Privacy-Preserving Location Assurance Protocols for Mobile Applications

Genqiang Wu[a,b,c], Yeping He[a], Yi Lu[a], Liping Ding[a]
[a] NFS, Institute of Software Chinese Academy of Sciences, Beijing 100190, China
Email: genqiang80@gmail.com, {yeping, luyi, liping}@nfs.iscas.ac.cn
[b] Graduate University Chinese Academy of Sciences, Beijing 100190, China
[c] School of Information Engineering, Lanzhou University of Finance and Economics, Lanzhou 730020, China

Abstract— **Location-based applications require a user's location data to provide customized services. However, location data is a sensitive piece of information that should not be revealed unless strictly necessary which induces the emerging of a number of location privacy protection methods, such as anonymity and obfuscation. However, in many applications, one needs to verify the authenticity and other properties (e.g. inclusion to an area) of location data which becomes an intractable problem because of the using of location privacy protection. How to achieve both location assurance, i.e. assuring the authenticity and other properties of location data, and location privacy protection seems to be an intangible problem without complex trusted computing techniques.**

By borrowing range proof techniques in cryptography, however, we achieve them both successfully with minimized trusted computing assumptions. The Pedersen commitment scheme is employed to give location data a commitment which would be used for possibly future location assurance. Area proof, testing whether a private location is within some area, is employed to test whether or not the location data having the commitment is within any definite area. Our system model do not rely on third trusted party and we give reasonable explanations for our system model and for the trusted computing assumptions.

We present a new range proof protocol and a new area proof protocol which are based on a new data structure, i.e. Perfect k-ary Tree (PKT). Some deeper properties of PKT are presented which are used to analyze our protocols' complexity. The analysis results show that our protocols are more efficient that the former and are flexible enough to support some existing mobile applications, such as tracking services and location-based access control.

Index Terms— **location privacy, range proof, area proof, location assurance, tracking services, location-based access control**

I. Introduction

A. Background and Motivation

With the rapid development of wireless sensor networks and smart phone techniques, many applications have employed users' location data as their essential elements, such as location-based services [1], [2].

There is one class of application which needs users providing the authentic location data in order to test whether it satisfies some confinement, such as tracking services or location-based access control. These applications are unable to rely solely on users to transmit the correct location data to the server since users have great incentive to spoof. Instead, these applications require their users to be able to assure the server of the reliability of their locations thereby eliminating, or at least vastly reducing, the possibility of spoofing. Moreover, user's location privacy needs to be protected. Intuitively, location authenticity can be achieved through hashing and signing the location data, such as in location proof schemes [3], [4]. However, in signing or hashing schemes the only way to verify the authenticity and other (inclusion) properties of location data is to input the location data which makes it impossible to protect location privacy. One possible alternative scheme is the using of trusted computing techniques [5] which is yet far from to be in practice.

Our research is mainly motivated by two location-based applications. The first is tracking services [5] where user's moving area are confined, such as court may demand that a person should not go out of the country or the city region, or the parents need to confirm that their children be in safe areas which are adjacent to their home. The second is location-based access control [6]–[10] where users need to prove that their locations are within some area before they can get some services.

In the tracking services application, one party needs to be confirmed that the monitored party is always within some valid area. Obviously, the easiest way to solve the problem is to monitor the party and to get the real-time location data of the monitored party. However, it is obvious that this is intrusive to the monitored party's location privacy. One scheme is needed not only to confirm the validity of the monitored party's location information, the location of the monitored party is within the valid area, but also to protect the location privacy of the monitored party.

In the location-based access control, the server also needs to be confirmed that the user is within some

Manuscript received August 21, 2013; revised January 9, 2014; accepted January 21, 2014. © 2005 IEEE.

The research is supported by following funds: National Science and Technology Major Project under Grant No.2010ZX01036-001-002 & 2010ZX01037-001-002, and the Knowledge Innovation Key Directional Program of Chinese Academy of Sciences under Grant No.KGCX2-YW-125 & KGCX2-YW-174.

doi:10.4304/jsw.9.8.1991-2001

definite valid area when the user sends a request to the server. Meanwhile, the user's location privacy needs to be protected.

We now summarize the above two applications and give the meaning of "Privacy-preserving location assurance". In the paper "location assurance" has two meanings; (1) assuring the authenticity of location data. (2) assuring the claimed properties of location data, e.g., whether it is within valid area (VA). "Privacy-preserving location assurance" means that the method not only protects the location privacy with controllable (any) degree but also have assurance to location data.

We borrow range proof techniques [11] in zero-knowledge proof to solve our problems. In range proof there have two parties A, B. A has a private integer σ and its commitment C. A needs to prove to B that σ is in a publicly known interval $[a, b]$ and that it is the committed secret integer of C. The range proof is very suitable to solve our problem. First, the commitment scheme can be used to cope with the authenticity of location data. Second, there are many range proof protocols we can choose to extend them to design area proof and to solve our problems.

However, it needs to be noted that most range proof protocols have relative high communication and computational complexity, which is a fatal weakness to mobile application where mobile devices have limited resources. Moreover, These range proof protocols are designed to be used for a specified integer range which would not meet our flexible applications, such as multi-area proof in location-based access control. Therefore, our new schemes should be able to cope with the new situations.

B. Our Contribution

We list our contribution as follows.

- We introduce range proof techniques in cryptography to treat location assurance and location privacy problems.
- We generalize the binary-tree index method in [12] to the k-ary tree index and use it to design a secure and efficient range proof protocol and a secure and efficient area proof protocol. Our range proof protocol, to the best of our knowledge, is the first to treat union of multi-range proof problem, which is especially useful in some applications.
- We compare the complexity of our protocols with the one in [11]. The result shows that our protocols are especially efficient than that in [11]. We give some deeper properties of the k-ary tree index and use these properties to reduce the complexity of corresponding protocols by choosing suitable parameter k. Our protocols have the property of balancing the complexity of each party and of the whole. In the extreme case, our protocol can reduce the computational complexity of Prover (the mobile user) into very smaller level or even into constant level which is especially useful to mobile users with limited resources.
- We introduce a system model which do not rely on trusted third party. We compare our model with other models and give reasonable explanations for our model.
- We use our privacy-preserving location assurance schemes to analyze two applications; tracking services and location-based access control.

C. Organization

The rest of the paper is organized as follows: Section II gives the related works. Section III and IV present the system model and some preliminaries. In Section V area proof protocol is illustrated. Next, in Section VI and Section VII we discuss the security and complexity of the area proof protocols. Section VIII extends the area proof protocol to give the privacy-preserving location assurance protocols. In Section VIII, possible applications of our protocols are also discussed. Section IX illustrates the implementation of the system and Section X gives the conclusions.

II. Related Work

The paper is mainly related to three domains; range proof in cryptography, location proof and localization assurance service provider, and secure integer comparison in secure multiparty computation.

A. Range Proof

The range proof [11], [13], [14] is to show that a committed private integer lies in a specified integer range. It is frequently used in anonymous credential and e-cash scenarios [15]–[17].

We borrow range proof protocols to treat location privacy problems. Although our area proof protocol is mainly based on range proof protocols, it has several differences from them. First, our area proof protocol treat area data, the two dimensional data, whereas the range proof protocols treat interval data, the one dimensional data. Intuitively, $(x, y) \stackrel{?}{\in} I_1 \times I_2$ can be achieved simply through first testing $x \stackrel{?}{\in} I_1$ and then testing $y \stackrel{?}{\in} I_2$. However, the later may leak some vital information about (x, y) , exactly whether $x \stackrel{?}{\in} I_1$ or $y \stackrel{?}{\in} I_2$, than the former and so we need to choose suitable range proof protocol which can be efficiently transformed to area proof protocol without leaking more secrets.

Second, our range proof protocol (and related area proof protocol) is mainly aimed at (not specified) arbitrary interval or union of intervals, whereas theirs are aimed at a specified interval.

Furthermore, our protocols are aimed at mobile users who have limited resources. Therefore, the efficiency of protocols, especially to the client side (mobile users), would be a vital point.

B. Location Proof and Localization Assurance Service Provider

There are a number of papers study location proof [3], [18]. A location proof [3], [18] is a piece of data that certifies a geographical location. In general, a user's location data is obtained from Global Positioning System (GPS) traces or cell phone triangulation. However, due to its vulnerability to malicious users to support false location claims, some papers suggest using records of access points (AP) where she visits as a provenance to the location data. That scheme seems to alleviate the false location claims. However, it introduces an auditor to check the validity of the records users submitted which makes the trust chain more complex [4]. Furthermore, most of these location proof schemes do not provide location privacy protection methods.

There are some researches by completely using trusted computing techniques to solve the location proof and the privacy protection [5]. However, trusted computing needs the hardware's support which made it somewhat costly and is hard to implement. Therefore, we should minimize the use of the trusted computing techniques where it could be.

Localization assurance service provider [19] is to provide assurance of GPS signals.

Our work is different from these schemes. Our work emphasize mainly on the location assurance with privacy-preserving methods after the GPS receiver has received correct signals and has calculated correct location data.

C. Secure Integer Comparison and Secure Multiparty Computation

Our work is also related to the secure integer comparison - the well-known "Millionaires' Problem" [12], [20]–[24]. Due to its significant applications, such as electronic auction [23], secure interval check [12], location-based access control [2], [25], and proximity test [1], [26]–[31], it has been studied extensively. Our range proof and area proof is similar with the secure interval check where two parties securely check whether $x \in [a, b]$ where one party holds x and the other holds $[a, b]$ without leaking each other's secrets.

However, our schemes are different from the secure integer comparison schemes; almost all secure integer comparison schemes are based on the semi-honest assumption, where each party follows the protocol with the exception that it keeps all its intermediate computations, which is not practical in our application scenarios. Our schemes don't rely on the semi-honest model but assume the existence of malicious parties which makes our schemes more practical than the formers. Furthermore, their schemes only solve the location privacy of two party but ours is to not only solve the location privacy of one party but also the location assurance which seems impossible by using their methods.

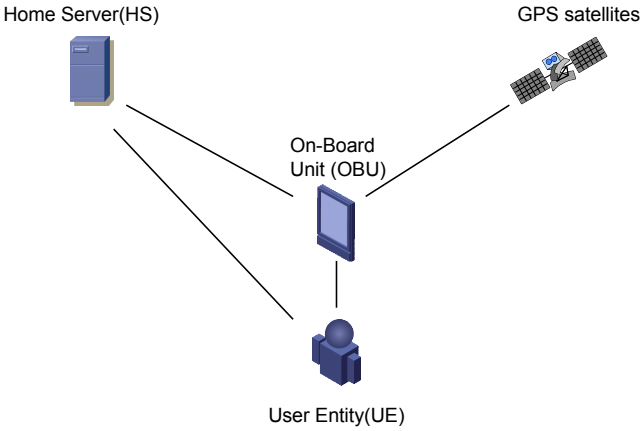

Figure 1. Entities in our system model

III. SYSTEM MODEL

The system model has three parties.

User Entity(UE). UE normally equips a unique On-Board Unit(OBU) embedded in UE's mobile device.

Home Server(HS). HS manages UE's subscription data, including subscriber services and global location handling. The HS assigns the permanent UE identity and the HS-UE security credentials.

On-Board Unit(OBU). OBU is an electronic device installed in UE subscribed to HS. It is in charge of collecting GPS data and doing some other operations, such as evaluating signatures and encoding data.

Figure 1 shows entities in our system model.

We suppose that OBU's operation can't be altered by UE and UE should not be able to spoof the GPS signal. These assumptions needs trusted computing techniques to assure the claimed properties. However, our aim is trying to minimize the trusted computing assumptions in our system model. We then would try our best to minimize the function of the OBU and so reduce the use of trusted computing.

Our system model is mainly derived from two applications, i.e. tracking services and location-based access control. They both have some common nature which could be defined in formal model. First, the authenticity of location is vital. Second, there must exist some way to assure that location data is within some area. Third, UE's location privacy needs to be protected.

There are some areas which are defined by HS solely or by HS and UE jointly. UE needs to prove to HS that its location data is within the area. We called these areas Valid Area(VA). VA, a rectangular area or the union of several rectangular areas in the paper, is publicly known to UE and HS. It is vital that the system must assure that (σ, τ) is the true location data of UE at time t since UE have great motivation to alter his location data. Then we should first confirm that (σ, τ) couldn't be altered by UE. On the other hand, HS should only be confirmed that $(\sigma, \tau) \stackrel{?}{\in} VA$ but no any other knowledge about (σ, τ). Then the problem is illustrated as follows.

- There exists VA which is defined by HS solely or

by HS and UE jointly. VA is public to UE and HS. It can be defined arbitrarily which means that it can be any rectangular area or union of rectangular area and that it can be defined at any time in protocol.
- UE obtains his current location data (t, σ, τ) from OBU embedded in her mobile device.
- UE needs to assure HS that (σ, τ) is her authentic location at time t.
- UE proves that $(\sigma, \tau) \stackrel{?}{\in} VA$ to HS but no any other knowledge of (σ, τ) to HS.
- UE can't alter the value of location data (σ, τ). Otherwise, her malicious behavior would be detected by HS.

Although our system model is similar to [32], they have major difference. The OBU in [32] is in charge of all communication and computation of the vehicle it is embedded, including calculating the location data from GPS signals and fee computation and fee proof, whereas ours is only in charge of calculating the location data and limited other computations, i.e. commitment computation and signature computation etc. All other costly steps are computed by UE('s mobile device), which makes our OBU a relatively simple chip and is easy to be implemented.

Our system model is also different from [33] where a trusted third party is in charge of verifying and signing location data. However, in our model they are computed by the OBU through trusted computing techniques which escapes the need of trusted third party, a party that is not available in practice.

Remark 1: The aim of our system model is to assure location data with privacy-preserving mode without trusted computing assumption which seems hard to be implemented. However, we find that it seems impossible to securely calculate UE's location data by using so called "inside-out" sensing [34], in which UE to be located looks outside itself for location beacons, without trusted computing assumptions. The only way to escape the dilemma seems to introduce a trusted third party which, to a certain degree, downgrades to so called "outside-in" sensor [34], depending on measurements made by the surrounding infrastructure, which has more location privacy problems. Therefore, our aim is to use as little trusted computing assumptions as possible on which our protocols are based.

IV. Preliminaries

A. Basic Notation

In the paper, the GPS coordinates of UE and the area data are all in the form of integers. Let (σ, τ) denote the GPS coordinates of UE. The notation $[x, y]$, for the two integers x, y, denotes an interval of integers, i.e., $[x, y] = \{z \in \mathbb{Z} : x \leq z \leq y\}$. The notation $[(x_1, y_1), (x_2, y_2)]$, for the four integers x_1, x_2, y_1 and y_2, denotes the area $\{(x, y) : x \in [x_1, x_2] \wedge y \in [y_1, y_2]\}$.

We denote $S \times B$ as the Cartesian product of two sets S and B, i.e. $S \times B = \{(s, b) : s \in S \wedge b \in B\}$.

B. Computational Assumptions

As in [11], our protocols require bilinear groups and associated hardness assumptions. Let PG be a pairing group generator that on input 1^k outputs descriptions of multiplicative groups $\mathbb{G}_1$ and $\mathbb{G}_T$ of prime order p where $|p| = k$. Let $\mathbb{G}_1^* = \mathbb{G}_1 \backslash \{1\}$ and let $g \in \mathbb{G}_1^*$. The generated groups are such that there exists an admissible bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_T$, meaning that (1) for all $a, b \in \mathbb{Z}_p$ it holds that $e(g^a, g^b) = e(g, g)^{ab}$; (2) $e(g, g) \neq 1$; and (3) the bilinear map is efficiently computable.

C. Strong Diffie-Hellman Assumption and Boneh-Boyen Signatures

Definition 1 (Strong Diffie-Hellman assumption): We say that the q-SDH assumption associated to a pairing generator PG holds if for all p.p.t. adversaries A, the probability that $A(g, g^x, \ldots, g^{x^q})$ where $(\mathbb{G}_1, \mathbb{G}_T) \leftarrow PG(1^k)$, $g \leftarrow \mathbb{G}_1^*$ and $x \leftarrow \mathbb{Z}_p$, outputs a pair $(c, g^{1/(x+c)})$ where $c \in \mathbb{Z}_p$ is negligible in k.

Our scheme, as in [11], relies on the Boneh-Boyen signature scheme which we briefly summarize. The signer's secret key is $x \leftarrow \mathbb{Z}_p$, the corresponding public key is $y = g^x$. The signature on a message m is $\sigma \leftarrow g^{1/(x+m)}$; verification is done by checking that $e(\sigma, y \cdot g^m) = e(g, g)$.

D. Perfect Binary Index

The paper [12] presents a binary tree index. We summarize their main results as follows.

Definition 2 (Tree node): A *tree node* is a data structure that consists of a unique label (h, o) and possibly two pointers *left* and *right* to other tree nodes. The label has two components: the height h and the order o. A *leaf node* is a tree node with a label $(0, o)$ and no pointers.

Definition 3 (ℓ_n Perfect Binary Tree (ℓ_n PBT)): An ℓ_n *perfect binary tree* is a binary tree with a set of leaf nodes $L = \{(0, 0), (0, 1), \ldots, (0, n-1)\}$ and a set of non-leaf nodes NL such that $(\ell_n, 0) \in NL$ and for all $(h, o) \in NL$, there exists $(h-1, 2o), (h-1, 2o+1) \in (L \cup NL)$ with $(h, o).left = (h-1, 2o), (h, o).right = (h-1, 2o+1)$. In an ℓ_n PBT, the root node is the node $(\ell_n, 0)$.

Figure 2 shows 3 PBT.

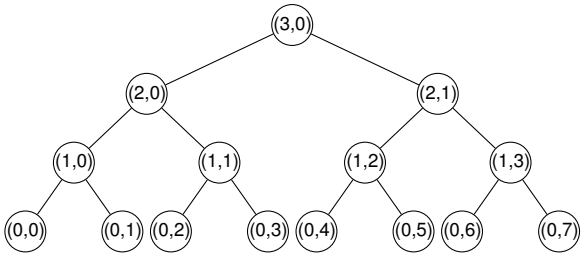

Figure 2. 3 Perfect Binary Tree

Definition 4 (Coverage): Given an ℓ_n PBT, we say a tree node (h_1, o_1) covers a leaf node $(0, o_2)$ if there exist

a path from (h_1, o_1) to $(0, o_2)$ in the tree (e.g., if $o_1 \cdot 2^{h_1} \leq o_2 < (o_1+1) \cdot 2^{h_1}$). The *covering set* of a given leaf node v is the set of all nodes in the PBT that cover v (e.g., all nodes on the path from v to the root). The *coverage* of a tree node $v = (h_1, o_1)$ is the set of all leaf nodes covered by v. $v.leftLeaf$ [$v.rightLeaf$] returns the left [right] most leaf node in the coverage of v ($v.leftLeaf = (0, o_1 \cdot 2^{h_1})$, $v.rightLeaf = (0, (o_1+1) \cdot 2^{h_1} - 1)$).

Definition 5 (Representer set and minimal representer set): Given an ℓ_n PBT, a *representer (set)* of a set of leaf nodes $L' \subseteq L$ is a set of nodes $R \subseteq (L \cup NL)$ such that

- for all nodes $v \in L'$, there exists a node in R that covers v, and
- for all nodes $v \in R$, there is no leaf node $v' \notin L'$ that is covered by v.

A representer R for the set of leaf nodes L' is minimal, if there is no other representer R' of L' with $|R'| < |R|$.

The notion of PBT gives a bijection between a set of integers and a set of leaf nodes in the PBT, such as the range $[a_1, a_2]$ is injected to the set $\{(0, a_1), (0, a_1 + 1) \ldots, (0, a_2)\}$. In the following sections, we will use the notations $[a_1, a_2]$ and $\{(0, a_1), (0, a_1+1) \ldots, (0, a_2)\}$ interchangeably without ambiguity.

Lemma 1 ([12](Lemma 6)): Let R be a minimal representer of the set of leaf nodes L. For each node $n \in R$, no other node $n' \in R$ is a descendant of n.

Lemma 2: Let R be a minimal representer for $L' = \{(0,0), ..., (0,a)\}$ in an ℓ_n PKT. For each level $0 \leq i \leq \ell_n$, there can be at most $k-1$ nodes $v \in R$ such that $v.h = i$.

Proof: The proof is similar with the proof of Lemma 3 in [12]. ■

Lemma 3: Given an ℓ_n PKT, let R be a representer for the set of leaf nodes $S = \{(0, a_1), \ldots, (0, a_2)\}$ and let B be the covering set for the leaf node $(0, b)$. Then $a_1 \leq b \leq a_2$ if and only if $R \cap B \neq \emptyset$, if and only if $|R \cap B| = 1$ exactly.

Lemma 3 is the generalization of Lemma 5 in [12]. Lemma 3 changes a set membership problem into a set intersection problem.

V. AREA PROOF

In this section, we present a protocol to solve area proof, i.e., UE proves that $(\sigma, \tau) \in VA$ to HS but without any other knowledge of (σ, τ) to HS.

Definition 6 (Area Proof): Let $C = (Gen, Com, Open)$ be the generation, the commitment and the open algorithm of a string commitment scheme. For an instance c, a proof with respect to commitment scheme C and area A is a proof of knowledge for the following statement:

$$PK\{((\sigma, \tau), \rho) : c \leftarrow Com((\sigma, \tau); \rho) \wedge (\sigma, \tau) \in A\},$$

where $A = [(x_1, y_1), (x_2, y_2)]$.

In this section, we present an area proof protocol based on a k-ary tree index. The protocol in this section has the advantages of less complexity, extensibility and flexibility which is suitable for smart phone applications.

A. Perfect k-ary Index

Now we generalize the notion of perfect binary index in Section IV-D into perfect k-ary index. The generalization is direct. We only give the details of the notion of ℓ_n perfect k-ary tree (ℓ_n PKT). Other notions can be treated similarly.

Definition 7 (ℓ_n Perfect k-ary tree (ℓ_n PKT)): An *ℓ_n perfect k-ary tree*, $k \geq 2$, is a k-ary tree with a set of leaf nodes $L = \{(0,0), (0,1), \ldots, (0, n-1)\}$ and a set of non-leaf nodes NL such that $(\ell_n, 0) \in NL$ and for all $(h, o) \in NL$, there exists $(h-1, ko), (h-1, ko+1), \ldots, (h-1, ko+k-1) \in (L \cup NL)$ with $(h, o).c_1 = (h-1, ko)$,..., $(h, o).c_k = (h-1, ko+k-1)$. In an ℓ_n PKT, the root node is the node $(\ell_n, 0)$.

Figure 3 shows 2 Perfect 3-ary Tree(P3T).

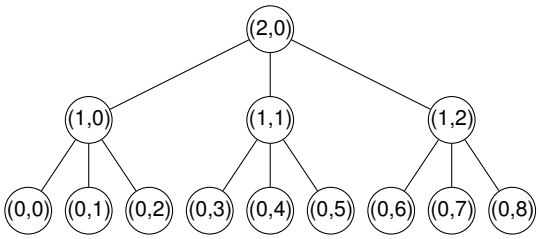

Figure 3. 2 Perfect 3-ary Tree

Lemma 1 and Lemma 3 also hold for k-ary tree which can be proved similarly as [12].

We define a bijection function $g(h, o) = \ell_n \cdot o + h$ where $0 \leq h < \ell_n$ and $0 \leq o < 2^{\ell_n}$, which maps a node (h, o) into an integer $g(h, o)$. In Figure 4 and Figure 5, we would substitute the integer $g(h, o)$ for the node (h, o) where needed.

B. Range Proof

Before giving area proof, we first present a range proof which is based on Lemma 3. The basic idea is that we can employ the set membership protocol in [11] to solve the set intersection problem. Let R be a representative set for the set of leaf nodes $S = \{(0, a_1), \ldots, (0, a_2)\}$ and let B be the covering set for the leaf node $(0, b)$.

The range proof is presented in Figure 4 where we assume that $b \in [a_1, a_2]$ and so that $|R \cap B| = 1$. For the case of $b \notin [a_1, a_2]$, Prover will find that $R \cap B = \emptyset$. Therefore, Prover would not complete the protocol in this case.

The range proof protocol in Figure 4 can easily be extended to the case of the union of ranges. The extension needs the following corollary.

Corollary 1: Let $[a_i, b_i]$ for $1 \leq i \leq m$ be m intervals in an ℓ_n PKT. Let R_i be the minimal representer set of $[a_i, b_i]$ for $1 \leq i \leq m$. Let R be the minimal representer set of $\cup_{i=1}^{m} L_i$. If $[a_i, b_i]$ are pairwise non-adjacent, i.e. $b_i + 1 < a_{i+1}$ for $1 \leq i \leq m-1$, then $R = \cup_{i=1}^{m} R_i$.

Proof: Since $[a_i, b_i]$ for $1 \leq i \leq m$ are pairwise non-adjacent, We get that $R_i \cup R_j = \emptyset$ for different i, j from the definition of minimal representer. Then we can easily get that $R = \cup_{i=1}^{m} R_i$. ■

Common Input:	g, h and commitments C_i for $1 \leq i \leq \|B\|$.
Prover Input:	i, r_i such that $C_i = g^i h^{r_i}\ \forall i \in B$. (By our assumption, there exists one and only one element in $\in R \cap B$.) We denote the only node in $R \cap B$ as σ and set $C = g^\sigma h^r$.
$P \xleftarrow{y,\{A_i\}} V$	Verifier picks $x \in_R \mathbb{Z}_p$ and sends $y \leftarrow g^x$ and $A_i = g^{\frac{1}{x+i}}, \forall i \in R$.
$P \xrightarrow{V} V$	Prover picks $v \in_R \mathbb{Z}_p$ and sends $V \leftarrow A_\sigma^v = g^{\frac{v}{x+\sigma}}$.
Prover and Verifier run $PK\{(\sigma, r, v) : C = h^r g^\sigma \wedge V = g^{\frac{v}{x+\sigma}}\}$.	
$P \xrightarrow{a,D} V$	Prover picks $s, t, m \in_R \mathbb{Z}_p$ and sends $a \leftarrow e(V,g)^{-s} e(g,g)^t$ and $D \leftarrow h^m g^s$.
$P \xleftarrow{c} V$	Verifier sends a random challenge $c \in_R \mathbb{Z}_p$.
$P \xrightarrow{z_\sigma, z_v, z_r} V$	Prover sends $z_\sigma = s - \sigma c$, $z_v = t - vc$, $z_r = m - rc$.
	Verifier checks that $D \stackrel{?}{=} C^c h^{z_r} g^{z_\sigma}$ and that $a \stackrel{?}{=} e(V,y)^c e(V,g)^{-z_\sigma} e(g,g)^{z_v}$.

Figure 4. Range proof protocol for range $[a_1, a_2]$

C. Area Proof

Now we can transform the range protocol in Figure 4 into area proof protocol. Let $A = [(x_1, y_1), (x_2, y_2)]$ be a rectangular area. Let R, R' denote the minimal representer of $[x_1, x_2]$ and $[y_1, y_2]$ respectively. Let B, B' denote the covering set of $(0, x), (0, y)$ respectively. Then we can get the following corollary from Lemma 3.

Corollary 2: $(x, y) \in A$ if and only if $(B \times B') \cap (R \times R') \neq \emptyset$, if and only if $|(B \times B') \cap (R \times R')| = 1$ exactly.

As in Section V-B, we assume that $(x, y) \in A$ and so that $|(B \times B') \cap (R \times R')| = 1$ in Figure 5 and we would substitute the integer $g(h, o)$ for the node (h, o) where needed.

The basic idea of our area proof is that, as in [11], the verifier first sends the prover a signature of every element in the set $R \times R' = \{(\sigma, \tau)\}$, which is a set of *two-tuples*. Thus, the prover receives a signature on the particular element (σ, τ) to which C is a commitment. The prover then"blind" this received signature. They perform a proof of knowledge that she possesses a signature on the committed (two tuples) element.

In Figure 5 the signature of (σ, τ) is in fact the signature of $\sigma + b\tau$ in Boneh-Boyen Signature scheme, where b is a security parameter and must be larger than $\max\{\sigma\}$. The value of b needn't to be known to the prover and at best to be secret to the prover.

The area proof is presented as in Figure 5.

D. Extension to Multi-Area Proof

Similarly as in Section V-B, we can extend the area proof protocol in Figure 5 to multiple areas case. For the multi-area proof protocol, we only need slightly revision to the area proof protocol in Figure 5.

Corollary 3: Let $A_i = [(x_{1i}, y_{1i}), (x_{2i}, y_{2i})]$ for $1 \leq i \leq n$ be n pairwise non-adjacent rectangular areas, i.e. $[x_{1i}, x_{2i}]$ for $1 \leq i \leq n$ are pairwise non-adjacent as well as $[y_{1i}, y_{2i}]$ for $1 \leq i \leq n$ are pairwise non-adjacent. Let R_i, R_i' for $1 \leq i \leq n$ denote the minimal representer of $[x_{1i}, x_{2i}]$ and $[y_{1i}, y_{2i}]$ respectively. Let R be the minimal representer set of $\cup_{i=1}^n A_i$. Then $R = \cup_{i=1}^n (R_i \times R_i')$.

Proof: This is a direct result of Corollary 1. ■

Let A_i, R_i, R_i', R be as in Corollary 3. Let B, B' denote the covering set of $(0, x), (0, y)$ respectively. Then we have the following result by Corollary 3 and Lemma 3.

Corollary 4: $(x, y) \in \cup_i A_i$ if and only if $(B \times B') \cap R \neq \emptyset$, if and only if $|(B \times B') \cap R| = 1$ exactly.

The revision for area proof in Figure 5 is then apparent. The multi-area proof protocol is very useful in some applications, such as location-based access control as interpreted in Section VII-B.

VI. Security analysis

In Figure 4, there have $|B|$ commitments which are the commitments of all nodes in B, the covering set of the leaf node $(0, b)$. Only when the $|B|$ commitments is the commitments of each element in B, that the protocol in Figure 4 is correct and secure. Otherwise, if the $|B|$ commitments is not the commitments of all nodes in B but an another set E, the protocol only get the conclusion that $E \cap S \stackrel{?}{=} \emptyset$ which has no direct relation with the problem of $b \stackrel{?}{\in} [a_1, a_2]$.

However, in our application scenarios which will be illustrated in Section VIII in details, the assumption is practical.

Theorem 1: Assume that $\{C_{ij}\}$ are $|B| \cdot |B'|$ commitments of all nodes in $B \times B'$ and that B, B', R, R' are evaluated correctly. If the q-Strong Diffie-Hellman assumption associated with a pairing generator PG(1^k) holds, then the protocol in Figure 5 is a zero-knowledge area argument for area $[x_1, x_2] \times [y_1, y_2]$.

Proof: This is a direct result of Theorem 1 in [11]. ■

In Figure 5 data privacy of Prover is protected completely since Verifier only gets the commitments of locations. When protocol is done, Verifier only gets that whether $(x, y) \in A$ or not, no other information will be obtained. Moreover, A is public and changeable. If Prover considers it is too small, then she can reject to complete the protocol.

VII. Complexity analysis

A. Communication and Computation Complexity

For simplicity, we only compare the complexity of the protocol in Figure 4 with the range proof protocol

Common Input:	g, f, h and commitments C_{ij} where $1 \leq i \leq \|B\|$ and $1 \leq j \leq \|B'\|$, where $f = g^b$ and b is secret to Prover.
Prover Input:	$(i,j), r_{ij}$ such that $C_{ij} = g^i f^j h^{r_{ij}}$, $\forall (i,j) \in B \times B'$. (By our assumption, there exists one and only one element in $(R \cap B) \times (R' \cap B')$.) We denote the only node in $(R \cap B) \times (R' \cap B')$ as (σ, τ) and set $C = g^\sigma f^\tau h^r$.
$P \xleftarrow{y,\{A_i\}} V$	Verifier picks $x \in_R \mathbb{Z}_p$ and sends $y \leftarrow g^x$ and $A_{ij} = g^{\frac{1}{x+i+bj}}, \forall (i,j) \in (R, R')$.
$P \xrightarrow{V} V$	Prover picks $v \in_R \mathbb{Z}_p$ and sends $V \leftarrow A_{\sigma,\tau}^v = g^{\frac{1}{x+\sigma+b\tau}}$.
Prover and Verifier run $PK\{(\sigma, r, v) : C = h^r g^\sigma f^\tau \wedge V = g^{\frac{v}{x+\sigma+b\tau}}\}$.	
$P \xrightarrow{a,D} V$	Prover picks $s, t, m, n \in_R \mathbb{Z}_p$ and sends $a \leftarrow e(V,g)^{-s} e(V,f)^{-n} e(g,g)^t$ and $D \leftarrow h^m g^s f^n$.
$P \xleftarrow{c} V$	Verifier sends a random challenge $c \in_R \mathbb{Z}_p$.
$P \xrightarrow{z_\sigma, z_\tau, z_v, z_r} V$	Prover sends $z_\sigma = s - \sigma c$, $z_\tau = n - \tau c$, $z_v = t - vc$, $z_r = m - rc$.
	Verifier checks that $D \stackrel{?}{=} C^c h^{z_r} g^{z_\sigma} f^{z_\tau}$ and that $a \stackrel{?}{=} e(V,y)^c e(V,g)^{-z_\sigma} e(V,f)^{-z_\tau} e(g,g)^{z_v}$.

Figure 5. Area proof protocol for area $[x_1, x_2] \times [y_1, y_2]$

in [11] in the case of $k = 2$ (i.e. $u = 2$ in [11]) for range $[a, b]$ with $2^{\ell-1} < b < 2^\ell$ in the subsection. For the general case, i.e. the parameter k is arbitrary, since our protocol is for arbitrary intervals but not a specified interval, it would be meaningless to compare their complexity. Therefore, in the latter case we give their difference and corresponding properties instead of complexity comparison. The complexity of area proof protocol in Figure 5 is similar with the one in Figure 4. The comparison of complexity is shown in Figure 6. Figure 6 is evaluated by the following notes.

- Only costly operations, i.e., modular exponentiations and parings, are counted in computation complexity. The relatively costless operations, such as the modular multiplication, are omitted.
- Complexity is evaluated in the multiplier of parameter ℓ. Constant numbers of operations, such as modular exponentiations, are omitted.
- The (average) computation cost of a modular exponentiation in group $\mathbb{G}$ is denoted $c_\mathbb{G}$, such as $c_{\mathbb{G}_1}, c_{\mathbb{G}_T}$ and $c_{\mathbb{Z}_p}$. The computation cost of a paring $e(g_1, g_2)$ is denoted c_e.
- Recall that the computation of commitments are ascribed to Prover.
- Recall that $|R| \leq 2\ell$ by Corollary 5.

From Figure 6 we can see that the protocol in Figure 5 has much smaller communication and computational complexity than the range proof protocol in [11]. Especially, the former is very efficient in the knowledge proof steps where the computation cost is in the constant numbers of modular exponentiation and paring (no relation with the parameter ℓ), e.g. the integer a compared with the integer sequence $\{a_{ij}\}$, and is omitted in the evaluation of overall computation cost.

Moreover, in Figure 4 Prover's computation cost is mainly the evaluation of commitments ($\ell c_{\mathbb{G}_1}$) which is much smaller than the one in [11] and which can be further reduced by raising k. This property is especially useful when Prover is resource-limited, such as a smart phone user in our application scenarios.

The protocol in Figure 5 needs to encode area data $[x_1, x_2] \times [y_1, y_2]$ and location data (x, y) using k-ary tree which seems to add protocol's computational complexity. However, the proofs of Lemma 6 and Lemma 7 give us ideas to design algorithms to efficiently evaluate the minimal representer sets. Because of the page limit, we don't give these algorithms in the paper.

B. Further Properties of Minimal Representer and Parameter Optimization

In this section we discuss some further properties of k-ary tree index and give some optimization to choose the parameter k.

Lemma 4: Given an ℓ_n PKT and a non-empty set of leaf nodes $L' \subseteq L$, there is one and only one minimal representer for L'.

Proof: It can be verified that L' itself is a representer of L' and so the existence of minimal representer is obvious. The following part proves the uniqueness. Assume that R, R' are two distinct minimal representer of L'. Then there must exist a node $r \in R$ such that $r \notin R'$. Let $T \subseteq L'$ be the coverage of node r and so r is a minimal representer of T. Assuming that $e \in T$, then there must exist a node $r' \in R'$ that covers e. Therefore, r, r' are both in the covering set of the leaf node e and so one is an ancestor of the other. Without loss of generality, we suppose that r is an ancestor of r'. Then there is at least one element $e' \in T$ not covered by r' and so there exists $s' \in R'$ such that s' covers e'. Therefore, r, s' are both in the covering set of e' and then one must be an ancestor of the other. However, the node s' should not be an ancestor of r, or else s' is an ancestor of r' which is contrary to Lemma 1. Therefore, the node r is an ancestor of s' and so r is an ancestor of both s' and r' where $s', r' \in R'$. Then we can substitute r for s', r' in R' and so get a new representer R'' of L' with $|R''| < |R'|$, which is contrary to the fact that R' is a minimal representer of L'. Hence, the assumption is wrong and so there is only one minimal representer. ■

Lemma 5: Let $R = \{r_1, \ldots, r_t\}$ be the minimal representer of a set of leaf nodes in an ℓ_n PKT. Let T_i be

	Computation of Prover	Computation of Verifier	Total computation	Communication
Ours	$\ell c_{\mathbb{G}_1}$	$2\ell c_{\mathbb{G}_1}$	$3\ell c_{\mathbb{G}_1}$	$3\ell\|\mathbb{G}_1\|$
[11]	$\ell(4c_{\mathbb{G}_1}+4c_{\mathbb{G}_T}+2c_e)$	$\ell(2c_{\mathbb{G}_1}+6c_{\mathbb{G}_T}+2c_e)$	$\ell(6c_{\mathbb{G}_1}+10c_{\mathbb{G}_T}+4c_e)$	$\ell(3\|\mathbb{G}_1\|+2\|\mathbb{G}_T\|+4\|\mathbb{Z}_p\|)$

Figure 6. Comparison in computation cost and communication cost

the coverage of node r_i for $1 \le i \le t$. Then $T_i \cap T_j = \emptyset$ for $1 \le i, j \le t$ with $i \ne j$.

Proof: It is easy to verify that r_i is the minimal representer of T_i. Without loss of generality, suppose that there exists a leaf node e such that $e \in T_1 \cap T_2$. Then r, r' both cover e and so one is an ancestor of the other, which is a contradiction to Lemma 1. ∎

In a PBT there exists a bijection between the elements in the minimal representer of the range $[0, b]$ and the digits 1 in the binary notation of $b+1$. Setting $b_t b_{t-1} \cdots b_0$ bing the binary notation of $b+1$, if $b_i = 1$ the minimal representer of $[0, b]$ has one and only one element in the level i of the PBT, or else there is none. In detail, $b_i = 1$ maps to which element in level i is defined by the digits of $b+1$ important than i. Similarly, in a PKT, there is a map between the elements in the minimal representer of $[0, b]$ and the no-zero digits of the k-ary notation of $b+1$. In detail, there are exactly b_i elements of the minimal representer of $[0, b]$ in level i. The same as in PBT, b_i maps to which b_i elements in S is related to no-zero digits of $b+1$ with level lager than i. The detail is presented in Lemma 6.

Lemma 6: In an ℓ_n PKT, the size of the minimal representer set of $L = \{(0,0),(0,1),\ldots,(0,b)\}$ is $\sum_{i=0}^{n} b_i$, where b_i is the ith digit in the k-ary notation of $b+1$ with b_0 the least significant digit, i.e., $b+1 = \sum_{i=0}^{n} b_i \cdot k^i$.

Proof: Let $R = \{r_1, \ldots, r_t\}$ be the minimal representer of L and T_i be the coverage of node r_i for $1 \le i \le t$. Then for $1 \le i \le t$, r_i is the minimal representer of T_i. According to Lemma 5, $T_i \cap T_j = \emptyset$ for $1 \le i, j \le t$ with $i \ne j$. Then $b+1 = |L| = \sum_{i=1}^{t} |T_i|$. Since each $r_i \in R$ covers nodes of which the number is of the form k^{τ_i} for a definite non-negative integer τ_i, then $b+1 = \sum_{i=1}^{t} |T_i| = \sum_{i=1}^{t} k^{\tau_i} = \sum_{\tau_i, 1 \le i \le t} n_i k^{\tau_i}$, where n_i is the times of k^{τ_i} appeared in $\sum_{i=1}^{t} k^{\tau_i}$. By Lemma 2 there are at most $k-1$ nodes $r \in R$ such that $r.h = i$. Since each node $r \in R$ covers $k^{r.h}$ nodes in L, which says that each node in R at the same level covers the same number of leaf nodes, then $0 \le n_i \le k-1$. Then, $\sum_{\tau_i, 1 \le i \le t} n_i k^{\tau_i}$ is the k-ary expansion of $b+1$. Therefore, $|R| = \sum_{i=1}^{n} b_i = \sum_{\tau_i, 1 \le i \le t} n_i$. The proof is complete. ∎

Similarly, there exists a map between the elements in the minimal representer of $[a, b]$ and the no-zero digits of the k-ary notation of $a, b+1$. Let R be the minimal representer of $[a, b]$. Let $a_t a_{t-1} \cdots a_0$ and $b_t b_{t-1} \cdots b_0$ be the k-ary representation of a and $b+1$ respectively and $b_t \ne 0$. If $a_i < b_i$, there must have at least one element in R in level i or none otherwise. Especially, in a PBT the size of S equals to the sum of the digits of $b+1$ and $-a$ minus 1. The detail is presented in Lemma 7.

Lemma 7: Let a, b be two non-negative integers with $a \le b$. Let $a_t a_{t-1} \cdots a_0$ and $b_t b_{t-1} \cdots b_0$ be the k-ary representations of a and $b+1$ respectively and $b_t \ne 0$. Then

1) if $a_t < b_t$, $|R_{[a,b]}| = \sum_{i=0}^{t-1} b_i + ((a_t+1)k^t - a)_k + (b_t - a_t - 1) = (b+1)_k + ((a_t+1)k^t - a)_k - a_t - 1$
2) otherwise (i.e., $a_t = b_t$), $|R_{[a,b]}| = |R_{[a-a_t \cdot k^t, b-b_t \cdot k^t]}|$

where $R_{[x,y]}$ denotes the minimal representer of $\{(0,x),(0,x+1),\ldots,(0,y)\}$ in the ℓ_n PKT and $(x)_k$ denotes the sum of every digits in the k-ary notation of non-negative integer x.

Proof: (1) If $a_t < b_t$, there exist two cases. (i) If $b_t = a_t + 1$, then $L = \{(0,a),(0,a+1),\ldots,(0,(a_t+1)k^t - 1)\} \cup \{(0,b_t k^t),\ldots,(0,b)\}$ while $b > b_t k^t - 1$, or else $L = \{(0,a),(0,a+1),\ldots,(0,b_t k^t - 1)\}$ while $b = b_t k^t - 1$. While $b > b_t k^t - 1$, $R_{[a,b]} = R_{[a,b_t k^t - 1]} \cup R_{[b_t k^t, b]}$. By the symmetric property of PKT, $|R_{[a,b_t k^t-1]}| = |R_{[0,b_t k^t - a - 1]}|$ and $|R_{[b_t k^t, b]}| = |R_{[0, b-b_t k^t]}|$. Then $|R_{[a,b]}| = |R_{[a,b_t k^t-1]}| + |R_{[b_t k^t, b]}| = ((a_t+1)k^t - a)_1 + (b - b_t k^t + 1)_1 = \sum_{i=0}^{t-1} b_i + ((a_t+1)k^t - a)_1 + (b_t - a_t - 1) = (b+1)_1 + ((a_t+1)k^t - a)_1 - a_t - 1$. While $b = b_t k^t - 1$, $|R_{[a,b]}| = 1 = \sum_{i=0}^{t-1} b_i + ((a_t+1)k^t - a)_1 + (b_t - a_t - 1)$.

(ii) If $b_t > a_t + 1$, $L = \cup_{i=1}^{b_t - a_t - 1}\{(0,(a_t+i)k^t),\ldots,(0,(a_t+i+1)k^t - 1)\} \cup \{(0,a),(0,a+1),\ldots,(0,(a_t+1)k^t - 1)\} \cup \{(0,b_t k^t),\ldots,(0,b)\}$ while $b > b_t k^t - 1$, or else $L = \cup_{i=1}^{b_t - a_t - 1}\{(0,(a_t+i)k^t),\ldots,(0,(a_t+i+1)k^t - 1)\} \cup \{(0,a),(0,a+1),\ldots,(0,(a_t+1)k^t - 1)\}$ while $b = b_t k^t - 1$. While $b > b_t k^t - 1$, $R_{[a,b]} = \cup_{i=1}^{b_t - a_t - 1} R_{[(a_t+i)k^t, (a_t+i+1)k^t - 1]} \cup R_{[a,(a_t+1)k^t - 1]} \cup R_{[b_t k^t, b]}$. Then $|R_{[a,b]}| = \sum_{i=0}^{b_t - a_t - 1} |R_{[(a_t+i)k^t, (a_t+i+1)k^t - 1]}| + |R_{[a,(a_t+1)k^t - 1]}| + |R_{[b_t k^t, b]}| = (b_t - a_t - 1) + ((a_t+1)k^t - a)_1 + (b - b_t k^t + 1)_1 = \sum_{i=0}^{t-1} b_i + ((a_t+1)k^t - a)_1 + (b_t - a_t - 1) = (b+1)_1 + ((a_t+1)k^t - a)_1 - a_t - 1$. While $b = b_t k^t - 1$, the conclusion can be proved similarly as in the case (i).

(2) If $a_t = b_t$, which says that $(0,a)$ and $(0,b)$ are both covered by the node $(t, a_t - 1)$ in level t, then $(t, a_t - 1)$ is an ancestor of all nodes in $R_{[a,b]}$ and $(t, a_t - 1) \notin R_{[a,b]}$ (If $(t, a_t - 1) \in R_{[a,b]}$, then $(t, a_t - 1)$ is the minimal representer of $[a, b]$. We have $a = a_t k^t$ and $b = a_t k^t + k^t - 1$ and so $b + 1 = a_t k^t + k^t$, which says that $b_t = a_t + 1$, a contradiction to $a_t = b_t$). Therefore, $R_{[a,b]} = R_{[a - a_t \cdot k^t, b - b_t \cdot k^t]}$ and so $|R_{[a,b]}| = |R_{[a-a_t \cdot k^t, b-b_t \cdot k^t]}|$.

In all, the proof is complete. ∎

We then have Theorem 2.

Theorem 2: Let a, b be two non-negative integers with $a \le b$. Let $a_t a_{t-1} \cdots a_0$ and $b_t b_{t-1} \cdots b_0$ be the k-ary notations of a and $b+1$ respectively and $b_t \ne 0$. Let s be the first integer i such that $a_i < b_i$ for i from t to 0. Then, $|R_{[a,b]}| = \sum_{i=0}^{s} b_i + ((a_s+1) \cdot k^s - \sum_{i=0}^{s} a_i \cdot k^i)_k - a_s - 1$.

From Theorem 2 we have Corollary 5.

Corollary 5: $|R_{[a,b]}| \leq 2(k-1)\log_k b$.

According to Theorem 2, $|R_{[a,b]}| = \sum_{i=0}^{s} b^i + ((a_s+1)\cdot k^s - \sum_{i=0}^{s} a_i \cdot k^i)_k - a_s - 1$ which displays that the minimal representer of $[a,b]$ in an PKT is determined entirely by the digits of the k-ary notation of a, b. Since different k will result in the different $R_{[a,b]}$ and so the different $|R_{[a,b]}|$, we should choose suitable k to minimize $|R_{[a,b]}|$ and then to minimize the number of signature in Figure 4 and in Figure 5.

However, minimization of $|R_{[a,b]}|$ will probably result in a relatively large $|B|$, where B is the covering set of the node $(0,\sigma)$, which is another important parameter for the complexity of the protocols in Figure 4 and in Figure 5. Therefore, we should leverage the communication and computational complexity among UE, HS and the whole of protocol by adjusting the value of k.

Since our protocols' application scenarios are mobile device applications, there are three different complexity aspects - the overall communication complexity, the overall computational complexity and Prover's computational complexity - among which we should balance when we choose the parameter k.

For a specified interval $[a,b]$, Prover's computational complexity is determined by the number of commitments ($\log_k b$). The overall computational and communication complexity are determined by the number of commitments ($\log_k b$) and the number of signatures ($|R_{[a,b]}| \leq (k-1)\log_k b$). By raising k, we can lower the number of commitments and so Prover's computational complexity. In the extreme case, such as $k = b/c$ for a constant c, Prover's computational complexity becomes constant, i.e. $1 + \log_k c$. On the other hand, raising k will increase the number of signatures which increases the overall computational and communication complexity. This is a dilemma. Since our protocols are not aimed at a fixed interval or area but a plenty of intervals or areas, then we can optimize the complexity by using the statistical data of intervals or areas. Let $p(x,y)$ be a probability distribution function of interval $[x,y]$ where $A \leq x \leq y \leq B$ with $A \geq 0$. Then the following formula (1) would give a relatively optimized parameter k by the perspective of statistical data of intervals.

$$\arg\min_k E_p|R_{[x,y]}| = \arg\min_k \sum_{x,y} p(x,y)|R_{[x,y]}| \quad (1)$$

In formula (1), $|R_{[x,y]}|$ can be calculated by the result of Theorem 2. Solving the above optimization problem would be costly and so sampling techniques would be suitable.

The above choosing of parameter k is practical. For example, the frequent areas where people emerge are much smaller than the overall area of a city. Therefore, it is reasonable to give smaller minimal representer set for these frequent areas which is the principle of coding theory [35].

VIII. Privacy-preserving location assurance protocols

Now we discuss how to use our area proof protocols to design privacy-preserving location assurance schemes. As in Section III, there are three parties: UE, HS and OBU. We assume that each UE equips a unique On-Board Unit(OBU) embedded in UE's mobile device. The OBU is mainly in charge of collecting UE's GPS data from GPS satellites.

The protocol proceeds as follows.

- OBU calculates its current GPS data (σ,τ) (like a GPS receiver).
- OBU sends $(ID, t, (\sigma,\tau), C, r, Sig(ID,t,C))$ to UE (and sends Sig(ID, t, C) to HS where needed), where
  - $C = g^\sigma f^\tau h^r$, r is a randomly chosen integer.
  - $Sig()$ is the signature function of OBU.
- UE needs to assure HS that C is the commitment of the authentic location (σ,τ) of UE at time t OBU issued.
  - UE sends $(ID, t, C, Sig(ID,t,C))$ to HS who use the public key of OBU to verify the authenticity of data by verifying the validity of OBU's signature.
- UE proves to HS that (σ,τ) with its commitment C has the properties UE claimed, e.g., within some place, without disclosing any other information of (σ,τ).
  - the proof is achieved by area proof protocols.

The above scheme can both protect location privacy of UE and give a location assurance about UE to HS. The signature of OBU assures the authenticity and validity of the commitment C which can prevent UE to spoof location (x,y) to which C is committed. By employing area proof protocols, the location assurance of (x,y) can be achieved with location privacy protection.

Location data granularity is an important factor to implement our schemes. High granularity of certified location data can induce high communication and computational complexity. We should choose suitable schemes for different application with different location granularity.

History data storage location data and the certification $(ID, t, (\sigma,\tau), C, r, Sig(ID,t,C))$ needs to be stored by UE for possible future verification or assurance.

A. Temporal Obfuscation

Temporal obfuscation [34] with assurance property is also supported by our protocols. In temporal obfuscation with assurance property protocols, the commitment of location data becomes $C = f^t h^r$. The location data of UE becomes $(ID, T, t, (\sigma,\tau))$, where T stands for the time of staring point of a period(such as one day or one week etc.), t stands for the time when an event happens at that period(such as 12 o'clock at one day etc.). The time assurance proof is achieved by range proof protocols.

If both temporal obfuscation and location obfuscation are needed, the commitment of location data becomes

$C = f^{\sigma} g^{\tau} d^{t} h^{r}$, where $d = g^{c}$ for a definite integer c. In the case, three-dimensional spatial-temporal proof protocol is needed.

B. Applications

Now we discuss how to exploit the scheme presented in Section VIII to support tracking services, location-based access control.

1) Tracking Services: A tracking service allows HS (or a customer) to track UE. When UE has left a boundary area or has been adjacent to the margin of the area, HS is warned. Tracking services applications may be of that: (1)the court may demand that a person should not go out of a city or a district; (2) the parents need to be confirmed that their children be in safe areas which is adjacent to their home; (3) some staffs in security offices will be confined in some areas in case of secret leaking.

In these applications, UE has great motivation to change location data to spoof HS. On the other hand, the privacy of UE is a vital factor in order to obtain UE's cooperation. Furthermore, HS must get the confirmation information about UE's location data frequently which requires the high efficiency of scheme.

Our privacy-preserving location assurance schemes in Section VIII is suitable to cope with problems in these applications. The location privacy and location assurance can be solved by our schemes.

2) Location-based Access Control: In location-based access control [6]–[10], UE can possibly get access permission only when UE has been within some valid area(VA). In this application scenario, as analyzed in Section VIII-B.1, spoofing motivation and location privacy of UE are vital factors. However, in location-based access control, the authentication is not as frequent as tracking services.

Especially, our schemes also support the case where Valid area is not a contiguous area but a union of multiple valid areas(VAs). In this case, the scheme based on multi-area proof protocol in Section V-D would function.

IX. Implementations

In this section, we give some hints on implementation to the prototype of our system.

The OBU is the most important unit of our system. At high-level, the elements of our OBU prototype are: a GPS receiver and a trusted platform module (TPM). The GPS receiver is in charge of receiving the GPS signal and of evaluating the location data. The TPM is in charge of assuring the honesty of the GPS receiver and of evaluating the commitment of the location data.

We are currently implementing the range proof protocol in Figure 4 and the area proof protocol in Figure 5 in C++ with NTL[1]. In the future we will add these protocols to the system architecture.

[1] http://www.shoup.net/ntl/

X. Conclusion

In this paper, we present some privacy-preserving location assurance schemes. These schemes achieve both location assurance and location privacy protection. In detail, by borrowing Pedersen commitment scheme and area proof protocols, we can not only achieve location authenticity and validity with privacy-preserving way but also achieve the verification of the committed location being within a public area with privacy-preserving way.

We present a new range proof protocol and a new area proof protocol which are based on a new data structure, i.e. Perfect k-ary Tree (PKT). Some deeper properties of PKT are presented which are used to analyze our protocols' complexity. The analysis results show that our protocols are more efficient that the former and is flexible enough to support some existing mobile device applications, such as tracking services, location-based access control.

In most cases, it is hard to prove the properties of a committed value in privacy-preserving way, such as proving the committed value being in a (public or secret) set or evaluating the distance of two committed secret points in privacy-preserving way etc. Location data, a two tuple (σ, τ) in its simple case, has its own physical meaning which we care about, e.g. private proximity testing in mobile social networks [1], [26]–[31] and secure interval check [12]. The private proximity testing and secure interval check problem are intuitively similar with the area proof problem. However, it's actually not the case. This is due to the fact that our method can only assure a committed value being within a publicly known area, whereas the formers are not the case.

Moreover, our schemes try to minimize the trust assumptions to which are rooted the trusted computation of a small OBU. Our analysis in Section III shows that it is reasonable to use the trusted computing techniques to achieve the OBU. How to achieve that OBU with the least trusted computing techniques needs to be solved.

Our one future work would extend our method to assuring more properties of location data and apply it to other applications such as in mobile participatory sensor and in mobile social networks and so on. The other future work would be to further minimize the trust assumptions of the OBU and to do implementation.

Acknowledgment

The authors are grateful to the anonymous referees for their valuable comments and suggestions to improve the presentation of this paper.

References

[1] G. Zhong, I. Goldberg, and U. Hengartner, "Louis, lester and pierre: Three protocols for location privacy," in *Privacy Enhancing Technologies*, 2007, pp. 62–76.

[2] G. M. Køien and V. A. Oleshchuk, "Location privacy for cellular systems; analysis and solution," in *Privacy Enhancing Technologies*, 2005, pp. 40–58.

[3] S. Saroiu and A. Wolman, "Enabling new mobile applications with location proofs," in *HotMobile*, 2009.

[4] R. Hasan and R. C. Burns, "Where have you been? secure location provenance for mobile devices," *CoRR*, vol. abs/1107.1821, 2011.
[5] U. Hengartner, "Hiding location information from location-based services," in *MDM*, 2007, pp. 268–272.
[6] I. Ray and M. Kumar, "Towards a location-based mandatory access control model," *Computers & Security*, vol. 25, no. 1, pp. 36–44, 2006.
[7] I. Ray and M. Toahchoodee, "A spatio-temporal role-based access control model," in *DBSec*, 2007, pp. 211–226.
[8] M. Toahchoodee and I. Ray, "On the formalization and analysis of a spatio-temporal role-based access control model," *Journal of Computer Security*, vol. 19, no. 3, pp. 399–452, 2011.
[9] C. A. Ardagna, M. Cremonini, S. D. C. di Vimercati, and P. Samarati, "Privacy-enhanced location-based access control," in *Handbook of Database Security*, 2008, pp. 531–552.
[10] ——, "Access control in location-based services," in *Privacy in Location-Based Applications*, 2009, pp. 106–126.
[11] J. Camenisch, R. Chaabouni, and A. Shelat, "Efficient protocols for set membership and range proofs," in *ASIACRYPT*, 2008, pp. 234–252.
[12] A. E. Nergiz, M. E. Nergiz, T. Pedersen, and C. Clifton, "Practical and secure integer comparison and interval check," in *SocialCom/PASSAT*, 2010, pp. 791–799.
[13] K. Peng, "A secure and efficient proof of integer in an interval range," in *IMA Int. Conf.*, 2011, pp. 97–111.
[14] K. Peng and F. Bao, "Batch range proof for practical small ranges," in *AFRICACRYPT*, 2010, pp. 114–130.
[15] J. Camenisch and T. Groß, "Efficient attributes for anonymous credentials," in *ACM Conference on Computer and Communications Security*, 2008, pp. 345–356.
[16] ——, "Efficient attributes for anonymous credentials," *ACM Trans. Inf. Syst. Secur.*, vol. 15, no. 1, p. 4, 2012.
[17] R. Henry and I. Goldberg, "Formalizing anonymous blacklisting systems," in *Proceedings of the 2011 IEEE Symposium on Security and Privacy*, May 2011.
[18] S. Saroiu and A. Wolman, "I am a sensor, and i approve this message," in *HotMobile*, 2010, pp. 37–42.
[19] X. Chen, C. Harpes, G. Lenzini, M. Martins, S. Mauw, and J. Pang, "Implementation and validation of a localisation assurance service provider," in *NAVITEC*, 2012, pp. 1–8.
[20] A. C.-C. Yao, "How to generate and exchange secrets (extended abstract)," in *FOCS*, 1986, pp. 162–167.
[21] ——, "Protocols for secure computations (extended abstract)," in *FOCS*, 1982, pp. 160–164.
[22] H.-Y. Lin and W.-G. Tzeng, "An efficient solution to the millionaires' problem based on homomorphic encryption," in *ACNS*, 2005, pp. 456–466.
[23] I. Damgård, M. Geisler, and M. Krøigaard, "Efficient and secure comparison for on-line auctions," in *ACISP*, 2007, pp. 416–430.
[24] J. A. Garay, B. Schoenmakers, and J. Villegas, "Practical and secure solutions for integer comparison," in *Public Key Cryptography*, 2007, pp. 330–342.
[25] F. Zhang, A. Kondoro, and S. Muftic, "Location-based authentication and authorization using smart phones," in *TrustCom*, 2012, pp. 1285–1292.
[26] S. Chatterjee, K. Karabina, and A. Menezes, "A new protocol for the nearby friend problem," in *IMA Int. Conf.*, 2009, pp. 236–251.
[27] B. Zan, T. Sun, M. Gruteser, F. Hu, and Y. Zhang, "A privacy preserving system for friend locator applications," in *MOBIWAC*, 2011, pp. 93–100.
[28] S. Mascetti, D. Freni, C. Bettini, X. S. Wang, and S. Jajodia, "Privacy in geo-social networks: proximity notification with untrusted service providers and curious buddies," *VLDB J.*, vol. 20, no. 4, pp. 541–566, 2011.
[29] D. Freni, S. Mascetti, C. Bettini, and M. Cozzi, "Pcube: A system to evaluate and test privacy-preserving proximity services," in *Mobile Data Management*, 2010, pp. 273–275.
[30] A. Narayanan, N. Thiagarajan, M. Lakhani, M. Hamburg, and D. Boneh, "Location privacy via private proximity testing," in *NDSS*, 2011.
[31] Z. Lin, D. F. Kune, and N. Hopper, "Efficient private proximity testing with gsm location sketches," in *Financial Cryptography*, 2012, pp. 73–88.
[32] J. Balasch, A. Rial, C. Troncoso, B. Preneel, I. Verbauwhede, and C. Geuens, "Pretp: Privacy-preserving electronic toll pricing," in *USENIX Security Symposium*, 2010, pp. 63–78.
[33] G. Lenzini, S. Mauw, and J. Pang, "Selective location blinding using hash chains," in *Security Protocols Workshop*, 2011, pp. 132–141.
[34] J. Krumm, "A survey of computational location privacy," *Personal and Ubiquitous Computing*, vol. 13, no. 6, pp. 391–399, 2009.
[35] T. M. Cover and J. A. Thomas, *Elements of information theory (2. ed.)*. Wiley, 2006.