JID: COMCOM [m5G;April 6, 2016;13:52]

Computer Communications xxx (2016) xxx–xxx

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

ABAKA: A novel attribute-based k-anonymous collaborative solution for LBSs

Q1 Tooska Dargahi [a], Moreno Ambrosin [b], Mauro Conti [b,*], N. Asokan [c]

[a] Department of Computer Engineering, West Tehran Branch, Islamic Azad University, Tehran, Iran
Q2 [b] Department of Mathematics, University of Padua, Padua, Italy
[c] Department of Computer Science, Aalto University and University of Helsinki, Finland

ARTICLE INFO

Article history:
Received 19 August 2015
Revised 17 February 2016
Accepted 7 March 2016
Available online xxx

Keywords:
Location-based services
Privacy
k-anonymity
p-sensitivity
Ciphertext-policy attribute-based encryption

ABSTRACT

The increasing use of mobile devices, along with advances in telecommunication systems, increased the popularity of Location-Based Services (LBSs). In LBSs, users share their exact location with a potentially untrusted Location-Based Service Provider (LBSP). In such a scenario, user privacy becomes a major concern: the knowledge about user location may lead to her identification as well as a continuous tracing of her position. Researchers proposed several approaches to preserve users' location privacy. They also showed that hiding the location of an LBS user is not enough to guarantee her privacy, i.e., user's profile attributes or background knowledge of an attacker may reveal the user's identity. In this paper we propose ABAKA, a novel collaborative approach that provides identity privacy for LBS users considering users' profile attributes. In particular, our solution guarantees *p*-sensitive *k*-anonymity for the user that sends an LBS request to the LBSP. ABAKA computes a cloaked area by collaborative multi-hop forwarding of the LBS query, and using Ciphertext-Policy Attribute-Based Encryption (CP-ABE). We ran a thorough set of experiments to evaluate our solution: the results confirm the feasibility and efficiency of our proposal.

1. Introduction

With the rapid development of mobile devices and advances of telecommunications, mobile users tend to have ubiquitous access to information such as traffic prediction or location map data. Location-Based Services (LBSs) are the best examples of this new trend, allowing mobile users to receive information based on their geographical position [1]. Based on their location, mobile users can access several types of information and services, e.g., getting the position of the nearest gas station, restaurant or hospital.

An LBS consists of two major entities: a user (from now on referred also as *issuer* of a query) who is interested in acquiring location-based service, and a Location-Based Service Provider (LBSP) which provides the desired location-based service to the issuer. To obtain such a service, the issuer sends her geographical location, along with her identity and the query to the LBSP. Unfortunately, some queries (such as searching for the nearest hospital specialized in a particular disease) may reveal privacy-sensitive information about the issuer.

The growing interest of smartphone users in using LBSs leads to two major privacy concerns: *location privacy* and *identity privacy* (also known as *query privacy*). The former refers to preventing the disclosure of the exact location of an issuer, while the latter is the ability of concealing the link between her identity and her query. These two concepts are complementary, and therefore, guaranteeing both location and identity privacy for an issuer becomes a challenging task. Researchers proposed several solutions providing location and identity privacy in the context of LBSs (examples can be found in [2]). The location privacy problem has also been studied extensively in other contexts such as sensor networks [3], and cloud computing [4].

A popular tool used in the literature to guarantee user's identity privacy, in the context of LBSs, is the concept of *k*-anonymity [5]. This concept refers to a set of *k* users in which a target user is indistinguishable (with respect to her location) from the other $k-1$ individuals in the set. However, according to [6], in the presence of an attacker with background knowledge about a user's profile attributes, we can only guarantee *k*-anonymity by considering anonymity sets in which all the users have the same profile attributes. Furthermore, the authors in [7] proved that *k*-anonymity is not sufficient to protect the privacy of an individual's attributes in a dataset, and might not prevent the disclosure of sensitive attributes for the user. With respect to *sensitive attributes*, we refer to a precise definition in [8]: "*an attribute whose values may be confidential for an individual (subject to her/his preferences)*".

Q3 * Corresponding author. Tel.: +390498271488.
E-mail address: conti@math.unipd.it (M. Conti).

http://dx.doi.org/10.1016/j.comcom.2016.03.002

Please cite this article as: T. Dargahi et al., ABAKA: A novel attribute-based k-anonymous collaborative solution for LBSs, Computer Communications (2016), http://dx.doi.org/10.1016/j.comcom.2016.03.002

Indeed, in the context of LBSs, the semantics of an issued query might allow the LBSP to infer sensitive attributes of an issuer's profile, or even her identity [9].

In order to address this problem, researchers proposed a solution called p-sensitive k-anonymity [7,9,10], in which at least p different values for each group of sensitive attributes are used. In the context of LBSs, this translates in ensuring that the anonymity set for an issuer contains individuals with diverse values for a specific set of privacy-sensitive attributes. In this paper, inspired by the concept of "personalized privacy preservation" by Xiao and Tao in [8], we give the opportunity to the issuer of a query to decide her preferences in sensitive attributes, based on her query content and physical location. We provided this feature for the issuer, due to the fact that an attribute could be sensitive for a query in special location, and insensitive for another query in another location- (we will further clarify this matter in the following). Before introducing the key contribution of the paper, we present a running example.

Medical help example. Consider a set of smartphone users in a geographical area. We assume that each user is assigned a profile that consists of five attributes: {*Gender*, *Age*, *Nationality*, *Job*, *Zip-code*}. Suppose a user u_1 is a 19-year-old Finnish girl living in Italy. She is looking for a pregnancy help center near her house, where the doctors are able to speak English. She sends an LBS query $Q =$ *"where is the nearest pregnancy help center with English speaking doctors?"* and wants to cloak her location while being 9-anonymous. In this example, based on the content of the query, the attributes *Gender* and *Zip-code* should be identical between all the users in the anonymity set (i.e., providing profile k-anonymity). Moreover, based on the semantics of the issued query, *Age* and *Nationality* are sensitive attributes of u_1. It should be noted that age and nationality are not sensitive attributes per se, but due to the fact that the issuer is in Italy, her nationality could reveal her identity. Moreover, her query semantics (i.e., being pregnant) strongly relates to her age. Therefore, we consider these two attributes to be her sensitive attributes. Assume that she computes a cloaked area using one of the existing k-anonymity preserving methods, and sends her query to the LBSP. Given the fact that she is looking for an English speaking doctor, a malicious LBSP can infer that the issuer is foreigner. Moreover, suppose that there are only two foreign users in her cloaked area: one 19 years old (u_1) and the other 50 years old. In such case, if the attacker has this background knowledge, he can infer that the issuer is likely to be u_1. This example emphasizes the fact that, based on the query semantics and considering the attacker's background knowledge, some attributes could be sensitive in specific scenarios and reveal the identity of the issuer. A proper privacy preserving solution should take into account sensitive attributes of u_1, according to the semantics of the query. For example, a solution could provide an anonymity set in which all the k users are non-Italian (i.e., providing profile k-anonymity) and there are enough diversity in age attribute (i.e., providing p-sensitivity considering the more probable values for being pregnant).

Contribution. In this paper, we propose ABAKA (Attribute-Based k-Anonymous collaborative solution for LBSs), a novel solution to provide both identity, and location privacy for LBS users taking into account the profile attributes of the users. Our motivation is the existing limitations of the prior research in the area of LBS users' privacy: on the one hand, those researches which attempt to ensure k-anonymity considering the profile of the users (such as in [6]) are centralized; and on the other hand, the existing distributed approaches do not consider profile attributes of the LBS users (such as in [11]).

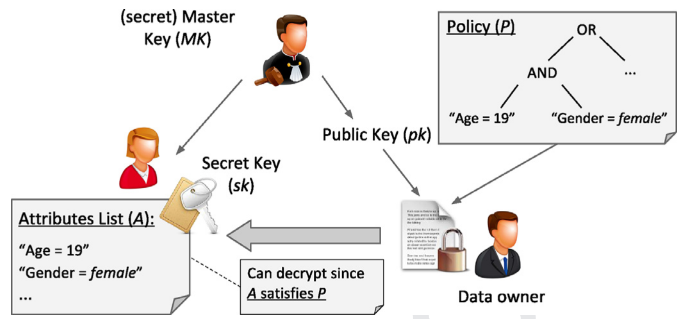


Fig. 1. Example of CP-ABE encryption and decryption.

In this paper, we make the following contributions:

- We propose ABAKA, the *first* privacy-preserving LBS system that guarantees p-sensitive k-anonymity running a TTP-free protocol between participating users (Section 4). In particular, ABAKA has the following features:
  - It cloaks the exact location of a user into a cloaked area of arbitrary size, by ensuring that (at least) $k-1$ collaborating users will forward a query in a random multi-hop path within the cloaked area.
  - ABAKA guarantees p-sensitivity by ensuring that the collaborating users in the anonymity set, which will forward the query, have specific attributes selected by the issuer. Each issuer can select a desired set of attributes based on the semantics of the query she wants to send. In particular, with ABAKA she can decide: (i) which attributes need to be identical within an anonymity set; and (ii) which attributes are sensitive, and thus need to have p different values within the anonymity set.
  - ABAKA adopts Ciphertext-Policy Attribute-Based Encryption (CP-ABE) [12], in order to apply fine-grained access control over encrypted data, by defining high-level access policies as a combination of attributes. CP-ABE allows the issuer to specify attribute-based policies on the query; in this way, she ensures that other $k-1$ collaborative users have the desired attributes.
  - ABAKA ensures the confidentiality of the query, by using public key encryption.
- We run a systematic performance evaluation of ABAKA using two different datasets (Section 5.1) and a thorough evaluation of the computational overhead imposed by cryptographic processing required by ABAKA (Section 5.2). Our evaluation demonstrates that ABAKA is feasible on both smartphone and PC platforms.

2. Background on attribute-based encryption

In what follows, we introduce the fundamental concepts about Attribute-Based Encryption (ABE), and Ciphertext-Policy Attribute-Based Encryption (CP-ABE) in particular. In 2005, Sahai and Waters introduced a Fuzzy Identity-Based Encryption scheme [13], called ABE. This scheme is a public key encryption protocol that allows an encryptor to specify fine-grained access control policies over data. In this scheme, each user is assigned a set of *attributes* (e.g., *Gender*, *Age*, or *Job*). The data owner encrypts a plaintext in such a way that all the users that have a specific set of attributes will be able to decrypt the ciphertext (i.e., if user's attributes *satisfy* the policy over the data). CP-ABE [12] is a type of ABE in which the access policy is included into the ciphertext, and expressed as a combination of attributes. An example of such a policy is: $(Age = 19 \wedge Gender = female) \vee (Nationality = Italian)$ (see Fig. 1).

Each user has a private decryption key, which represents the set of attributes she owns. She will be able to decrypt a ciphertext

Table 1
Notation table.

Notation	Description
Q, *R*	Location-based query and response, respectively
s, *r*	Issuer-generated random numbers
pk_L, sk_L	Respectively, public and private key pair of the LBSP
k_u, sk_u	Respectively, symmetric key and private CP-ABE key of *u*
k_r	Symmetric key of collaborating users
pk	Public CP-ABE key
$\textsc{CpabeEnc}_{pk}(ptxt, p)$	Encryption of a plaintext *ptxt* applying a policy *p*, with CP-ABE
$\textsc{Enc}_k(ptxt)$	Symmetric encryption of a plaintext *ptxt*, using key *k*

if and only if a subset of her attributes satisfies the access policy on the data. By construction, in the CP-ABE scheme only the key issuer (i.e., a Certificate Authority) is able to generate new private keys, therefore preventing collusion attacks [12].

In general, a CP-ABE scheme provides the following functions:

- **Setup.** It takes as input an implicit security parameter, and outputs the public key *pk*, and the master key *MK*.
- **Encryption.** It takes as input a message *M*, an access policy *A*, and the public key *pk*, and outputs the corresponding ciphertext *E*.
- **KeyGen.** It takes as input a set of attributes $\mathcal{A} = \{A_1, A_2, \ldots, A_n\}$, the master key *MK* and the public key *pk*. It outputs a decryption key *D* reflecting the given attributes.
- **Decryption.** It takes as input the ciphertext *E* that is encrypted under the access policy *P*; the decryption key *D* representing a set of attributes γ; and the public key *pk*. It outputs the message *M* if and only if $\mathcal{A}$ "satisfies" the access policy *P*.

Several researches on LBS context adopt ABE to provide either access control or location privacy. For example, in [4], Zhu et al. used KP-ABE scheme in order to: (i) protect the privacy of the issuer against LBSP by enforcing the user authentication process to be accomplished on the client-side, and (ii) control the access to exchanged data between the issuer and the LBSP through defining access policies. In another work, Yang et al. [14] proposed a privacy preserving method for vehicular location based services. In this scheme, each user encrypts her location information using ABE, while defining desired access policy, and shares her encrypted location in online social sites. Leveraging ABE, the authors protect the location information of the users against third party attackers. Different from the state-of-the-art, for the first time, we adopt ABE in ABAKA in order to find $k-1$ collaborating users who have our desired attributes in their profiles, to provide *p*-sensitivity as well as *k*-anonymity.

3. Model and assumptions

In this section, we provide some definitions and assumptions that will be used in the remainder of the paper. Table 1 reports the used notation.

3.1. System model

We consider a set of users $U = \{u_1, u_2, \ldots, u_m\}$ in a geographical area. Each user can be a potential LBS user (i.e., an issuer) and is equipped with a location-aware wireless device (e.g., smartphone or tablet) that is able to retrieve the coordinates associated with its position. We assume the users to be mostly stationary (from the time the issuer sends out the query until when she receives the response back), or to have limited mobility. Users can communicate with their neighboring users over a wireless medium (e.g., via WiFi) via a single-hop or a multi-hop route. Moreover, we assume that users ignore received packets that are not intended for them (which they could receive due to the broadcast nature of the wireless communication). We consider the ad hoc model due to the increasing trend in opportunistic networks and device-to-device communications, where several mobile devices (e.g., smartphones) collaborate in order to forward messages using wireless technologies, such as Bluetooth or WiFi [15,16]. This model has been extensively used and analyzed in several works in the literature, such as [15,17–20].

We assume that each user is assigned a profile which consists of a set of attributes $\mathcal{A} = \{A_1, A_2, \ldots, A_n\}$. These attributes can be of different types: personal information (e.g., gender), employment information (e.g., job), and contact information (e.g, Zip-code). In our *medical help* example, we consider the following profile attributes: {A_1: *Gender*, A_2: *Age*, A_3: *Nationality*, A_4: *Job*, A_5: *Zip-code*}. We also assume that none of the users have exact information about the number of users in her vicinity, and their profile attributes. We consider the LBSP to be untrusted, and assume that each LBS user does not want to share her exact location and identity (*ID*) with the LBSP. In our model, the issuer sends her request to the LBSP through a multi-hop path, to anonymize her location and identity. Our multi-hop approach is similar to the work in [19,21], however in ABAKA the issuer looks for a set of *collaborating users* having specific attributes, who cooperate with each other to anonymize the location of the issuer. We also assume that each user, based on its own policy, decides whether to participate in the anonymizing process. One may think of an incentive mechanism in order to motivate users to participate in our collaborative scheme. There are several monetary and non-monetary incentive schemes in the literature [22], which could be considered to be a complement for ABAKA. One possible approach, to be used, could be the privacy-aware incentive mechanism proposed in [23], which is a TTP-free scheme based on blind signature. However, an encouraging mechanism is out of the scope of this paper (and an orthogonal open research problem, as pointed out by Conti et al. [24]), and we leave it as future work.

We assume that the LBSP has a pair of keys: a public key pk_L, and a private key sk_L that are used to preserve confidentiality and integrity of the message sent by the issuer to the LBSP. Moreover, we suppose that there could be multiple *Certification Authorities* (CAs) [25], each of which being responsible for a specific geographical area (e.g., states or municipalities), to authenticate the users and assign them CP-ABE private keys (users key management is out of the scope of this paper). Each user obtains a CP-ABE private key based on her profile attributes, from the CA nearest to her location. The CP-ABE private key will be used for authentication of collaborating users, and fulfilling the requirement of *p*-sensitivity. Furthermore, CAs provide the CP-ABE public key, that the issuer uses to encrypt her query specifying an access policy. In our solution, we assume each user to contact the nearest CA when her profile attributes change, in order to retrieve a new CP-ABE private key. Note that this does not change the collaborative nature of our approach. We also assume each user u_i has a symmetric key, k_{u_i}, which can be a random number defined by u_i. The user u_i will use this key to encrypt/decrypt a special field of the packet during the packet forwarding procedure. Moreover, the issuer generates a random group secret key, k_r, for the collaborating users.

Finally, in our model each user can specify her privacy requirements in terms of size *k* of the anonymity set, number of users with specific issuer-defined attributes *p*, and the largest and smallest desired cloaked area size. Also, we assume the issuer to not issue any query that the query content could lead to her identification or reveal information about her exact location (otherwise the use of anonymity preserving approaches would not make much sense).

Please cite this article as: T. Dargahi et al., ABAKA: A novel attribute-based k-anonymous collaborative solution for LBSs, Computer Communications (2016), http://dx.doi.org/10.1016/j.comcom.2016.03.002

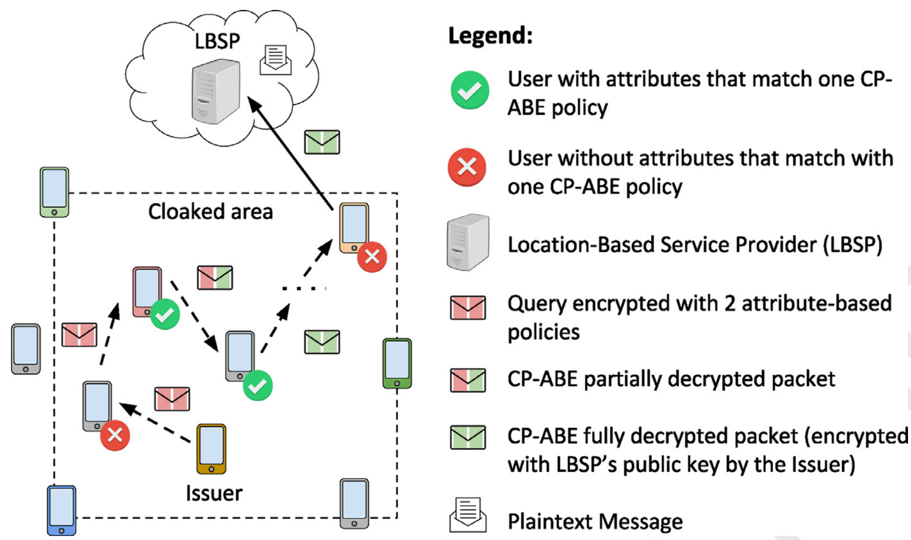


Fig. 2. Multi-hop CP-ABE based routing to form a rectangle cloaked area, example with $k = 3$.

3.2. Adversary model

We consider two types of adversaries: passive and active. A passive adversary can be one of the following three entities [11,26]: (i) the untrusted LBSP, which collects information about LBS users such as their location, identity or activities, based on their queries; (ii) an outsider eavesdropper on wireless communication, which is interested in identifying location and identity of the issuer; (iii) the users that collaborate in computing the k-anonymity set. The collaborating users are not fully trusted; we consider them to be honest-but-curious (we observed that this assumption is consistent with several works in the literature, such as the ones in [27–29]): i.e., users honestly follow the ABAKA protocol, and neither drop nor modify the packets. However, they are curious to learn location and identity of the issuer, or of the other users in the k-anonymity set. We assume that a malicious user cannot generate fake profiles in order to participate in our protocol and decrease the privacy level of the issuer, since the CAs authenticate the users upon joining the network and assign them CP-ABE private keys (we found this assumption consistent with [30,31]).

An active adversary can be one of the non-collaborating users who is not able to satisfy the access policy on the encrypted packet (i.e., the user who does not have the issuer-defined attributes). He is interested in identifying the issuer, modifying the LBS request, or reducing the issuer's privacy level. In the last case, he aims at reducing the number of users in the cloaked area (i.e., reducing the value of k). We assume that both passive and active adversaries have some background knowledge about the users [26]. This background information could be about profile attributes of the users, such as location information (e.g., office address), personal information (e.g., age or nationality), or even the exact or estimated number of users in a geographical location. The adversary aims at using his background knowledge to attack the privacy of the issuer. In our model, we address the collusion attack of non-collaborating users and we assume that collaborating users do not collude (as they are semi-trusted). Finally, in this paper we do not consider other types of attacks, such as, Denial of Service, which is inevitable in all the collaborative approaches in wireless networks.

4. Our solution: ABAKA

In this section, we present ABAKA, our TTP-free solution that provides identity privacy for LBS users. ABAKA deals with both generating and sending the LBS query to the LBSP (Section 4.1), as well as generating and forwarding the requested location-based service to the issuer.

First, the issuer u_i divides the encrypted query into $k - 1$ parts, and on each part enforces a specific access policy by means of CP-ABE [12]. Then, the issuer sends the packet to the LBSP through a multi-hop path. This way, she conceals her identity among other $k - 1$ neighboring users who are able to decrypt the CP-AB encrypted parts of the packet. Fig. 2 provides a high-level example of our multi-hop attribute-based solution, considering $k = 3$. As Fig. 2 shows, the protocol cloaks the position of the issuer (by collaboration of both users with green tick icon and red cross icon in Fig. 2) and computes a k-anonymity set based on the issuer-defined attributes. Using CP-ABE allows us to address two important issues:

- *Finding $k - 1$ collaborating users (users with green tick icon in Fig. 2) having specific attributes, which could be issuer's sensitive attributes.* Enforcing a policy on each of the $k - 1$ parts of the message, the issuer will be sure that only the users with attributes satisfying the policy, are able to decrypt one part. Thus, we guarantee that the collaborating users in the k-anonymity set satisfy p-sensitivity (recall that collaborating users are honest-but-curious). We assume that each collaborating user uses her CP-ABE private key only one time for each received packet. In other words, we assume that if she is able to decrypt some of the CP-AB encrypted parts of the packet with her private key (satisfying more than one policy), she will process just one part. We consider this assumption to ensure that all the $k - 1$ parts of the message will be processed by $k - 1$ different collaborating users and hence ensuring the k-anonymity.
- *Addressing privacy attack form non-collaborating users, i.e., users outside the cloaked area in Fig. 2.* As non-collaborating users are not able to satisfy any of the access policies, they will not be able to decrypt any of the query parts. Therefore, they will not be able to reduce the privacy level of the issuer by collaboration in computing the cloaked area.

In our *medical help* example, user u_1 wants to be 9-anonymous between eight other users who are female and have the same four digit prefix Zip-code, i.e., *Gender* = *female* and *Zip-code* = 0019. Moreover, due to her sensitive attributes, she is looking for eight other users who are not Italian and have diverse values for the age attribute which fall in three different age categories, i.e., 15∼24, 25∼34, and 35∼44. User u_1 uses ABAKA to conceal her identity. She encrypts the query Q with the public key of the LBSP,

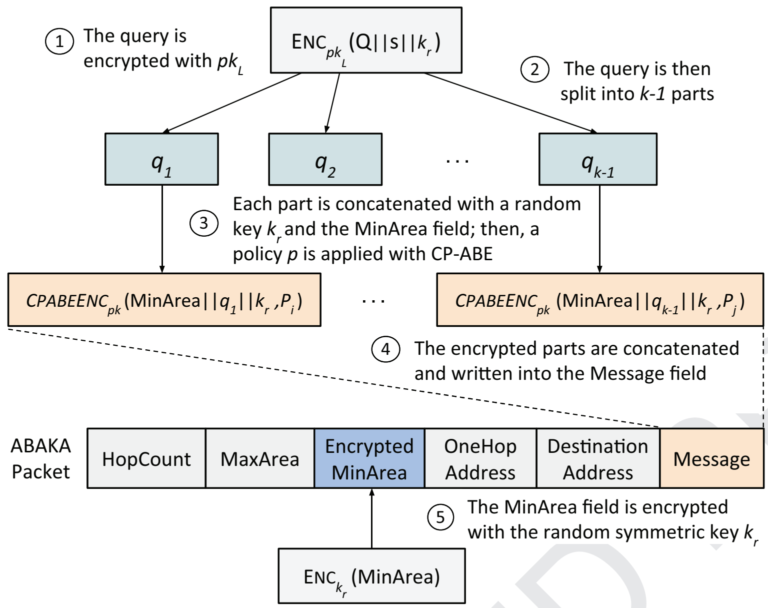


Fig. 3. LBS request packet format generated by the issuer.

splits it into eight equally sized parts and applies an access policy on each part using CP-ABE, such as $(A_1 = female) \wedge (A_5 = 0019) \wedge (A_3\ NOT\ Italian) \wedge (15 \leq A_2 < 25)$. This way, she is sure that only the user with the following attributes will be able to decrypt the corresponding part: who is female, lives in an area with the same Zip-code prefix as u_1, is not Italian, and her age is between 15 and 24. By defining three different categories for the age attribute (A_2), the final 9-anonymity set will be 3-sensitive. As users in the 9-anonymity set have diverse values from three different categories for sensitive attribute of u_1, the probability that the attacker can identify the issuer's age category is $\frac{1}{3}$.

Upon receiving an LBS request packet (the packet with two green parts in Fig. 2), the LBSP decrypts the query with its private key (sk_L) obtaining: Q; a random number s, and random symmetric key k_r generated by the issuer; and the encrypted cloaked area. Then, the LBSP decrypts the cloaked area field by the obtained k_r and generates a response message R considering the cloaked area, which comprises the location information requested by the issuer. To provide confidentiality of the response message, the LBSP encrypts R with s. Finally, the LBSP sends the generated response packet back to the user that delivered the query (the user in right top corner of the cloaked area in Fig. 2). All the collaborating users in the k-anonymity set use a semi-onion routing approach [32] to send the response packet back to the issuer. In particular, semi-onion routing allows us to deliver the response packet to the issuer, following the reverse path, without the need for all the nodes in the path to keep track of the path locally. This approach is not intended to hide the path from the LBSP to the issuer; indeed, we leave this as a future work.

4.1. Generate and forward a request

In this section, we describe how a query issuer, u_i, is generating and forwarding an LBS request to the LBSP. In particular, an LBS request packet is composed of the six fields illustrated in Fig. 3 and discussed in the following.

The `Message` field contains the query Q, a random number s, and a randomly generated symmetric key k_r encrypted with the public key, pk_L, of the LBSP. This message is then split into $k - 1$ parts, each encrypted with CP-ABE applying a certain policy, and finally recomposed. The `HopCount` field denotes the maximum number of hops that the packet should pass through other users. Its value should be greater than $k - 1$. The `MaxArea` field denotes the maximum size of the desired cloaked area in the form of a rectangle, which is defined by two points (x_l, y_l) and (x_r, y_r) for bottom left and top right corners of the rectangle, respectively. The `MinArea` field represents the minimum size of the desired cloaked area in the form of a rectangle, which is defined by two points (x'_l, y'_l) and (x'_r, y'_r) for bottom left and top right corners of the rectangle, respectively. The content of this field is encrypted with the randomly generated symmetric key k_r. After completing the cloaking procedure, this field represents the actual cloaked area dimensions. `OneHopAddress` is used for routing back the LBSP response to the issuer of the query. The initial value of this field is $\text{ENC}_{k_{u_i}}(r)$, where r is a random number generated by the issuer u_i. Upon receiving the LBS request packet, each user encrypts the address of the previous hop with her symmetric secret key (k_{u_i}) and appends this encrypted layer to the current content of the `OneHopAddress` field. Finally, `DestinationAddress` contains the address of the LBSP.

4.1.1. Packet generation

An issuer u_i generates a packet executing the Algorithm 1, which comprises the following steps:

Step 1. The query issuer, u_i, generates a *Message* which comprises her query, Q, a random number, s, and a randomly generated symmetric key k_r encrypted with the public key, pk_L, of the LBSP (Algorithm 1, lines 2–3).

Step 2. The issuer splits the encrypted *Message* into $k - 1$ parts (e.g., in chunks of equal size), where k is the k-anonymity parameter (Algorithm 1, line 4). Then, she defines the minimum size of the desired cloaked area, `MinArea` field (Algorithm 1, line 5). She appends the `MinArea` field and also the symmetric key k_r to each part and encrypts that part with CP-ABE, specifying an access policy, i.e., a combination of desired attributes (Algorithm 1, lines 6-8). The reason behind including `MinArea` field in each part is to provide each collaborating user with the means of checking whether the actual minimum desired cloaked area defined by the issuer has been modified during the path by intermediate nodes (we will provide a further discussion in Section 4.2).

Step 3. The issuer creates an empty packet (Algorithm 1, line 9), as illustrated in Fig. 3. Then, she concatenates the $k - 1$ parts generated in the previous step to form a complete message

Please cite this article as: T. Dargahi et al., ABAKA: A novel attribute-based k-anonymous collaborative solution for LBSs, Computer Communications (2016), http://dx.doi.org/10.1016/j.comcom.2016.03.002

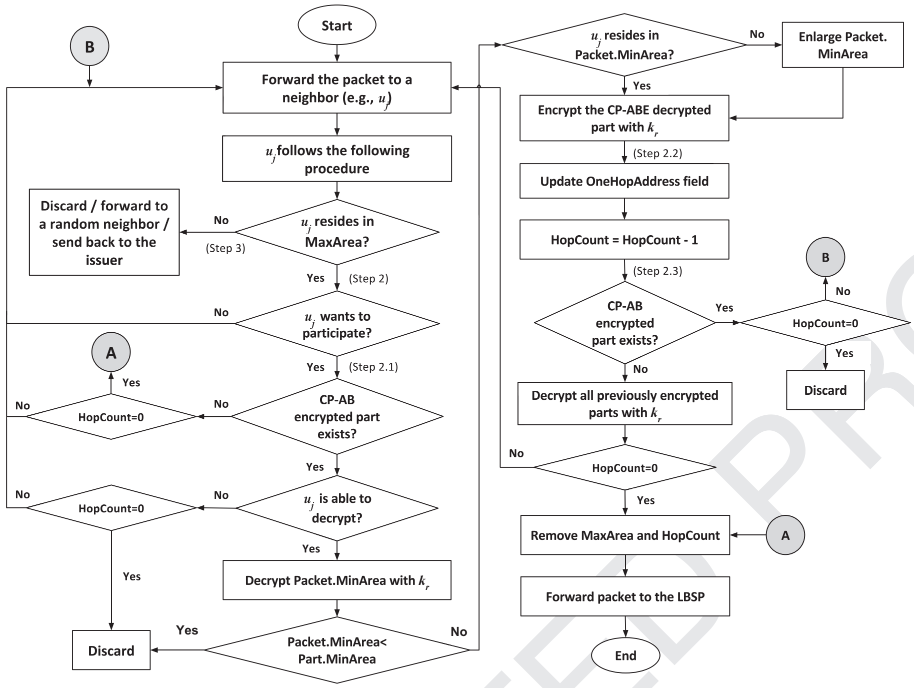


Fig. 4. Packet forwarding flowchart.

Algorithm 1 LBS Packet Generation.

Input: The LBS query Q, the anonymity parameter k, an array of policies, the maximum hop count max, the largest cloaked area limits $((x_l, y_l), (x_r, y_r))$, the smallest cloaked area limits $((x'_l, y'_l), (x'_r, y'_r))$, and the Destination address $Destination$.

1: **procedure** GenerateRequest(k, $policies[]$, Q, max, (x_l, y_l), (x_r, y_r), (x'_l, y'_l), (x'_r, y'_r))
2: $k_r \leftarrow$ RandomKey(); $s \leftarrow$ RandomNumber();
3: $Message \leftarrow \text{Enc}_{pk_L}(Q||s)$;
4: $parts[] \leftarrow \text{Split}(Message_{enc}, k-1)$;
5: $minArea \leftarrow \text{Area}((x'_l, y'_l),(x'_r, y'_r))$;
6: **for** $i \in [1 : k-1]$ **do**
7: $parts[i] \leftarrow \text{CpabeEnc}_{pk}(minArea||parts[i]||k_r, policies[i])$;
8: **end for**
9: $packet \leftarrow$ GenerateEmptyPacket();
10: $packet.Message \leftarrow$ Concatenate($parts[]$);
11: $packet.HopCount \leftarrow max$;
12: $packet.MaxArea \leftarrow \text{Area}((x_l, y_l),(x_r, y_r))$;
13: $packet.MinArea \leftarrow \text{Enc}_{k_r}(minArea)$;
14: $packet.DestinationAddress \leftarrow Destination$;
15: $r \leftarrow$ Random();
16: $packet.OneHopAddress \leftarrow \text{Enc}_{k_{u_i}}(r)$;
17: Forward($packet$, $neighbors[]$);
18: **end procedure**

(Algorithm 1, line 10). Afterward, u_i defines her privacy requirements in terms of maximum number of neighbors that the message should pass through, the maximum and minimum size of the desired cloaked area, and the destination address, i.e., the address of the LBSP (Algorithm 1, lines 11–14). The issuer u_i encrypts the `MinArea` field of the header with k_r, to avoid eavesdroppers or non-collaborating users to be able to read (or modify) such information (Algorithm 1, line 13).

Step 4. Before sending the packet to a next hop, u_i encrypts a random number r with her symmetric secret key (k_{u_i}), and attaches it to the packet (Algorithm 1, lines 15–16). Finally, u_i sends the generated packet to one of her neighbors. The choice of the next-hop can be done in several ways, e.g., selecting randomly or based on the proximity with the issuer (Algorithm 1, line 17).

In the *medical help* example, user u_1 splits the encrypted query into eight parts. Then, she defines her desired smallest cloaked area (`MinArea`) which could be 100 m × 100 m rectangle including her house (the house is not necessarily placed in the center of the defined area). She concatenates the `MinArea` to each part along with a random symmetric key k_r, and applies the aforementioned policies on each part. Afterward, she determines her largest desired cloaked area, `MaxArea`, which is a 600 m × 600 m rectangle including her geographical position and the maximum number of hops (e.g., `HopCount=15`). Then she encrypts a random number r with her symmetric key (k_{u_1}) and specifies the address of the LBSP. Finally, she forwards the generated packet to one of her neighbors.

4.1.2. Packet forwarding

Once received a packet, a user u_j performs the following operations (the packet forwarding procedure's flowchart is depicted in Fig. 4):

Step 1. User u_j checks whether she resides in the largest desired cloaked area defined in the `MaxArea` field of the packet.

Step 2. If u_j resides in the defined area, she peruses the packet fields to decide, based on her own policies, whether she wants to participate in the cloaking algorithm. If she does not want to collaborate, she forwards the packet to another user. Otherwise, she performs the following actions:

- *Step 2.1:* The user u_j checks the `Message` field of the packet, to verify whether there is any encrypted part, and if she is able to decrypt one of them. User u_j will be able to decrypt one part, if and only if the attributes associated to her profile (i.e, attributes associated to her private key sk_{u_j}) satisfy the policy enforced on that part. If able to decrypt, u_j decrypts the `MinArea` field of the packet header, i.e., `Packet.MinArea`, using the key k_r obtained from the CP-ABE decrypted part. Then, u_j compares such field with the `Part.MinArea` field: if `Packet.MinArea` < `Part.MinArea`, it means that an attacker has decreased the

Please cite this article as: T. Dargahi et al., ABAKA: A novel attribute-based k-anonymous collaborative solution for LBSs, Computer Communications (2016), http://dx.doi.org/10.1016/j.comcom.2016.03.002

original value defined by the issuer. In such a case, u_j discards the packet. Otherwise, u_j continues by checking whether she resides in the area defined by the `Packet.MinArea`. If not, u_j enlarges the area to include also her location. Then, she updates the part she is currently processing, by removing the `Part.MinArea` field and k_r and encrypting such part with k_r.

- *Step 2.2:* The user u_j updates the current value of the `OneHopAddress` concatenating the address of the previous hop, and encrypting the whole content of the field with her symmetric secret key (k_{u_j}). This way she adds a new "onion layer" that will be used to route the response message back to the issuer. Then, u_j decrements the value of the `HopCount` field. If u_j is the one who decrypted the last part with her CP-ABE key, she decrypts all the previous parts with the key k_r. Then, if `HopCount`$= 0$, u_j removes the `MaxArea` and `HopCount` fields of the packet header, and sends the query to the LBSP. The coordinates (x'_l, y'_l) and (x'_r, y'_r) in the `Packet.MinArea` field represent the actual cloaked area, i.e., the smallest area covering the positions of all the collaborating users. If `HopCount` > 0, u_j continues forwarding the packet to one of her neighbors.
- *Step 2.3:* If there are other encrypted parts (i.e., the packet did not pass enough users to guarantee k-anonymity), or if the user was not able to decrypt one of the parts of the message, u_j continues forwarding the packet to one of her neighbors. Before forwarding the packet, u_j checks the `HopCount` value. If `HopCount`$= 0$, u_j discards the packet. Otherwise, forwards the packet again.

Step 3. If u_j does not reside in the defined largest cloaked area, she can perform one of the following actions: drop the packet, forward it to a random neighbor, or send the packet back to the previous user.

The protocol explained in this section ensures that the query is forwarded through, at least, $k - 1$ neighboring users having specific attributes, ensuring k-anonymity and p-sensitivity.

4.2. Discussion

In this section we briefly discuss issues related to packet generation and forwarding, as well as the privacy level provided by ABAKA.

4.2.1. Packet generation

To ensure that the smallest cloaked area specified by the issuer will be respected, we introduced the `MinArea` field in the ABAKA packet. This field is of extreme importance in order to guarantee the desired privacy level for the query issuer. Indeed, on one hand, an attacker might want to increase such area to reduce the quality of service; and, on the other hand, the attacker might also want to reduce the value of the `MinArea` field, in this case attempting to reduce the privacy guarantees of the ABAKA. In order to prevent these two attacks, we place the `MinArea` field inside each of the CP-ABE encrypted parts of the query. We also encrypt the `MinArea` field of the packet header with a secret symmetric key (k_r), which can be accessed only by the collaborating users after decrypting a CP-ABE part. This way, only the collaborating users are able to modify this field as well as verifying the possible malicious modifications to the packet, and eventually discarding it. Similarly, also the `MaxArea` and `HopCount` fields might be targeted by an attacker, who may want to enlarge or reduce their values. However, such possible attacks would lead to a Denial of Service, that is out of the scope of this work.

4.2.2. Packet forwarding

During the packet forwarding process, we may have some concerns. First, participating in the ABAKA protocol may threaten the privacy of the collaborating users. Indeed, the issuer could infer that there are people with specific attributes in the cloaked area, simply by issuing several ABAKA messages adopting different policies. We addressed this concern by allowing each user who receives the packet to decide whether to participate in the protocol or not. Therefore, if a user receives a packet, which has some parts that specify her own sensitive attributes, she can decide to not decrypt such part and just forward the packet to a neighbor. Another possible solution for this problem could be considering each collaborating user to be able to influence the packet, e.g., enlarging the minimum cloaked area and then decrypting the packet. In this way, she can cloak herself in a larger area.

The second concern is the participation of users with revoked attributes. This issue is mainly related to the key revocation mechanisms for CP-ABE, and therefore is out of the scope of this paper. We will leave such concern as a future work.

A third issue is the collusion of non-collaborating users, that might want to send the packet to the LBSP when only a portion of CP-ABE parts are already decrypted. In such a scenario, the LBSP may be able to extract some useful information from the currently decrypted parts. We addressed this issue introducing a random symmetric key (k_r) that each collaborating user will obtain after decrypting a CP-ABE part; after processing the `MinArea` field (as explained in Section 4.1.2), each collaborating user will encrypt with k_r the part she decrypted with her CP-ABE private key. In this way, even in case of collusion attack, the LBSP receives an encrypted packet and cannot infer any useful information.

Another privacy concern is the mobility of the collaborating users which may lead to a reduction of the k-anonymity level, in a case that some of the collaborating users leave the cloaked area. Although we assumed users to be in a limited mobility scenario, we could integrate mobility and movement directions in computing the cloaked area to support also dynamic networks (e.g., taking into account the speed of the collaborating users, and computing how much they could move by the time the response comes back, and computing whether they will still be reachable). However, such integration is not trivial, since it depends on several parameters (e.g., its domain of application), and requires a trade-off between privacy level, overhead, and trust to some central entities (such a trade-off is a common issue in collaborative approaches, such as in [33]). We leave the management of nodes' mobility as a future work.

The other issue could be continuous request of a same LBS by a user u in a cloaked area. In this case, the LBSP might identify the user by correlation of the requests over time. In such case, overtime if the other individuals in the anonymity set are changed, then the user u could be the one who is requesting the same query. This attack can happen in two cases: (i) if the attacker has a general view over the path, which could be solved by using some kind of anonymous routing, (ii) if the attacker has local real-time knowledge about the individuals in the set and the query content, and also have historical information about the previous same requests and the individuals in that sets. We leave a thorough study of the latter attack as future work.

Finally, another issue is the delay imposed by the multi-hop forwarding, and finding $k - 1$ users with specific attributes. ABAKA is most effective in dense environments (in which the probability of finding collaborating users in vicinity is high) and non real-time scenarios. It provides a strong privacy protection considering the issuer profile attributes varying for each user and query, with the cost of imposing delay to the system. In many applications, the issuer is willing to accept a trade-off between strong privacy protection (by defining strict access policies) and latency (or not receiving response at all). We could also define a maximum time bound for the reception of the response: if the issuer does not receive the response within a certain time frame, she can decide to relax the

Table 2
Statistics on the considered datasets (data extracted from [34–37]).

City	Inhabitants per km^2	Smartphone Users (%)	ABAKA Users (%)	Neighboring users Average	Neighboring users Std. Dev.
New York	27,733	64	50	20.00	4.82
			60	23.89	5.31
			70	27.85	5.79
Milan	7382	41	50	2.99	1.99
			60	3.37	2.08
			70	4.00	2.24

Table 3
Considered attributes and their distribution, according to the data in [34].

Attribute	Attribute value	Presence in the population (%)
Sex (*S*)	male (*m*)	47.5
	female (*f*)	52.5
Race (*R*)	white (*w*)	33
	black (*b*)	25.5
	latino or hispanic (*h*)	28
	asian (*s*)	12.7
	american indian (*a*)	0.8
Origin (*O*)	foreign born (*f*)	37
	local born (*l*)	63
Age (*A*)	<18	21.6
	between 18 and 65	66.3
	≥65	12.1

privacy constraints and re-issue the query. It is worth mentioning that, as a design choice, we attributed higher priority to users' privacy, with respect to the quality of service. Therefore, in the case of not finding enough collaborating users, the issued query will not be submitted to the LBSP and the issuer will still be anonymous, but we do not ensure that she will receive her requested service.

4.2.3. Privacy discussion

As introduced in Section 3.2, we consider the following adversaries separately: (i) the untrusted LBSP; (ii) an outsider eavesdropper; (iii) the semi-trusted collaborating users; (iv) the untrusted non-collaborating users. We now discuss how ABAKA protects users against these adversaries.

(i) Consider the *medical help* example. Based on the content of the query, the LBSP could infer that the sender is a foreign woman, probably between 15 and 45 years old. However, even with background knowledge about profile attributes of women in that area, it could not infer which of these women could be the issuer. In fact, there are at least nine women in the age range between 15 and 44, with different nationalities.

(ii) The outsider eavesdropper observes the communication between the users. He is not able to access the content of the packet since it is encrypted with CP-ABE, and with the public key of the LBSP. If he can observe all the path, he can find out the issuer and if he has background knowledge about what could be the issuer's query, he may only be able to infer some attributes of the collaborating users; however, it is a strong assumption about the adversary. One can think about an on top anonymized routing layer which could be an orthogonal solution to be used along with the ABAKA, and we leave it as a future work.

(iii) There is no useful information inside the LBS packet for honest-but-curious collaborating users; the content of the message is encrypted with the public key of the LBSP, and both location and identity of the issuer are hidden. A curious collaborating user could obtain only knowledge about attributes of all the collaborating users, or, at least, attributes of a subset of collaborating users.

(iv) Non-collaborating users may try to reduce the privacy level of the issuer (e.g., in the previous example, a man could try to collaborate in computing the cloaked area to decrease the value of *k*) or to modify the packet. Using CP-ABE, users without specific attributes are not able to decrypt the packet. Therefore, they can neither modify the packet nor collaborate in the *k*-anonymity set to reduce the privacy level for the issuer.

5. Experimental results

In this section, we present an experimental evaluation of ABAKA, using two different datasets. In Section 5.1 we provide performance evaluation of ABAKA in terms of success rate considering different scenarios; while in Section 5.2 we investigate the overhead imposed by the cryptographic operations in our proposed approach.

5.1. Performance evaluation

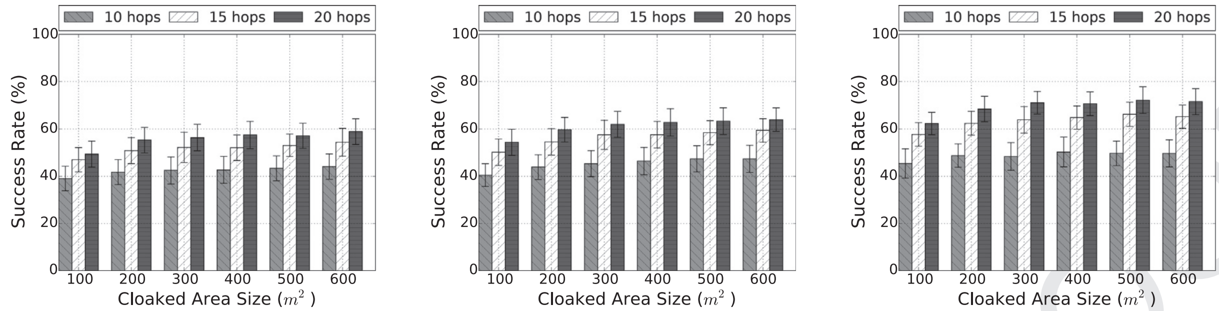
For the purpose of evaluating ABAKA in a realistic scenario, we created two synthetic datasets based on real world statistics of the population of two cities: New York (USA), focusing on the Manhattan island, and Milan (Italy). In particular, we estimated the average number of ABAKA users in an area of 1 km^2, based on: (1) the average population density in such cities, obtained from [34] and [35]; (2) the statistics on the smartphone penetration in the state of belonging, i.e., the percentage of population owning a smartphone, according to [36] and [37]; and (3) a hypothetical percentage of the smartphone users with the ABAKA application installed (50%, 60%, and 70% were considered). Moreover, in our evaluation we assumed a WiFi range of 25 meters for each device [38]. Table 2 shows some statistics about the considered datasets, in particular the number of users per km^2, the percentage of considered collaborating users, and the average number of neighboring collaborators for each user. As we can see form Table 2, the Milan dataset represents a non-dense scenario. Indeed, the average collaborating neighbors per ABAKA user, spans, on average, form 2.99 to only 4.00, with a percentage of ABAKA users in the smartphone-users population of 50% and 70%, respectively. The New York dataset, instead, represents a “best case” scenario, where the average connection degree per ABAKA user is high, e.g., some 23.89 neighbors on average, considering a 60% ABAKA users in the smartphone-users population.

To evaluate the performance of ABAKA, we measured the average success rate for a query packet to be received by the LBSP, varying the maximum allowed size of the cloaked area, from 100 m^2, to 600 m^2, with steps increase of 100 m^2, as well as the maximum allowed hops number, i.e., 10, 15 and 20 hops.

In our evaluation, we performed our experiments considering two possibilities for a user to forward a message to a neighbor, i.e., she can forward the packet to: (1) the closest neighbor, or (2) a random one. We also considered different possible actions that a user can perform when receiving a packet outside of the largest possible cloaked area. In this case, she can decide to: (i) drop the packet, (ii) forward it to a random neighbor, or (iii) return the packet back to the previous user, which in turn will select another user to which forward the message. However, in our experiments we did not consider option (i), since it would reduce the probability for a message to complete the protocol.

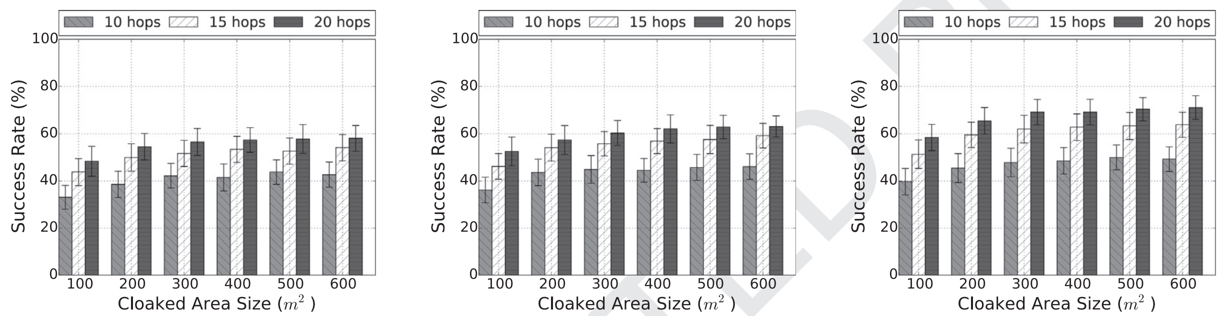
We considered four different types of attributes for the population, reported in Table 3. The table reports also the distribution of attribute values in the population, extracted from [34]. We

Please cite this article as: T. Dargahi et al., ABAKA: A novel attribute-based k-anonymous collaborative solution for LBSs, Computer Communications (2016), http://dx.doi.org/10.1016/j.comcom.2016.03.002



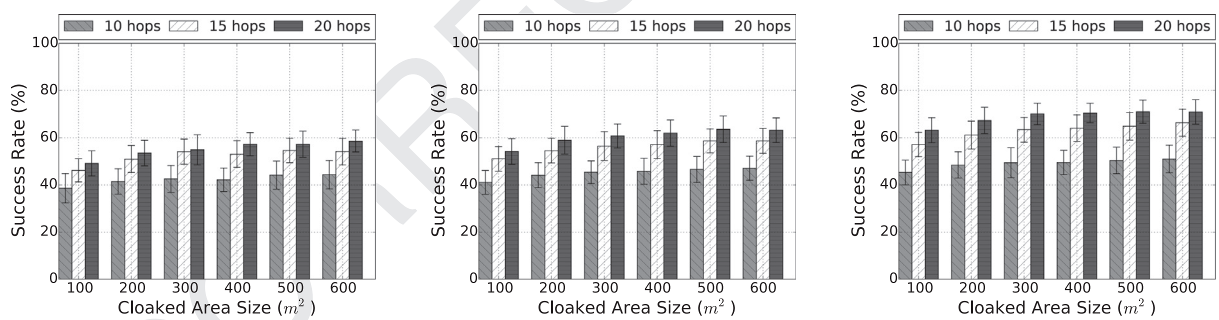
(a) 50% ABAKA users. (b) 60% ABAKA users. (c) 70% ABAKA users.

Fig. 5. Success rate of ABAKA simulating policies combination (a) on the Milan dataset. Each user forwards the message to its closest neighbor; outside the cloaked area, user returns the message to previous user.



(a) 50% ABAKA users. (b) 60% ABAKA users. (c) 70% ABAKA users.

Fig. 6. Success rate of ABAKA simulating policies combination (a) on the Milan dataset. Each user forwards the message to its closest neighbor; outside the cloaked area, user forwards the message to a random neighbor.



(a) 50% ABAKA users. (b) 60% ABAKA users. (c) 70% ABAKA users.

Fig. 7. Success rate of ABAKA simulating policies combination (a) on the Milan dataset. Each user forwards the message to a random neighbor; outside the cloaked area, user returns the message to previous user.

performed 1000 runs of the ABAKA protocol, each time randomly initializing the configuration according to the values in Table 3, and randomly selecting a different issuer.

Our evaluation of ABAKA considers the following two different policy combinations, where parentheses delimit a policy enforced on a single message part (considered notation is consistent with the reported attributes in Table 3):

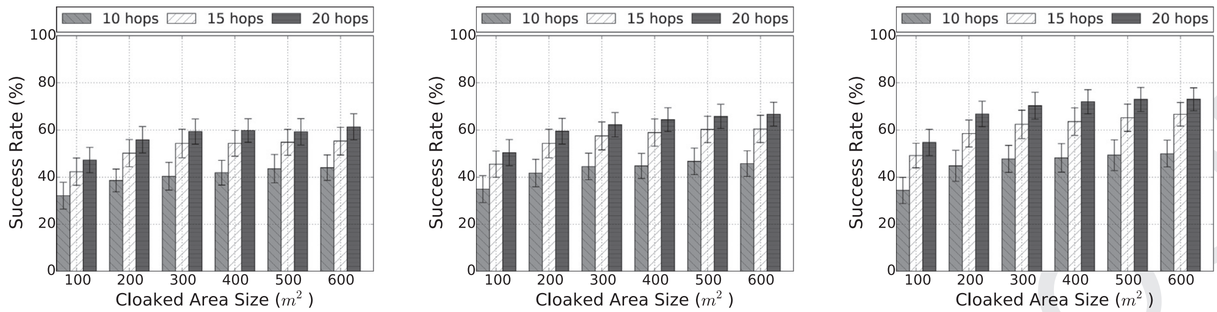
(a) $[(A \geq 18 \wedge S = f), (A \geq 18 \wedge S = f), (A \geq 18 \wedge S = f), (A \geq 18 \wedge S = f)]$
(b) $[(A \geq 18 \wedge O = l), (A \geq 18 \wedge R = h)]$

Policies combination (a) provides at least 5-anonymity, and 1-sensitivity, while polcies combination (b) provides at least 3-anonymity and 2-sensitivity.

Figs. 5–8 present the results of our simulation, adopting the different strategies introduced above, with set of policies (a) on the Milan dataset; Figs. 9–12 presents the results of our simulation with set of policies (a) on the New York dataset. For the sake of brevity, for policies combination (b) we report only the results obtained on both datasets, with strategy (1) for selecting the next collaborating user, and strategy (iii) to handle the out-of-area case. We report these results in Figs. 13 and 14.

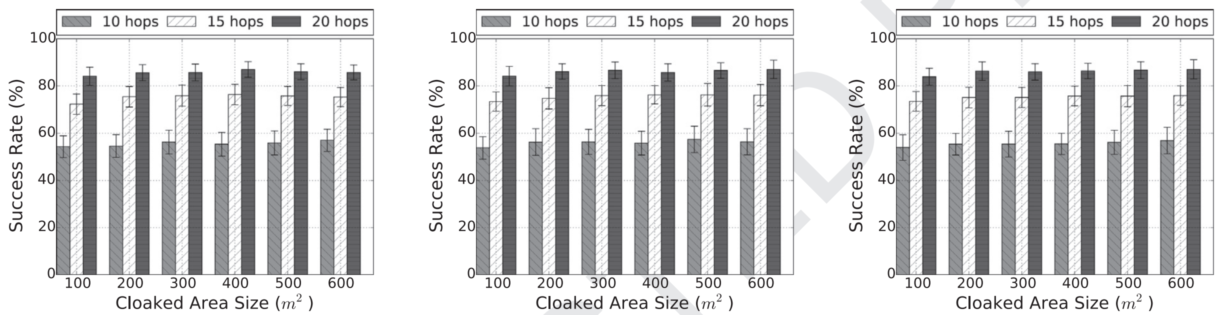
From our results, we can derive some useful observations. First of all, we notice that, unsurprisingly, the average number of collaborating neighbors per ABAKA user (listed in Table 2) influences the success rate of our proposal. This is more evident if we consider the Milan dataset. As an example, Fig. 5 shows a significative increase of the success rate, i.e., from a maximum of some 60% to a maximum of some 70%, as the number of ABAKA users (and consequently the number of neighbors per user) grows. However, note that even in non-dense scenarios, ABAKA achieves a reasonable success rate, e.g., in Fig. 5(c) we can observe that ABAKA is capable to achieve a success rate of some 70%, considering a maximum of 20 hops and a maximum cloaked area size of 200 m^2.

Second, we can observe that both the maximum number of allowed hops, as well as the maximum cloaked area size, play an important role. The effect of the maximum number of hops is evident from the results of the experiment performed on the New



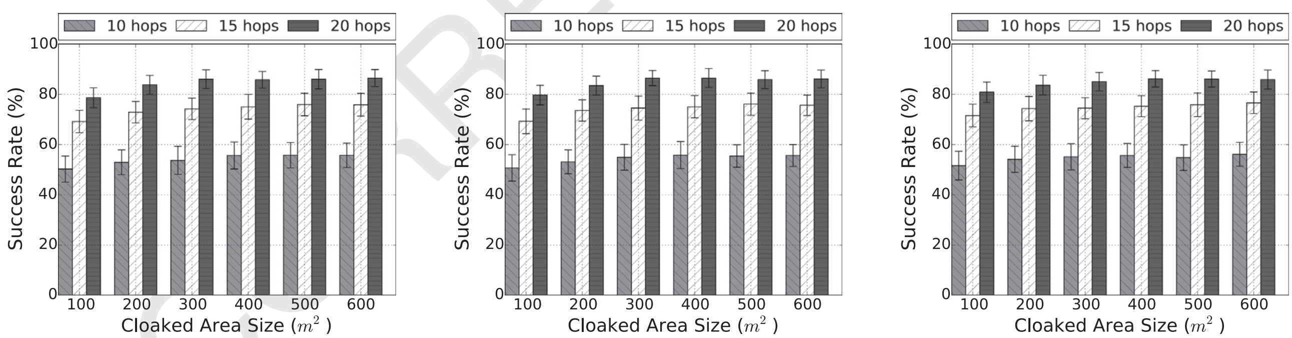
(a) 50% ABAKA users. (b) 60% ABAKA users. (c) 70% ABAKA users.

Fig. 8. Success rate of ABAKA simulating policies combination (a) on the Milan dataset. Each user forwards the message to a random neighbor; outside the cloaked area, user forwards the message to a random neighbor.



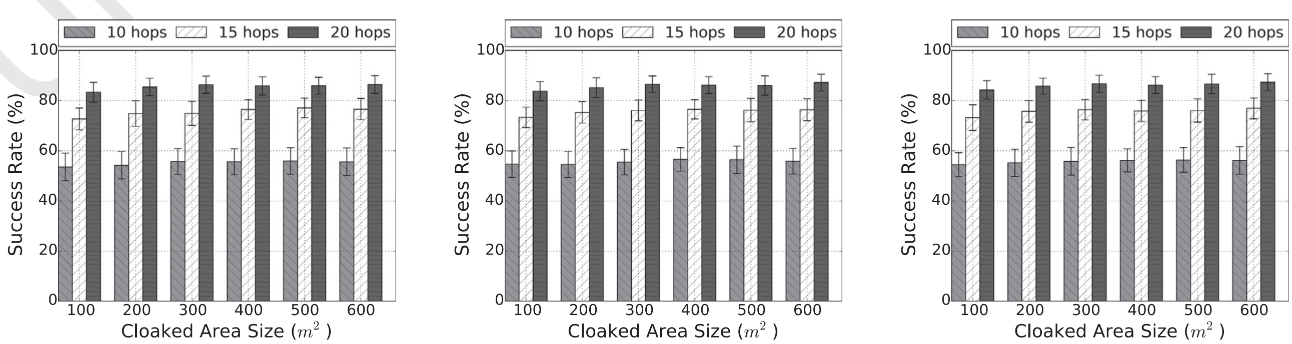
(a) 50% ABAKA users. (b) 60% ABAKA users. (c) 70% ABAKA users.

Fig. 9. Success rate of ABAKA simulating policies combination (a) on the New York dataset. Each user forwards the message to its closest neighbor; outside the cloaked area, user returns the message to previous user.



(a) 50% ABAKA users. (b) 60% ABAKA users. (c) 70% ABAKA users.

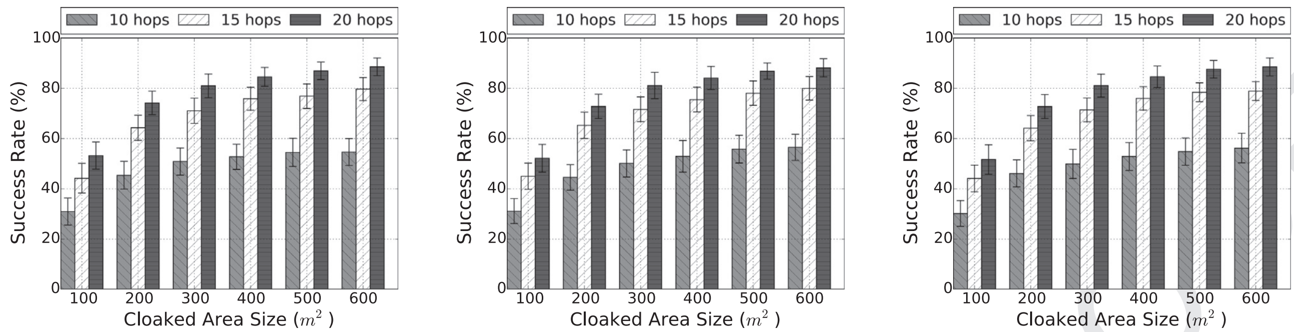
Fig. 10. Success rate of ABAKA simulating policies combination (a) on the New York dataset. Each user forwards the message to its closest neighbor; outside the cloaked area, user forwards the message to a random neighbor.



(a) 50% ABAKA users. (b) 60% ABAKA users. (c) 70% ABAKA users.

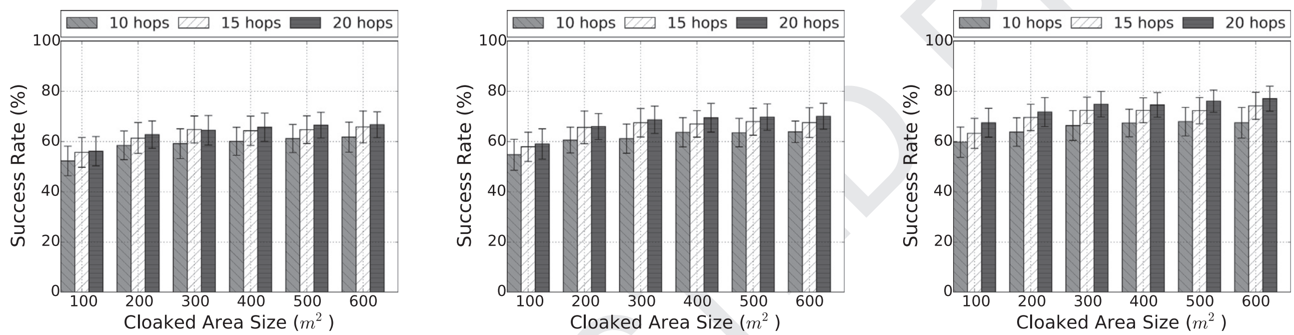
Fig. 11. Success rate of ABAKA simulating policies combination (a) on the New York dataset. Each user forwards the message to a random neighbor; outside the cloaked area, user returns the message to previous user.

Please cite this article as: T. Dargahi et al., ABAKA: A novel attribute-based k-anonymous collaborative solution for LBSs, Computer Communications (2016), http://dx.doi.org/10.1016/j.comcom.2016.03.002



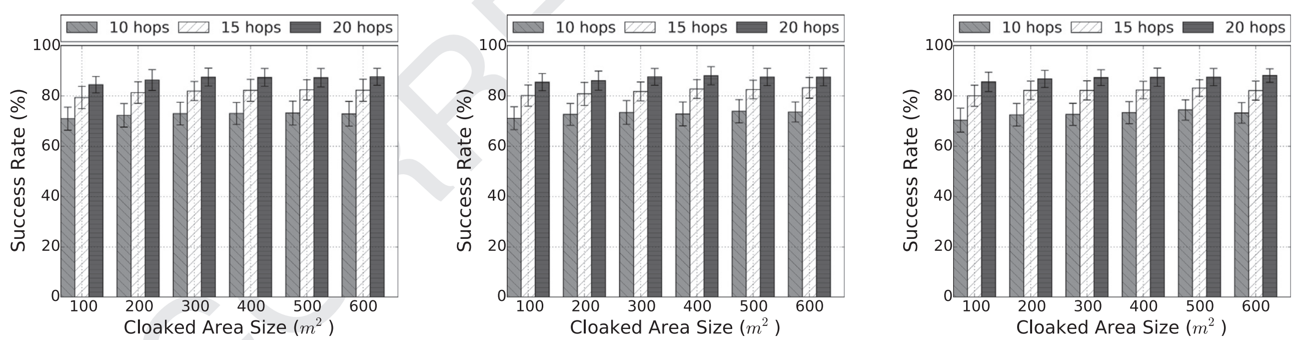
(a) 50% ABAKA users. (b) 60% ABAKA users. (c) 70% ABAKA users.

Fig. 12. Success rate of ABAKA simulating policies combination (a) on the New York dataset. Each user forwards the message to a random neighbor; outside the cloaked area, user forwards the message to a random neighbor.



(a) 50% ABAKA users. (b) 60% ABAKA users. (c) 70% ABAKA users.

Fig. 13. Success rate of ABAKA simulating policies combination (b) on the Milan dataset. Each user forwards the message to its closest neighbor; outside the cloaked area, user returns the message to previous user.



(a) 50% ABAKA users. (b) 60% ABAKA users. (c) 70% ABAKA users.

Fig. 14. Success rate of ABAKA simulating policies combination (b) on the New York dataset. Each user forwards the message to its closest neighbor; outside the cloaked area, user returns the message to previous user.

York dataset. For example, from Fig. 12 we can see that adopting a maximum number of hops of 20, brings the success rate of the protocol to greater than 90%, while a maximum of 10 hops leads to a success rate lower than 60%. Analogously, the effect of the adopted bigger maximum cloacked area size can be observed from Fig. 5 to Fig. 12; as an example, Fig. 5(a) shows that, with a maximum of 20 hops, a maximum cloacked area size of 100 m^2 leads to an average success rate of some 50%, while when the maximum cloacked area size is 600 m^2, the success rate is some 60% an average.

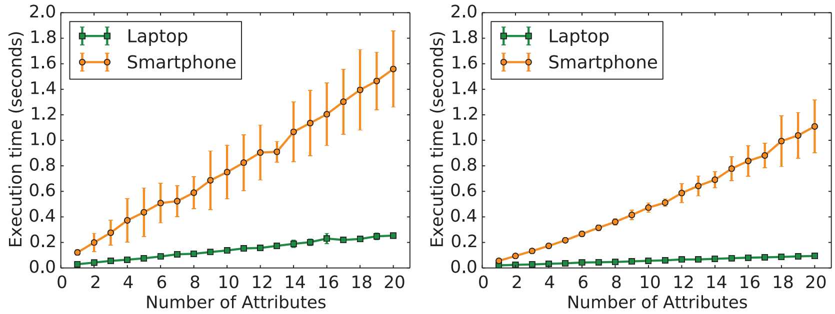
5.2. Cryptographic overhead

For a thorough evaluation of ABAKA, we estimated the overhead introduced by the cryptographic tools used in our protocol. In particular, we measured the average time required for encryption and decryption with CP-ABE, RSA, and AES-CBC. We considered two different platforms: a laptop equipped with 4x1.8 GHz Intel Core i7-4500U processor, and 8 GB RAM, running Ubuntu 14.04; and a smartphone equipped with a 1.2 GHz dual-core ARM Cortex-A9 CPU processor, and 1 GB RAM, running Android 4.3 “Jelly Bean”.

On both platforms, we evaluated CP-ABE using the ABE implementation for Android devices we proposed in [39][1]. Fig. 15 shows the results of our measurements on a 250 KB file (we believe that this is a reasonable size assumption for a piece of query encrypted in the protocol). Since the time required by CP-ABE mainly depends on the number of attributes employed in the cryptographic operations [12], we considered a varying number of attributes for policies and keys from one to 20.

As we can see from Fig. 15, even adopting a large number of attributes, the time required by CP-ABE implementation for encryption and decryption is low, on both smartphone and laptop.

[1] The code of the library is available at http://spritz.math.unipd.it/projects/andraben/



(a) Encryption (b) Decryption

Fig. 15. Average time required for encryption and decryption operations using CP-ABE on an Android smartphone and a Laptop device.

Table 4
Average encryption/decryption time for RSA/AES-CBC on Smartphone and Laptop.

Scheme	Smartphone		Laptop	
	Encrypt	Decrypt	Encrypt	Decrypt
RSA	7.5101 ms	0.0156 ms	0.153 ms	0.001 ms
AES-CBC*	26.199 ms	26.517 ms	2.809 ms	3.953 ms
AES-CBC**	110.179 ms	109.574 ms	11.072 ms	15.526 ms

* Encryption/decryption of a 250 KByte file.
** Encryption/decryption of a 1 MByte file.

For a more comprehensive overview of the performance of ABE on smartphone devices, the reader may refer to our recent work [39]. Additionally, we measured the average encryption and decryption time for RSA, with key size of 4096 bits, and AES-CBC with key size of 256 bits. On both platforms, we employed the openssl library [40], that we cross compiled for Android. We measured RSA encryption and decryption for a key of size 256 bits; while for AES-CBC, we considered a file of size 1 MB. Table 4 shows the results of our measurements. As we can see, for both RSA and AES-CBC, the imposed overhead is very small.

The results we obtained confirm the applicability of ABAKA not only on powerful devices such as laptops, but also on smartphone devices. As an example, consider an anonymity level $k = 5$, and policies composed by three attributes (which we believe are expressive enough to successfully guarantee p-sensitivity). In this case, the average overhead on an Android smartphone would be approximately $(0.27613 \times 5) + 0.00751 + 0.11018 = 1.49834$ s for the issuer, who has to encrypt the query with a symmetric key, that in turn is encrypted with LBSP's public key (this is a common usage of public key encryption), and encrypt each part of the split message with CP-ABE. Each collaborating user has to decrypt a part of the query with her CP-ABE private key, and immediately encrypt it with AES-CBS. Therefore, the approximate overhead will be $0.13275 + 0.26199 = 0.15894$ s. Finally, the last collaborating user have to decrypt all the parts that are previously encrypted with AES-CBC. Therefore, she will incur in an additional overhead of $0.02651 \times 5 = 0.13255$ s.

6. Related work

The concept of k-anonymity was first introduced for databases applications [41], and later applied in the context of LBSs [5]: the user's position is translated into a cloaked area and provided to the LBSP along with the requested query. The concept of k-anonymity has been extended in several aspects, e.g., *l-diversity* [42], and *t-closeness* [43]. Moreover, in [9] the authors proposed a p-sensitive approach for LBSs, which provides query l-diversity by classifying queries into sensitive and non-sensitive groups. However, unlike our work, none of these approaches considered both (i) query semantics, and (ii) sensitive profile attributes of each user, at the same time.

Bamba et al. [44] proposed an approach to provide k-anonymity and location l-diversity for LBS users. In this scheme, mobile users are not identifiable from $k-1$ other users in a set of l different physical locations such as hospitals, bars and university. This scheme utilizes one or more anonymization servers between users and LBSP to perform spatio-temporal cloaking.

In traditional approaches for k-anonymity in LBSs, the computation of the cloaked area is carried out by an *anonymization server* to which the query is first forwarded. Such solutions are typically referred as *TTP-based* schemes. However, the use of a centralized anonymizer offers a single point of attack, and may represent a serious bottleneck for the overall system. To overcome these limitations, researchers proposed several distributed solutions that compute the cloaked area in a collaborative way, referred to as *TTP-free* solutions. For an overview of the main existing TTP-free solutions, the reader can refer to [45].

Unfortunately, most of the existing schemes (both TTP-free and TTP-based) do not consider the background knowledge of the attackers, except from only a few recently proposed approaches [11]. However, an attacker with background information about a user's profile might be able to identify her, even if her location is hidden [46]. k-anonymity preserving solutions try to overcome the above issues, by considering user profiles information [6,47]. However, unlike our work, all the aforementioned profile-based schemes are centralized, and might be subject to the limitations introduced before. To the best of our knowledge, our proposal is the first TTP-free approach for p-sensitive profile k-anonymity in LBS that considers user's profile attributes.

7. Conclusions

Location and identity privacy in Location-Based Services are major concerns for users who want to protect their privacy from a malicious LBSP, as well as from an eavesdropper. While several solutions for guaranteeing privacy in LBSs have been proposed in the literature, they are often centralized, or do not take into account the prior knowledge of the attacker about user profiles. In this paper we present ABAKA, our collaborative solution that guarantees k-anonymity, as well as p-sensitivity in LBSs, taking into account the issued query semantics. In our approach, users have a set of attributes associated to their profile. Their attributes are bound to a CP-ABE private key. An LBS message is first processed by the issuer, and then forwarded through a multi-hop route to the LBSP. ABAKA enables each issuer to delimit a cloaked area within which she wants to be anonymous, and to specify a list of $k-1$ policies, i.e., attribute combinations, that users in the multi-hop path must satisfy in order to forward the query message to the LBSP. ABAKA provides the possibility of performing a trade-off between the stringency of privacy protection and quality of service for the issuer in her current location, based on the query semantics. We addressed the threat of active and passive adversaries by means of CP-ABE and multi-hop routing approaches. We simulated our protocol on synthetic datasets derived from real population statistics (considering two cities: New York (USA), and Milan (Italy)), and demonstrated that our approach is feasible and efficient.

Acknowledgments

Mauro Conti is supported by a Marie Curie Fellowship funded by the European Commission (agreement PCIG11-GA-2012-321980). This work is also partially supported by the EU TagItSmart! Project (agreement H2020-ICT30-2015-688061), the EU-India REACH Project (agreement ICI+/2014/342-896), the Italian MIUR-PRIN TENACE Project (agreement 20103P34XC), and by the projects “Tackling Mobile Malware with Innovative Machine Learning Techniques”, “Physical-Layer Security for Wireless Communication”, and “Content Centric Networking: Security and Privacy Issues” funded by the University of Padua.

References

[1] H.A. Karimi, Advanced Location-based Technologies and Services, CRC Press, 2013.
[2] M. Wernke, P. Skvortsov, F. Dürr, K. Rothermel, A classification of location privacy attacks and approaches, Person. Ubiquitous Comput. 18 (1) (2014) 163–175.
[3] M. Conti, J. Willemsen, B. Crispo, Providing source location privacy in wireless sensor networks: a survey, IEEE Commun. Surv. Tutor. 15 (3) (2013) 1238–1280.
[4] Y. Zhu, D. Ma, D. Huang, C. Hu, Enabling secure location-based services in mobile cloud computing, in: Proceedings of the Second ACM SIGCOMM Workshop on Mobile Cloud Computing, MCC'13, 2013, pp. 27–32.
[5] M. Gruteser, D. Grunwald, Anonymous usage of location-based services through spatial and temporal cloaking, in: Proceedings of the 1st International Conference on Mobile Systems, Applications and Services, MobiSys'03, 2003, pp. 31–42.
[6] H. Shin, J. Vaidya, V. Atluri, A profile anonymization model for location-based services, J. Comput. Secur. 19 (5) (2011) 795–833.
[7] T.M. Truta, B. Vinay, Privacy protection: p-sensitive k-anonymity property, in: Proceedings of the 22nd IEEE International Conference on Data Engineering, ICDE'06, 2006, p. 94.
[8] X. Xiao, Y. Tao, Personalized privacy preservation, in: Proceedings of the ACM SIGMOD international conference on Management of data, SIGMOD'06, ACM, 2006, pp. 229–240.
[9] Z. Xiao, J. Xu, X. Meng, p-sensitivity: A semantic privacy-protection model for location-based services, in: Proceedings of the 9th IEEE International Conference on Mobile Data Management Workshops, MDMW '08, IEEE, 2008, pp. 47–54.
[10] A. Solanas, F. Sebé, J. Domingo-Ferrer, Micro-aggregation-based heuristics for p-sensitive k-anonymity: one step beyond, in: Proceedings of the 2008 International Workshop on Privacy and Anonymity in Information Society, PAIS'08, 2008, pp. 61–69.
[11] R. Shokri, G. Theodorakopoulos, P. Papadimitratos, E. Kazemi, J.-P. Hubaux, Hiding in the mobile crowd: location privacy through collaboration, IEEE Trans. Depend. Secure Comput. 11 (3) (2014) 266–279.
[12] J. Bethencourt, A. Sahai, B. Waters, Ciphertext-policy attribute-based encryption, in: Proceedings of the IEEE Symposium on Security and Privacy, S&P'07, 2007, pp. 321–334.
[13] A. Sahai, B. Waters, Fuzzy identity-based encryption, in: Advances in Cryptology – EUROCRYPT 2005, Lecture Notes in Computer Science, 3494, 2005, pp. 457–473.
[14] T. Yang, C. Tang, L. Yu, W. Xin, Y. Deng, J. Hu, Z. Chen, VLSP: enabling location privacy in vehicular location based services, in: Proceedings of the 2011 First International Conference on Instrumentation, Measurement, Computer, Communication and Control, MCCC'11, 2011, pp. 462–465.
[15] C. Boldrini, M. Conti, F. Delmastro, A. Passarella, Context-and social-aware middleware for opportunistic networks, J. Netw. Comput. Appl. 33 (5) (2010) 525–541.
[16] H. Nishiyama, M. Ito, N. Kato, Relay-by-smartphone: realizing multihop device-to-device communications, IEEE Commun. Mag. 52 (4) (2014) 56–65.
[17] M. Conti, F. Delmastro, G. Minutiello, R. Paris, Experimenting opportunistic networks with wifi direct, in: Wireless Days, WD'13, IEEE, 2013, pp. 1–6.
[18] E. Biondi, C. Boldrini, A. Passarella, M. Conti, Optimal duty cycling in mobile opportunistic networks with end-to-end delay guarantees, in: Proceedings of the 20th European Wireless Conference, European Wireless'14, VDE, 2014, pp. 1–6.
[19] C.A. Ardagna, M. Conti, M. Leone, J. Stefa, An anonymous end-to-end communication protocol for mobile cloud environments, IEEE Transactions on Services Computing 7 (3) (2014) 373–386.
[20] X. Bao, Y. Lin, U. Lee, I. Rimac, R.R. Choudhury, Dataspotting: Exploiting naturally clustered mobile devices to offload cellular traffic, in: Proceedings of the IEEE International Conference on Computer Communications, INFOCOM'13, IEEE, 2013, pp. 420–424.
[21] M. Conti, L. Zhang, S. Roy, R. Di Pietro, S. Jajodia, L.V. Mancini, Privacy-preserving robust data aggregation in wireless sensor networks, Secur. Commun. Netw. 2 (2) (2009) 195–213.
[22] H. Gao, C.H. Liu, W. Wang, J. Zhao, Z. Song, X. Su, J. Crowcroft, K.K. Leung, A survey of incentive mechanisms for participatory sensing, Commun. Surv. Tutor., IEEE 17 (2) (2015) 918–943.
[23] Q. Li, G. Cao, Providing privacy-aware incentives for mobile sensing, in: Proceedings of the International Conference on Pervasive Computing and Communications, PerCom'13, IEEE, 2013, pp. 76–84.
[24] M. Conti, C. Boldrini, S.S. Kanhere, E. Mingozzi, E. Pagani, P.M. Ruiz, M. Younis, From manet to people-centric networking: milestones and open research challenges, Comput. Commun. 71 (2015) 1–21.
[25] J. Li, Q. Huang, X. Chen, S.S.M. Chow, D.S. Wong, D. Xie, Multi-authority ciphertext-policy attribute-based encryption with accountability, in: Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security, ASIACCS'11, 2011, pp. 386–390.
[26] R. Shokri, J. Freudiger, J.-P. Hubaux, A unified framework for location privacy, Technical Report, 2010.
[27] C.A. Ardagna, A. Stavrou, S. Jajodia, P. Samarati, R. Martin, A multi-path approach for k-anonymity in mobile hybrid networks, in: Proceedings of the International Workshop on Privacy in Location-Based Applications, PILBA'08, 2008, pp. 82–101.
[28] H. Takabi, J.B. Joshi, H. Karimi, et al., A collaborative k-anonymity approach for location privacy in location-based services, in: Proceedings of the 5th International Conference on Collaborative Computing: Networking, Applications and Worksharing, CollaborateCom'09, 2009, pp. 1–9.
[29] X. Gong, X. Chen, K. Xing, D.-H. Shin, M. Zhang, J. Zhang, Personalized location privacy in mobile networks: a social group utility approach, in: INFOCOM'15, IEEE, 2015, pp. 1008–1016.
[30] G. Zhong, U. Hengartner, Toward a distributed k-anonymity protocol for location privacy, in: Proceedings of the 9th Annual ACM Workshop on Privacy in the Electronic Society, WPES'08, 2008, pp. 33–38.
[31] G. Zhong, U. Hengartner, A distributed k-anonymity protocol for location privacy, in: IEEE International Conference on Pervasive Computing and Communications, PerCom'09, 2009, pp. 1–10.
[32] M.G. Reed, P.F. Syverson, D.M. Goldschlag, Anonymous connections and onion routing, IEEE J. Select. Areas Commun. 16 (4) (1998) 482–494.
[33] D. Rebollo-Monedero, J. Forné, A. Solanas, A. Martínez-Ballesté, Private location-based information retrieval through user collaboration, Comput. Commun. 33 (6) (2010) 762–774.
[34] United states census bureau - quick facts, http://quickfacts.census.gov/qfd/states/36/36061.html.
[35] Focus on milan, 2012, http://allegati.comune.milano.it/Statistica/AnnuariStatistici/MilanoInBreve2012/FocusOnMilano2012.pdf.
[36] Pew Research Center – U.S. Smartphone Use in 2015, http://www.pewinternet.org/2015/04/01/us-smartphone-use-in-2015/.
[37] Lo scenario social, digital e mobile in europa e in italia, http://www.wired.it/internet/social-network/2014/02/17/lo-scenario-social-digital-e-mobile-europa-e-italia/.
[38] M. Gielen, Ad hoc networking using wi-fi during natural disasters: overview and improvements, in: Proceedings of the 17th Twente Student Conference on IT, TSConIT '12, Vol. 17, 2012.
[39] M. Ambrosin, M. Conti, T. Dargahi, On the feasibility of attribute-based encryption on smartphone devices, in: Proceedings of the 2015 Workshop on IoT Challenges in Mobile and Industrial Systems, IoT-Sys'15: MobiSys'15 workshop, 2015, pp. 49–54.
[40] OpenSSL library., https://www.openssl.org/.
[41] P. Samarati, L. Sweeney, Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression, Technical Report, SRI International, 1998.
[42] A. Machanavajjhala, D. Kifer, J. Gehrke, M. Venkitasubramaniam, L-diversity: privacy beyond k-anonymity, ACM Trans. Knowl. Discov. Data 1 (1) (2007) 3.
[43] N. Li, T. Li, S. Venkatasubramanian, t-closeness: privacy beyond k-anonymity and l-diversity., in: Proceedings of the 23rd IEEE International Conference on Data Engineering, ICDE'07, 2007, pp. 106–115.
[44] B. Bamba, L. Liu, P. Pesti, T. Wang, Supporting anonymous location queries in mobile environments with privacygrid, in: Proceedings of the 17th International Conference on World Wide Web, WWW'08, 2008, pp. 237–246.
[45] A. Khoshgozaran, C. Shahabi, A taxonomy of approaches to preserve location privacy in location-based services, Int. J. Comput. Sci. Eng. 5 (2) (2010) 86–96.
[46] R. Shokri, C. Troncoso, C. Diaz, J. Freudiger, J.-P. Hubaux, Unraveling an old cloak: K-anonymity for location privacy, in: Proceedings of the 9th Annual ACM Workshop on Privacy in the Electronic Society, WPES'10, 2010, pp. 115–118.
[47] X. Chen, J. Pang, Protecting query privacy in location-based services, GeoInformatica 18 (1) (2014) 95–133.

Please cite this article as: T. Dargahi et al., ABAKA: A novel attribute-based k-anonymous collaborative solution for LBSs, Computer Communications (2016), http://dx.doi.org/10.1016/j.comcom.2016.03.002