Protecting Location Privacy against Location-Dependent Attack in Mobile Services

Xiao Pan
School of Information
Renmin University of China
smallpx@ruc.edu.cn

Jianliang Xu
Dept of Computer Science
Hong Kong Baptist University
xujl@comp.hkbu.edu.hk

Xiaofeng Meng
School of Information
Renmin University of China
xfmeng@ruc.edu.cn

ABSTRACT
Privacy preservation has recently received considerable attention for location-based mobile services. In this paper, we present *location-dependent attack* resulting from continuous and dependent location updates and propose an incremental clique-based cloaking algorithm, called ICliqueCloak, to defend against location-dependent attack. The main idea is to incrementally maintain maximal cliques for location cloaking in an un-directed graph that takes into consideration the effect of continuous location updates.

Categories and Subject Descriptors
H.2.m [DATABASE MANAGEMENT]: Miscellaneous

General Terms
Algorithms, Performance, Information Privacy.

Keywords
Location Privacy, Location-dependent Attack, LBS

1. INTRODUCTION
With advances in wireless communication and mobile positioning technologies, location-based mobile services have been gaining increasingly popular in recent years. A lot of research has been carried out on about how to enjoy location-based services while protecting location privacy of mobile users [1, 2, 3]. An important technique for location privacy protection is location cloaking. It blurs user locations into cloaked regions by reducing their spatial and temporal resolutions. However, most of the existing location cloaking algorithms work with *snapshot* locations only and do not consider the effect of continuous location updates, which may introduce serious privacy compromise. If an attacker (e.g., the service provider) can collect the user's historical cloaked regions as well as the user's mobility pattern (e.g., speed limit), the location privacy might be disclosed. For example, as shown in Figure 1, users A, B, and C are cloaked together at time t_i, and their cloaked region is $R_{A,ti}$. If the attacker knows the maximum speed v_A, the **m**aximum **m**ovement **b**oundary (MMB) of A is a round rectangle at t_{i+1}, denoted by $MMB_{A,ti,ti+1}$. Then at t_{i+1}, A is cloaked with E and F, with cloaked region $R_{A,ti+1}$. We can see that there is an overlap (i.e., the shaded area) between $MMB_{A,ti,ti+1}$ and $R_{A,ti+1}$. As a consequence, the attacker can infer that A must reside in the shaded area at t_{i+1}, which fails to meet the expected privacy requirement. In the worst case, the overlapped area is just a location point, which discloses the exact location. We call this kind of attack *location-dependent attack*.

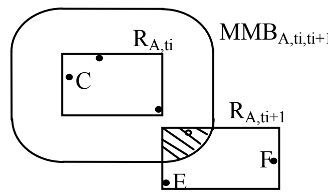

Figure 1. Location-dependent attack

In order to deal with this problem, in this paper we propose a new location cloaking algorithm, called *ICliqueCloak*, to incorporate the effect of continuous location updates in the process of location cloaking. We use a graph model to formulate the problem. Each mobile user is represented by a node in the graph; an edge exists between two nodes/users only if they are within the MMB of each other and can be potentially cloaked together. To meet the location *k*-anonymity requirement, the problem becomes finding *k*-node cliques in the graph such that all the nodes within a clique form a cloaking set. To reduce the computational complexity, we propose to maintain the maximal cliques incrementally.

2. PRELIMINARIES
We employ an un-directed graph model to formalize the location cloaking problem.

DEFINITION 1 (*Graph Modeling*). Let G(V, E) be an undirected graph where V is the set of nodes/users who submitted location-based query requests, and E is the set of edges. Assume that the last cloaked region of user u is $R_{u,ti-1}$ at t_{i-1}. The current time is t_i. There exists an edge e_{vw} between two nodes/users v and w, if and only if

1) $v \neq w$
2) v is covered by $MMB_{w,ti-1,ti}$
3) w is covered by $MMB_{v,ti-1,ti}$
4) $Area(MBR(v, w)) < A_{max}$

Conditions 1), 2), and 3) collectively ensure that two users can be cloaked together if and only if they have different ids and are within each other's MMB (and hence free of location-dependent attack). To find the cloaked region satisfying location k-anonymity, it has been shown in [2] that this problem is equivalent to the problem of finding k-node cliques (i.e., k-node complete subgraphs) in the corresponding graph G(V, E). Once a k-node clique is found, all the users within the clique may form a cloaking set and the minimum bounding rectangle (MBR) of their

CIKM'08, October 26-30, 2008, Napa Valley, California, USA.
ACM 978-1-59593-991-3/08/10.

spatial locations can be used as the cloaked region. Note that Condition 4) above is a necessary condition to guarantee that the cloaked region is smaller than the user-tolerable maximum area A_{max}. Yet it is not a sufficient condition. When a k-node clique is found, we should still check if the area of the MBR is smaller than A_{max}.

3. ICLIQUECLOAK ALGORITHM

We define *maximal clique* as a clique that is not contained in any other clique. The main idea of our proposed ICliqueCloak approach is as follows. We start with a graph without any edges. All nodes themselves constitute a set of 1-node cliques. Then we add the edges to the graph one by one and incrementally update the set of *maximal cliques*. In the following, we shall discuss how to incrementally maintain the maximal cliques and find cloaking sets based on the maximal cliques.

DEFINITION 2 *(t-Parameterized Graph*). Consider an undirected graph G=(V, E), where V is the set of nodes and E is the set of edges. Define G_0=(V, ϕ). Adding each edge $e_{vw} \in E$ to G_0, G can be parameterized as

$$G_t = (V, E_t) \qquad t = 1, \dots, |E|,$$

where E_t is the set of edges added so far, E_t-E_{t-1}=e_{vw}, and $E_0 = \phi$.

For a t-parameterized graph G_t, let C_t be the set of maximal cliques and C_{v_t} be the set of maximal cliques which contain node v. Before a new edge e_{vw} is added, the cliques in C_t can be partitioned into three classes: 1) the cliques containing node v (C_{v_t}); 2) the cliques containing node w (C_{w_t}); 3) the cliques containing neither v nor w. It has been proved in [4] that adding the edge e_{vw} to the graph can only alter the maximal cliques in C_{v_t} or C_{w_t}. Thus, for incremental updating of maximal cliques, we only need to consider the cliques in C_{v_t} and C_{w_t}. First, for any clique in $C_{v\_t} \cap C_{w\_t}$, all its nodes are connected to v and w. Therefore, it will upgrade to a new larger clique after the edge e_{vw} is added. Next, we need to check whether the cliques in C_{v_t} and C_{w_t} are still maximal. Specifically, for any clique $c_i \in C_{v\_t}$ and $c_k \in C_{v\_t} \cap C_{w\_t}$, if c_i- c_k = {v}, c_i is no longer a maximal clique in C_{t+1} for G_{t+1}, because $c_k \cup$ {v, w} will take place to be a new maximal clique. Similarly, for any clique $c_j \in C_{w\_t}$ and $c_k \in C_{v\_t} \cap C_{w\_t}$, if c_j- c_k = {w}, c_j is no longer a maximal clique in C_{t+1} for G_{t+1}.The time complexity of this algorithm is $O(nE^2)$, where n is the number of users/nodes, and E is the number of maximal cliques that a graph can have.

After updating the maximal-clique set, the cliques where the user of the new request is involved might be candidate cloaking sets. They can be classified to three classes: *positive candidates, negative candidates,* and *not candidates*. For a positive candidate, all users in it can be cloaked together since they satisfy both k-anonymity and maximum area requirements. Therefore, the MBR of all users in the positive candidate can be returned as the cloaking region. For a negative candidate, to find the cloaking set, the algorithm first sorts the users in the clique by their privacy level k. Then, it repeatedly removes the user with the highest privacy level until the number of remaining users is larger than or equal to the maximum privacy level k and the area of their MBR is smaller than the user-tolerable maximum area A_{max}. Then this MBR is returned as the cloaking region. For more details, please refer to our technical report [4].

We have implemented the proposed ICliqueCloak algorithm in C++ and evaluated its performance in terms of average cloaking time[4]. As shown in Figure 2, ICliqueCloak is very fast with average cloaking time shorter than 0.5 ms under all privacy level settings tested.

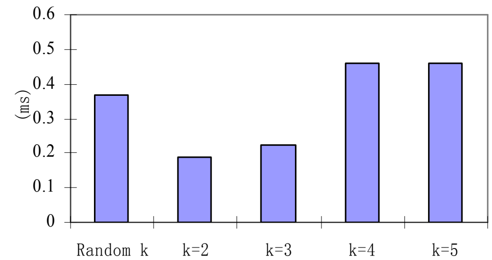

Figure 2. Average cloaking time

4. CONCLUSION

In this paper, we investigated cloaking algorithms that can protect location privacy against location-dependent attack. We have employed a graph model to formalize the problem and transformed it to the problem of finding k-node cliques in the graph. An incremental clique-based cloaking algorithm called ICliqueCloak has been proposed to generate cloaked regions.

5. ACKNOWLEDGMENTS

This research was partially supported by the grants from the Natural Science Foundation of China under grant number 60573091; China 863 High-Tech Program(No:2007AA01Z155); China National Basic Research and Development Program's Semantic Grid Project (No. 2003CB317000). Jianliang Xu's work was supported in part by the Research Grants Council, Hong Kong SAR, China (Project Nos. HKBU211206 and HKBU211307).

6. REFERENCES

[1] J. Du, J. Xu, X. Tang, and H. Hu. iPDA: enabling privacy-preserving location-based services. In *Proceedings of the International Conference on Mobile Data Management (MDM)*, 2007.

[2] B. Gedik and L. Liu. Location privacy in mobile systems: a personalized anonymization model. In *Proceedings of the International Conference on Distributed Computing Systems (ICDCS)*, Columbus, OH, USA, 2005, pp. 620–629.

[3] H. Hu and D. Lee. Range nearest-neighbor query. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 18(1):78–91, 2006.

[4] X.Pan, J.Xu, and X.Meng. Protecting Location Privacy against Location-Dependent Attack in Mobile Services. Technical Report, Renmin University, 2008