An Efficient Mobile Location-based Security Service Framework for Resource-constrained Devices

Yinghui Zhang[1,2], Xinwei Ma[1], Axin Wu[3], Fangyuan Ren[1], and Dong Zheng[1,2]
(Corresponding author: Yinghui Zhang)
School of Cyberspace Security, Xi'an University of Posts and Telecommunications[1]
Xi'an 710121, China
Westone Cryptologic Research Center, Beijing 100070, China[2]
College of Cybersecurity, Jinan University, Guangzhou 510632, China[3]
(Email: yhzhaang@163.com)
(Received June 5, 2019; Revised and Accepted Sept. 4, 2019; First Online Sept. 21, 2019)

Abstract

In this paper, we propose a mobile location-based security service framework supporting Cloud-side Matching and Preliminary Decryption, which is called CMPD. A user only needs to send part of his attribute key to the cloud server and doesn't need any computation in the matching phase. Then the cloud service provider (CSP) tests whether the attributes of a user meet the corresponding access policy. After passing the matching phase, the user needs to send his attribute key to the CSP for decryption. The CSP sends the result of the computation to the user after having completed the preliminary decryption. The user can get the plaintext through a few decryption operations. CMPD neither leakages the attributes and geographic coordinate of the user, nor reveals the information of access policies. Besides, CMPD achieves the confidentiality of the mobile location-based service data. Finally, security and performance analysis show that our proposed framework has achieved privacy protection, data security and efficiency improvement for resource-constrained mobile devices.

Keywords: Attribute-based Encryption; Cloud Computing; Fast Decryption; Mobile Location-based Service; Privacy Protection

1 Introduction

Thanks to the progress of cloud storage and computing, numerous cloud service providers have emerged, i.e., Huawei Cloud, Microsoft Azure, Google Cloud, Amazon Web Services and Tencent Cloud. They can provide cloud services to the government, companies, and individuals. Cloud service users can send their data files to the cloud server.

The CSP can not only store the data files from users but also compute and process it. Mobile location-based service (MLBS) has been further developed under the technology of cloud storage and computing. MLBS includes two phases. First, the MLBS provider receives geographic coordinate from users. Mobile devices can get geographic coordinate through GPS, mobile networks and WIFI. Second, the MLBS provider will provide kinds of information services according to the geographic coordinate.

For example, finding friends around your location on the chat software, finding nearby restaurants and hotels, checking in at a meeting or conference. However, more and more people are concerned about cloud security [1] and care about security and privacy of the data. When MLBS data files are outsourced to the cloud server, the MLBS provider hopes that no one can access the files except the authorized users. Besides, flexible access control and fine-grained framework are required because MLBS is a model [9] which provides data services. Some location-based service frameworks [5, 14] employ attribute-based encryption to solve the above problems.

Anonymous ciphertext-policy attribute-based encryption (CP-ABE) was proposed in [10]. Data owners can formulate an access policy and encrypt the plaintext according to the access policy using the encryption algorithm. Access policy is encrypted in the ciphertext. A person can obtain the plaintext from the encrypted ciphertext when the attributes of this person meet the access policy and this person cannot guess what the access policy is. However, it is very complex for the resource-constraint mobile devices, because heavy computation is needed during the decryption in CP-ABE algorithm. In existing ABE algorithms, a user can only know whether he or she has permission to access data after fully decrypting the ciphertext. Hence, the existing ABE algorithms

is a bit inefficient, because even when a user does not have an access right, he or she can know the fact after decrypting the ciphertext. However, most mobile devices are resource-constraint, the above ABE algorithms are not suitable for these devices.

Recently, Zhang *et al.* [27] proposed a scheme of anonymous CP-ABE, which includes a matching phase and a decryption phase. The proposed scheme is an improved one of the preliminary version [26]. The user's attributes first need to satisfy access policy in the match phase and then the user can perform the decryption phase. However, these two phases are performed on the user side. To further reduce the computing cost of resource-constraint mobile devices and improve the efficiency of MLBS, we modify and innovate the basic scheme of Zhang *et al.* [27] and then apply it to MLBS. In our scheme, the match and preliminary decryption are operated and computed by the CSP. Because of the application of MLBS, we add a geographic coordinate match in the match phase. Users can complete decryption with only a few computation operations.

In this paper, we propose a mobile location-based security service framework supporting Cloud-side Matching and Preliminary Decryption, which is called CMPD. The following are our contributions to this paper.

In our framework, which is shown in Figure 1, the user only needs to send part of his attribute key which is used to match and geographic coordinate to the cloud server and doesn't need any computation in the match phase. Then the cloud service provider tests whether the attributes of a certain person meet the corresponding access policy. After passing the match phase, the user needs to send an attribute key which is used to decrypt to the cloud server. Then the CSP sends the result of the computation to the user after having completed the preliminary decryption. The user can get the plaintext through a few decryption operations.

To make our CP-ABE more suitable for MLBS, we add a geographic coordinate attribute to the algorithm, which is used to enable users to obtain corresponding MLBS. The proposed framework neither leakages the attributes and geographic coordinate of the user, nor reveals the information of access policy. Besides, the proposed framework achieves the confidentiality of the mobile location-based service data. We guarantee data security, user privacy security and access policy security. At the same time, our proposed framework improves the efficiency of the scheme and reduces the computation cost for the resource-constraint mobile devices.

Now, we introduce the structure of the rest of our paper. We describe the related work in the next section. We describe the system model, security model and design goals in Section 3. Some preliminaries are given in Section 4. We introduce our CP-ABE scheme in Section 5. We introduce our CMPD framework in Section 6. Finally, we analyze the security and performance of our CMPD framework in Sections 7 and 8 respectively.

2 Related Work

Since Sahai *et al.* [20] proposed the ABE scheme, many types of research have been studied on the schemes of various ABE. This cryptographic algorithm has two forms, where key-policy ABE (KP-ABE) and key-ciphertext (CP-ABE). KP-ABE and CP-ABE were both proposed in [8]. In the KP-ABE scheme, the access policy is connected with the private key and hidden in it. In the CP-ABE scheme, the access policy is connected with ciphertext and hidden in it. Goyal *et al.* [8] first presented the KP-ABE algorithm and the algorithm realized the goal of monotonic access policy structure. A flexible access policy was presented by Ostrovsky *et al.* [18] which realizes the goal of non-monotonic access policy structure.

The first CP-ABE scheme was presented by Bethencourt *et al.* [2]. He only proved secure of the scheme with the condition of a generic group model. A more secure scheme was presented by Cheung *et al.* [7] which is proved secure with the condition of the standard model. CP-ABE [2] encryption allows a person to establish an access policy and the plaintext is encrypted according to it. Encryption could be completed if the attributes meet the access policy. The CP-ABE algorithm could realize a fine-grained access control framework. Lee *et al.* [12] proposed a new convertible encryption scheme based on the ElGamal algorithm. To protect attribute privacy and renew access policy, Zhang *et al.* [28] proposed an anonymous CP-ABPRE framework, in which a matching phase is added before the proxy re-encryption phase. Liu *et al.* [17] proposed an online and offline CP-ABE scheme in an electronic health record system. Zhang *et al.* [29] proposed an attribute-based data sharing system which realizes the function of offline key generation and encryption. Li *et al.* [13] proposed a lightweight protocol without public-key encryption and decryption. Zhang *et al.* [30] proposed a policy-hiding CP-ABE scheme and used it to design smart health security and privacy system. Based on the behavior of the receiver, Cao *et al.* [6] analyzed attribute-based encryption. Attribute-based encryption evolved from identity-based encryption. Liu *et al.* [16] improved an anonymous identity-based encryption scheme by removing one decrypting helper and the strong simulator.

Khoshgozaran *et al.* [11] presented a privacy-preserving location-based service (LBS) framework which needs a trusted institution to convert the original geographic coordinate into a new space. Avoiding a trusted institution keeping the location privacy, Paulet *et al.* [19] presented a novel method in which location privacy is obtained by retrieving private information. In order to avoid a large amount of computing consumption of mobile terminals, Lien *et al.* [15] presented a private circular query protocol that solves the problem of privacy and accuracy for privacy-preserving LBS. However, when the number of points becomes enormous, the scheme isn't appropriate for resource-constraint mobile Internet of Things devices. Jung *et al.* [14] proposed a privacy-preserving LBS scheme

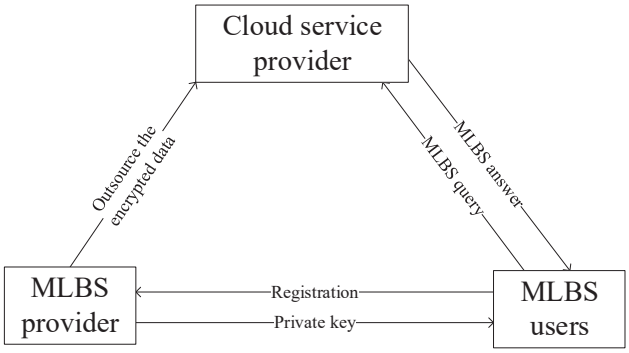

Figure 1: The system model of our proposed framework

according to the ABE algorithm, which doesn't keep the access policy secret. Yu *et al.* [25] first presented a framework that achieves fine-grained access control, scalability and data security in cloud computing. Wang *et al.* [22] presented a framework which achieves practicability and privacy protection search in the cloud environment. But the scheme is not suitable for LBS. Ye *et al.* [24] presented a novel location privacy-preserving framework according to l-queries for continuous LBS using the Paillier public-key cryptosystem. Zhao *et al.* [31] presented a secure and highly efficient LBS framework in which users can retrieve information related to the current geographic coordinate without revealing the geographic coordinate privacy to the CSP. Shao *et al.* [21] presented an LBS framework for mobile devices according to the CP-ABE algorithm, which realizes a fine-grained scheme and privacy-preserving.

3 Model and Design Goals

3.1 System Model

Our system model is the same as the model [9] which provides data services. The system model of our proposed CMPD framework consists of three parts: the MLBS provider, plenty of MLBS users and the CSP which are shown in Figure 1. The following is a detailed description of these three parts.

MLBS provider computes the system public key and system master key. MLBS provider also produces the MLBS data files which have the information of MLBS. The MLBS data files are sent to the cloud server after being encrypted by the MLBS provider. The MLBS users register at the MLBS provider and obtain the attribute private keys from it.

The MLBS users are resource-constraint who want to obtain the MLBS according to his or her attribute private key and geographic coordinate. And our proposed framework doesn't reveal any information about attributes and geographic coordinate.

The cloud service provider storages and computes the data files outsourced from the MLBS provider. We suppose that the cloud service provider could store a large amount of data files and perform fast computing and is always online.

3.2 Security Model

In our proposed framework, we assume that the CSP is honest and curious which is stated in [21]. Specifically, the cloud service provider will faithfully obey our proposed framework, but it can attack the framework as much as possible to get private information. The CSP may try to obtain the plaintext or access policy of the encrypted MLBS data files and the geographic coordinate of the MLBS user by colluding malicious users. But the cloud service provider will store data correctly, perform computing correctly, and send data to the users correctly. The users would access the data files without permission or obtaining the access rights of data files. And the users may attack the framework independently or cooperatively to obtain private information.

3.3 Design Goals

The design goals of our proposed framework are as follows.

1) MLBS data files should be secret for anyone who doesn't have access permissions. Unauthorized MLBS users cannot obtain any information from the ciphertext of the encrypted MLBS data files.

2) Our proposed framework should have the capacity of the collusion-resistance in which the cloud service provider and the MLBS users may collude to obtain the information from the ciphertext of the encrypted MLBS data files when the attributes of a user don't meet the access policy.

3) The privacy of the MLBS users and MLBS data should be protected, where anyone including the cloud service provider cannot obtain any information of user's attributes, geographic coordinate and access policy, even when the cloud service provider is in the match and preliminary decryption phase.

4) High Efficient match and preliminary decryption are needed, because of resource-constraint mobile devices. Since the match and preliminary decryption phases are at the cloud, users only need a small and constant computation before obtaining the plaintext.

5) The CSP should send accurate MLBS data files to the user after it received the user's query which is composed of the geographic coordinate and part of the attribute private key.

6) A flexible access control and fine-grained framework are required because MLBS is a model that provides data services, where data files with different access policies should be decrypted by the MLBS users with different attributes.

4 Preliminaries

In this section, we briefly describe some concepts of cryptogram used in our paper, including the bilinear pairing and some complexity assumptions and access structure.

4.1 Bilinear Pairings

We assume that $\mathbb{G}$ and $\mathbb{G}_T$ are two multiplicative cyclic groups which have a same large prime order p. Assume the generator of $\mathbb{G}$ is g. Suppose the identity of $\mathbb{G}_T$ is $1_{\mathbb{G}_T}$. There is a bilinear pairing if $\mathbf{e} : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is a map which satisfies the following properties:

1) Bilinear: $\mathbf{e}(\beta_1^a, \beta_2^b) = \mathbf{e}(\beta_1, \beta_2)^{ab}$ for all $\beta_1, \beta_2 \in \mathbb{G}$ and $a, b \in \mathbb{Z}_p$.

2) Non-degenerate: There exists $\beta_1, \beta_2 \in \mathbb{G}$ such that $\mathbf{e}(\beta_1, \beta_2) \neq 1_{\mathbb{G}_T}$.

3) Computable: There exists an efficient algorithm to calculate $\mathbf{e}(\beta_1, \beta_2)$ for all $\beta_1, \beta_2 \in \mathbb{G}$.

4.2 Complexity Assumptions

The Discrete Logarithm (DL) assumption: The DL assumption is right if for any probabilistic polynomial-time (PPT) algorithm, we can obtain the a from the β^a with non-negligible advantage, where unknown a is randomly selected from $\mathbb{Z}_p$ and β is randomly selected from $\mathbb{G}$.

The Decisional Diffie-Hellman (DDH) assumption: The DDH assumption is right if for any PPT algorithm, we can make a distinction between the tuple

$$[\beta, \beta^a, \beta^b, \beta^{ab}]$$

and the tuple

$$[\beta, \beta^a, \beta^b, \beta^z]$$

with non-negligible advantage, where a, b, z are randomly selected from $\mathbb{Z}_p$ and β is randomly selected from $\mathbb{G}$.

The Decisional Bilinear Diffie-Hellman (DBDH) assumption: The DBDH assumption [4] is right if for any PPT algorithm, we can make a distinction between the tuple

$$[\beta, \beta^a, \beta^b, \beta^c, \mathbf{e}(\beta, \beta)^{abc}]$$

and the tuple

$$[\beta, \beta^a, \beta^b, \beta^c, \beta^z]$$

with non-negligible advantage, where a, b, c, z are randomly selected from $\mathbb{Z}_p$ and β is randomly selected from $\mathbb{G}$.

The Decisional Linear Diffie-Hellman (D-Linear) assumption: The D-Linear assumption [3] is right if for all any PPT algorithm, we can make a distinction between the tuple

$$[\beta, \beta^{z_1}, \beta^{z_2}, \beta^{z_1 z_3}, \beta^{z_2 z_4}, \beta^{z_3 + z_4}]$$

and the tuple

$$[\beta, \beta^{z_1}, \beta^{z_2}, \beta^{z_1 z_3}, \beta^{z_2 z_4}, \beta^{z}]$$

with non-negligible advantage, where z, z_1, z_2, z_3, z_4 are randomly selected from $\mathbb{Z}_p$ and β is randomly selected from $\mathbb{G}$.

4.3 Access Structure

Suppose all the users' attributes set is $\mathbb{U} = \{U_1, U_2, \ldots, U_n\}$ in universe, where $|\mathbb{U}| = n$. Each attribute U_i has n_i multiple values, where $U_i = \{u_{i,1}, u_{i,2}, \ldots, u_{i,n_i}\}$ for $1 \leq i \leq n$. We assume the $L = [l_1, l_2, \ldots, l_n]$ is a user's attribute list. In our anonymous CP-ABE scheme, we suppose the access policy structure is a single AND-gate which supports multiple values and wildcards. Formally, there are a user's attribute list $L = [l_1, l_2, \ldots, l_n]$ and a access policy $W = [W_1, W_2, \ldots, W_n]$. For $1 \leq i \leq n$, $L \vDash W$ if $l_i \in W_i$ or $W_i = *$, otherwise $L \nvDash W$. The symbol $\vDash$ and $\nvDash$ mean that L meets or doesn't meet W respectively. And the wildcard $*$ means that the multiple values in this attribute of W is inconsequential.

For example, we assume that the attributes set is $\mathbb{U} = \{U_1, U_2, \ldots, U_6\}$ and the access policy is $W = [u_{1,2}, u_{2,4}, *, u_{4,1}, u_{5,4}, *]$ which the attributes U_3 and U_6 are inconsequential. The attributes satisfies access policy if the user has the multiple values $u_{1,2}$ for U_1, $u_{2,4}$ for U_2, $u_{4,1}$ for U_4, $u_{5,4}$ for U_5 and no matter what the attributes U_3 and U_6 are.

5 Our Anonymous CP-ABE

In this section, we describe our cloud-side matching and preliminary decryption scheme for anonymous CP-ABE, which is improved from the basic scheme of Zhang *et al.* [27]. The scheme protects the user's attributes of privacy and improves decryption efficiency for the resource-constraint mobile devices. More concretely, our scheme has two phases which need the cloud service provider and users to participate in.

In particular, the user needs to send part of the attribute private key and geographic coordinate to the cloud server and the final decryption cost is small and constant, no matter how many attributes exist and how complex the access policy is.

5.1 Proposed Scheme

Our anonymous CP-ABE scheme is as follows.

Setup (1^λ): We assume that $\mathbb{G}$ and $\mathbb{G}_T$ are two multiplicative cyclic groups which have a same large

prime order p and $\mathbf{e} : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is a bilinear map. Assume the generator of $\mathbb{G}$ is g. Suppose the identity of $\mathbb{G}$ is $1_{\mathbb{G}}$. We define a hash function $H : \{0,1\}^* \to \mathbb{G}$. Suppose all the users' attributes set is $\mathbb{U} = \{U_1, U_2, \ldots, U_n\}$ in universe, where $|\mathbb{U}| = n$. And each attribute U_i has n_i multiple values, where $U_i = \{u_{i,1}, u_{i,2}, \ldots, u_{i,n_i}\}$ for $1 \le i \le n$. Then the $\mathbb{U} = \{u_{i,1}, u_{i,2}, \ldots, u_{i,n_i}\}$ is a set containing all the multiple values. And only MLBS provider knows the index order of elements in the set $\mathbb{U}$ which should be secret. MLBS randomly chooses $y, \in \mathbb{Z}_p$ and $\beta_1, \beta_2, \beta_3, \beta_4 \in \mathbb{G}$. Then MLBS provider computes $Y = \mathbf{e}(\beta_1, \beta_2)^y$. The system public key is $PK =< H, \beta_3, \beta_4 >$ and the system master key is $MK =< y, \beta_1, \beta_2 >$. MLBS provider public the multiple values of all attributes, but the index order of each multiple value in the $\mathbb{U} = \{u_{i,1}, u_{i,2}, \ldots, u_{i,n_i}\}$ is secret.

KeyGen (PK, MK, L): After obtaining a certain user's attributes, MLBS provider makes the user's attribute list $L = [l_1, l_2, \ldots, l_n]$ according to the index order of elements in $\mathbb{U}$. MLBS provider randomly chooses $r_1, r_2, \ldots, r_{n-1} \in \mathbb{Z}_p$ and computes $r_n = y - \sum_{i=1}^{n-1} r_i \mod p$. MLBS randomly chooses $r, \lambda, \hat{\lambda} \in \mathbb{Z}_p$ and $\hat{r}_1, \hat{r}_2, \ldots, \hat{r}_n \in \mathbb{Z}_p$ and computes $\hat{r} = \sum_{i=1}^{n} \hat{r}_i$. MLBS provider randomly chooses $a \in \mathbb{Z}_p$ and then computes $A = \mathbf{e}(\beta_1, \beta_2)^a$. MLBS provider computes $D_{\Delta,0} = \beta_1^r, \hat{D}_{\Delta,0} = \beta_2^{y-\hat{r}}, D_0 = \beta_2^{a\lambda}, \hat{D}_0 = \beta_1^{a\hat{\lambda}}$.

For $1 \le i \le n$, we assume the indexes satisfy $l_i = u_{i,t}$, MLBS provider computes

$$\begin{aligned} D_{i,t,\Delta} &= \beta_2^{\hat{r}_i} H(i \| u_{i,t})^r, \\ D_{i,t} &= \beta_1^{ar_i} H(0 \| i \| u_{i,t})^{a\lambda}, \\ \hat{D}_{i,t} &= \beta_2^{ar_i} H(1 \| i \| u_{i,t})^{a\hat{\lambda}} \end{aligned}$$

Then the attribute private key is $PK_L =< A, D_{\Delta,0}, \hat{D}_{\Delta,0}, D_0, \hat{D}_0, \{D_{i,t,\Delta}, D_{i,t}, \hat{D}_{i,t}\}_{1 \le i \le n} >$.

Encrypt (PK, M, W): MLBS provider encrypts a message $M \in \mathbb{G}_T$ under a ciphertext access policy $W = [W_1, W_2, \ldots, W_n]$. MLBS provider randomly chooses $s, s^{'}, s^{''} \in \mathbb{Z}_p$ and computes $C = MY^s, C_\Delta = Y^{s^{'}}, \hat{C}_0 = \beta_1^{s^{'}}, C_1 = \beta_2^{s^{''}}, \hat{C}_1 = \beta_1^{s-s^{''}}$.

Then for $1 \le i \le n$ and $1 \le t \le n_i$, MLBS provider randomly chooses $\kappa_{i,\Delta}, \kappa_{i,0}, \kappa_{i,1}$ such that $\prod_{i=1}^{n} \kappa_{i,\Delta} = \prod_{i=1}^{n} \kappa_{i,0} = \prod_{i=1}^{n} \kappa_{i,1} = 1_{\mathbb{G}}$, and computes $C_{i,t,\Delta}, C_{i,t}, \hat{C}_{i,t}$ as follows:

1) If $u_{i,t} \in W_i$ or $W_i = *$, then MLBS provider computes

$$\begin{aligned} C_{i,t,\Delta} &= \kappa_{i,\Delta} H(i \| u_{i,t})^{s^{'}}, \\ C_{i,t} &= \kappa_{i,0} H(0 \| i \| u_{i,t})^{s^{''}}, \\ \hat{C}_{i,t} &= \kappa_{i,1} H(1 \| i \| u_{i,t})^{s-s^{''}} \end{aligned}$$

2) If $u_{i,t} \notin W_i$, then MLBS provider randomly chooses $C_{i,t,\Delta}, C_{i,t}, \hat{C}_{i,t} \in \mathbb{G}$.

For simplicity, we denote $[L_x^{file}, L_y^{file}]$ as geographic coordinate of MLBS data files which have the information of location services. Then MLBS provider randomly chooses $z, z^{'} \in \mathbb{Z}_p$ computes $C_{n+1} = \beta_3^{z^{'}}, C_{n+1}^{'} = \beta_4^{z-z^{'}}$ and $C^{'} = \mathbf{e}(\beta_3, H(0\|\beta_3))^{z^{'} L_x^{file}} \cdot \mathbf{e}(\beta_4, H(1\|\beta_4))^{(z-z^{'}) L_y^{file}}$.

Then the ciphertext of M is $CT_W =< C, C^{'}, C_{n+1}, C_{n+1}^{'}, C_\Delta, \hat{C}_0, C_1, \hat{C}_1, \{\{C_{i,t,\Delta}, C_{i,t}, \hat{C}_{i,t}\}_{1 \le t \le n_i}\}_{1 \le i \le n} >$ which is encrypted by MLBS provider under the access policy W.

Decrypt (PK, CT_W, PK_L): The ciphertext CT_W is matched and preliminarily decrypted by cloud service provider. Then the final decryption is performed by the user. Process is as follow:

1) Cloud-side matching phase: The user computes $D_{n+1} = H(0\|\beta_3)^{L_x^{user}}, D_{n+1}^{'} = H(1\|\beta_4)^{L_y^{user}}$, where L_x^{user} and L_y^{user} are the user's geographic coordinate obtained from his or her mobile devices, i.e., mobile phone or smart watch. Then the user sends $D_{n+1}, D_{n+1}^{'}$ and part of the attribute private key PK_L except A to the cloud service provider.

   $L \vDash W$ if and only if

$$\mathbf{e}(\hat{C}_0, \hat{D}_{\Delta,0} \cdot \prod_{i=1}^{n} D_{i,t,\Delta}) = C_\Delta \cdot \mathbf{e}(\prod_{i=1}^{n} C_{i,t,\Delta}, D_{\Delta,0}) \quad (1)$$

   and

$$C^{'} = \mathbf{e}(D_{n+1}, C_{n+1}) \cdot \mathbf{e}(D_{n+1}^{'}, C_{n+1}^{'}). \quad (2)$$

   Otherwise, $L \nvDash W$. The cloud service provider chooses the $C_{i,t,\Delta}$ according to the indexes $[i,t]$ of the $D_{i,t,\Delta}$.

2) Cloud-side preliminary decryption phase: If $L \vDash W$, then cloud service provider computes

$$C^p = \frac{\mathbf{e}(C_1, \prod_{i=1}^{n} D_{i,t}) \cdot \mathbf{e}(\hat{C}_1, \prod_{i=1}^{n} \hat{D}_{i,t})}{\mathbf{e}(\prod_{i=i}^{n} C_{i,t}, D_0) \cdot \mathbf{e}(\prod_{i=1}^{n} \hat{C}_{i,t}, \hat{D}_0)}. \quad (3)$$

   Then the CSP sends C^p and C to the user. Similarly, the CSP chooses the $C_{i,t}$ and $\hat{C}_{i,t}$ according to the indexes $[i,t]$ of the $D_{i,t}$.

3) Final decryption: The user can obtain the plaintext M after computes

$$\frac{C \cdot A}{C^p} \to M. \quad (4)$$

5.2 Consistency of Proposed Scheme

Consistency of Formula (1):

$$\begin{aligned}
&\frac{\mathbf{e}(\hat{C}_0, \hat{D}_{\Delta,0} \cdot \prod_{i=1}^{n} D_{i,t,\Delta})}{\mathbf{e}(\prod_{i=1}^{n} C_{i,t,\Delta}, D_{\Delta,0})} \\
= &\frac{\mathbf{e}(\beta_1^{s'}, \beta_2^{y-\hat{r}} \cdot \prod_{i=1}^{n} \beta_2^{\hat{r}_i} H(i\|u_{i,t})^r)}{\mathbf{e}(\prod_{i=1}^{n} \kappa_{i,\Delta} H(i\|u_{i,t})^{s'}, \beta_1^r)} \\
= &\mathbf{e}(\beta_1^{s'}, \beta_2^{y-\hat{r}} \cdot \prod_{i=1}^{n} \beta^{\hat{r}_i}) \\
= &\mathbf{e}(\beta_1^{s'}, \beta_2^{y-\hat{r}} \cdot \beta^{\hat{r}}) \\
= &\mathbf{e}(\beta_1^{s'}, \beta_2^{y}) \\
= &C_\Delta
\end{aligned}$$

Consistency of Formula (2):

If the following equation holds:

$$\begin{aligned}
&\mathbf{e}(D_{n+1}, C_{n+1}) \cdot \mathbf{e}(D'_{n+1}, C'_{n+1}) \\
= &\mathbf{e}(H(0\|\beta_3)^{L_x^{user}}, \beta_3^{z'}) \cdot \mathbf{e}(H(1\|\beta_4)^{L_y^{user}}, \beta_4^{z-z'}) \\
= &\mathbf{e}(\beta_3, H(0\|\beta_3))^{z' L_x^{user}} \cdot \mathbf{e}(\beta_4, H(1\|\beta_4))^{(z-z')L_y^{user}} \\
= &C'
\end{aligned}$$

It means that the user's geographic coordinate is equal to the MLBS data file's.

Consistency of Formula (3):

$$\begin{aligned}
C^p = &\frac{\mathbf{e}(C_1, \prod_{i=1}^{n} D_{i,t}) \cdot \mathbf{e}(\hat{C}_1, \prod_{i=1}^{n} \hat{D}_{i,t})}{\mathbf{e}(\prod_{i=i}^{n} C_{i,t}, D_0) \cdot \mathbf{e}(\prod_{i=1}^{n} \hat{C}_{i,t}, \hat{D}_0)} \\
= &\frac{\mathbf{e}(\beta_2^{s''}, \prod_{i=1}^{n} \beta_1^{ar_i} H(0\|i\|u_{i,t})^{a\lambda})}{\mathbf{e}(\prod_{i=1}^{n} \kappa_{i,0} H(0\|i\|u_{i,t})^{s''}, \beta_2^{a\lambda})} \\
&\cdot \frac{\mathbf{e}(\beta_1^{s-s''}, \prod_{i=1}^{n} \beta_2^{ar_i} H(1\|i\|u_{i,t})^{a\hat{\lambda}})}{\mathbf{e}(\prod_{i=1}^{n} \kappa_{i,1} H(1\|i\|u_{i,t})^{s-s''}, \beta_1^{a\hat{\lambda}})} \\
= &\mathbf{e}(\beta_2^{s''}, \prod_{i=1}^{n} \beta_1^{ar_i}) \cdot \mathbf{e}(\beta_1^{s-s''}, \prod_{i=1}^{n} \beta_2^{ar_i}) \\
= &\mathbf{e}(\beta_1, \beta_2)^{y \cdot s \cdot a}
\end{aligned}$$

Consistency of Formula (4):

$$\frac{C \cdot A}{C^p} = \frac{MY^s \cdot \mathbf{e}(\beta_1, \beta_2)^a}{\mathbf{e}(\beta_1, \beta_2)^{y \cdot s \cdot a}} = \frac{MY^s}{Y^s} = M.$$

6 Our CMPD Framework

In this section, we describe our mobile location-based service framework supporting Cloud-side Matching and Preliminary Decryption. Now, we give the description of our CMPD framework as follows.

System Initialization: The MLBS provider chooses a security parameter λ with which a system public key PK and a system master key MK are generated by running the algorithm **Setup** (1^λ). Then the MLBS provider publics the PK and keeps MK secret.

MLBS Data Files Encryption: The MLBS provider produces MLBS data files and then encrypts the data files as follows.

The MLBS provider chooses a symmetric encryption algorithm and randomly chooses a secret key py from the key space. Different MLBS data files could have different secret keys. Then the MLBS provider encrypts the MLBS data file with the py using the symmetric encryption algorithm. We denote the ciphertext as C^{file} .

The MLBS provider defines an access policy W for an MLBS data file. Different MLBS data files could have different access policies. Then the MLBS provider encrypts the secret key py by running the algorithm **Encrypt** (PK, py, W). We denote the ciphertext as CT_W.

Finally, the format of each MLBS data file is (C_{file}, CT_W) as shown in Table 1. Then the MLBS provider outscores (C_{file}, CT_W) to the CSP.

Table 1: Format of a MLBS data file

MLBS data file	CT_W	C_{file}

Users Registration: The MLBS user who wants to obtain mobile location-based service sends his or her attributes to the MLBS provider. After running the algorithm **KeyGen** (PK, MK, L), the MLBS provider distributes corresponding the attribute private key PK_L to the MLBS user.

Users Query: We assume the geographic coordinate range within which the user wants to obtain MLBS is $[L_x^{R-}, L_x^{R+}]$ and $[L_y^{R-}, L_y^{R+}]$ as shown in Figure 2. There are many MLBS data files in this range. The MLBS user can obtain the corresponding MLBS data file if

$$L_x^{R-} \leq (L_x^{user} - L_x^{file}) \leq L_x^{R+}$$

and

$$L_y^{R-} \leq (L_y^{user} - L_y^{file}) \leq L_y^{R+}.$$

When an MLBS user who has registered in the MLBS provider wants to obtain mobile location-based service, the user needs to send D_{n+1}, D'_{n+1}, $[L_x^{R-}, L_x^{R+}]$, $[L_y^{R-}, L_y^{R+}]$ and part of attribute private key PK_L except A to the cloud service provider.

Cloud-Side Matching and Preliminary Decryption: After receiving the user's query, the cloud service provider does the following computation:

$$C_\Delta^{test} = \frac{\mathbf{e}(\hat{C}_0, \hat{D}_{\Delta,0} \cdot \prod_{i=1}^{n} D_{i,t,\Delta})}{\mathbf{e}(\prod_{i=1}^{n} C_{i,t,\Delta}, D_{\Delta,0})}$$

and for each value $R_x \in [L_x^{R-}, L_x^{R+}]$ and $R_y \in [L_y^{R-}, L_y^{R+}]$,

$$C'_{test} = \mathbf{e}(D_{n+1} \cdot H(0\|\beta_3)^{R_x}, C_{n+1})$$

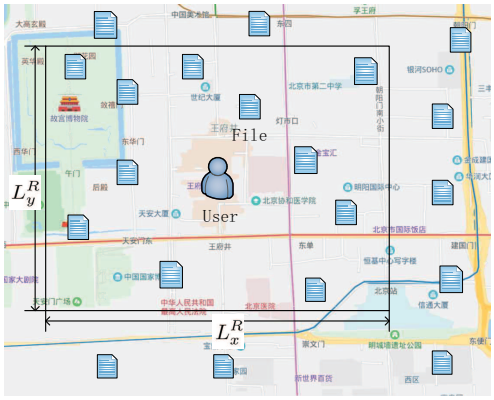

Figure 2: The MLBS geographic coordinate range

$$\cdot \mathbf{e}(D'_{n+1} \cdot H(1\|\beta_4)^{R_y}, C'_{n+1}).$$

The MLBS user can obtain corresponding MLBS data file if and only if $C_{\Delta}^{test} = C_{\Delta}$ and $C'_{test} = C'$. Otherwise, the user cannot obtain any MLBS.

If the matching phase is successful, then the CSP does the preliminary decryption phase. Finally, the CSP sends C^p, C and C_{file} of corresponding MLBS data file to the user.

Getting MLBS: The MLBS user can obtain the secret key pk after computing $\frac{C \cdot A}{C^p} \rightarrow py$. Then the MLBS user can obtain corresponding MLBS after using py to decrypt C_{file}.

7 Security Analysis

In this section, we describe the security of our proposed CMPD framework.

The confidentiality of MLBS data: In our proposed CMPD framework, we first encrypt the MLBS data file using a symmetric encryption algorithm. Then we encrypt the secret key py using our CP-ABE algorithm. Only the MLBS user knows the transformation key A. As analyzed in [21], the cloud service provider transforms our CP-ABE ciphertext into ElGamal type ciphertext according to the transformation key A. The cloud service provider can obtain no information about the access policy without knowing A, because of the security of ElGamal type ciphertext. Hence, no one except the user can obtain information about plaintext from the corresponding encrypted MLBS data file.

The privacy of access policy and user's attributes: We analyze the privacy of access policy in our proposed CMPD framework by comparing it with the basic anonymous CP-ABE scheme presented by Zhang *et al.* [27], which is proved the selective ciphertext-policy and chosen-plaintext secure (CPA-secure) under the condition of the DBDH assumption and the D-Linear assumption.

Our anonymous CP-ABE partial ciphertext components

$$< C_{\Delta}, \hat{C}_0, C_1, \hat{C}_1, \{\{C_{i,t,\Delta}, C_{i,t}, \hat{C}_{i,t}\}_{1 \le t \le n_i}\}_{1 \le i \le n} >$$

and partial attribute private key components

$$< D_{\Delta,0}, \hat{D}_{\Delta,0}, \{D_{i,t,\Delta}\}_{1 \le i \le n} >$$

are computed the same as the basic anonymous CP-ABE scheme presented by Zhang *et al.* [27]. As analyzed in [27], these above components don't reveal any information of the access policy. The matching and preliminary decryption phases of our framework perform at the cloud service provider, the corresponding components of ciphertext are chosen based on the indexes $[i,t]$ of the attribute private key. Since the index order of each multiple value in the $\mathbb{U} = \{u_{i,1}, u_{i,2}, \ldots, u_{i,n_i}\}$ is secret, people who only know the indexes $[i,t]$ don't know what the corresponding multiple value $u_{i,t}$ is. Since MLBS provider keeps β_1, β_2 secret, our proposed framework avoids the combination of CSP and users guessing access policies and user's attributes. Hence, our framework don't reveal any information of the access policy and user's attributes.

The confidentiality of geographic coordinate: The geographic coordinate L_x^{file}, L_y^{file} of the MLBS data file are in the formula

$$\mathbf{e}(\beta_3, H(0\|\beta_3))^{z' L_x^{file}}, \mathbf{e}(\beta_4, H(1\|\beta_4))^{(z-z') L_y^{file}}.$$

Hence, the information of L_x^{file}, L_y^{file} cannot be revealed even in the matching phase, because of the complexity assumptions. Similarly, the geographic coordinate L_x^{user}, L_y^{user} of the MLBS user are in the formula $H(0\|\beta_3)^{L_x^{user}}, H(1\|\beta_4)^{L_y^{user}}$ and the information of L_x^{file}, L_y^{file} cannot be revealed even in the matching phase, because of the complexity assumptions. Hence, our framework cannot reveal any information of geographic coordinate whether it belongs to MLBS data files or MLBS users.

8 Performance Analysis

In this section, we compare our proposed framework with anther three LBS frameworks [21, 23, 32] in terms of efficiency.

Table 2 mainly introduces the efficiency comparison including PK size, pairings in decryption, whether there is a matching phase and the computation on the user-side. We denote the bit length of an element in $\mathbb{G}$ and $\mathbb{G}_T$ as $|\mathbb{G}|$ and $|\mathbb{G}_T|$ respectively. We denote by $\mathcal{M}$ a multiplication operation, by $\mathcal{D}$ a division operation, by $\mathcal{E}$ an exponentiation operation, by $\mathcal{H}$ a hash operation, by $\mathcal{P}$ a public key cryptographic algorithm operation and by

Table 2: The efficiency comparison of LBS schemes

Schemes	Fine-grained	Pairings in decryption	Decryption matching	User-side computation
[21]	✓	$8N+2$	×	$2\mathcal{M}+\mathcal{D}+3\mathcal{E}+2\mathcal{H}$
[32]	×	8	×	$2\mathcal{M}+7\mathcal{E}+3\mathcal{H}+2\mathcal{P}+2\mathcal{B}$
[23]	✓	8	×	$(\|S_x\|+1)\mathcal{M}+4\mathcal{D}+3\mathcal{B}$
Our	✓	8	✓	$\mathcal{M}+\mathcal{D}+2\mathcal{E}+2\mathcal{H}$

$\mathcal{B}$ a bilinear pairing operation. In [23], S_x is a set of child nodes and we denote the size of S_x as $|S_x|$. We assume $N=\sum_{i=1}^{n} n_i$.

Our scheme has a matching phase before the preliminary decryption phase in the cloud, which reduces unnecessary computational consumption and improves the efficiency of decryption. In our scheme, the users don't need to compute bilinear pairings, which reduces the computational cost for resource-constraint mobile devices. It means that the user only needs to do some simple calculations during the query MLBS and the final decryption phase. A detailed comparison is shown in the Table 2. In summary, our scheme is more efficient than another.

9 Conclusion

In this paper, we propose a mobile location-based security service framework supporting Cloud-side Matching and Preliminary Decryption. The proposed framework takes advantage of cloud storage and cloud computing and ensures data confidentiality, access policy privacy, user's attribute privacy and geographic coordinate privacy. In particular, the computation cost of terminals is reduced significantly.

Acknowledgments

This research is supported by the National Key R&D Program of China under Grant 2017YFB0802000, the Innovation Capability Support Program of Shaanxi under Grant 2020KJXX-052, the Shaanxi Special Support Program Youth Top-notch Talent Program, the Key Research and Development Program of Shaanxi under Grants 2019KW-053 and 2020ZDLGY08-04, the Natural Science Basic Research Plan in Shaanxi Province of China under Grants 2019JQ-866 and 2018JZ6001. Yinghui Zhang is supported by New Star Team of Xi'an University of Posts and Telecommunications under Grant 2016-02.

References

[1] M. Ali, S. U. Khan, and A. V. Vasilakos, "Security in cloud computing: Opportunities and challenges," *Information Sciences*, vol. 305, pp. 357–383, 2015.

[2] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *IEEE Symposium on Security and Privacy (SP'07)*, pp. 321–334, 2007.

[3] D. Boneh, X. Boyen, and H. Shacham, "Short group signatures," in *Annual International Cryptology Conference*, pp. 41–55, 2004.

[4] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in *Annual International Cryptology Conference*, pp. 213–229, 2001.

[5] D. Boneh and B. Waters, "Conjunctive, subset, and range queries on encrypted data," in *Theory of Cryptography Conference*, pp. 535–554, 2007.

[6] Z. Cao, L. Liu, and Z. Guo, "Ruminations on attribute-based encryption," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 9–19, 2018.

[7] L. Cheung and C. Newport, "Provably secure ciphertext policy ABE," in *Proceedings of the 14th ACM Conference on Computer and Communications Security*, pp. 456–465, 2007.

[8] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM Conference on Computer and Communications Security*, pp. 89–98, 2006.

[9] H. Hu, Q. Chen, and J. Xu, "Verdict: Privacy-preserving authentication of range queries in location-based services," in *IEEE 29th International Conference on Data Engineering (ICDE'13)*, pp. 1312–1315, 2013.

[10] A. Kapadia, P. P. Tsang, and S. W. Smith, "Attribute-based publishing with hidden credentials and hidden policies," in *NDSS*, vol. 7, pp. 179–192, 2007.

[11] A. Khoshgozaran and C. Shahabi, "Blind evaluation of nearest neighbor queries using space transformation to preserve location privacy," in *International Symposium on Spatial and Temporal Databases*, pp. 239–257, 2007.

[12] C. C. Lee, M. S. Hwang, and S. F. Tzeng, "A new convertible authenticated encryption scheme based on the elgamal cryptosystem," *International Journal of Foundations of Computer Science*, vol. 20, no. 02, pp. 351–359, 2009.

[13] C. T. Li and M. S. Hwang, "A lightweight anonymous routing protocol without public key en/decryptions for wireless ad hoc networks," *Information Sciences*, vol. 181, no. 23, pp. 5333–5347, 2011.

[14] X. Y. Li and T. Jung, "Search me if you can: Privacy-preserving location query service," in *Proceedings IEEE Infocom*, pp. 2760–2768, 2013.

[15] I. T. Lien, Y. H. Lin, J. R. Shieh, and J. L. Wu, "A novel privacy preserving location-based service protocol with secret circular shift for k-nn search," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 6, pp. 863–873, 2013.

[16] L. Liu, Z. Guo, Z. Cao, and Z. Chen, "An improvement of one anonymous identity-based encryption scheme," *International Journal of Electronics and Information Engineering*, vol. 9, no. 1, pp. 11–21, 2018.

[17] Y. Liu, Y. Zhang, J. Ling, and Z. Liu, "Secure and fine-grained access control on e-healthcare records in mobile cloud computing," *Future Generation Computer Systems*, vol. 78, pp. 1020–1026, 2018.

[18] R. Ostrovsky, A. Sahai, and B. Waters, "Attribute-based encryption with non-monotonic access structures," in *Proceedings of the 14th ACM Conference on Computer and Communications Security*, pp. 195–203, 2007.

[19] R. Paulet, M. G. Kaosar, X. Yi, and E. Bertino, "Privacy-preserving and content-protecting location based queries," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 5, pp. 1200–1210, 2013.

[20] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 457–473, 2005.

[21] J. Shao, R. Lu, and X. Lin, "Fine: A fine-grained privacy-preserving location-based service framework for mobile devices," in *IEEE Conference on Computer Communications*, pp. 244–252, 2014.

[22] C. Wang, K. Ren, S. Yu, and K. M. R. Urs, "Achieving usable and privacy-assured similarity search over outsourced cloud data," in *Proceedings IEEE Infocom*, pp. 451–459, 2012.

[23] Y. Xue, J. Hong, W. Li, K. Xue, and P. Hong, "Labac: A location-aware attribute-based access control scheme for cloud storage," in *IEEE Global Communications Conference (GLOBECOM'16)*, pp. 1–6, 2016.

[24] A. Ye, Y. Li, and L. Xu, "A novel location privacy-preserving scheme based on l-queries for continuous LBS," *Computer Communications*, vol. 98, pp. 1–10, 2017.

[25] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in *2010 Proceedings IEEE Infocom*, pp. 1–9, 2010.

[26] Y. Zhang, X. Chen, J. Li, D. S. Wong, and H. Li, "Anonymous attribute-based encryption supporting efficient decryption test," in *Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security*, pp. 511–516, 2013.

[27] Y. Zhang, X. Chen, J. Li, D. S. Wong, H. Li, and I. You, "Ensuring attribute privacy protection and fast decryption for outsourced data security in mobile cloud computing," *Information Sciences*, vol. 379, pp. 42–61, 2017.

[28] Y. Zhang, J. Li, X. Chen, and H. Li, "Anonymous attribute-based proxy re-encryption for access control in cloud computing," *Security and Communication Networks*, vol. 9, no. 14, pp. 2397–2411, 2016.

[29] Y. Zhang, A. Wu, and D. Zheng, "Efficient and privacy-aware attribute-based data sharing in mobile cloud computing," *Journal of Ambient Intelligence and Humanized Computing*, vol. 9, no. 4, pp. 1039–1048, 2018.

[30] Y. Zhang, D. Zheng, and R. H. Deng, "Security and privacy in smart health: Efficient policy-hiding attribute-based access control," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 2130–2145, 2018.

[31] X. Zhao, H. Gao, L. Li, H. Liu, and G. Xue, "An efficient privacy preserving location based service system," in *IEEE Global Communications Conference*, pp. 576–581, 2014.

[32] H. Zhu, R. Lu, C. Huang, L. Chen, and H. Li, "An efficient privacy-preserving location-based services query scheme in outsourced cloud," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 9, pp. 7729–7739, 2015.

Biography

Yinghui Zhang is a professor of National Engineering Laboratory for Wireless Security (NELWS), Xi'an University of Posts & Telecommunications since 2018. He has published over 80 research articles in ACM ASIACCS, IEEE Transactions on Dependable and Secure Computing, IEEE Transactions on Services Computing, Computer Networks, IEEE Internet of Things Journal, Computers & Security, IEEE Transactions on Industrial Informatics, etc. His research interests include public key cryptography, cloud security and wireless network security.

Xinwei Ma received the B.S. degree from the Northwestern Polytechnical University Mingde College in 2017. He is currently pursuing the M.Eng. degree with the Xi'an University of Post and Telecommunications, Xi'an, China. His research interests include attribute-based encryption.

Axin Wu received B.S. degree from Zhengzhou University of Light Industry in 2016, and M.Eng. degree from Xi'an University of Post and Telecommunications in 2019. Since then, he is currently in Ph.D program in Jinan University, Guangzhou, China. His research interests include cloud security and wireless network security.

Fangyuan Ren received the B.S. degree from the Xi'an University of Post and Telecommunications in 2017. She is currently pursuing the M.Eng. degree with the Xi'an University of Post and Telecommunications, Xi'an, China.

His research interests include authentication in 5G.

Dong Zheng received his Ph.D. degree in communication engineering from Xidian University, China, in 1999. He was a Professor at the School of Information Security Engineering, Shanghai Jiao Tong University. He is currently a Professor at National Engineering Laboratory for Wireless Security, Xi'an University of Posts & Telecommunications. He has published over 100 research articles including CT-RSA, IEEE Transactions on Industrial Electronics, Information Sciences, etc. His research interests include cloud computing security, public key cryptography.