A Distortion-Based Metric for Location Privacy

Reza Shokri, Julien Freudiger, Murtuza Jadliwala, and Jean-Pierre Hubaux
Laboratory for Computer Communications and Applications, EPFL, Switzerland
firstname.lastname@epfl.ch

ABSTRACT

We propose a novel framework for measuring and evaluating location privacy preserving mechanisms in mobile wireless networks. Within this framework, we first present a formal model of the system, which provides an efficient representation of the network users, the adversaries, the location privacy preserving mechanisms and the resulting location privacy of the users. This model is general enough to accurately express and analyze a variety of location privacy metrics that were proposed earlier. By using the proposed model, we provide formal representations of four metrics among the most relevant categories of location privacy metrics. We also present a detailed comparative analysis of these metrics based on a set of criteria for location privacy measurement. Finally, we propose a novel and effective metric for measuring location privacy, called the *distortion-based* metric, which satisfies these criteria for privacy measurement and is capable of capturing the mobile users' location privacy more precisely than the existing metrics. Our metric estimates location privacy as the expected distortion in the reconstructed users' trajectories by an adversary.

Categories and Subject Descriptors

C.2.0 [**Computer-Communication Networks**]: General—*Security and protection*; K.4.1 [**Computers and Society**]: Public Policy Issues—*Privacy*

General Terms

Security, Measurement

1. INTRODUCTION

The recent popularity of location-based data sharing applications, together with the rapid proliferation of wireless mobile devices equipped with advanced communication capabilities, has fueled the emergence of context-aware, location-based applications and services on these devices. Applications that take advantage of the device-to-device communication capabilities of these devices in order to share location-based data among each other, e.g., [1, 2, 3], or the device-to-infrastructure capabilities to request/report location-based data from/to the service providers, e.g., [25, 30], are now available and widely used. Despite their popularity, privacy issues such as leakage of one's location information to service providers, without the user's consent, or to external eavesdroppers [9, 10] is a major concern in these applications.

Privacy concerning the users' locations, also termed *location privacy*, has been formally defined in the literature in the context of such pervasive location-based applications and services. Beresford and Stajano [7] define location privacy as *the ability to prevent other parties from learning one's current or past location.* In order to protect location privacy, several mechanisms have been proposed in the literature. These privacy preserving mechanisms provide various methods (e.g., anonymization [10], pseudonym change [6], path perturbation [20]) that could be implemented either centrally on the trusted third-party servers or on the individual mobile devices themselves to prevent an adversary from trivially learning the users' past or current locations.

In order to evaluate the effectiveness of these location privacy preserving mechanisms, in terms of correctness in measuring or quantifying the notion of "location privacy", several metrics [12, 15, 19, 20, 21, 22, 32] are proposed in the literature. Ideally, a location privacy metric should capture the amount of information an adversary has about users' actual trajectories (or positions). In other words, the location privacy metric should be able to measure the inability of a given adversary in accurately tracking the mobile users over space and time. However, existing location privacy metrics in the literature do not completely capture this notion of location privacy and are too specific to particular privacy preserving mechanisms. Moreover, some of these metrics for location privacy preserving mechanisms inspired from other domains such as *anonymous communication* or *database security* are not able to correctly capture the notion of location privacy because privacy in these domains is defined in a slightly different way than in the context of mobile networks.

The intuition behind location privacy can easily be conveyed by drawing an analogy to a general communication system in which some information sent by a transmitter is first encoded before being sent over a communication channel to the receiver who tries to decode it in order to reconstruct the encoded information. The quality of this communication depends upon how accurately the receiver is able to decode the information (even in the presence of errors) from the transmitted data. The analogy here is that a lo-

WPES'09, November 9, 2009, Chicago, Illinois, USA.

cation privacy preserving mechanism can be viewed as the communication channel encoder that transforms the passing location data. The transmitter is the mobile network of users who transmit their location information, whereas the adversary is analogous to the receiver who wants to reconstruct the users' trajectories or locations after their transformation through the privacy preserving mechanism. Hence, the quality of the location privacy provided by a location privacy preserving mechanism depends on how unsuccessful the adversary is in reconstructing the users' locations over time. We argue that the higher the *distortion* between the expected reconstructed trajectory and the actual trajectory of the users is, the better their location privacy will be. The location privacy metric that we propose in this paper, unlike earlier metrics, considers this distortion in order to measure the location privacy of mobile users.

In this paper, we have two main contributions. First, we propose an analytical framework for formally representing our location privacy metric, which includes representation of system elements such as the mobile device network, the privacy preserving mechanism and the adversary. The proposed framework is also general enough to formally express other location privacy metrics in the literature. We then evaluate the effectiveness of these metrics in correctly quantifying location privacy based on a set of criteria that we derive from the definition of location privacy in mobile networks. Based on this evaluation, we argue that existing location privacy metrics do not effectively capture the notion of location privacy, especially from the point of view of the accuracy of an adversary in tracking users. Our second contribution is a novel location privacy metric, called *the distortion-based metric*, that measures a user's location privacy at any time by considering the level of distortion in each part of the user's reconstructed trajectory by the adversary. We believe that such a distortion-based metric captures the notion of location privacy much more accurately and satisfies most of the criteria for accurate representation of location privacy in mobile networks.

2. SYSTEM MODEL

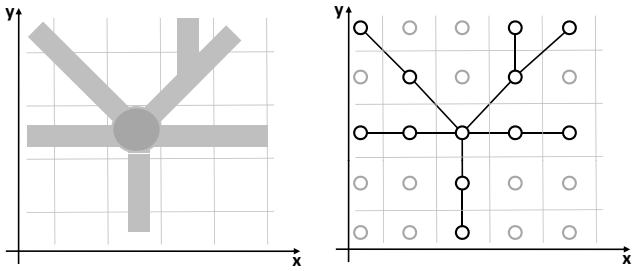
In this section, we present a formal framework for modeling mobile users, location privacy preserving mechanisms and the adversary. In the following section, we will use this model to express and analyze the location privacy metrics.

2.1 Mobile Networks

A mobile wireless network consists of a set of *mobile users* equipped with wireless devices. The set of users in the system is denoted by $\mathcal{U}$, where $n = |\mathcal{U}|$ is the number of users. All the users are associated with two types of *identities*: *actual identities* and *pseudonyms*.

A user's actual identity is any subset of his attributes (e.g., his name, his national identity number, his user name or his private key) that uniquely identifies the user within any set of users in the system [29]. We assume that each user has one, unique, actual identity. The set of all the users' actual identities is denoted by $\mathcal{I}$. We define a bijective function $\mathbf{name} : \mathcal{U} \rightarrow \mathcal{I}$ that maps each user with his actual identity.

Each user is also associated with a set of *pseudonyms*. A user's pseudonyms are different from his actual identity. The user to whom a particular pseudonym refers is called the *holder* of the pseudonym [29]. A pseudonym is called a *group pseudonym* if it refers to a set of holders [11]. Therefore, the set of pseudonyms held by a user is not necessarily disjoint from that of other users. The set of all pseudonyms used by all the users is denoted by $\widehat{\mathcal{I}}$, where $\widehat{\mathcal{I}} \cap \mathcal{I} = \emptyset$. We define the function $\mathbf{nyms} : \mathcal{U} \rightarrow \mathcal{P}(\widehat{\mathcal{I}})$ to give the set of pseudonyms associated with each user, where $\mathcal{P}(\widehat{\mathcal{I}})$ is the power set of $\widehat{\mathcal{I}}$. The *null pseudonym*, denoted by $\ominus$, is a pseudonym in $\widehat{\mathcal{I}}$ that represents the status of being identity-less (when a user's identity is removed from his communicating messages without being replaced by one of his pseudonyms, for example in full anonymization by a proxy). We will show its use later when we explain the privacy preserving mechanisms.



(a) Geographic space (b) Graph G^s

Figure 1: Example of geographic space. The space is divided into grids and contains the road network which is modeled in a discrete way by graph G^s.

The notion of time that we employ in this model is discrete, using a global clock, where each unit of time is called a *time instance*. The set of considered time instances is denoted by $\mathcal{T} = \{1, 2, \cdots, T\}$, where T represents the last considered time instance in the system.

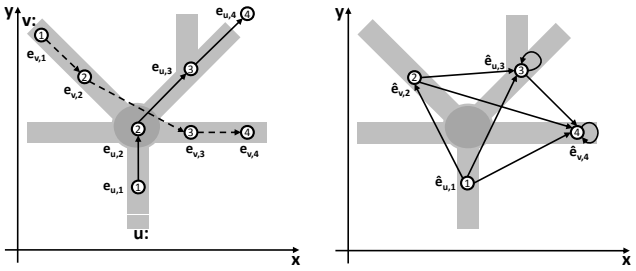
We model the geographic *space* in which users move, in a discrete way, as a graph $G^s = (V^s, E^s)$, where V^s is the set of vertices and each vertex denotes a location (region) in the users geographic space. The set E^s is the set of edges of the graph and is defined as follows. There is an edge between two vertices if the location pairs represented by those vertices are directly connected to each other, for example, by means of direct roads or direct transport links. In this model, there is a physical *route* between two locations if there is a path in the graph that connects their associated vertices to each other. Figure 1 illustrates a geographic space and its corresponding graph that is used as an example throughout the paper. A set of locations can be combined to a *location area*. Thus, a location area is visualized as a subset of V^s.

Note that the granularity of time depends on the level of accuracy we expect from the model. The same holds for the considered locations in the modeled geographic space. For example, time units can be hours, or days, and the space can also be divided into arbitrarily sized regions in the map.

As users are mobile, their location is a time-dependent value. Let $\mathbf{whereis} : \mathcal{U} \times \mathcal{T} \rightarrow V^s$ be a function that gives the *actual location* of a user at any time instance. Note that this function reflects the very exact location of each user, regardless of the knowledge of any entity about it.

2.2 Events and Traces

In this section, we present part of our formal framework that models the users' spatiotemporal status. It can represent both the users' actual status (defined in this section)



(a) (b)

Figure 2: Actual and observed events for two users u and v crossing at an intersection. The circles show the events and the associated number indicates their time-stamp. (a) Actual events. The arrows indicate the users' movement direction. (b) Observed events and the linkability graph G^l. The arrows indicate the edges in graph G^l.

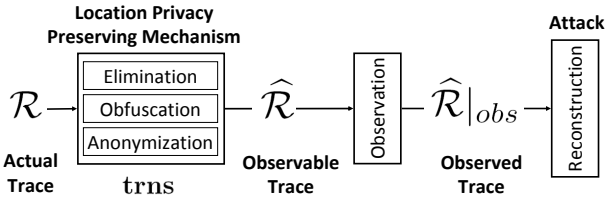


Figure 3: Actual trace $\mathcal{R}$ is transformed to $\widehat{\mathcal{R}}$ by a location privacy preserving mechanism. After the observation, a specific adversary obtains a subset of $\widehat{\mathcal{R}}$, denoted by $\widehat{\mathcal{R}}|_{obs}$. A location privacy metric must reflect the degree of the adversary's success in reconstructing $\mathcal{R}$ from $\widehat{\mathcal{R}}|_{obs}$.

and also the status of the users from an observer's perspective (defined in Sections 2.3 and 2.4).

An *event* is defined as a 3-tuple (i, t, l), where $i \in \mathcal{I} \cup \widehat{\mathcal{I}}$ is a user's identity (either actual or pseudonym), t represents the time instance at which the event occurred (referred to as the *time-stamp* of the event), and $l \in \mathcal{P}(V^s)$ is the location area associated with the event (referred to as the *location-stamp* of the event), where $\mathcal{P}(V^s)$ is the power set of V^s. Let $\mathcal{E}$ denote the set of all possible events. Let $\mathbf{id} : \mathcal{E} \to \mathcal{I} \cup \widehat{\mathcal{I}}$, $\mathbf{tm} : \mathcal{E} \to \mathcal{T}$, and $\mathbf{loc} : \mathcal{E} \to \mathcal{P}(V^s)$ be the functions that give the identity, time-stamp, and location-stamp of an event, i.e., for any event $e \equiv (i, t, l)$ we have $\mathbf{id}(e) = i$, $\mathbf{tm}(e) = t$, and $\mathbf{loc}(e) = l$. The relation *happens before*, denoted by $e_i \prec e_j$, exists between events e_i and e_j only if $\mathbf{tm}(e_i) < \mathbf{tm}(e_j)$.

A *trace* is defined as a non-empty set of events. Consider a trace $\Upsilon \in \mathcal{P}(\mathcal{E}) \setminus \{\emptyset\}$. For any two distinct events $e_i, e_j \in \Upsilon$, if $e_i \prec e_j$ and $\nexists\, e_k \in \Upsilon$ s.t. $e_i \prec e_k \prec e_j$ then the event e_i is called an *immediate predecessor* of the event e_j, and is denoted by $e_i \preccurlyeq_\Upsilon e_j$. In this case, the event e_j is called the *immediate successor* of the event e_i. We use $e_i \preccurlyeq e_j$ instead of $e_i \preccurlyeq_\Upsilon e_j$ if the trace Υ is clear from the context.

Let $\mathbf{f}$ be any function that is defined over events (e.g., $\mathbf{tm}$, $\mathbf{id}$, $\mathbf{loc}$). Its image function $\overrightarrow{\mathbf{f}}$ for a trace Υ is defined as $\overrightarrow{\mathbf{f}}(\Upsilon) = \{\mathbf{f}(e) \,|\, e \in \Upsilon\}$. The set of events in a trace Υ with time-stamp less than or equal to t is denoted by $\Upsilon_{\leq t} = \{e \in \Upsilon \,|\, \mathbf{tm}(e) \leq t\}$. For a trace Υ, we define $\mathbf{head}(\Upsilon) = \{e \in \Upsilon \,|\, \mathbf{tm}(e) = \min(\overrightarrow{\mathbf{tm}}(\Upsilon))\}$ and similarly $\mathbf{tail}(\Upsilon) = \{e \in \Upsilon \,|\, \mathbf{tm}(e) = \max(\overrightarrow{\mathbf{tm}}(\Upsilon))\}$. In other words, the head (or tail) of the trace is the set of events with the smallest (or largest) time-stamp.

An event e is called an *actual event* associated with a user u if $\mathbf{id}(e) = \mathbf{name}(u)$ and $\mathbf{loc}(e) = \{\mathbf{whereis}(u, t)\}$, where $\mathbf{tm}(e) = t$. Note that there is no uncertainty and inaccuracy about where such events occur. The *actual trace* of a user u, denoted by $\mathcal{R}_u$, represents his entire actual trajectory. It is defined as the set of the user's actual events, i.e., $\mathcal{R}_u = \{(\mathbf{name}(u), t, \{\mathbf{whereis}(u, t)\}) \,|\, t \in \mathcal{T}\}$. Figure 2(a) represents an example for the actual events of two users u and v in the geographic space previously shown in Figure 1 during time instances $\{1, 2, 3, 4\}$.

We define relation $\sim$ between two actual events e_i and e_j, denoted by $e_i \sim e_j$, if and only if they are associated with the same user (i.e., $\exists u \in \mathcal{U}$ s.t. $e_i, e_j \in \mathcal{R}_u$).

The actual trace of a system is the union of all users' actual traces and is denoted by $\mathcal{R}$, i.e., $\mathcal{R} = \bigcup_u \mathcal{R}_u$. Notice that $\forall u, v \in \mathcal{U}$ we have $\mathcal{R}_u \cap \mathcal{R}_v = \emptyset$. Therefore, the equivalence relation $\sim$ partitions the trace $\mathcal{R}$ into subsets, each representing the actual trace of one user. We refer to the set partition associated with $\sim$ as the *actual set partition* of $\mathcal{R}$.

2.3 Location Privacy Preserving Mechanisms

In this section, we formalize the location privacy preserving mechanism in the system. We formally define *location privacy preserving mechanisms* as transformation functions that modify or hide the users' actual events before they could be observable by any observer. Such a mechanism can work in a distributed way by being implemented on individual mobile devices, in a centralized manner by using a proxy server that modifies the users' messages, or by using a hybrid technique. In our model, we abstract away the implementation details of privacy preserving mechanisms.

Let the location privacy preserving mechanisms be denoted by function $\mathbf{trns} : \mathcal{R} \to \mathcal{E} \cup \{\text{HIDDEN}\}$, where HIDDEN stands for a *hidden event* (i.e., an unobservable event). The output of the transformation function $\overrightarrow{\mathbf{trns}}(\mathcal{R})$ on the actual trace is called the system's *observable trace* and is denoted by $\widehat{\mathcal{R}}$, i.e., $\widehat{\mathcal{R}} = \overrightarrow{\mathbf{trns}}(\mathcal{R}) \setminus \{\text{HIDDEN}\}$. As it is shown in Figure 3, privacy preserving mechanisms perform the transformation function by means of three methods: *elimination*, *obfuscation*, and *anonymization*.

One method is the elimination, by which a subset of events in the actual trace $\mathcal{R}$ is selected to be eliminated and thus to become hidden (i.e., **trns** replaces the eliminated events by HIDDEN element in the actual trace $\mathcal{R}$).

Another method is the obfuscation, by which the location-stamp or time-stamp of a subset of the events in the actual trace $\mathcal{R}$ is altered. Using the obfuscation methods results in the *inaccuracy* or *imprecision* of the location- or time-stamp of the events [13]. The location/time information obfuscation can be achieved mostly through methods such as *perturbation* [19, 22] or *generalization* (e.g., [4] or k-anonymity [16, 19]). In this paper, we only focus on location obfuscation, leaving time obfuscation as future work.

Lastly, using the anonymization method, the identity of a subset of events in the actual trace $\mathcal{R}$ is altered. In this phase, the actual identity of a user on an event is replaced by

one of his pseudonyms and can change over time. We capture the state of being identity-less by the *null pseudonym* $\ominus$. In other words, for an identity-less event e we have $\mathbf{id}(e) = \ominus$. Making the events identity-less is implemented in proxy-based mechanisms where the users' identities are removed before the events become observable to any distrusted entity.

We denote the observable trace of a user $u \in \mathcal{U}$ by $\widehat{\mathcal{R}}_u = \overrightarrow{\mathbf{trns}}(\mathcal{R}_u)$. Thus, $\widehat{\mathcal{R}} = \bigcup_u \widehat{\mathcal{R}}_u$, where $\widehat{\mathcal{R}}$ is the outcome of the transformation function (i.e., the observable trace).

If two actual events e_i and e_j are associated with the same user (i.e., $e_i \sim e_j$) and $\hat{e}_i = \mathbf{trns}(e_i)$ and $\hat{e}_j = \mathbf{trns}(e_j)$, then we define a relation $\sim_o$ between $\hat{e}_i$ and $\hat{e}_j$. In other words, we define $\hat{e}_i \sim_o \hat{e}_j$ if $\hat{e}_i, \hat{e}_j \in \widehat{\mathcal{R}}_u$ for some $u \in \mathcal{U}$.

2.4 Adversary

In our framework, the adversary is any entity that observes events after the transformation process. Thus, it has access to a subset of $\widehat{\mathcal{R}}$, called the *observed trace* and denoted by $\widehat{\mathcal{R}}|_{obs}$ (i.e., $\widehat{\mathcal{R}}|_{obs} \subseteq \widehat{\mathcal{R}}$). We denote by $\widehat{\mathcal{R}}_u|_{obs}$ the observed trace associated with user $u \in \mathcal{U}$, i.e., the transformed version of his actual trace. The set of all time instances in which the adversary observes any event in the system is denoted by $\mathcal{T}|_{obs} = \overrightarrow{\mathbf{tm}}(\widehat{\mathcal{R}}|_{obs})$. Accordingly, we define the time instances for $\widehat{\mathcal{R}}_u|_{obs}$ as $\mathcal{T}_u|_{obs} = \overrightarrow{\mathbf{tm}}(\widehat{\mathcal{R}}_u|_{obs})$.[1]

The extent to which an adversary can observe the observable events depends on its capabilities. For the most powerful adversary (known as the *global* adversary) the observed trace is equal to the observable trace, i.e., $\widehat{\mathcal{R}}|_{obs} = \widehat{\mathcal{R}}$.

With access to the observed trace $\widehat{\mathcal{R}}|_{obs}$, the adversary's objective is to reconstruct the actual trace $\mathcal{R}$ as accurately as possible and to eventually identify the users to whom each trace belong. For any given user u, the more accurately the adversary can reconstruct his actual trace $\mathcal{R}_u$, especially in location/time pairs containing more information about the user (e.g., the user's home place in the evening, or his workplace in the morning), the higher the probability of disclosing his actual identity $\mathbf{name}(u)$ is. [18, 27]

The adversary is assumed to have prior knowledge about the events and traces that may be possible in the system. More precisely, the adversary knows the mobility model of the network. It knows the geographic space in which the mobile users move and also the possible routes that connect the different locations together (i.e., graph G^x). The adversary can model the users' movement in the considered space and subsequently assign probabilities to the following events: (i) some user or a specific user is in a region at a specific time instance, and (ii) a user moves from one location to another location at specific time instances. Moreover, the adversary knows what kind of privacy preserving mechanism is used in the system. We do not assume anything about how the adversary obtains this knowledge and how accurate it is.

Given the capabilities of the adversary and the observed trace $\widehat{\mathcal{R}}|_{obs}$, the adversary's goal is to carry out the reconstruction attack (i.e., to reconstruct the actual trace), which is probabilistic in nature. The adversary assigns probabilities to possible related events in order to reconstruct the users' trajectories. Throughout the paper, we denote the probability with which the adversary considers a statement s to be true by $\Pr_{\mathrm{A}}(s)$. For example, $\Pr_{\mathrm{A}}(\hat{e}_i \sim_o \hat{e}_j)$ gives the probability that the adversary believes that two observed events $\hat{e}_i$ and $\hat{e}_j$ are associated with the same user (i.e., $\hat{e}_i, \hat{e}_j \in \widehat{\mathcal{R}}_u|_{obs}$ for some user u). In fact, we represent the adversary's knowledge by $\Pr_{\mathrm{A}}(.)$. However, the construction of an adversary's knowledge (i.e., computing the values of $\Pr_{\mathrm{A}}(.)$) is out of the scope of this paper.

[1]Note that subscript u is used in our model to refer to the events and traces associated with user u and it does not mean that the adversary is able to make that association.

2.5 Location Privacy Measurement

In order to represent various location privacy metrics in our model, we need a new data structure. To this end, we define a probabilistic graph called the linkability graph that represents the linkability of observed events based on the adversary's knowledge .

The *linkability graph* is a directed graph $G^l = (V^l, E^l, \pi^l)$, where V^l and E^l are the set of vertices and edges, respectively, and $\pi^l : V^l \times V^l \rightarrow [0, 1]$ is a weight function that defines the edges of the graph. The set of vertices in the graph is equal to the set of observed events by the adversary, i.e., $V^l = \widehat{\mathcal{R}}|_{obs}$. In other words, there is a vertex in the graph G^l corresponding to each observed event in $\widehat{\mathcal{R}}|_{obs}$. The edges of the graph are defined based on the weight function π^l as follows: there is an edge between two vertices $\hat{e}_i$ and $\hat{e}_j$ if and only if $\pi^l(\hat{e}_i, \hat{e}_j) > 0$. For distinct observed events $\hat{e}_i$ and $\hat{e}_j$, weight function $\pi^l(\hat{e}_i, \hat{e}_j)$ represents the probability with which the adversary believes that both events are associated with the same user (i.e., $\hat{e}_i \sim_o \hat{e}_j$) and $\hat{e}_i$ is an immediate predecessor of $\hat{e}_j$ in some user's observed trace (i.e., $\hat{e}_i \preccurlyeq \hat{e}_j$). Hence, $\pi^l(\hat{e}_i, \hat{e}_j) = \Pr_{\mathrm{A}}((\hat{e}_i \sim_o \hat{e}_j) \wedge (\hat{e}_i \preccurlyeq \hat{e}_j))$. Moreover, $\pi^l(\hat{e}_i, \hat{e}_i)$ gives the probability that $\hat{e}_i$ is the last event observed from a user. The sum of the linkability probabilities assigned to the outgoing edges from each vertex is equal to 1, i.e., $\forall \hat{e}_i \in V^l$ we have $\sum_j \pi^l(\hat{e}_i, \hat{e}_j) = 1$. Figure 2(b) represents an example of the observed events from u and v (shown in Figure 2(a)) and the associated linkability graph.

The set of vertices in any path in the linkability graph is a trace (based on the definition of traces). Let trace Υ_{ij} be a path between $\hat{e}_i$ and $\hat{e}_j$ in the linkability graph G^l. The weight (i.e., probability) assigned to Υ_{ij}, i.e., the probability that all the events in the trace are associated with the same user and the trace covers all the observed events $\hat{e}_k$ of that particular user such that $\mathbf{tm}(\hat{e}_i) < \mathbf{tm}(\hat{e}_k) < \mathbf{tm}(\hat{e}_j)$, is denoted by $\pi^{l*}(\Upsilon_{ij})$.

Finally, we give the notations that will be used to denote the location privacy measured by different metrics at different granularities. Let $\mathbf{x}$ be a metric that is used to measure location privacy. The overall system level location privacy is denoted by $\mathbf{LP^x}$. The overall location privacy of any user u is denoted by $\mathbf{LP}^{\mathbf{x}}_u$. The location privacy of any user u at time instance t is denoted by $\mathbf{LP}^{\mathbf{x}}_u(t)$.

3. EXISTING METRICS

We outline four relevant categories of location privacy metrics in this section. First, we describe these metrics by using the formalization that we establish in Section 2. Next, we study the effectiveness of these metrics in capturing the location privacy using a set of criteria.

3.1 Description of the Metrics

In this section, we describe the following classes of location privacy metrics: *uncertainty-based* metrics (u-metrics),

"clustering error"-based metrics (c-metrics), *traceability-based* metrics (t-metrics), and *k-anonymity* metrics (k-metrics).

3.1.1 Uncertainty-Based Metric

This metric was originally proposed by Diaz *et al.* [12] and Serjantov and Danezis [32] to measure privacy in anonymous communication systems. In their setting, the adversary's aim is to identify the sender and/or receiver of a transferred message. Each user is assigned a probability for being the possible sender/receiver of the message. The entropy [33] of the random variable that is associated with the users' probabilities is considered as the system's anonymity level. Thus, the metric captures the uncertainty (measured by the entropy) of the adversary in the identification process.

This metric has been widely used for measuring location privacy as well (e.g., [6, 8, 23, 24, 26, 28]). The location privacy of a given user is computed as the uncertainty of the adversary in linking the user's observed events.

Let u be a user in $\mathcal{U}$ and consider an event $\hat{e}_i \in \widehat{\mathcal{R}}_u|_{obs}$. Let a random variable X_i represent the probability that any $\hat{e}_j$, as another observed event, is the immediate successor of $\hat{e}_i$. In other words, we have $\Pr_{\mathrm{A}}(X_i = j) = \pi^l(\hat{e}_i, \hat{e}_j)$ for any j such that $(\hat{e}_i, \hat{e}_j) \in E^l$ in the linkability graph G^l. The entropy of X_i, denoted by $\mathbb{H}(X_i)$, is computed as follows.

$$\mathbb{H}(X_i) = -\sum_j \Pr_{\mathrm{A}}(X_i = j) \cdot \log_2(\Pr_{\mathrm{A}}(X_i = j))$$

In the u-metrics, as shown below, this entropy value is considered as the privacy of user u at time instance $\mathbf{tm}(\hat{e}_i)$.

$$\mathbf{LP}^{\mathbf{u}}_u(\mathbf{tm}(\hat{e}_i)) = \mathbb{H}(X_i) \tag{1}$$

3.1.2 "Clustering Error"-Based Metrics

Observing that u-metrics are not fully appropriate for measuring location privacy, Hoh and Gruteser [20] and Fischer *et al.* [15] propose two metrics, which are similar to each other, to compute system location privacy. The latter metric is proposed for measuring data privacy in general, yet focuses on location data as sensitive information, whereas the former is more specialized for location privacy.

The c-metrics measure location privacy based on the adversary's success in attacking the system in the following way. The adversary's goal here is to partition the observed events into multiple subsets, each for one user. As the adversary is not necessarily certain about the actual set partitions, it hypothesizes a number of set partitions in a probabilistic manner. The adversary's expected partitioning error represents the system level location privacy.

The adversary partitions the set of observed events $\widehat{\mathcal{R}}|_{obs}$ into k subsets, where each subset hypothesizes the observed events associated with one user. The set of all possible set partitions for $\widehat{\mathcal{R}}|_{obs}$ is denoted by Ψ. Let ψ be one such set partition in Ψ. Two observed events $\hat{e}_i$ and $\hat{e}_j$ are equivalent in ψ, i.e., $\hat{e}_i \sim_\psi \hat{e}_j$, if both belong to the same subset in ψ. Thus, the equivalence relation $\sim_\psi$ partitions $\widehat{\mathcal{R}}|_{obs}$ into k disjoint subsets denoted by $\widehat{\mathcal{R}}'_i|^{\psi}_{obs}$, where $\widehat{\mathcal{R}}'_i|^{\psi}_{obs}$ represents the set of observed events associated with a user indexed by i in the set partition ψ.

A distance function, defined separately by each of the two metrics, estimates the adversary's error by comparing each hypothesized set partition with the actual one (i.e., the one associated with relation $\sim$). In both metrics, it is assumed that there is no uncertainty about the location of the observed events (i.e., the location-stamp of an event is a single location and not an area). The two metrics mainly differ in the way the adversary's expected error is computed. Thus, in order to distinguish between them, we refer to the metric in [20] as $\dot{c}$-metric and that in [15] as $\ddot{c}$-metric.

$\dot{\mathbf{c}}$-metric.

To compute the location privacy based on $\dot{c}$-metric, Hoh and Gruteser [20] assume that the adversary knows the number of users in the system (i.e., $k = n$). It is also assumed that $\forall u \in \mathcal{U},\ \mathcal{T}_u|_{obs} = \mathcal{T}|_{obs}$, which means that the observed traces for all the users are synchronized. These assumptions also imply that, for every time instance in $\mathcal{T}|_{obs}$, the adversary observes a set of n events and each event is associated with one user. Hence, each set partition $\psi \in \Psi$ is composed of n subsets each with cardinality $|\mathcal{T}|_{obs}|$.

Each subset in a given set partition is identified by an index. The index function $\mathbf{ix}$ maps the set of users to the set of indices. Each user $u \in \mathcal{U}$ is manually assigned an index $i \in \{1...n\}$. In fact, for a given user u, $\mathbf{ix}_\psi(u)$ is the set partition ψ that includes the $\mathbf{head}(\widehat{\mathcal{R}}_u|_{obs})$ event.

In order to partition the observed trace, the adversary has to find perfect matches between each two sets of observed events belonging to two consecutive time instances in $\mathcal{T}|_{obs}$.[2] It is assumed that the users velocity at each observed event is known and that the mobile users move smoothly in the space (i.e., they do not have sharp turns). Then, inspired from a multi-target tracking (MTT) algorithm [5], a linear Kalman model is used to estimate the probability of the possible set partitions and also the linkability between observed events (modeled in our framework by $\pi^l(.)$).

Let $\hat{e}^{\psi}_{u,t}$ be the event in $\widehat{\mathcal{R}}'_i|_{obs}$ that is observed at time instance $t \in \mathcal{T}|_{obs}$ and is assigned to user u in the set partition ψ. Due to the adversary's uncertainty in partitioning the observed trace $\widehat{\mathcal{R}}|_{obs}$, the event $\hat{e}^{\psi}_{u,t}$ could be different from the observed event of user u at time t. Let $\Pr_{\mathrm{A}}(\hat{e}^{\psi}_{u,t})$ be the probability that $\hat{e}^{\psi}_{u,t}$ is the observed event associated with user u at time t. This probability is equal to $\pi^{l*}(\widehat{\mathcal{R}}'_i|^{\psi}_{obs\leq t})$, which is computed in the $\dot{\mathbf{c}}$-metric by the MTT algorithm.

The adversary's error at time instance $t \in \mathcal{T}|_{obs}$, averaged over all the users, is computed as follows.

$$\dot{D}_\psi(t) = \frac{1}{n} \cdot \sum_u ||\mathbf{loc}(\hat{e}^{\psi}_{u,t}) - \mathbf{whereis}(u,t)||$$

where $||.||$ denotes the Euclidean distance, and as mentioned before, $\mathbf{loc}(\hat{e}^{\psi}_{u,t})$ is considered to be a single location.

Subsequently, the expected error of the set partition ψ, over all the users and through all the observed time instances, is computed as follows.

$$\mathrm{E}[\dot{D}_\psi] = \frac{1}{\left|\mathcal{T}|_{obs}\right|} \cdot \sum_{t \in \mathcal{T}|_{obs}} \left(\dot{D}_\psi(t) \cdot \prod_u \Pr_{\mathrm{A}}(\hat{e}^{\psi}_{u,t})\right)$$

Finally, as shown in (2), the system level location privacy is computed as the expected error averaged over all probable set partitions.

$$\mathbf{LP}^{\dot{\mathbf{c}}} = \frac{1}{|\Psi|} \cdot \sum_{\psi \in \Psi} \mathrm{E}[\dot{D}_\psi] \tag{2}$$

[2] Similar methods can be found in the context of anonymous communication, e.g., [14, 17].

c̈-metric.

Fischer *et al.* [15] propose a slightly different approach to computing the adversary's expected error in partitioning the observed events. Instead of using the Euclidean distance to compute the error of a hypothesized set partition, a set dissimilarity measure is used. This measure is the normalized number of different features between any two set partitions. It is computed as the number of event pairs that are in the same subset in one set partition but not in the other. Hence, for a set partition ψ it is estimated as follows.

$$\ddot{D}_{\psi} = \frac{\left|\{(\hat{e}_i,\hat{e}_j)|\ (\hat{e}_i \sim_{\psi} \hat{e}_j \wedge \hat{e}_i \nsim_o \hat{e}_j) \vee (\hat{e}_i \nsim_{\psi} \hat{e}_j \wedge \hat{e}_i \sim_o \hat{e}_j)\}\right|}{\left|\{(\hat{e}_i,\hat{e}_j)|\ \hat{e}_i \nsim_o \hat{e}_j\}\right|}$$

where $\hat{e}_i, \hat{e}_j \in \widehat{\mathcal{R}}|_{obs}$.

Each set partition ψ is assigned a probability $\Pr_{\mathrm{A}}(\psi)$ that indicates how probable it is for the adversary to select this set partition. Finally, the system level location privacy is computed as the expected value of the partitioning error.

$$\mathbf{LP}^{\ddot{\mathbf{c}}} = \sum_{\psi \in \Psi} \Pr_{\mathrm{A}}(\psi) \cdot \ddot{D}_{\psi} \tag{3}$$

3.1.3 Traceability-Based Metrics

Traceability-based metrics, t-metrics, capture the extent to which the adversary can track a user with high certainty. The traceability is estimated as the length of the *time period* or the *distance in location space* in which the adversary can continuously and successfully track a user. Hoh and Gruteser propose two similar t-metrics [21, 22] that are built upon u-metrics. More precisely, the success of the adversary in tracking the users is estimated based on u-metrics. These two metrics are called *Mean time to confusion* and *Mean location to confusion*. We refer to them as $\dot{t}$-metric and $\ddot{t}$-metric, respectively.

An event in the observed trace of a user is called a *confusion point* if the adversary's uncertainty is above a given threshold, $\mathbb{H}_{cf}$, at that point. More precisely, an observed event $\hat{e} \in \widehat{\mathcal{R}}_u|_{obs}$ is called a confusion point if $\mathbf{LP}^{\mathbf{u}}_u(\mathbf{tm}(\hat{e})) > \mathbb{H}_{cf}$. Subsequently, the time to confusion is defined as the period of time before reaching a confusion point, during which the adversary's uncertainty remains below $\mathbb{H}_{cf}$. Location to confusion is defined in the same way. Then, the average value of time/location to confusion for each user represents his lack of location privacy. To outline it formally, let $\widehat{\mathcal{R}}_u|^{cf}_{obs}$ represent the set of all confusion points (events) of user u. Let the union set of the last observed event of user u and his confusion events be denoted by C_u.

$$C_u = \{\mathbf{tail}(\widehat{\mathcal{R}}_u|_{obs})\} \cup \widehat{\mathcal{R}}_u|^{cf}_{obs}$$

Let B_u be the set of events that are not confusion points but are immediate successors of each confusion point in the observed trace of user u. In addition to these events, B_u also contains the first observed event from u.

$$B_u = \{\mathbf{head}(\widehat{\mathcal{R}}_u|_{obs})\} \cup$$
$$\{\hat{e} \in \widehat{\mathcal{R}}_u|_{obs} \mid (\hat{e} \notin C_u) \ \wedge \ (\exists\, \hat{e}' \in C_u \text{ s.t. } (\hat{e}' \prec \hat{e}) \wedge (\hat{e}' \sim_o \hat{e}))\}$$

Subsequently, a traceable period can be defined as the time period between an event in B_u and an event in C_u such that there is no other event in B_u in that time period. Let Z_u be the set of all such traceable periods.

$$Z_u = \{(\hat{e}_i,\hat{e}_j)|\ (\hat{e}_i \in B_u) \wedge (\hat{e}_j \in C_u) \wedge (\hat{e}_i \prec \hat{e}_j) \wedge (\nexists\, \hat{e}_k \in B_u \text{ s.t. } \hat{e}_i \prec \hat{e}_k \prec \hat{e}_j)\}$$

Finally, the location privacy of user u based on mean time to confusion ($\mathbf{LP}^{\dot{\mathbf{t}}}_u$) and mean location to confusion ($\mathbf{LP}^{\ddot{\mathbf{t}}}_u$), respectively, are computed as follows.

$$\mathbf{LP}^{\dot{\mathbf{t}}}_u \propto \left(\frac{\sum_{(\hat{e}_i,\hat{e}_j)\in Z_u} |\mathbf{tm}(\hat{e}_i) - \mathbf{tm}(\hat{e}_j)|}{|Z_u|}\right)^{-1} \tag{4}$$

$$\mathbf{LP}^{\ddot{\mathbf{t}}}_u \propto \left(\frac{\sum_{(\hat{e}_i,\hat{e}_j)\in Z_u} ||\mathbf{loc}(\hat{e}_i) - \mathbf{loc}(\hat{e}_j)||}{|Z_u|}\right)^{-1} \tag{5}$$

As shown in (4) and (5), the users' location privacy is inversely proportional to *mean time to confusion* and *mean location to confusion* values.

3.1.4 K-Anonymity Metric

The concept of k-anonymity was originally proposed by Samarati and Sweeny [31, 34] as a way to release public information, and ensure both data privacy and integrity, by using generalization and suppression techniques: A (data) release provides k-anonymity protection if the information for each person contained in the release cannot be distinguished from at least $k-1$ individuals whose information also appears in the release. Gruteser and Grunwald [19] extended this concept to the field of location privacy for mobile users through spatial and temporal cloaking of location and time information. In this scheme, the users are indistinguishable from each other and the precision of location-stamps (time-stamps) of events are decreased to a much larger area (time period) to satisfy the k-anonymity conditions.

Now, we formalize k-anonymity in our framework. Let $e_{u,t} \in \mathcal{R}_u$ be the actual event of user u at time t (i.e., $\mathbf{tm}(e_{u,t}) = t$ and $\mathbf{id}(e_{u,t}) = \mathbf{name}(u)$). Let its corresponding observed event be denoted by $\hat{e}_{u,t} = \mathbf{trns}(e_{u,t})$. Note that $\mathbf{tm}(\hat{e}_{u,t})$ is a time period (i.e., a set of time instances). Location privacy of user u based on the k-metric at any time t such that $\hat{e}_{u,t} \neq$ HIDDEN is the number of distinct users v (including u himself) for whom there exists one time instance $t' \in \mathbf{tm}(\hat{e}_{u,t})$ at which the following three conditions are satisfied: (1) The actual location of user v at time t' is in the location area of $\hat{e}_{u,t}$, (2) The time period and location area of $\hat{e}_{v,t'}$ are the same as those of $\hat{e}_{u,t}$, and (3) The pseudonyms of any such user v is equal to that of user u (i.e., $\mathbf{id}(\hat{e}_{u,t})$ is a group pseudonym, e.g., $\ominus$ [19]).

$$\begin{aligned}\mathbf{LP}^{\mathbf{k}}_u(t) \ = \ & |\{v \in \mathcal{U} \mid \exists\, t' \in \mathbf{tm}(\hat{e}_{u,t}) \text{ s.t. } (\mathbf{id}(\hat{e}_{v,t'}) = \mathbf{id}(\hat{e}_{u,t})) \\ & \wedge(\mathbf{tm}(\hat{e}_{u,t}) = \mathbf{tm}(\hat{e}_{v,t'})) \wedge (\mathbf{loc}(\hat{e}_{u,t}) = \mathbf{loc}(\hat{e}_{v,t'})) \\ & \wedge(\mathbf{whereis}(v,t') \in \mathbf{loc}(\hat{e}_{u,t}))\}|\end{aligned} \tag{6}$$

Finally, user u is k-anonymous if $\mathbf{LP}^{\mathbf{k}}_u(t) \geq k$.

3.2 Discussion

In this section, we propose a set of criteria for analyzing location privacy metrics. These criteria are derived from the definition of location privacy.

Criterion 1 – **Probability of Error**: The adversary faces a classification problem in its reconstruction attack (e.g., he has to answer the following questions: "To whom should an observed event be linked?"; or, "Which observed events are linkable to each other?"). Due to the uncertainty of the adversary, the attack is probabilistic in nature with the adversary assigning probabilities to different choices. The adversary's objective is to select the most probable option, thus

minimizing its probability of error. Therefore, the user's location privacy is more related to the adversary's probability of error in choosing the right option rather than to its uncertainty in the selection process. In order to clarify the case, let us consider the following examples. Let $\hat{e}_i, \hat{e}_j, \hat{e}_k \in \widehat{\mathcal{R}}|_{obs}$, $\hat{e}_i \in \widehat{\mathcal{R}}_u|_{obs}$ for user $u \in \mathcal{U}$, $\pi^l(\hat{e}_i, \hat{e}_j) \ll \pi^l(\hat{e}_i, \hat{e}_k)$ and $\pi^l(\hat{e}_i, \hat{e}_j) + \pi^l(\hat{e}_i, \hat{e}_k) = 1$. It means that it is much more probable for the adversary to select $\hat{e}_k$, rather than $\hat{e}_j$, as the immediate successor of $\hat{e}_i$. Obviously, the location privacy of user u is very low if in fact $\hat{e}_k \in \widehat{\mathcal{R}}_u|_{obs}$, and conversely, the user enjoys a high level of privacy if $\hat{e}_j \in \widehat{\mathcal{R}}_u|_{obs}$ because the adversary fails to track u successfully. Uncertainty-based metrics cannot capture this factor because the adversary's uncertainty depends on the the aggregation of $\pi^l(\hat{e}_i, \hat{e}_j)$ and $\pi^l(\hat{e}_i, \hat{e}_k)$ values (when the entropy is computed) and therefore is the same whether $\hat{e}_i \sim_o \hat{e}_k$ or not. As another example, consider two users $u, v \in \mathcal{U}$ who are k-anonymous with $k = 2$. The users' privacy is considered to be the same no matter what is the probability the adversary assigns to users u and v to be associated with the observed events from them. In fact, because k-metrics measure the privacy based on the size of the anonymity set, they implicitly assume that users in the anonymity set are equally likely to be linked to the observed events of that set. Hence, k-anonymity metrics also do not capture the probability of error.

Criterion 2 – **Tracking Error**: The distance between the actual location of a user and his location predicted by the adversary represents the accuracy of the attacker in tracking the user. Therefore, a location privacy metric should take this tracking error into account. Metrics that do not consider the distance between the actual location and the adversary's determined location of users are not able to capture this criterion. Consider again the first example mentioned in Criterion 1. Let $\hat{e}_j = \mathbf{trns}(e_j)$ and $\hat{e}_k = \mathbf{trns}(e_k)$. No matter what the distance between $\mathbf{loc}(\hat{e}_j)$ and $\mathbf{loc}(e_j)$ (or the distance between $\mathbf{loc}(\hat{e}_k)$ and $\mathbf{loc}(e_k)$) is, the measured location privacy of user u using uncertainty-based metrics is the same. This was already observed and mentioned in [15, 20]. The k-metric also does not capture the adversary's error in localizing a user based on the event observed from him: a user with anonymity set size k has the same privacy level, irrespective of the size of its observed-event's location-stamp and the distance between the user's actual location and the location predicted by the adversary.

Criterion 3 – **Actual Trace**: In order to estimate the tracking success of an adversary, the users' actual traces must be taken into account. Apart from $\dot{c}$-metric that considers only a partial set of users' actual traces, for those events with time-stamp in $\mathcal{T}|_{obs}$, none of the other metrics considers this factor. Moreover, the $\dot{c}$-metric does not convey anything about the adversary's success in time instances in which no event from a user is observed. This is crucial in order to capture the effects of the number and the distribution of the observed events associated with a user. If a metric does not capture this criterion, it is not possible for a location privacy preserving mechanism to select the best events to be eliminated in order to maximize the privacy of the users. It becomes more obvious, especially if the gap between two subsequent observed events is large (e.g., before entering and after exiting a city), where the user might enjoy a high level of privacy yet it is not captured in the metrics outlined in Section 3.1.

Criterion 4 – **Location/Time Sensitivity**: The amount of information that the location/time pair in an event reveals about the identity of its associated user differs from one user to another. Therefore, location/time sensitivity plays an important role in a user's location privacy. Any location privacy metric has to take this into account in order to accurately illustrate the privacy of each user. Yet, none of the studied metrics satisfy this criterion.

Criterion 5 – **Measuring the Traceability**: A location privacy metric must be able to capture when, where, how accurately, and for how long the adversary is able to track a user. The longer a user is trackable, the lower his location privacy is; hence the higher the chance of his actual identity disclosure will be. "Clustering error"-based metrics, for example, do not assess traceability of the users. As an example, consider two anonymous users that are initially far from each other, move towards a meeting point and then move away again. At the meeting point, the adversary is uncertain about how to make a link between the observed events immediately before and after the meeting. However, the adversary's uncertainty in reconstructing other parts of their trajectories is low (because their trajectories are far from each other in other parts). In this case, the location privacy of both users is very high according to c-metrics, especially if the adversary's probability of error is high at the meeting point. But, in reality both users are fully trackable during most of the time (from the start to the meeting point and from the meeting point to the end). Similarly, the k-metric also does not capture this criterion because it focuses on observed events individually rather than the relationship of the events with each other.

Criterion 6 – **Genericity**: One of the most important characteristics of a good metric is its applicability in measuring the effectiveness of *different* privacy preserving techniques (explained in Section 2). Virtually all of the metrics in Section 3.1 are designed for specific privacy preserving mechanisms. These mechanisms are composed of three methods: Elimination, obfuscation, and anonymization. In order to evaluate how a metric captures the effects of a privacy preserving method, we consider each one individually.

- *Elimination*: Eliminating more events increases the users' privacy. However, none of the metrics outlined in Section 3.1 can capture this effect. This is because in this case (when there is elimination and no anonymity and obfuscation) the adversary has no error or uncertainty, no matter what the rate of elimination is.
- *Obfuscation*: No uncertainty-based metric (or any metric derived from it) can capture the effects of this method. This is because these metrics do not take the adversary's error into account. Even the k-metrics cannot effectively measure the privacy provided by this method, because they only consider the size of the anonymity set rather than the location area (time-period) in which a user's location (time) is obfuscated.
- *Anonymization*: All of the studied metrics can capture this method's effects on location privacy.

Criterion 7 – **Measurement Granularity**: A metric is more effective if it can measure the location privacy at different granularities, e.g., user level, and system level. "Clustering error"-based metrics, as they give the location privacy only at the system level, cannot reflect the location privacy of the each user individually.

4. DISTORTION-BASED METRIC

In this section, we first give a formal description of distortion-based metric. Next, we present a comparative analysis of our metric based on the set of criteria discussed earlier.

4.1 Description of the Metric

The adversary's reconstruction process is the reverse of the privacy preserving mechanism employed in a system (i.e., the transformation function). It is about finding the relationship among the observed events and recreating any information that has been lost during the transformation. This is done by hypothesizing the set of actual events that are eliminated or obfuscated. Considering the possible routes in the geographic space in which the users move and the adversary's knowledge about the users' mobility patterns, some of the events are more likely than others to be the actual events. Hence, the adversary attempts to remove the uncertainty of the obfuscated events by replacing them with their *representative* events that have the highest probability of being the actual events. It also predicts the probable events between any two consecutive events that are eliminated in the transformation. We capture this by defining the notion of *routes* between the representatives of the observed events. Next, we extend the linkability graph to include such hypothesized events. This graph reflects the possible events that the adversary could consider as the actual events of the users. Using this graph, we estimate the location privacy by computing the expected distortion in the reconstruction attack with regards to the actual trace of the users.

We define function $\mathbf{rep} : \widehat{\mathcal{R}}|_{obs} \rightarrow \mathcal{E}$ to represent the representative of the observed events. An event $\hat{e}' = \mathbf{rep}(\hat{e})$ is called the *representative* of an observed event $\hat{e}$ if it contains the most probable location in $\mathbf{loc}(\hat{e})$ in which $\hat{e}$ could have happened. The representative of any event is computed from the adversary's knowledge about the probability distribution of users. The identity of the representative of an observed event is the same as that of the observed event. In the following, whenever we refer to the observed events we mean their corresponding representative events.

A trace Υ_{ij} is called a *route* between two events e_i and e_j, if $\mathbf{head}(\Upsilon_{ij}) = \hat{e}_i, \quad \mathbf{tail}(\Upsilon_{ij}) = \hat{e}_j, \quad |\overrightarrow{\mathbf{tm}}(\Upsilon_{ij})| = |\Upsilon_{ij}|$, and also $\forall \hat{e}_x \in \Upsilon_{ij}$, the following holds: $\exists\, e_y \in \Upsilon_{ij}$ s.t. $\mathbf{tm}(e_x) = \mathbf{tm}(e_y) + 1 \ \wedge \ \Pr_{\mathrm{A}}((e_y \preccurlyeq e_x) \wedge (e_y \sim_o e_x)) > 0$. This means that the set of events in a route are continuous in time and each event is the immediate successor of its predecessor in the route. We denote by $Routes(e_i, e_j)$ the set of all the possible routes between events e_i and e_j.

Let $\hat{e}_i, \hat{e}_j \in V^l$ be two events such that $\pi^l(\hat{e}_i, \hat{e}_j) > 0$. This implies that it is probable that the adversary believes $\hat{e}_i \sim_o \hat{e}_j$ and $\hat{e}_i \preccurlyeq \hat{e}_j$ (i.e., both of the observed events are associated with the same user and $\hat{e}_j$ is the immediate successor of $\hat{e}_i$). If $\mathbf{tm}(\hat{e}_j) \neq \mathbf{tm}(\hat{e}_i) + 1$, then there must be a set of events (i.e., a trace) between $\mathbf{tm}(\hat{e}_i)$ and $\mathbf{tm}(\hat{e}_j)$ associated with the same user, which are eliminated during the transformation. A route in $Routes(e_i, e_j)$ is a possible hypothesis for such an eliminated trace.

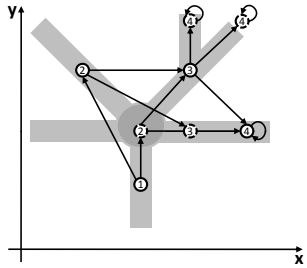


Figure 4: Example of extended graph G^x. Arrows indicate the edges in the graph and the circles indicate the vertices. Solid circles represent the observed events' representatives and dashed circles represent the events added in the extension procedure.

We extend the linkability graph G^l by considering the routes between the observed events. Let $G^x = (V^x, E^x, \pi^x)$ denote the extended linkability graph. The set of vertices and edges of the graph are represented by V^x and E^x, respectively. The weight function π^x in G^x is defined similarly to that of graph G^l based on the adversary's knowledge, and there is the following relation between them.

$$\pi^l(\hat{e}_i, \hat{e}_j) = \sum_{\Upsilon \in Routes(\hat{e}_i, \hat{e}_j)} \pi^{x*}(\Upsilon)$$

where $\pi^{x*}(\Upsilon)$ is the probability assigned to a trace Υ. It says that the probability assigned to any edge in the routes between two observed events $\hat{e}_i$ and $\hat{e}_j$ collectively determine the probability the adversary assigns to the link $(\hat{e}_i, \hat{e}_j)$.

The representatives of the events in V^l are in V^x. All events that belong to any route between any pair of linked events in V^l are added to V^x.

$$V^x = \overrightarrow{\mathbf{rep}}(V^l) \cup \Big\{ \hat{e} \in \Upsilon_{ij} \,\big|\, \Upsilon_{ij} \in Routes(\hat{e}_i, \hat{e}_j), (\hat{e}_i, \hat{e}_j) \in E^l \Big\}$$

Subsequently, E^x is defined by creating links between events in V^x based on π^x, as we did previously for graph G^l. Figure 4 represents an example of the extended graph G^x for the scenario represented in Figure 2(b).

In graph G^x, all the events in V^x with time-stamp equal to T are the only events that are potentially the tails of traces, because there is no event that can happen afterwards. Hence, $\pi^x(e, e) = 0$ for every $e \in V^x$ such that $\mathbf{tm}(e) \neq T$. Moreover, there is a self loop for every event $e \in V^x$ such that $\mathbf{tm}(e) = T$. Besides, for every $e \in V^x$ we have $\sum_{e' \in V^x} \pi^x(e, e') = 1$.

Now, we define a specific path in graph G^x, called *Tpath*, based on which we estimate the possible set of events that might be hypothesized for each user. This will be used in computing the distortion of the hypothesized traces and eventually the users' location privacy. Let $t \in \mathcal{T}$, $e \in V^x$ s.t. $\mathbf{tm}(e) < t$ and Υ be a path in G^x. The path Υ is called a *Tpath* from e until t, if the following holds.

$$\mathbf{head}(\Upsilon) = e, \quad \mathbf{tm}\big(\mathbf{tail}(\Upsilon)\big) = t, \quad |\Upsilon| = t - \mathbf{tm}(e) + 1$$

The set of *all* Tpaths from e until t is denoted by $Tpaths(e, t)$. Any path $\Upsilon \in Tpaths(e, t)$ represents a possible trace that

can be hypothesized by the adversary when it tracks a user from one of his observed events e until time instance t.

We define the expected distortion in a reconstructed trace of user $u \in \mathcal{U}$ at time instance $t \in \mathcal{T}$ denoted by $\mathrm{ED}(u,t)$ as follows. Let $e_t = \mathbf{tail}(\widehat{\mathcal{R}}_u|_{obs \leq t})$ be the *last* observed event from user u *before* time instance t. Then, we have:

$$\mathrm{ED}(u,t) = \sum_{\Upsilon} D\Big(\mathbf{whereis}(u,t), \mathbf{loc}\big(\mathbf{tail}(\Upsilon)\big)\Big) \cdot \pi^{x*}(\Upsilon)$$

where $\Upsilon \in Tpaths(e_t, t)$ and function $D : V^s \times V^s \rightarrow [0,1]$ is a normalized distance function between two locations.

Thus, for estimating the distortion of the reconstructed event of user u at time instance t, first we find the latest event from u observed at or before time instance t, which is denoted by e_t. Then, we take all the paths that start from e_t and end in any event with time-stamp equal to t. The location-stamp of these events represents the hypothesized locations of the user u at time instance t, each assigned a probability. Considering these probabilities we can compute the expected distortion of the location of u at instance t.

To compute a user's location privacy, we also take his personal location/time sensitivity factor into account. This is because the amount of information that can be inferred from the observed events of the users for finding their actual identities is highly user dependent [18, 27]. For example, if an event is observed from a user when he is at home at midnight, then it is easy for the adversary to find the actual identity associated with that event. In other words, de-anonymization of those events that belong to sensitive loction/time pairs are easier for the adversary. A user's sensitivity to his own location information over time is captured by the *location/time sensitivity* function $\mathbf{lts} : \mathcal{U} \times V^s \times \mathcal{T} \rightarrow [0,1]$. The higher the level of distortion around sensitive locations of a user is, the higher his location privacy is. Subsequently, we compute the distortion-based location privacy of a user $u \in \mathcal{U}$ at time $t \in \mathcal{T}$ as follows.

$$\mathbf{LP}^{\mathbf{d}}_u(t) = 1 - \mathbf{lts}\big(u, \mathbf{whereis}(u,t), t\big) \cdot \big(1 - \mathrm{ED}(u,t)\big) \quad (7)$$

Given the location privacy of a user u at any time instance t, the overall location privacy of the user can be computed as his average location privacy over $\mathcal{T}$, as follows.

$$\mathbf{LP}^{\mathbf{d}}_u = \frac{1}{T} \cdot \sum_{t \in \mathcal{T}} \mathbf{LP}^{\mathbf{d}}_u(t) \quad (8)$$

The traceability of a user can be computed similarly to time/location to confusion metrics outlined in Section 3.1.3. To compute the distortion-based traceability, first, we redefine the confusion point in a user's trace as the time instance in which the amount of distortion is above a threshold. Moreover as opposed to t-metrics, we take into account all the time instances in $\mathcal{T}$, and not only those in $\mathcal{T}|_{obs}$.

The system-level location privacy, i.e., the overall privacy of users, is computed as $\mathbf{LP}^{\mathbf{d}} = \frac{1}{n}\sum_u \mathbf{LP}^{\mathbf{d}}_u$. The minimum level of privacy provided to the users, i.e., $\min_u \left(\mathbf{LP}^{\mathbf{d}}_u\right)$, can also be computed for the worst case analysis.

4.2 Discussion

In the proposed distortion-based metric, a user's location privacy is estimated as the level of distortion in his actual trace reconstructed by reversing the privacy preserving mechanisms. The metric also considers the sensitivity of a user's location information over time.

We propose a list of criteria for location privacy metrics in Section 3.2 and show that existing metrics capture only a subset of the criteria as they consider only part of the adversary's knowledge and part of the information contained in the actual events. Hence, these metrics effectively capture location privacy only in some specific scenarios. Our metric leverages on the strengths of existing metrics outlined in Section 3.1 and also fulfills all the proposed criteria. More specifically, our metric considers the probability of a hypothesized trajectory for a user at any time instance and its expected distance to the actual location of the user and hence satisfies the first three criteria (i.e., Probability of error, Tracking error, and Actual trace). This distortion is weighted based on the users' personal sensitivity to different locations at different time instances. This is crucial for computing the users' location privacy and thus fulfilling criterion 4 (i.e., Location/Time sensitivity). Additionally, our metric computes the users traceability in the same way as t-metrics; hence fulfilling criterion 5 (i.e., Measuring the traceability). As our metric is based on the process of reconstruction by the adversary and the uncertainty associated with it, it is able to effectively capture the effects of the various location privacy preserving methods, namely, elimination, obfuscation, and anonymization. Thus, our metric also satisfies criterion 6 (i.e., Genericity). Lastly, criterion 7 (i.e., Measurement granularity) is fulfilled as we compute the users' privacy at any time instance, their overall privacy and also the system level privacy.

From our analysis of the existing location privacy metrics presented in Section 3 and the description of our distortion-based metric in Section 4.1, it is clear that the proposed formal framework, specifically the linkability graph model (Section 2.5), clearly captures the required criteria to measure location privacy. In particular, the adversary model assumed in existing metrics and the proposed distortion-based metric can also be captured within the linkability graph-based model. The distortion-based metric measures the location privacy of a user at a given time instance by computing the expected distance between the actual and the hypothesized path in the extended graph, beginning from the last observed event before that time instance. Hence, in practice, only a small part of the extended linkability graph is required at each step of the privacy computation. Consequently, irrespective of the overall size of the extended graph, such a piece-wise computation is an important efficiency feature of the distortion-based metric. The u-metrics and k-metrics are even simpler to compute within the current framework. The c-metrics and the t-metrics, on the contrary, generally require much more information about the linkability graph structure for the time period during which the user privacy has to be estimated. This can have computation space (and time) implications depending on the size of the linkability graph used in these metrics.

5. CONCLUSION AND FUTURE WORK

In this paper, we proposed a formal framework to model location privacy; it is general enough to capture the notion of privacy under different types of location privacy preserving mechanisms. In order to provide a common platform for evaluating the existing location privacy metrics, we shortlisted a set of criteria derived from the location privacy requirements in a mobile network setting. Based on these criteria and with the help of some representative counter-

examples, we showed that existing metrics do not capture location privacy completely in all cases. Finally, we proposed a novel distortion-based metric for measuring location privacy under a variety of privacy preserving mechanisms, which also fulfills all the proposed set of criteria.

We would like to extend the existing framework in order to model time obfuscation methods for location privacy. Further, we would like to show some concrete scenarios in which the proposed metric can be applied and address the implementation and computation specific issues in such target scenarios.

Acknowledgments

We would like to thank Marcin Poturalski, Marco Gruteser, Maxim Raya, Hossein Manshaei and also the anonymous reviewers for their insightful feedback on earlier versions of this work.

6. REFERENCES

[1] http://en.wikipedia.org/wiki/Bluedating.

[2] http://www.aka-aki.com.

[3] http://csg.ethz.ch/research/projects/Blue_star.

[4] C. Ardagna, M. Cremonini, E. Damiani, S. D. C. di Vimercati, and P. Samarati. Location privacy protection through obfuscation-based techniques. In *Data and Applications Security XXI*, 2008.

[5] D. B. Reid. An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, 24(6), 1979.

[6] A. R. Beresford. *Location privacy in ubiquitous computing*. PhD thesis, University of Cambridge Computer Laboratory, 2005.

[7] A. R. Beresford and F. Stajano. Location privacy in pervasive computing. *IEEE Pervasive Computing*, 2003.

[8] A. R. Beresford and F. Stajano. Mix zones: User privacy in location-aware services. *IEEE PerCom Workshops*, 2004.

[9] A. T. Campbell, S. B. Eisenman, N. D. Lane, E. Miluzzo, and R. A. Peterson. People-centric urban sensing. In *ACM WICON*, 2006.

[10] C. Cornelius, A. Kapadia, D. Kotz, D. Peebles, M. Shin, and N. Triandopoulos. Anonysense: privacy-aware people-centric sensing. In *ACM MobiSys*, 2008.

[11] G. Danezis and C. Diaz. A survey of anonymous communicaiton channels. Technical Report MSR-TR-2008-35, Microsoft Research, Cambridge, UK, 2008.

[12] C. Diaz, S. Seys, J. Claessens, and B. Preneel. Towards measuring anonymity. In *PET*, 2002.

[13] M. Duckham and L. Kulik. A formal model of obfuscation and negotiation for location privacy. In *Lecture Notes in Computer Science 3468*, 2005.

[14] M. Edman, F. Sivrikaya, and B. Yener. A combinatorial approach to measuring anonymity. In *IEEE Intelligence and Security Informatics*, 2007.

[15] L. Fischer, S. Katzenbeisser, and C. Eckert. Measuring unlinkability revisited. In *ACM WPES*, 2008.

[16] B. Gedik and L. Liu. Protecting location privacy with personalized k-anonymity: Architecture and algorithms. *IEEE Trans. on Mobile Computing*, 2008.

[17] B. Gierlichs, C. Troncoso, C. Diaz, B. Preneel, and I. Verbauwhede. Revisiting a combinatorial approach toward measuring anonymity. In *ACM WPES*, 2008.

[18] P. Golle and K. Partide. On the anonymity of home/work location pairs. 2009.

[19] M. Gruteser and D. Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. In *ACM MobiSys*, 2003.

[20] B. Hoh and M. Gruteser. Protecting location privacy through path confusion. In *SECURECOMM*, 2005.

[21] B. Hoh, M. Gruteser, R. Herring, J. Ban, D. Work, J.-C. Herrera, A. M. Bayen, M. Annavaram, and Q. Jacobson. Virtual trip lines for distributed privacy-preserving traffic monitoring. In *ACM MobiSys*, 2008.

[22] B. Hoh, M. Gruteser, H. Xiong, and A. Alrabady. Preserving privacy in gps traces via uncertainty-aware path cloaking. In *ACM CCS*, 2007.

[23] L. Huang, H. Yamane, K. Matsuura, and K. Sezaki. Towards modeling wireless location privacy. In *PET*, 2005.

[24] L. Huang, H. Yamane, K. Matsuura, and K. Sezaki. Silent cascade: Enhancing location privacy without communication qos degradation. In *Security of Pervasive Computing (SPC)*, 2006.

[25] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, and S. Madden. Cartel: a distributed mobile sensor computing system. In *ACM SenSys*, 2006.

[26] T. Jiang, H. J. Wang, and Y.-C. Hu. Preserving location privacy in wireless lans. In *ACM MobiSys*, 2007.

[27] J. Krumm. Inference attacks on location tracks. In *Pervasive*, 2007.

[28] M. Li, K. Sampigethaya, L. Huang, and R. Poovendran. Swing & swap: user-centric approaches towards maximizing location privacy. In *ACM WPES*, 2006.

[29] A. Pfitzmann and M. Köhntopp. Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management - a consolidated proposal for terminology., 2008.

[30] O. Riva and C. Borcea. The urbanet revolution: Sensor power to the people! *IEEE Pervasive Computing*, 6(2), 2007.

[31] P. Samarati and L. Sweeney. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. In *IEEE Symposium Research in Security and Privacy*, 1998.

[32] A. Serjantov and G. Danezis. Towards an information theoretic metric for anonymity. In *PET*, 2002.

[33] C. E. Shannon. A mathematical theory of communication. *The Bell system technical journal*, 1948.

[34] L. Sweeney. k-anonymity: a model for protecting privacy. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, 10(5), 2002.