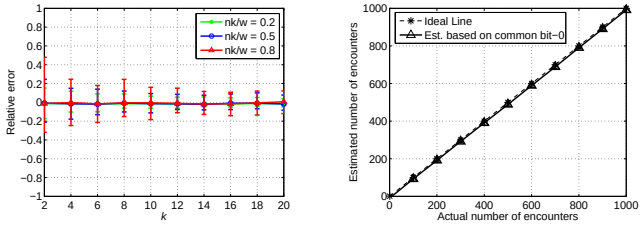
(a) Relative error (b) Privacy

Fig. 3. The impact of n, the cardinality of profiles.



(a) Relative error (b) Privacy

Fig. 4. The impact of w, the length of the Bloom filter.



(a) Relative error (b) Privacy

Fig. 5. The impact of k, the total number of hash functions.

location indexes.

Fig. 5 shows the impact of k (the total number of hash functions for Bloom filter construction) on the performance of advanced protocol, when $n = 1000$, $m = 500$, and the ratios l/k and nk/w are both fixed. It is obvious that the relative error decreases as k increases. The reason is that when k increases, l and w also increase proportionally with fixed l/k and nk/w. Recall that the variance of the $\hat{m}$ is inversely proportional to w/l^2 for fixed ρ (cf. Eq. (27)). As l increases, the variance of estimation error decreases. In addition, Bob's remaining privacy against Alice decreases as k increases. The reason is that the probability that at least l bit positions have been set decreases as k increases, which leads to lower remaining entropy.

From the above figures, a general conclusion we can draw is that there is an inherent tradeoff between matching accuracy and spatiotemporal privacy: the more accuracy Alice wants, the lower spatiotemporal privacy Bob can enjoy, and vice versa.

VII. Related Work

In this section, we discuss work in several areas which is most germane to our work in this paper.

There is some work on encounter-based matching [23], [24]. Manweiler *et al.* [23] discussed the privacy concerns for some missed-connection sites, which allows anonymous users to rediscover strangers that they ever encountered. In their follow-on work [24], they proposed to let mobile users exchange spatiotemporal credentials when encountering each other and later attempt to discover each other via a third-party server which acts as a rendezvous point for the users. In contrast, our protocols focus on a more general problem and are completely distributed without requiring mobile users to interact with a third-party server in most scenarios.

Existing proposals for private matching can be generally classified into two categories. The first category such as [6], [25], [7], [25], [26], [27], [28], [29] assumes that each participant's personal profile consists of multiple attributes chosen from a public set of attributes [26], which can be various interests [7], disease symptoms [25], or friends [6] in different contexts. Private matching is then converted into Private Set Intersection (PSI) [30], [31], Private Set Intersection Cardinality (PSI-CA) [20], [12], or their variations, whereby two mutually mistrusting parties, each holding a private data set, jointly compute some function over the two sets without leaking any additional information to either party. The second category such as [32], [8], [33], [34], [35], [36] assumes that user profile can be modeled as a multi-dimensional vector, where each element is an integer indicating the priority level, knowledge level [35], or interest level [34] of users on the corresponding attribute. Private matching is then converted into the secure computation of various functions over two vectors. Our work belongs to the first category but does not rely on computationally expensive PSI-CA.

Private proximity testing aims at testing the physical proximity of two users at some discrete time points in a privacy-preserving fashion. In [9], private proximity test is reduced to private equality test based on some location tags often sent by third parties, and the sketches of GSM location tags [10] are for efficient private proximity test. In contrast, our protocols evaluate the proximity of two users for any desired continuous time period. Moreover, our most efficient protocol does not involve expensive cryptographic operations unlike [9], [10].

VIII. Conclusion

In this paper, we have motivated and formulated privacy-preserving spatiotemporal matching as a fundamental primitive for supporting secure D2D communications. We presented

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2549046, IEEE Internet of Things Journal

a novel privacy-preserving spatiotemporal matching protocol and a novel weighted privacy-preserving spatiotemporal matching protocol based on a novel use of the Bloom filter. Detailed performance analysis and evaluation confirmed the high efficacy and efficiency of our solutions.

APPENDIX A
PROOF OF THEOREM 1

Proof: For each bit position of either Bloom filter, the probability that it is set to bit-1 by a common element with l common hash functions is given by

$$p = 1 - (1 - \frac{1}{w})^{ml} \approx 1 - e^{-\frac{ml}{w}}. \tag{19}$$

The probability that it is set to bit-1 in all the other cases is given by

$$q = 1 - (1 - \frac{1}{w})^{nk-ml} \approx 1 - e^{-\frac{nk-ml}{w}}. \tag{20}$$

Therefore, the probability that a position is bit-0 in both BF_A and BF_B (i.e., common bit-0 position) is given by

$$P_0 = (1-p)(1-q)^2 = e^{-\frac{ml}{w}} e^{-\frac{2(nk-ml)}{w}}. \tag{21}$$

Since Alice can count the number of common bit-0 positions n_0 in BF_A and BF_B, the following equation can be established

$$P_0 = e^{-\frac{ml}{w}} e^{-\frac{2(nk-ml)}{w}} = \frac{n_0}{w} . \tag{22}$$

Solving this equation, we have

$$\hat{m} = \frac{2nk - w(\ln w - \ln n_0)}{l} . \tag{23}$$

Next, we derive the variance. We cast the problem into RFID tag estimation and refer to the results in [37]. The RFID system with t tags divides a time period into f slots and let each RFID tag randomly select one of f slots to respond. One slot may be responded by zero, one, or multiple tags. The expected number of zero-response slots is nearly $fe^{-t/f}$. Knowing the number of zero-response slots, the system administrator can estimate the number of present RFID tags. Our estimation method based on the Bloom filter is similar to RFID tag estimation if we consider common bit-1 positions and common bit-0 positions as multiple-response and zero-response slots in the RFID system, respectively. The expected number of common bit-0 positions of BF_A and BF_B is nearly $we^{-(2nk-ml)/w}$. Knowing the number of common bit-0 positions, we can estimate the intersection size m.

Let $\rho = \frac{2nk-ml}{w}$. According to Theorem 1 in [37], we have $n_0 \sim \mathcal{N}(\mu, \sigma^2)$, where

$$\mu = w(1 - \frac{1}{w})^{2nk-ml} = we^{-\rho} , \tag{24}$$

$$\sigma^2 = we^{-\rho}(1 - (1+\rho)e^{-\rho}) . \tag{25}$$

We can view μ as a function of the true number of common elements, denoted by $\mu(m)$. Since $\mu(m)$ is monotonic continuous functions of m, it has a unique inverse, denoted by $g()$, i.e., $g(\mu(m)) = m$. Let $2nk - ml \to \infty$ and $w \to \infty$, while maintaining $\frac{2nk-ml}{w} = \rho$. Since $g(\mu(m)) = m$, differentiating this equation with respect to m, we get $g'(\mu(m))\mu'(m) = 1$. it follows that $g'(\mu(m)) = \frac{1}{\mu'(m)}$. According to Theorem 6 in [37], the variance of common bit-0 estimation of m is given by

$$\delta_0 = \sigma^2(m)[g'(\mu(m))]^2 = \frac{\sigma^2(m)}{[\mu'(m)]^2} . \tag{26}$$

Since $\mu = we^{-\frac{2nk-ml}{w}}$ and $\sigma^2 = we^{-\rho}(1-(1+\rho)e^{-\rho})$. Differentiating $\mu(m)$ with respect to m, we can obtain $\frac{d\mu(m)}{dm} = le^{-\rho}$. Therefore we have

$$\delta_0 = \frac{we^{-\rho}(1-(1+\rho)e^{-\rho})}{l^2 e^{-2\rho}} = \frac{w(e^{\rho} - (1+\rho))}{l^2}. \tag{27}$$

In addition, since $\frac{d\delta_0}{d\rho} = \frac{w}{l^2}(e^{\rho} - 1) > 0$, we know that δ_0 is monotonic increasing with ρ. Since $0 \le m \le n$, we have $\frac{n(2k-l)}{w} \le \rho \le \frac{2nk}{w}$. Therefore when $\rho = \frac{2nk}{w}$, we have

$$\delta_{0\max} = \frac{w(e^{\frac{2nk}{w}} - (1 + \frac{2nk}{w}))}{l^2} . \tag{28}$$

We thus have $\hat{m} \sim \mathcal{N}(m, \delta_0)$. According to the Chebyshev's inequality, we have

$$Pr(|\hat{m} - m| \le \epsilon m) \ge 1 - \frac{\delta_0}{\epsilon^2 m^2} \ge 1 - \delta . \tag{29}$$

Therefore, $\hat{m}$ is an (ϵ, δ) estimation of m if

$$\begin{aligned} \delta &\ge \frac{\delta_{0\max}}{\epsilon^2 m^2} \\ &= \frac{w(e^{\frac{2nk}{w}} - (1 + \frac{2nk}{w}))}{l^2 \epsilon^2 m^2}. \end{aligned} \tag{30}$$

■

APPENDIX B
PROOF OF THEOREM 2

Proof: In the passive approach, since Alice and Bob's spatiotemporal profiles have the same size, we have $\Psi'_A = \Psi_A$ and $\Psi'_B = \Psi_B$. Bob's privacy disclosure is caused by transmitting BF_B and the indexes $\mathcal{H}_B$ of k hash functions to Alice. In particular, Alice can exploit BF_B and the knowledge that Bob inserts every element in $\overline{\mathcal{P}}_{B,\alpha\to\beta}$ using l random hash functions from $\mathcal{H}_B$ and $k-l$ unknown hash functions to deduce some information about $\overline{\mathcal{P}}_{B,\alpha\to\beta}$. Consider an arbitrary element $\langle i, j, \bar{p}_{B,i}[j]\rangle$ as an example. For each of the N possible cell indexes, say cID, Alice can test whether it is a viable candidate for the unknown $\bar{p}_{B,i}[j]$ by using all the k hash functions in $\mathcal{H}_B$ to compute the k corresponding positions for the resulting element $\langle i, j, cID\rangle$. If there are at least l out of k corresponding positions set to bit-1 in BF_B, we have $cID = \bar{p}_{B,i}[j]$ with probability P; otherwise, we must have $cID \ne \bar{p}_{B,i}[j]$.

We now estimate P. After inserting all the $\lambda(\alpha+\beta-1)$ elements in $\overline{\mathcal{P}}_{B,\alpha\to\beta}$ into BF_B, the expected number of bit-1 positions is $w(1-(1-\frac{1}{w})^{\lambda(\alpha+\beta-1)k})$. For a random hash function applied to cID, the probability of the corresponding bit position having been set to bit-1 is

$$p = 1 - (1 - \frac{1}{w})^{\lambda(\alpha+\beta-1)k} \approx 1 - e^{-\frac{\lambda(\alpha+\beta-1)k}{w}} . \tag{31}$$

The probability that at least l corresponding bit positions corresponding to cID have been set to bit-1 is then given

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2549046, IEEE Internet of Things Journal

by

$$P = \sum_{i=l}^{k} \binom{k}{i} p^i (1-p)^{k-i} \ . \tag{32}$$

Let $X_{i,j}$ denote the number of valid candidate cell indexes for $\bar{p}_{B,i}[j]$. The remaining entropy for interval i in epoch j is then $\log_2 X_{i,j}$. Since $X_{i,j}$ is randomly distributed in $[1, N]$ ($N = 4^{\xi-1}$), we have the mean remaining entropy for interval i in epoch j as

$$\begin{aligned} \mathbf{E}[i,j] &= \sum_{x=1}^{N} Pr(X_{i,j} = x) \log_2 x \\ &= \sum_{x=1}^{N} \binom{N}{x} P^x (1-P)^{N-x} \log_2 x \ . \end{aligned} \tag{33}$$

Assuming that the $\lambda(\beta-\alpha+1)$ intervals are independent from each other, the total remaining entropy is given by

$$\mathbf{E} = \sum_{i=\alpha}^{\beta} \sum_{j=1}^{\lambda} \mathbf{E}[i,j] = \lambda(\alpha+\beta-1)\mathbf{E}[i,j] \ . \tag{34}$$

■

APPENDIX C
PROOF OF THEOREM 3

Proof: Assume that Alice and Bob conduct spatiotemporal profile matching with profiles $\Psi_A = \mathcal{I}_{A,\alpha\to\beta}$ and $\Psi_B = \mathcal{R}_{A,\alpha\to\beta}$, respectively. For every element in Alice's spatiotemporal profile, Alice can test whether it is a viable candidate in Bob's spatiotemporal profile by using all the k hash functions in $\mathcal{H}_B$ to compute the k corresponding positions for the resulting element. If there are at least l out of k corresponding positions set to bit-1 in BF_B, we have the conclusion that Bob and Alice were at the the same location at the same time with probability P.

Let $n = \max(n_A, n_B)$, where $n_A = |\Psi_A|$ and $n_B = |\Psi_B|$. Similar to Theorem 2, the probability that at least l corresponding bit positions have been set to bit-1 is then given by

$$P = \sum_{i=l}^{k} \binom{k}{i} p^i (1-p)^{k-i} \ , \tag{35}$$

where

$$p = 1 - (1 - \frac{1}{w})^{nk} \approx 1 - e^{-\frac{nk}{w}} \ . \tag{36}$$

Let X denote the number of tokens which might be in Bob's spatiotemporal profile. The remaining entropy for Bob's spatiotemporal profile is given by

$$\begin{aligned} \mathbf{E} &= \sum_{x=0}^{n_A} Pr(X = x) \log_2 N^{n_B - x} \\ &= \sum_{x=0}^{n_A} \binom{n_A}{x} P^x (1-P)^{n_A - x} \log_2 N^{n_B - x} \ . \end{aligned} \tag{37}$$

■

APPENDIX D
PROOF OF THEOREM 5

Proof: Recall that Bob converts each of the elements in his profile to w new elements. For each of the N possible cell indexes, say cID, Alice wants to test whether its converted w elements $cID||1, cID||2, \ldots, cID||\mathsf{w}$ are in Bob's new profile Ψ''_B. Let $n = \max(w_A, w_B)$. For each element $cID||i, 1 \leq i \leq \mathsf{w}$, if there are at least l out of k corresponding positions set to bit-1 in BF_B, $cID||i$ is considered in Bob's new profile Ψ''_B with probability P, where

$$P = \sum_{i=l}^{k} \binom{k}{i} p^i (1-p)^{k-i} \ , \tag{38}$$

$$p = 1 - (1 - \frac{1}{w})^{nk} \approx 1 - e^{-\frac{nk}{w}} \ . \tag{39}$$

For any cID, it is considered in Bob's unconverted profile Ψ_B only if each of the w elements has at least l corresponding bit-1 positions, and the probability is P^{w}.

Let $X_{i,j}$ denote the number of candidate cell indexes. The remaining entropy for interval i in epoch j is then $\log_2 X_{i,j}$. Since $X_{i,j}$ is randomly distributed in $[1, N]$ ($N = 4^{\xi-1}$), we have the mean remaining entropy for interval i in epoch j as

$$\begin{aligned} \mathbf{E}[i,j] &= \sum_{x=1}^{N} Pr(X_{i,j} = x) \log_2 x \\ &= \sum_{x=1}^{N} \binom{N}{x} P^{x\mathsf{w}} (1-P^{\mathsf{w}})^{N-x} \log_2 x \ . \end{aligned} \tag{40}$$

Assuming that the $\lambda(\beta-\alpha+1)$ intervals are independent from each other, the total remaining entropy is given by

$$\mathbf{E} = \sum_{i=\alpha}^{\beta} \sum_{j=1}^{\lambda} \mathbf{E}[i,j] = \lambda(\alpha+\beta-1)\mathbf{E}[i,j] \ . \tag{41}$$

■

APPENDIX E
PROOF OF THEOREM 6

Proof: Consider an arbitrary element in Alice's profile Ψ_A as an example. Alice can convert it to w elements as what Bob does. For each of the elements in Alice's profile Ψ_A, Alice wants to know whether it is a viable candidate in Bob's profile Ψ_B by testing whether each of its w converted elements results in at least l corresponding bit-1 positions. Let $n = \max(w_A, w_B)$. Similar to Theorem 5, the probability that each of the w elements has at least l corresponding bit-1 positions is P^{w}, where P is the probability that at least l corresponding bit positions have been set to bit-1 and is then given by

$$P = \sum_{i=l}^{k} \binom{k}{i} p^i (1-p)^{k-i} \ , \tag{42}$$

$$p = 1 - (1 - \frac{1}{w})^{nk} \approx 1 - e^{-\frac{nk}{w}} \ . \tag{43}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2549046, IEEE Internet of Things Journal

Let X denote the number of candidate elements in Bob's profile Ψ_B. The mean remaining entropy of Bob's profile is

$$\begin{aligned}\mathbf{E} &= \sum_{x=0}^{n_A} Pr(X=x)\log_2 N^{n_B-x} \\ &= \sum_{x=0}^{n_A}\binom{n_A}{x}P^{x\mathsf{w}}(1-P^{\mathsf{w}})^{n_A-x}\log_2 N^{n_B-x}\ .\end{aligned} \tag{44}$$

■

REFERENCES

[1] J. Sun, R. Zhang, and Y. Zhang, "Privacy-preserving spatiotemporal matching," in *IEEE INFOCOM'13*, Turin, Italy, Apr. 2013.

[2] H. Nishiyama, M. Ito, and N. Kato, "Relay-by-smartphone: realizing multihop device-to-device communications," *IEEE Commun. Mag.*, vol. 52, no. 4, pp. 56–65, Apr. 2014.

[3] J. Liu, S. Zhang, N. Kato, H. Ujikawa, and K. Suzuki, "Device-to-device communications for enhancing quality of experience in software defined multi-tier lte-a networks," *IEEE Netw.*, vol. 29, no. 4, pp. 46–52, Jul. 2015.

[4] B. Bloom, "Space/time trade-offs in hash coding with allowable errors," *Comm. ACM*, vol. 13, no. 7, pp. 422–426, July 1970.

[5] R. Zhang, J. Sun, Y. Zhang, and C. Zhang, "Secure spatial top-k query processing via untrusted location-based service providers," *Dependable and Secure Computing, IEEE Transactions on*, vol. 12, no. 1, pp. 111–124, Jan. 2015.

[6] M. Arb, M. Bader, M. Kuhn, and R. Wattenhofer, "VENETA: Serverless friend-of-friend detection in mobile social networking," in *WIMOB'08*, Avignon, France, Oct. 2008, pp. 184–189.

[7] M. Li, N. Cao, S. Yu, and W. Lou, "FindU: Privacy-preserving personal profile matching in mobile social networks," in *INFOCOM'11*, Shanghai, China, Apr. 2011.

[8] R. Zhang, Y. Zhang, J. Sun, and G. Yan, "Fine-grained private matching for proximity-based mobile social networking," in *IEEE INFOCOM'12*, Orlando, FL, Mar. 2012.

[9] A. Narayanan, N. Thiagarajan, M. Lakhani, M. Hamburg, and D. Boneh, "Location privacy via private proximity testing," in *NDSS'11*, San Diego, CA, Feb. 2011.

[10] Z. Lin, D. Kune, and N. Hopper, "Efficient private proximity testing with GSM location sketches," in *FC'12*, Bonaire, Feb. 2012.

[11] J. Sun, R. Zhang, X. Jin, and Y. Zhang, "Securefind: Secure and privacy-preserving object finding via mobile crowdsourcing," *IEEE Trans. Wireless Commun.*, vol. PP, no. 99, pp. 1–1, 2015.

[12] E. Cristofaro and G. Tsudik, "Practical private set intersection protocols with linear complexity," in *FC'10*, vol. 6052, Tenerife, Canary Islands, Spain, Jan. 2010, pp. 143–159.

[13] R. Zhang, J. Sun, Y. Zhang, and X. Huang, "Jamming-resilient secure neighbor discovery in mobile ad hoc networks," *IEEE Trans. Wireless Commun.*, vol. PP, no. 99, pp. 1–1, Jun. 2015.

[14] Z. Yang, B. Zhang, J. Dai, A. Champion, D. Xuan, and D. Li, "E-SmallTalker: A distributed mobile system for social networking in physical proximity," in *ICDCS'10*, Genoa, Italy, June 2010, pp. 468–477.

[15] J. Sun, X. Chen, J. Zhang, Y. Zhang, and J. Zhang, "SYNERGY: A game-theoretical approach for cooperative key generation in wireless networks," in *IEEE INFOCOM'14*, Toronto, Canada, Apr. 2014.

[16] A. Broder and M. Mitzenmacher, "Network applications of bloom filters: A survey," pp. 636–646, 2002.

[17] Y. Zhao and J. Wu, "B-SUB: A practical bloom-filter-based publish-subscribe system for human networks," in *ICDCS '10*, 2010.

[18] A. Kirsch and M. Mitzenmacher, "Less hashing, same performance: Building a better Bloom filter," in *ESA'06*, Zurich, Switzerland, Sept. 2006.

[19] P. Dillinger and P. Manolios, "Bloom filters in probabilistic verification," in *FMCAD'04*, Austin, TX, USA, Nov. 2004.

[20] M. Freedman, K. Nissim, and B. Pinkas, "Efficient private matching and set intersection," in *EUROCRYPT'04*, Interlaken, Switzerland, May 2004, pp. 1–19.

[21] K. Christensen, A. Roginsky, and M. Jimeno, "A new analysis of the false positive rate of a bloom filter," *Inf. Process. Lett.*, 2010.

[22] P. Bose, H. Guo, E. Kranakis, A. Maheshwari, P. Morin, J. Morrison, M. Smid, and Y. Tang, "On the false-positive rate of bloom filters," *Inf. Process. Lett.*, 2008.

[23] J. Manweiler, R. Scudellari, Z. Cancio, and L. Cox, "We saw each other on the subway: secure, anonymous proximity-based missed connections," in *HotMobile'09*, Santa Cruz, California, Feb. 2009.

[24] J. Manweiler, R. Scudellari, and L. Cox, "SMILE: encounter-based trust for mobile social services," in *ACM CCS'09*, Chicago, Illinois, USA, Nov. 2009, pp. 246–255.

[25] R. Lu, X. Lin, X. Liang, and X. Shen, "A secure handshake scheme with symptoms-matching for mhealthcare social network," *Mobile Networks and Applications*, pp. 1–12, 2010.

[26] L. Zhang, X. Li, and Y. Liu, "Message in a sealed bottle: Privacy preserving friending in social networks," in *IEEE ICDCS'13*, Jul. 2013.

[27] B. Niu, X. Zhu, T. Zhang, H. Chi, and H. Li, "P-Match: Priority-aware friend discovery for proximity-based mobile social networks," in *IEEE MASS'13*, Oct 2013, pp. 351–355.

[28] M. Nagy, E. D. Cristofaro, A. Dmitrienko, N. Asokan, and A.-R. Sadeghi, "Do i know you?: Efficient and privacy-preserving common friend-finder protocols and applications," in *ACSAC'13*, Dec. 2013, pp. 159–168.

[29] A. Thapa, M. Li, S. Salinas, and P. Li, "Asymmetric social proximity based private matching protocols for online social networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 6, pp. 1547–1559, June 2015.

[30] L. Kissner and D. Song, "Privacy-preserving set operations," in *CRYPTO'05*, Santa Barbara, CA, Aug. 2005, pp. 241–257.

[31] Q. Ye, H. Wang, and J. Pieprzyk, "Distributed private matching and set operations," in *ISPEC'08*, vol. 4991, Sydney, Australia, Apr. 2008, pp. 347–360.

[32] W. Dong, V. Dave, L. Qiu, and Y. Zhang, "Secure friend discovery in mobile social networks," in *INFOCOM'11*, Shanghai, China, Apr. 2011.

[33] R. Zhang, J. Zhang, Y. Zhang, J. Sun, and G. Yan, "Privacy-preserving profile matching for proximity-based mobile social networking," *IEEE J. Sel. Areas Commun.*, 2012.

[34] X. Liao, S. Uluagac, and R. Beyah, "S-match: Verifiable privacy-preserving profile matching for mobile social services," in *DSN'14*, June 2014, pp. 287–298.

[35] X. Liang, X. Li, K. Zhang, R. Lu, X. Lin, and X. Shen, "Fully anonymous profile matching in mobile social networks," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 9, pp. 641–655, September 2013.

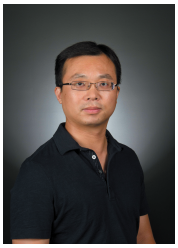
[36] H. Zhu, S. Du, M. Li, and Z. Gao, "Fairness-aware and privacy-preserving friend matching protocol in mobile social networks," *IEEE Trans. Emerg. Topics Comput*, vol. 1, no. 1, pp. 192–200, June 2013.

[37] M. Kodialam and T. Nandagopal, "Fast and reliable estimation schemes in RFID systems," in *ACM MOBICOM'06*, Los Angeles, CA, Sep. 2006, pp. 322–333.

Jingchao Sun received the B.E. in Electronics and Information Engineering and the M.E. in Communication and Information System from Huazhong University of Science and Technology, China, in 2008 and 2011, respectively. He is currently a Ph.D. student in School of Electrical, Computer, and Energy Engineering at Arizona State University. His primary research interests are network and distributed system security and privacy, wireless networking, and mobile computing.

Rui Zhang received the B.E. in Communication Engineering and the M.E. in Communication and Information System from Huazhong University of Science and Technology, China, in 2001 and 2005, respectively, and the PhD degree in Electrical Engineering from the Arizona State University, in 2013. He was a software engineer in UTStarcom Shenzhen R&D center from 2005 to 2007. He has been an assistant professor in the Department of Electrical Engineering at the University of Hawaii since July 2013. His research interests are the security and privacy issues in wireless networks, mobile crowdsourcing, mobile systems for disabled people, cloud computing, and social networks. He is an Associate Editor of IEEE Internet of Things Journal and a member of IEEE.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2016.2549046, IEEE Internet of Things Journal



Yanchao Zhang received the B.E. in Computer Science and Technology from Nanjing University of Posts and Telecommunications in 1999, the M.E. in Computer Science and Technology from Beijing University of Posts and Telecommunications in 2002, and the Ph.D. in Electrical and Computer Engineering from the University of Florida in 2006. He is an Associate Professor in School of Electrical, Computer and Energy Engineering at Arizona State University. His primary research interests are network and distributed system security, wireless networking, and mobile computing. He is an Editor of IEEE Transactions on Mobile Computing, IEEE Transactions on Vehicular Technology, and IEEE Wireless Communications. He was also a TPC Co-Chair of Communication and Information System Security Symposium, IEEE GLOBECOM 2010. He received the NSF CAREER Award in 2009 and is a senior member of IEEE.