Privacy-Preserving Computation for Location-based Information Survey via Mobile Cloud Computing

Hao Zhang†, Yonggang Wen*, Nenghai Yu† and Xinwen Zhang‡

*School of Computer Engineering, Nanyang Technological University, Singapore 639798
Email: ygwen@ntu.edu.sg
†Key Laboratory of Electromagnetic Space Information,
University of Science and Technology of China, ACS, Hefei 230027, PRC
Email: zhh1000@mail.ustc.edu.cn, ynh@ustc.edu.cn
‡Huawei Research Center, Santa Clara, CA, USA
Email: xinwen.zhang@huawei.com

***Abstract*—Location-Based Service (LBS) has become pervasive, riding along the trend of mobile application proliferation. Given its high utility value, LBS, however, present serious privacy concerns for cautious users. In this paper, we investigate privacy preserving for location-based information survey application, which calculates the geographic distribution of user's information. The design objective is twofold: i) calculate an information distribution for a pool of mobile users, and ii) protecting the location and value privacy of individual user, in the presence of malicious servers and possible corrupted users. Our proposed solution leverages a mobile cloud computing paradigm, in which each mobile device is replicated with a system-level clone in a proximate cloud environment. The computing of distribution function is distributed among the set of cloud clones, via a peer-to-peer (P2P) protocol. Compared to the approaches based on centralized server or aggregate proxy, our scheme is advantageous in avoiding single point of failure/attack, load balancing and overhead reduction. Simulation results verify these advantages and suggest that our proposed scheme is suitable for large scale applications.**

I. Introduction

Owing to the wide adoption of mobile Internet, location-based service (LBS) has gained significant attention from both academia [1] and industry [2]. Examples of widely-deployed LBSs include Foursquare, Yelp and etc. These signature LBSs serve important functions in the proliferation of mobile application. As a subclass of LBS, location-based information survey application (LB-ISA for short) also becomes more and more popular (e.g., CarTel [3] and BikeNet [4]), where individuals contribute their values and locations to a server for calculating some geographic distribution. For example, CarTel [3] uses mobile phones to collect road information to learn the traffic condition.

However, LB-ISAs are exposing the privacy of mobile users (e.g., location and value[1]) to potential exploitation, starting to cause concerns to cautious users. Indeed, location information carries huge value if used properly. One widely-touted example was that a suspected killer was apprehended via location tracking [5]. On the other hand, J. Krumm [6] suggested that it is possible to find out individual's home and identify the individual from anonymous location database. Such unintended usage could bring tremendous risks to the mobile user. At the same time, collecting of values (and mining them) associated with individuals, such as health condition, income and etc, could serve important functions (e.g.,Microsoft HealthVault [7]). Nevertheless, much of this information is too private for individual to report it in public, in the presence of malicious players in the system. As such, further success of LBS applications highly depends on the feasibility of preserving privacy for mobile users, protecting both individual's location and value information.

Previous solutions for LBS privacy preserving fall into two architectural paradigms. One approach is to leverage a centralized proxy, assumed to be trust-worthy, to accept information from mobile users and present the information in which individual privacy is erased to the LBS server [8], [9], [10], [11]. The LBS server cannot obtain individual user's privacy-sensitive information. However, the proxy could be the bottleneck of the system, suffering from a single point of failure/attack problem and capability limit. The other approach is to delegate the privacy protection scheme to mobile devices, as in [12], [13], [14], [15]. This solution, however, could drain the resources of smartphones (CPU, memory and power) quickly, in providing privacy protection such as encryption/decryption, caching and communication overheads. To protect the value privacy, SMC (Secure Multiparty Computation) [16], [17] is a good choice to calculate the statistics without any private information disclosure. However, the protocol could result in huge message overhead, limiting its practicality in large mobile systems with millions of users, even in a single metropolitan area.

In this paper, we leverage the emerging mobile cloud computing paradigm [18] to preserve privacy in LB-ISAs. Specifically, each mobile device is shadowed by a system-level clone in a proximate cloud infrastructure. The computation of distribution function is dynamically assigned to this set of cloud clones, which formulate a stabler P2P network than their physical counterparts. Moreover, these cloud clones have access to more system resources than their physical devices, offering extended capability for privacy-preserving computing.

[1]In terms of data privacy, both "location" and "value" are the attributes of data and can be treated equally. However, in the LBS applications "location" is an exceptive attribute that plays an important role. As such, it is singled out from the normal attributes of users.

978-1-4673-2815-9/13/$31.00 ©2013 IEEE

Under this architecture, we design a robust protocol for privacy preserving LB-ISAs and evaluate its effectiveness via analysis and numerical simulation. Our contributions include:

- We identify the problem of privacy-preserving information survey in LBSs and introduce a novel architecture to avoid the potential pitfalls in previous centralized and distributed solutions.
- Analytical investigation establishes the robustness of our protocol, defending against potential attackers.
- Numerical simulations verify the effectiveness of our protocol, providing better load-balancing, reduced message overhead, and receivable communication cost in large scale applications.

The rest of the paper is organized as follows. Related work is presented in Section II. We present our system model and state the LS-ISA problem in Section III. In Section IV we describe our proposed scheme in detail. In Section V the security of our scheme is analyzed and its effectiveness is verified via numerical simulations. We conclude the paper and propose the future work in Section VI.

II. Related Work

Location privacy. The most common method to protect location privacy is the k-anonymity [8], in which a cloaking region containing at least k unrecognizable users is used to hide the identity of each user. For example, in [9], a cell-divided method is exploited to generate the desiring regions. The privacy protection strength of k-anonymity can be further enhanced by advanced ideas, such as l-diversity [19]. Considering the uniqueness of road networks, Mixzone [10] and path confusion [20] are two special solutions to hide the identity of individual user.

Notice that these aforementioned methods normally adopt an architecture involved with a trusted proxy, which could be the bottleneck for the system. Lately new approaches are removing the trusted proxy. For example, cache method [12] stores the data on the mobile phone. However, this method could drain the onboard resource with the mobile device. Other methods such as Private Information Retrieval (PIR) [14] could provide the strongest privacy protection by cryptology methods, nevertheless suffering from its complexity.

Query/Value privacy. Theoretically secure multi-party computation (SMC) is an excellent solution for value privacy. However, its practicality is often challenged in real system. For example, [16] presents a secure sum protocol in which the summation is calculated serially. It would consume a large amount of time for a large-scale system. In [17], the summation is allowed on untrusted servers, in which it uses participates' keys. However, it requires the sum of all the parties' keys to be 0. If one of the party leaves in the process, which is a common case in large-scale system, the summation cannot be calculated. Previously, dummy query is used to hide the true query or value in multiple fake ones [21], which would generate too much useless information. PrivStats [15] solves a similar problem as the problem in this paper. However, it suffers from the channel varying in wireless environment.

TABLE I. Aggregation Function List

Aggregation Function	Construction with summation $sum(v)$
Count: $count(v)$	the value of each individual is 1
Average: $avg(v)$	$sum(v)/count(v)$
Standard Deviation: $std(v)$	$\sqrt{avg(v^2) - avg(v)^2}$

III. Problem Statement and System Architecture

In this section, we first present a problem statement for privacy-preserving location-based information survey service. Following that, we suggest a system architecture based on mobile cloud computing, in which each mobile device is shadowed by a system-level clone located in a proximate cloud. In this architecture, we highlight the fact that the computation is delegated into a subset of cloud clones in the system. Finally we list our assumptions about threat models and attack vectors.

A. Problem Statement

In order to calculate the required distribution of LB-ISA on an interesting area of unit square, we subdivide the area into $k_0 \times k_1$ grids, and calculate the aggregation statistic in each grid. We assume that n users are distributed in this area, in which user u_i, $i = 0, 1, \cdots, n-1$ has a value of v_i and a *location* at (x_i, y_i). In practice, the unit square can be scaled proportionally. We only consider a snapshot of the scenario and our approach can be extended to a slotted time horizon.

The aggregation computation is based on the requirement of distribution function. For example, if we want to survey the population distribution in a city, we can count the number of person located in each subdivided grid of the city. Previously, Popa *et al* [15] outlined a list of possible aggregation statistics, including summation, average, standard deviation and count. Specifically, summation is the basic one, in which the server processes the aggregation function $sum(v) = \sum v_i$. Other aggregation functions can be constructed by summation, as outlined in Table I. Therefore, the problem of interest in this paper can be stated as follows:

Problem: *A given unit area of* ***R*** *is subdivided into* $m = k_0 \times k_1$ *subregions* $\{r_i | 0 \leqslant i \leqslant m-1\}$. n *users with locations* $\{(x_i, y_i) | 0 \leqslant i \leqslant n-1\}$ *are distributed in* ***R****, each of which has a value* v_i*. A protocol is required to calculate the summation of values in each of the subregion and cannot disclose the individual's value and location.*

B. System Model

In this paper, we leverage the emerging mobile cloud computing paradigm to design privacy-preserving location-based information survey protocol. Our system architecture is illustrated in Fig. 1. In particular, each mobile device is replicated by a system-level clone located in a cloud infrastructure, acting on behalf of its physical counterpart. Compared to its physical device, a mobile clone has access to more system resources (e.g., CPU, RAM, bandwidth, etc). Moreover, the set of mobile clones can formulate a stabler P2P network than their corresponding mobile counterparts. We leverage the set of mobile clones as potential candidate for proxy to compute the summation of individual values in subregions and hide the location/value information of individual user.

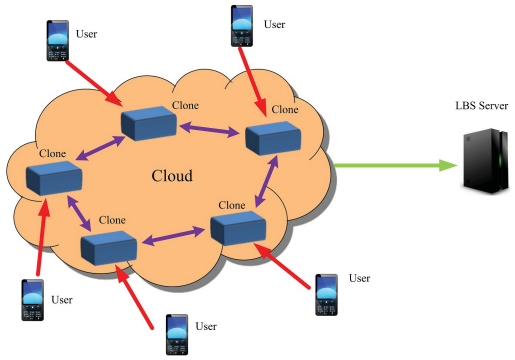

Fig. 1. System architecture based on mobile cloud computing: each mobile device is replicated by a clone in a proximate cloud, acting on behalf of its physical device.

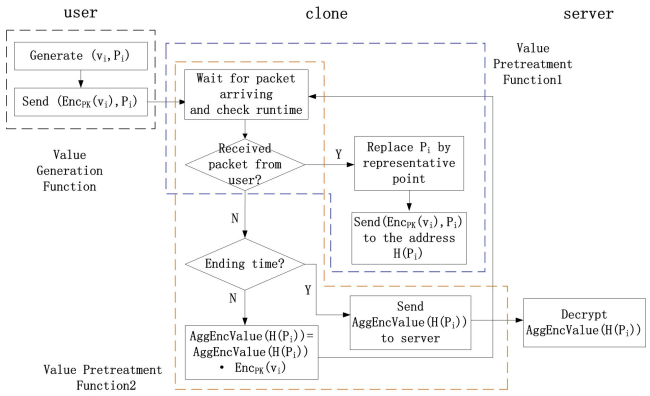

Fig. 2. Workflow of our scheme

Comparing to the existing dedicated proxy solution, our solution exhibits a few advantages. First, the amount of system resource allocated to mobile clone can be dynamically determined in response to the application demand. Second, over a period of time, the communication load can be shared among all the participating clones, achieving load balancing.

C. Threat Model

We assume that *the LBS server is honest but cannot be trusted*. The server executes the protocols honestly to get the statistics, but it also makes an effort to find out individuals' privacy information such as positions and values from the data it receives. Moreover, because the LBS server is in the form of centralization that all the reported data is stored on it, it is quite possible that hackers could steal it to find out some useful information about users by side information or background knowledge when they break into the server.

Cloud providers are semi-trusted. We assume that cloud providers are honest that they do not furtively monitor and record the data and operations on the clones. However, the security protection they provide is limited so that the attackers may compromise the clone. This assumption is reasonable because the main service that cloud providers offer is remote storing/computing. Monitoring of users' information will greatly worsen cloud providers' reputation. At the same time, there is no perfect security system in the cloud, so the clones could be compromised.

Malicious clone can get the data that passes and stores on it, and can combines with other clones. Malicious clone is the one belonging to a malicious user or compromised by an attacker. It participates in the pretreatment of raw data as other clones. The attacker can obtain all the data routed and preprocessed by the compromised clone. Furthermore, many malicious users may collaborate with others or the attacker can control many clones to try to disclose individual's privacy. Since both malicious users and attackers cause the same threaten, we do not distinguish "malicious user" and "attacker" by default.

Notice that we only consider the passive adversary which only eavesdrops information without modifying them. The more powerful active adversary which modifies the information it captures will be considered in future.

IV. Privacy-Preserving Information Survey Based on Mobile Cloud Computing

In this section we describe the details of our proposed scheme. Suppose the interesting area **R** is subdivided m subregions, and n users lying in it (see the **Problem** in Section III-A). The server wants to get the sum of values of users in each subregion. The workflow is shown in Fig. 2.

A. Scheme Initialization

Our scheme requires an asymmetric encryption scheme, which is defined by the following tuples:

- $(PK, SK) \leftarrow \mathsf{KeyGen}(\kappa)$. It is a probabilistic polynomial time (PPT) algorithm that takes a security parameter κ as input and outputs a key pair containing a public key PK and a private key SK.
- $C \leftarrow \mathsf{Enc}_{PK}(m)$. It is a PPT algorithm that takes a public key PK and a plaintext message m as inputs and outputs a ciphertext C.
- $m \leftarrow \mathsf{Dec}_{SK}(C)$. It is a deterministic algorithm that takes a secret key SK and a ciphertext C as inputs and outputs a plaintext message m.

For correctness, we require that for any plaintext message m, we have $m \leftarrow \mathsf{Dec}_{SK}(\mathsf{Enc}_{PK}(m))$ if $(PK, SK) \leftarrow \mathsf{KeyGen}(\kappa)$ for some security parameter κ. We additionally require the encryption scheme to have the homomorphic property:

- **Homomorphic Addition of plaintexts**: $\mathsf{Enc}_{PK}(m_1 + m_2) = \mathsf{Enc}_{PK}(m_1) \cdot \mathsf{Enc}_{PK}(m_2)$.

Paillier cryptosystem [22] is a good candidate to satisfy all the above requirements.

Server executes the KeyGen algorithm. It publishes PK while keeping SK private. It also claims the subregions $\{r_i | 0 \leqslant i \leqslant m-1\}$ it requires, where r_i=$\{<(X_i, Y_i), (X_i', Y_i') >\}$ satisfying $X_i < X_i'$ and $Y_i < Y_i'$. Suppose $r_i \cap r_j = \varnothing (i \neq j, 0 \leqslant i, j \leqslant m-1)$, which means each user only belongs to one subregion.

For users, each mobile phone has a clone in the cloud. Suppose the phone and clone are connected by a secure

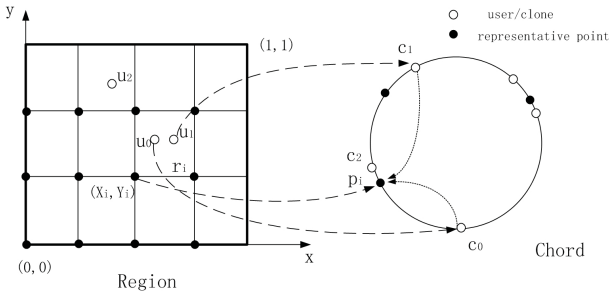

Fig. 3. Illustration the mapping from region to Chord

channel such as SSL, regardless of the connection way (Wi-Fi, GPRS, 3G, etc.).

For clones, all the clones form a structured P2P network. Here we use Chord which is one of the most classical P2P routing protocol to organize the clones. Each clone runs as a peer in Chord. The identity (ID) of peer is a pseudorandom number, such as the hash value of the phone number, guaranteeing that each peer has an unique ID, and its position on the ring is not selected elaboratively.

B. Scheme in Details

Our scheme is composed by three phases: value generation, value pretreatment and value aggregation. Value generation is responsible for the generation of individuals' information, which is implemented on the user side. Value pretreatment wipes off both location and value privacy of individuals, which is carried out on the clone side. Value aggregation computes the statistics on server. In the following the details of the phases are described.

1) Value Generation: The function Value Generation Function (Table II) processes at the user part. In this phase each user generates the encrypted value and location pair, and transmits the pair to his clone in the cloud. Suppose user $u_i(0 \leqslant i \leqslant n-1)$ generates value v_i, and the location is P_i (x_i, y_i). u_i encrypts v_i by server's public key PK, and sends $\big(\mathsf{Enc}_{PK}(v_i), P_i\big)$ to his clone c_i.

TABLE II. VALUE GENERATION FUNCTION

Value Generation Function Functions run by u_i	
1	generate (value, location) pair (v_i, P_i)
2	encrypt the value v_i using PK
3	send $\big(\mathsf{Enc}_{PK}(v_i), P_i\big)$ the pair to clone c_i

2) Value Pretreatment: At the beginning of value pretreatment phase, each clone c_i $(0 \leqslant i \leqslant n-1)$ contains one (value, location) pair. In value pretreatment phase there are two functions. One function protects the privacy of locations by fake location technology or rounding [6]. The other protects the privacy of values by aggregation.

The former function (Value Pretreatment Function1, Table III) is executed by every clone. For clone c_i, firstly it finds out the subregion that its location (x_i, y_i) belongs to. The subregions are claimed by server. Each clone downloads them to find the subregion locally. If c_i finds a subregion r_i satisfying $X_i \leqslant x_i \leqslant X_i'$ and $Y_i \leqslant y_i \leqslant Y_i'$, r_i is the subregion its location belongs to. Then c_i uses the *representative point* of r_i (X_i, Y_i) to displace his original location, where the *representative point* is a point delegating the subregion. Here we select the lower left point of the subregion as representative point ((X_i, Y_i) for subregion r_i in Fig. 3). In this way all the locations in r_i are changed to (X_i, Y_i). Then clone sends the pair $(\mathsf{Enc}_{PK}(v_i), (X_i, Y_i))$ with the destination address $\mathcal{H}((X_i, Y_i))$, where $\mathcal{H}$ is a collision-resistant hash function that maps any arbitrary string to a point on Chord ring. So the n pairs are sent to m destinations, which are maintained by m clones at most. In Fig. 3 u_0 and u_1 locate in the same subregion, so the clones of both take P_i as the destination, which is the mapping of the representative point (X_i, Y_i) of r_i. As the formulation of Chord protocol, the clockwise nearest peer c_2 takes charge of p_i. For the hash mapping from realistic region to the Chord ring, the corresponding user u_2 lies in the region randomly.

TABLE III. VALUE PRETREATMENT FUNCTION1

Value Pretreatment Function1 Functions run by c_i	
1	$< (X_i, Y_i), (X_i', Y_i') >$ = FindSubregion(P_i)
2	$P_i = (X_i, Y_i)$
3	send $\big(\mathsf{Enc}_{PK}(v_i), (X_i, Y_i)\big)$ to the address $\mathcal{H}(P_i)$

Another function Value Pretreatment Function2 (Table IV) is executed by every clone too, but only the processes on the clones receiving (value, location) pairs are active (*active clone* for short, such as c_2 in Fig. 3), while others are in the suspended state. When an active clone receives the (value, location) pairs by Chord protocol, he multiplies all the values with the same location. Suppose $\mathcal{H}(P_i)$ is between c_j and its predecessor, so the $\big(\mathsf{Enc}_{PK}(v_i), P_i\big)$ is transmitted to c_j which is an active clone. At first c_j initializes the aggregate result $\mathsf{AggEncValue}\big(\mathcal{H}(P_i)\big) \leftarrow \mathsf{Enc}_{PK}(0)$, which is the ciphertext of zero. Then the product of all encrypted values in the address $\mathcal{H}(P_i)$ is calculated when the packets generated by Value Pretreatment Function1 of some other clones come. This step lasts until time $T_i' + \epsilon$ where T_i' is the end time of collecting user's information and ϵ is time margin to guarantee all the packets can be received. At last c_j sends $\big(\mathsf{AggEncValue}\big(\mathcal{H}(P_i)\big), r_i\big)$ to the server. In order to avoid congestion at server, the sending time is selected randomly in the time interval $[0, T_s]$, where T_s is the maximum delay given by the scheme.

TABLE IV. VALUE PRETREATMENT FUNCTION2

Value Pretreatment Function2 Functions run by c_i	
1	$\mathsf{AggEncValue}\big(\mathcal{H}(P_i)\big) \leftarrow \mathsf{Enc}_{PK}(0)$
2	while get value v_i with address $\mathcal{H}(P_i)$ and nowtime $< T_i' + \epsilon$
3	$\mathsf{AggEncValue}\big(\mathcal{H}(P_i)\big) = \mathsf{AggEncValue}\big(\mathcal{H}(P_i)\big) \cdot \mathsf{Enc}_{PK}(v_i)$
4	endwhile
5	send $\big(\mathsf{AggEncValue}\big(\mathcal{H}(P_i)\big), r_i\big)$ to the server with random delay in $[0, T_s]$

3) Value Aggregation: In this phase, server receives all the (result, subregion) pairs. Server decrypts the results by his private key. There are m subregions in all, so server gets a set $\{\big(\mathsf{AggEncValue}\big(\mathcal{H}(P_i)\big), r_i\big) | 0 \leqslant i \leqslant m-1\}$. When server receives $\big(\mathsf{AggEncValue}\big(\mathcal{H}(P_i)\big), r_i\big)$ from c_j, he decrypts to get $\sum v_i \leftarrow \mathsf{Dec}_{SK}\Big(\mathsf{AggEncValue}\big(\mathcal{H}(P_i)\big)\Big)$, which is the sum of the values in subregion r_i. Now server gets the summation aggregation in each subregion.

V. Security Analysis and Performance Evaluation

In this section, at first we analyze the security of our scheme. It shows that our scheme can preserve the privacy of individuals from both untrusted server and malicious users. Then the performance, especially the communication cost of our scheme is evaluated. Shown by the simulation, our scheme balances the load efficiently, which is especially suitable for large-scale applications.

A. Theoretic Security Analysis

Our scheme aims both location and value privacy of individuals from untrusted server and malicious users. From the perspective of server, the data it receives only is the necessary information for statistics calculation, i.e. the encrypted aggregate results of the subregions without any individual information. It fundamentally prevents the attacks from server regardless of any side information the server obtains. Therefore server cannot violate the individuals' privacy.

Privacy of individuals' information on clone is another part we should concern. We also need to prevent privacy disclosure from malicious clones. We focus on the malicious user who has a clone participating in the Chord network and attacker who compromises clone. They can eavesdrop the packets in the P2P network. We use the hash of phone number as the ID of clone in Chord, so the ID is not easy to change arbitrarily and frequently, which means that the clone can not choose whether it is an active clone, or which subregion it maintains.

If the malicious clone is not an active clone, it can monitor all the passing packets, which contain encrypt values and locations of others. However, the clone does not obtain the private key of server, so the values cannot be decrypted. The location information can be obtained, but it has been rounded by user's clone before the packet is sent into Chord. So the attacker only knows the subregions the users belong to and nothing about the exact locations, which is not regarded as the individuals' privacy anymore. Therefore, the attacker does not compromise the privacy of users' locations and values.

If the malicious clone is an active clone of some subregion, it additionally has all the users' packets in the subregion besides the data in the former situation. Similarly, the values can not be decrypted and the locations have been formulated. The potential threaten is that the adversary knows the number of users in the subregion, which may be helpful to recognize the user when the number is small. But usually in survey applications there are plenty of individuals in one subregion which satisfies k-anonymity easily. Therefore malicious active clone also cannot violate the location and value privacy. Furthermore, collusion of multiple malicious clones does not weaken the strength of privacy protection, just having more encrypted data and rounded locations.

In our problem, the region is divided by server according to the application, which is a tradeoff of the privacy and utility. If the application only wants to calculate simple statistics such as the average of values in the whole region, one subregion (no division in another word) meets the requirement. However in this case too much traffic converges to the active clone. Dividing the region into several proper subregions can balance the traffic of clones, which is evaluated in Section V-B. However, if much subregions are subdivided, some subregions may have few locations, easily violating individual's privacy. The extremely case is each subregion only contains one person, which violates individual's privacy easily. Thus adequate size of subregion is necessary to meet the k-anonymity in each subregion. Here we do not focus on how to divide the region into proper subregions, which is out of the scope of the paper.

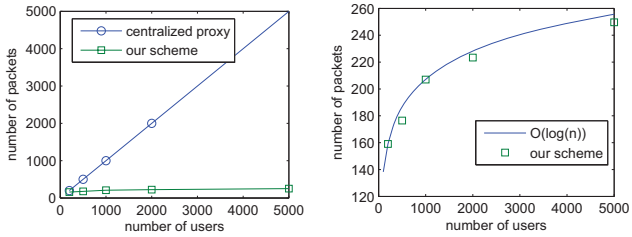

Fig. 4. In Fig.4(a), for the method with centralized proxy, the proxy receives all the data. The number of receive packets is $O(n)$. In our scheme, data is distributed to all the clones. Each clone receives much less packets, or rather $O(\log(n))$ packets in average (Fig. 4(b)).

B. Performance Evaluation

1) Simulation Setup: We simulate our scheme based on OverSim [23] to see the performance, especially the communication cost of clones. In the experiment, we adopt the default parameters of Chord in OverSim. We divide the interesting region into m subregions with equal size. We set the runtime of our protocol 200s and $m = 20$ as default. Users are uniformly distributed in the m subregions. Each user has one clone, and has one (value, location) pair.

2) Performance Investigation: We measure the communication cost of clones by the number of received packets. They can be divided into types. One is generated by the data procession in value pretreatment phase. The other is caused by the maintenance of Chord itself. In the P2P network, the packets a clone receiving are not only the packets to it, but also the packets routed by it. Fig. 4 illustrates the average packets the clone receiving. By way of contrast in the method with centralized proxy, the proxy receives all the data from users. So it receives $O(n)$ packets, where n is the number of users. Our scheme distributes the data by the P2P network. Compared to the method with centralized proxy, in our scheme the average number of packets each clone receiving (including the packets it routes) grows in logarithmic way (Fig. 4(b)). For example, the number is 159.035 when $n = 200$. It only increases to 249.680 when n reaches 5000. Thus our scheme is especially suitable for large-scale applications.

Actually, some clones, such as aggregation clones, are under the heavier load than others. Fig. 5 shows the difference between aggregation clones and other clones, where the communication cost of aggregation clone is larger than other clone's. The packets clone receiving can be divided into two categories. One contains the packets maintaining the P2P protocol. The other contains the packets for application. The communication cost of aggregation clone is caused by the application data, especially when large number of users exist.

Increasing the number of subregion can reduce the cost of aggregation clones. Fig. 6 shows the communication cost

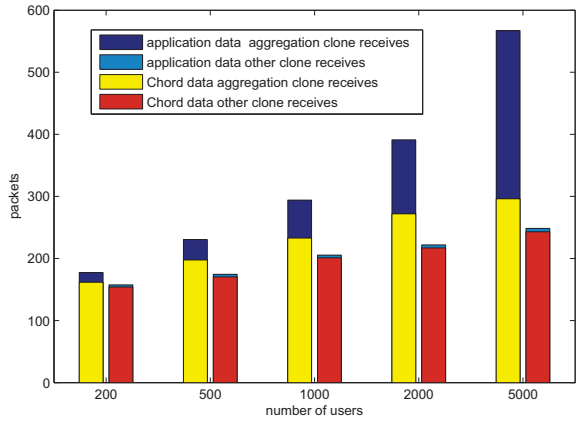

Fig. 5. Comparison of the communication cost of aggregation clone and other clone.

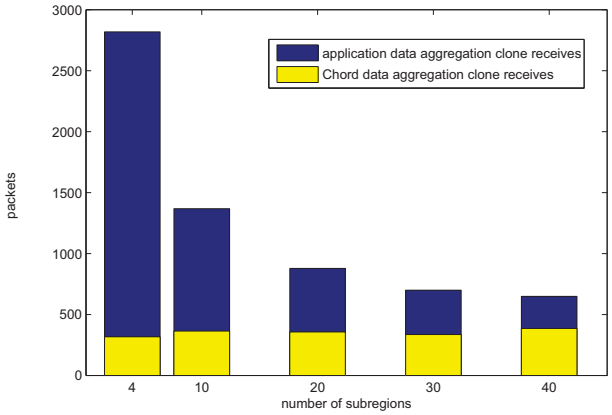

Fig. 6. The communication cost of aggregation clone with different number of subregions.

of aggregation clones, where there are 10000 users. The load can be distributed by adding the number of subregions. In our experiment, the average number of received packets is reduced from 2818.25 to 649.075 when m increases from 4 to 40. Thus the load can be balanced by splitting subregions.

VI. Conclusion and Future Work

In this paper, we proposed a novel structure to protect both individuals' location privacy and value privacy against untrusted server and other users/attackers in LB-ISAs. From the security analysis and performance evaluation, our scheme is efficient to defend the passive adversary without modifying the data, especially in large scale applications. In the future work, we would enhance the power of adversary, to protect the individual privacy against active adversary.

References

[1] D. Mohapatra and Jakkur, "Survey of location based wireless services," in *ICPWC'2005*, 2005.

[2] I. A. Junglas and R. T. Watson, "Loca tion-based services," vol. 51, no. 3, pp. 65–69, March 2008.

[3] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, A. M. Michel Goraczko, E. Shih, H. Balakrishnan, and S. Madden, "Cartel: A distributed mobile sensor computing system," in *Proceedings of the 4th ACM International Conference on Embedded Networked Sensor Systems*, November 2006.

[4] S. B. Eisenman, E. Miluzzo, N. D. Lane, R. A. Peterson, G.-S. Ahn, and A. T. Campbell, "The case for vm-based cloudlets in mobile computing," *ACM Transactions on Sensor Networks*, vol. 6, no. 1, December 2009.

[5] "Justice dept. loses round in warrantless phone tracking," http://news.cnet.com/8301-13506_3-20102518-17/justice-dept-loses-round-in-warrantless-phone-tracking/.

[6] J. Krumm, "Inference attacks on location tracks," in *Fifth International Conference on Pervasive Computing*, May 2007.

[7] "Healthvault," https://www.healthvault.com.

[8] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking," in *MobiSys '03*, May 2003.

[9] X. Pan, J. Xu, and X. Meng, "Protecting location privacy against location-dependent attacks in mobile services," *IEEE Transactions on Knowledge and Data Engineering*, May 2011.

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

[11] B. Palanisamy and L. Liu, "Mobimix: Protecting location privacy with mix-zones over road networks," in *The IEEE International Conference on Data Engineering*, April 2011.

[12] S. Amini, J. Lindqvist, J. Hong, J. Lin, E. Toch, and N. Sadeh, "Cach: caching location-enhanced content to improve user privacy," in *MobiSys '11*, June 2011.

[13] R. Shokri, P. Papadimitratos, G. Theodorakopoulos, and J.-P. Hubaux, "Collaborative location privacy," in *2011 IEEE 8th International Conference on Mobile Adhoc and Sensor Systems*, October 2011.

[14] S. Papadopoulos, S. Bakiras, and D. Papadias, "Nearest neighbor search with strong location privacy," in *The 36th International Conference on Very Large Data Bases*, September 2010.

[15] R. A. Popa, A. J. Blumberg, and H. Balakrishnan, "Privacy and accountability for location-based aggregate statistics," in *CCS'11*, October 2011.

[16] C. Clifton, M. Kantarcioglu, J. Vaidya, X. Lin, and M. Y. Zhu, "Tools for privacy preserving distributed data mining," *ACM SIGKDD Explorations Newsletter*, vol. 4, December 2002.

[17] E. Shi, T.-H. H. Chan, E. Rieffel, R. Chow, and D. Song, "Privacy-preserving aggregation of time-series data," in *Network & Distributed System Security Symposium(NDSS)*, February 2011.

[18] Y. Wen, W. Zhang, and H. Luo, "Energy-optimal mobile application execution: Taming resource-poor mobile devices with cloud clones," in *the 31st IEEE International Conference on Computer Communications*, March 2012.

[19] A. Machanavajjhla, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data*, vol. 1, March 2007.

[20] J. Meyerowitz and R. R. Choudhury, "Hiding stars with fireworks: Location privacy through camouflage," in *MobiCom09*, September 2009.

[21] A. Pingley, N. Zhang, X. Fu, H.-A. Choi, S. Subramaniam, and W. Zhao, "Protection of query privacy for continuous location based services," in *INFOCOM 2011*, April 2011.

[22] I. Damgard and M. Jurik, "A generalisation, a simplication and some applications of paillier probabilistic public-key system," in *Public Key Cryptography*, 2001.

[23] I. Baumgart, B. Heep, and S. Krause, "OverSim: A flexible overlay network simulation framework," in *Proceedings of 10th IEEE Global Internet Symposium (GI '07) in conjunction with IEEE INFOCOM 2007, Anchorage, AK, USA*, May 2007, pp. 79–84.