Enhancing Location Privacy through P2P Network and Caching in Anonymizer

Peiqian Liu, Shangchen Xie, Zihao Shen and Hui Wang*
College of Computer Science and Technology, Henan Polytechnic University
Henan Jiaozuo 454000 China
[e-mail: liupeiqian@hpu.edu.cn; xcartoon@126.com; szh@hpu.edu.cn; wanghui_jsj@hpu.edu.cn;]
*Corresponding author: Hui Wang

Received January 16, 2022; revised March 21, 2022; accepted April 24, 2022; published May 31, 2022

Abstract

The fear that location privacy may be compromised greatly hinders the development of location-based service. Accordingly, some schemes based on the distributed architecture in peer-to-peer network for location privacy protection are proposed. Most of them assume that mobile terminals are mutually trusted, but this does not conform to realistic scenes, and they cannot make requirements for the level of location privacy protection. Therefore, this paper proposes a scheme for location attribute-based security authentication and private sharing data group, so that they trust each other in peer-to-peer network and the trusted but curious mobile terminal cannot access the initiator's query request. A new identifier is designed to allow mobile terminals to customize the protection strength. In addition, the caching mechanism is introduced considering the cache capacity, and a cache replacement policy based on deep reinforcement learning is proposed to reduce communications with location-based service server for achieving location privacy protection. Experiments show the effectiveness and efficiency of the proposed scheme.

Keywords: Caching Mechanism, Deep Reinforcement Learning, Location Privacy Protection, Location-Based Service, Peer-to-Peer Network

http://doi.org/10.3837/tiis.2022.05.013 **ISSN : 1976-7277**

1. Introduction

With the remarkable development of wireless communication technology and space positioning technology, services that can be available by mobile terminals are becoming rich diversity. Location-based service(LBS) is one of the representative examples [1]. That is, the mobile terminal only needs to send the current location and request about the location that it wants to query to LBS server which will return the corresponding query results. For instance, during the COVID-19 pandemic, communication big data travel card based on this service make it easier for people to verify whether they have been in contact with positive cases and whether they have been to medium and high risk areas [2]. However, there are concerns about location privacy. If the query request is intercepted by malicious eavesdroppers, the most intuitive damage is the leakage of location information. With further consideration of timestamps and the correlations between different query requests from the same mobile terminal, implicit propensity such as living habits and health conditions will also be tapped. However, due to its specific features, even if users are aware of the disadvantages, they still cannot stop using it. Therefore, location privacy protection in LBS has aroused widespread attention, and many schemes have been proposed one after another.

Gruteser and Grunwald introduced the k-anonymity algorithm, which makes one record indistinguishable from at least $k-1$ other records when publishing data in the relational database, to location privacy protection in LBS and then proposed the concept of location k-anonymity [3]. That makes it difficult for an attacker to identify the mobile terminal which sends the location-dependent query request in a region that contains at least k mobile terminals. In addition, the region containing k mobile terminals is called the cloaked region. In the centralized architecture, a trusted anonymizer (aka trusted third party, TTP) is introduced between the mobile terminal and LBS server. It regionalizes the exact location in the request and forwards it to LBS server.

Yet with centralized architecture, TTP suffers from performance bottlenecks and large-scale deployment is very difficult. Sequentially, the distributed architecture begins to emerge, where the mobile terminal communicates with each other based on common communication protocols, using single-hop and multi-hop rules to find collaborative peers, forming an anonymous set, and then generating the cloaked region. On that basis, to deal with the different responsibilities of the initiator and the collaborative peer, Chow et al. proposed the CloakP2P with the on-demand mode and the proactive mode [4]. Liu et al. proposed a distributed negotiation algorithm that enables mobile terminals to fully participate in the anonymity process [5]. At the same time, some schemes began to employ the caching mechanism. Because the more records about query requests in LBS server, the greater the chance of an attacker uncovering sensitive information. It can reduce the number of connections between the mobile terminal and LBS server as much as possible to achieve location privacy protection. In addition, it can also improve the whole performance.

However, there are some drawbacks in these schemes. Distributed architecture, in practice, is unable to effectively guarantee that collaborative peers are all trusted. Therefore, the scheme cannot safely assume that they trust each other. The mobile terminal cannot customize the level of location privacy protection to generate the suitable cloaked region. In densely populated or sparsely populated areas, depending on k alone, it usually generates the cloaked region in P2P network that is too small or too large for the mobile terminal. Moreover, when considering the caching mechanism, cache capacity is a factor that cannot be

overlooked.

As a result, this paper proposes a scheme that could tackle these disadvantages. In the proposed scheme, collaborative peers of the initiator when enjoying LBS have registered and authenticated with CA-like server using their location, and are secure and trusted. Since messages are shared in the same P2P network, the concept of private sharing data group is proposed. Faced with the trusted but curious mobile terminal, the identity of requesting collaborative peers and the requested content can be further protected. The mobile terminal generates the identifier according to designed identifier scheme that includes wanted protection level and rough location information, The negotiation of collaborative peers performs based on the identifier, and eventually, all collaborative peers are with the same level of location privacy protection, which subsequently forms the cloaked region with k-anonymity. The query request will send to anonymizer to match whether the corresponding candidate result set is cached at first. If there is, this request will not be recorded in LBS server. And, using a deep reinforcement learning-based replacement strategy that can adapt to dynamic environments, location privacy protection is enhanced by improving the hit rate of cached query results. The contributions of this paper can be summarized as follows:

- A method of secure authentication based on location is proposed to make mobile terminals mutually trustworthy, and private sharing data group is proposed to prevent the casual sharing of messages in P2P network.
- An identifier scheme that allows customizing the level of location privacy protection is proposed so that the mobile terminal can form the suitable cloaked region.
- The caching mechanism is introduced, and an efficient replacement strategy based on deep reinforcement learning is proposed to improve the hit rate within the limited cache capacity, thereby reducing the communications with LBS server.

The remainder of the paper is organized as follows. In section 2, we introduce some related works. The preliminaries and the overview of the proposed scheme are presented in section 3. In section 4, we describe the details. Section 5 shows our experimental results and the last section draws the conclusion.

2. Related Work

There are lots of schemes that show us how to protect location privacy in LBS. This section summarizes schemes in relation to the proposed scheme.

P2P network, which could provide relay forwarding functions for all participants, thus providing users with better privacy protection [6]. CloakP2P is a typical scheme using k-anonymity to generate the cloaked region in P2P network [4]. But the region is much larger than the actual need, the attacker can identify the initiator with a probability much larger than $1/k$. To make the generated clocked region have stronger protection, Sui et al. proposed RPS algorithm which randomly and dynamically generates the region based on the number of mobile terminals, and adds noise to each region with a particular probability [7]. It will lead to low quality of service, and to balance the two, Zhang et al. proposed to scale polygon in the cloaked region and randomize fake locations according to density distribution [8]. But it neglected to consider the geographical semantic information about fake locations, which will be vulnerable to location semantic attack. On this basis, Zhang et al. proposed a scheme that considers semantic information between real locations and fake locations, and the scheme improves the efficiency of making dummies [9]. However, this process is often accompanied by the problems of high communication overhead and low success rate. With the help of distribution information and density information in the vicinity of mobile terminals, Xu et al.

proposed a scheme that achieves lower communication expenses and faster finding collaborative peers [10]. Li et al. proposed a client-server-to-user model to reduce overhead based on the principle of Geohash coding and Voronoi graph for the partitioning region in Internet of Things [11]. But it needs to be implemented in the situation where collaborative peers and P2P network are all fully trusted. Therefore, it has great limitations in real scenarios. Jagdale and Bakal used hash functions and multilayer protocols to improve the security of P2P network, but they do not address important access control problems [12]. Sen et al. created an assisted reputation manager named R-TTP for the initiator to judge collaborative peers when sending a query request and adopted fog computing to improve the robustness and protection [13]. A federated learning scheme supported by support vector machine and random forest was proposed to detect and remove collaborative peers and exclude malicious eavesdroppers from joining [14]. For the correlation of time, identifier, and location attributes in query requests, Ashraf et al. proposed a model called "Improved Dummy Position" that protects them simultaneously [15]. Li et al. investigated the effects of using machine learning to improve security and privacy in LBS, showing that machine learning is one of the most promising new technologies to address such challenges [16].

Caching can reduce the number of requests for service queries, thereby reducing the chances of the attacker discovering sensitive information [17]. Amini et al. introduced a caching framework in LBS. Before reaching a specific area, the mobile terminal will send a pre-query to LBS server to obtain the candidate result sets. Therefore, the mobile terminal can run locally without further querying [18]. Sequentially, according to the mobility of the mobile terminal, Zhang et al. used the Markov chain to predict the next position rather than the large area and then generated fake locations based on the predicted position and cached candidate result sets [19]. Aiming at the accuracy and privacy of candidate result sets, Yamin and Sen proposed a scheme to address it [20]. Alaradi and Innab proposed a safe cycle-based approach and combined it with caching to retrieve future query results utilizing bloom filter-based search, thereby improving the overall performance [21]. Cui et al. proposed a cache-based scheme that could protect both query privacy and location privacy named CBPP, in which each mobile terminal acts as a mini trusted third-party server [22]. Zhang et al. designed a converter that maps location to a uniform grid structure and combines it with caching mechanism to achieve location privacy protection, which is named UGC [23].

3. Preliminary and System Architecture

In this section, we first introduce the prerequisite knowledge involved in the proposed scheme, namely Kademlia algorithm and DQN algorithm, and then describe the location privacy protection architecture respecting P2P network and cache replacement strategy.

3.1 Kademlia

Many prevalent P2P applications adopt the decentralized architecture named Kademlia [24]. In Kademlia, each node is comprised of a unique identifier ID facilitating to locate and a routing table remembering the information about ID, IP address, and network port number of others. Similarly, each file has an identifier with the same length as the node. It will be stored in those nodes whose ID has the shortest distance to its ID. Incidentally, exclusive or operation is used to measure the distance between two identifiers.

According to the definition of Kademlia, there are W buckets and the i^{th} bucket ($0 \leq i < W$) of the node U will record N other node information whose distance to U is in

$[2^i, 2^{i+1})$. Here, W is the bitwidth of node ID and N is the system parameter which is usually small. Meanwhile, owing to the limitations of storage space and privacy, not all node information will be stored. In general, the closer to the node, the greater the possibility of being stored.

3.2 DQN

DQN proposed by DeepMind is the combination of deep neural network (DNN) and Q-Learning [25]. It mainly consists of neural network model and environment that includes state space, action space, and reward mechanism controlling the reward value. As shown in **Fig. 1**, the duty of DQN is to select the next action based on the current state and network parameters and then the chosen action updates environment. The new environment and corresponding reward will be returned after environment is updated.

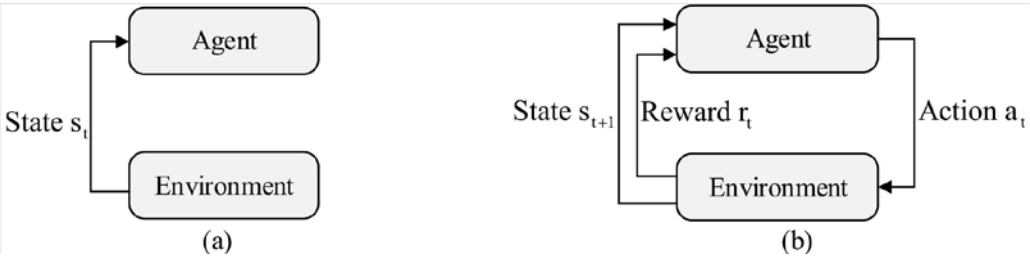

Fig. 1. DQN interaction process

Typically, temporal difference (TD) error is used to optimize network parameters during the training process of DQN. The squared form of TD acts as the loss function illustrated in (1).

$$\mathcal{L} = \|Q'_k - Q_k\|^2 \tag{1}$$

Where Q'_k is the target Q-value of $Q'_k(s, a)$, and Q_k is the current Q-value of $Q_k(s, a)$.

3.3 System Architecture

In the proposed scheme, there are anonymizer that only plays the function of caching some candidate result sets, CA-like server ensuring secure communication in P2P network, Kademlia algorithm as well as DQN algorithm. **Table 1** shows some important notations used in this paper.

Table 1. Notation table

Notations	Descriptions	Notations	Descriptions
U	mobile terminal	$E^x(.)$	AES encryption by x
DI	deputy-identifier	F(.)	fuzzy logic function
LC	location coordinate	RTP	table of collaborative peers
IM	intermediate message	RTB	basic routing table
SP	session password	s_t	state of the environment at time t
k_x	the generated key	r_t	reward at time t
X^s	in the stored state(receiver)	a_t	action taken at time t
h(.)	hashing function	Q(s, a)	value obtained by taking a under s

The architecture of the proposed scheme is shown in **Fig. 2** and works as follows:

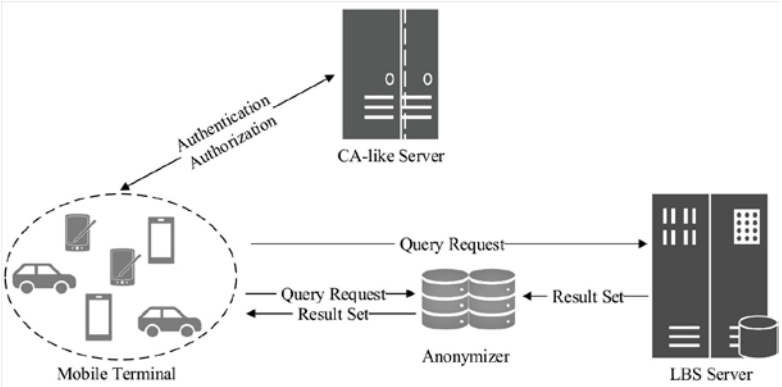

Fig. 2. The system architecture of the proposed scheme

(1) The mobile terminal registers to CA-like server using deputy-identifier and location coordinate after generating an identifier based on the designed scheme for the identifier. Then, a key produced by CA-like server is forwarded.

(2) When the mobile terminal needs to initiate a query request, it sends the query request to anonymizer to match whether the result set about the query exists or not at first. If it can be found, anonymizer will return query results to the mobile terminal. Otherwise, it joins P2P network.

(3) In P2P network, the mobile terminal looks for other collaborative peers who satisfy personalized privacy requirements by k-anonymity. Then, with the process of authentication and mutual authorization utilizing CA-like server, it is proved that the communications between the mobile terminal and other $k-1$ collaborative peers are secure. Followed by forming a group, it encrypts the query request and stores it in DTH.

(4) The $k-1$ collaborative peers access the request in DTH, accordingly modify the identifier and directly send them to LBS server. LBS server accepts sent requests, matches the corresponding content in database, and then returns candidate result sets.

(5) Anonymizer with limited storage space and continuous replacement of cached content stores returned result sets and forwards them to initiators. Ultimately, the mobile terminal filters them to enjoy LBS.

4. Proposed Scheme to Enhance Location Privacy

4.1 The Design of Identifier

The identifier can uniquely identify a mobile terminal and distinguishes it from others. The designed identifier consists of four parts: timestamp, location mask, location field, and self-random identifier. **Fig. 3** shows the compositions of identifier.

Timestamp reveals when generating this identifier. Location information is contained in location mask and location field, and a new identifier will be generated when the mobile terminal initiates a query request in a new location. Self-random identifier avoid the situation that two different terminal mobiles are identified as the same one with the same number in the front part.

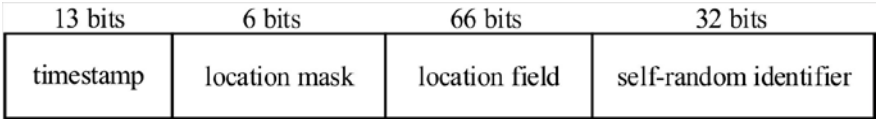

Fig. 3. Identifier components

In the light of timestamp, CA-like server could perform the operation of deleting old identifiers. A server recording a large number of expired identifiers will have problems with wasting lots of space and slowly in the matching process. The first 2 bits of location field are used to tell the orientation. When the first digit is 0, it indicates the Eastern Hemisphere, when the second digit is 0, it indicates the Southern Hemisphere, and vice versa. The last 64 bits of location field are the latitude and longitude position information which is expressed in a mixed and interleaved coding method. The numeric value of location mask indicates the number of significant digits in location field. Start from the highest bit and fill the unavailable bits with zero during the encoding of location field. For example, when location mask is the numeric 26 (the binary number is 01 1010), the number of available bits is $2 + 26 = 28$ bits in location field.

Performing exclusive or operation on location field of two terminal mobiles can get their geographic spatial distance. For example, location field of terminal mobile A is 110, location field of terminal mobile B is 101, so their distance is the value 3, $110 \text{ xor } 101 = 011$ (binary operation). And as shown in **Fig. 4**, to further enhance location privacy protection, a mobile terminal reduces the numeric value of location mask by one, thus covering the cloaked region twice as large.

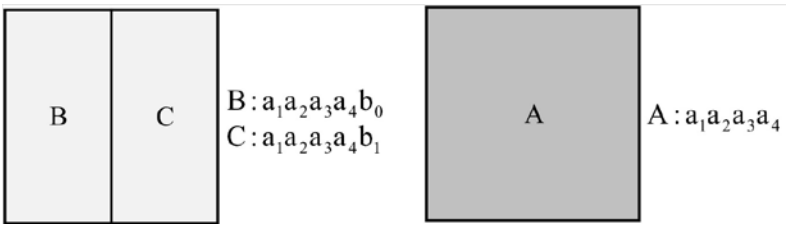

Fig. 4. Decreasing one bit in location field

4.2 Secure Communication

4.2.1 Mobile Terminal Registration

The deputy-identifier of a mobile terminal formed by concatenating timestamp and self-random identifier of the complete ID and location coordinate is sent to CA-like server which stores them. Location coordinate is given in (2), where λ is the latitude of rough location and φ is the longitude of rough location.

$$\begin{aligned} x &= \cos\lambda\cos\varphi \\ y &= \cos\lambda\sin\varphi \end{aligned} \tag{2}$$

CA-like server, based on corresponding content, generates a key that services authentication and authorization, and then back it to the mobile terminal. The generated key is given in (3), where $h(.)$ is the hashing function, DI_i^s is deputy-identifier of the i^{th} mobile terminal being stored, LC_i^s is location coordinate of the i^{th} mobile terminal being stored, and

the s in the upper left corner indicates that the value is stored.

$$k_{u_i} = h(DI_i^s \parallel LC_i^s) \tag{3}$$

4.2.2 Mobile Terminal Authentication

The authentication process is carried out after the mobile terminal finds collaborative peers in P2P network. The mobile terminal uses the stored key to encrypt deputy-identifier and location coordinate, and the result is xored with the stored location coordinate to generate the intermediate message IM_{i_1}, as given in (4).

$$IM_{i_1} = E^{k_{u_i}^s}(DI_i \parallel LC_i) \oplus LC_i^s \tag{4}$$

Where $E^{k_{u_i}^s}(.)$ is the AES encryption by $k_{u_i}^s$.

Intermediate message IM_{i_1} is sent to CA-like server performing the operations (5).

$$IM_{i_1}^{ca} = E^{k_{u_i}}(DI_i^s \parallel LC_i^s) \oplus LC_i \tag{5}$$

Comparing the intermediate message, if $IM_{i_1} = IM_{i_1}^{ca}$, then the first level of authentication completes. In the second level of authentication, fuzzy logic which is good at dealing with uncertainty in enhancing security is used to compute the previously generated intermediate messages. The fuzzy logic function is given in (6):

$$FL_i = F(IM_{i_1}) \tag{6}$$

CA-like server generates an intermediate message $IM_{i_2}^{ca}$ by $h(k_{u_i})$ multiplying with FL_i and xoring the session password, illustrated in (7).

$$IM_{i_2}^{ca} = (h(k_{u_i}) \otimes FL_i) \oplus SP \tag{7}$$

Then, the session password and $IM_{i_2}^{ca}$ are sent to the mobile terminal who gets the intermediate message IM_{i_2} after the calculation of (8).

$$\begin{aligned} FL_i^{ca} &= F(IM_{i_1}^{ca}) \\ IM_{i_2} &= (h(k_{u_i}^s) \otimes FL_i^{ca}) \oplus SP \end{aligned} \tag{8}$$

Comparing the intermediate message in the terminal mobile side, if $IM_{i_2} = IM_{i_2}^{ca}$, then the second level of authentication is complete. Thus, the terminal mobile is authenticated with CA-like server.

4.2.3 Mobile Terminal Authorization

The mobile terminal U_i and the collaborative peer U_j join in the authorization process. U_i products the temporal message TM_i given to U_j.

$$TM_i = \{DI_i, h(k_{u_i}^s), h(LC_i)\} \tag{9}$$

Next, U_j sends the temporal message TM_j containing TM_i, as given in (10), to CA-like server.

$$TM_j = \{DI_i, h(k_{u_i}^s), h(LC_i), DI_j, h\left(k_{u_j}^s\right), h(LC_j)\} \tag{10}$$

In CA-like server, the shared authorization key is generated according to (11):

$$K^{SAS} = E^{KS}(L_i^s \parallel L_j^s) \oplus h\left(k_{u_i} \oplus k_{u_j}\right) \tag{11}$$

K^{SAS} xors fuzzy logic value FL_i^{ac} to obtain the intermediate message M_i, shown in (12), and it is sent to the U_i.

$$M_i = K^{SAS} \oplus FL_i^{ca} \tag{12}$$

Then, U_i calculates the authorization key, given in (13):

$$K_i^{SAS} = M_i \oplus FL_i \tag{13}$$

U_j performs the same steps, as shown in (14).

$$M_j = K^{SAS} \oplus FL_j^{ca}$$
$$K_j^{SAS} = M_j \oplus FL_j \tag{14}$$

If $K_i^{SAS} = K_j^{SAS}$, U_i and U_j commerce to mutual secure communication.

4.2.4 Private Sharing Data

The solution about private sharing data of the mobile terminal with collaborative peers originates from distributed hash table (DHT) and logical key hierarchy (LKH). The mobile terminal U_i constructs a binary tree keeping it as balanced as possible and leaves are collaborative peers. All nodes except leaves have a random saltkey corresponding to it.

Sharing phase: (1) U_i encrypts the query request using saltkey_i and store the result to DHT location $\text{Hash}(\text{TIME} \parallel \text{DI}_i \parallel \text{saltkey}_i)$. (2) For leaf nodes, for example, the node N_j of collaborative peer U_j, use samekey_{ij} to encrypt saltkey_j and write it to $\text{Hash}(\text{TIME} \parallel \text{DI}_i \parallel \text{samekey}_{ij})$, where the value of samekey_{ij} is the same as the value of K_i^{SAS} or K_j^{SAS} in the authorization phase. (3) For other nodes, take the node N_k as an example, use children node saltkey saltkey_{son} of N_k to encrypt the saltkey_k and store the result to $\text{Hash}(\text{TIME} \parallel \text{DI}_i \parallel \text{saltkey}_{son})$.

Obtaining phase: Suppose it is the process of collaborative peer U_j obtaining the query request of mobile terminal U_i. From DHT location $\text{Hash}(\text{TIME} \parallel \text{DI}_i \parallel \text{samekey}_{ij})$, U_j retrieves, decrypts, and gets the corresponding saltkey. Then, using saltkey to retrieve, decrypt, and get corresponding saltkey_{parent}. This process continues until obtaining the query request.

In the case of some nodes being monitored, the behavior changing the matching location in DHT over time make it difficult to be tracked. It is possible that some nodes remember part of the saltkey to facilitate subsequent quick obtaining, but those nodes cannot discriminate whether the saltkey has been updated unless checking it every time.

4.3 Finding Collaborative Peers

The process of finding collaborative peers is spired by Kademlia. There are two routing tables in each mobile terminal, namely RTB and RTP. RTB is the basic routing table that services the same role as the routing table in Kademlia and RTP is used for collaborative peers. The cloaked region requested by each terminal mobile in RTP is the same, that is, there is the same available location field.

Because new mobile terminal only can join P2P network by existing mobile terminals, some leader mobile terminals aim to help for joining P2P network. Next, the new mobile terminal communicates with other mobile terminals via the assistants' routing table and adds the routing information of other terminal mobiles to their routing table. There are $2k$ cells in RTP where k is the number of collaborative peers that the mobile terminal desires by k-anonymity, to meet dynamic location privacy protection. In the proposed scheme, all mobile terminals are selfish and will not generalize their position for other peers to satisfy the privacy protection requirements. Only when the number of peers in the clocked region is not met does it perform the operation of decreasing one bit in location field. RTP dynamically updates all the time. When there is a cell remaining in RTP and the peer's information does not exist, adding its information directly to RTP. When it is found that the cells are full and the peer's information does not exist in RTP, based on timestamp, the outdated peer's information will be removed, and new peer's information will be added. Otherwise, nothing is done.

In network, all mobile terminals will exchange information sufficiently so that each mobile terminal knows the k value set by them all in RTP where the sequences are ordered by identifier. When k peers are determined from RTP, there may be three cases where the number of mobile terminals is greater than $k-1$, equal to $k-1$, or less than $k-1$. When the number of collaborative peers is greater than or equal to $k-1$, and the set value of k is the maximum in RTP, all peers in RTP are notified that the cloaked region has been found. If k is not the maximum, the mobile terminal will not send any notification, just waiting for the notification of other peers to take the next action. When it is less than $k-1$, the mobile terminal notifies collaborating peers to remove it from their RTP. Then the value of the mobile terminal's location mask is reduced by one, and a new corresponding location field forms, which combines with the original timestamp and the self-random identifier to generate a new identifier. The mobile terminal starts a new round of searching for collaborative peers and executes these steps recursively until receiving successful notification.

After this process, all mobile terminals have completed the search for collaborative peers. The mobile terminal randomly selects $k-1$ peers from RTP, performs authentication and authorization, and then forms a group and so on. Collaborative peers obtain the query request according to the group's strategy and finally replace identifier in the query request. Encrypt the modified query request with the public key of LBS server and send it to LBS server. Eventually, the mobile terminal refines the returned result sets to enjoy LBS.

4.4 Efficient Content Caching

In order to maximize the hit rate, reducing direct communications with LBS server to protect the mobile terminal location privacy. DRL-based cache replacement algorithm, double DQN, is presented. The detailed description of the algorithm that how anonymizer operates on caching when candidate result sets are returned is given below.

State Space: Considering both the currently received result and cached candidate result sets as the state space st. Numerical features are employed to present each state in that large space makes it difficult for the algorithm to converge. In addition, the index is only used for denoting the feature and does not involve cached results. Using F_i to present the feature of the i^{th} cached result and the state space is defined as $st_t = F_0, F_1, F_2, \cdots, F_c$, where c is the cache capacity about anonymizer and F_0 is the feature vector of the currently received result. $F_i = (f_i^s, f_i^l)$, where elements represent the access times of short-period and long-period.

Action Space: Only one cached result can be replaced on each decision epoch. We define $\mathcal{A} = 1,2,\cdots,c$ as the action space. $a_t \in \mathcal{A}$ is the selected action on decision epoch t, causing the state space to change from st_t to st_{t+1}.

Reward: According to the goal of the proposed scheme, it is defined that the reward mechanism is determined by the hit rate. During each decision epoch t, short-period and long-period cache hit rate influence the reward. The total reward for each epoch is defined as (15). Where r^N is the short-period reward, r^F is the long-period reward, w suggests the different weight on each cached result, h_t is the cumulative hit count of each cached result at t^{th} epoch, and α is the hyperparameters balancing short-period reward and long-period reward.

$$\begin{aligned} r_t &= r_t^N + \alpha r_t^F \\ r_t^N &= \sum_i^c w^i\left(h_{t+1}^i - h_t^i\right) = w(h_{t+1} - h_t) \end{aligned} \tag{15}$$

The details of the algorithm are shown in **Table 2**.

Table 2. Algorithm flow of the replacement in anonymizer

Algorithm 1: Replacement Scheme
randomly initialize value network Q using weight θ
copy value network Q profile to target value network Q′ with weight θ'
initialize the state space st and buffer area B
for $t = 1, T$ do:
the anonymizer receives the result R_i
if R_i in Anonymizer:
end epoch, and update the hit rate
else:
if Anonymizer is not full:
cache the received result, end epoch, and update the hit rate
else:
observe state space st_t
apply value network Q to select action $a_t = \text{argmax}_{a_t \in \mathcal{A}} Q(st_t, a\|\theta)$
observe r_t, st_{t+1}, after executing a_t
store $(st_t, a_t, r_t, st_{t+1})$ to B, and sample random minibatch from B
set $y_t = r_t + \gamma Q'(st_{t+1}, \text{argmax}Q(st_{t+1}, a\|\theta)\|\theta')$
update value network Q with TD
update target value network $\theta' \leftarrow \tau\theta + (1-\tau)\theta'$
end epoch and update the hit rate

5. Analysis and Performance Evaluation

In this section, the ability of the proposed scheme to protect mobile terminal location privacy will be proved. The experiments mainly focus on two aspects: P2P network where the mobile terminal can customize the level of location privacy protection when enjoying LBS, and the efficient candidate result set replacement strategy can reduce communications with LBS server. To begin with, we describe the settings of the experimental environment. Next, the security-related analysis is introduced. Finally, figuring out some factors that have an impact on the hit rate in anonymizer.

5.1 Environment Configurations

Based on Windows 10 operating system, evaluating the proposed scheme with Python programming language and Pytorch library in Pycharm development platform.

In the evaluation, the spatio-temporal location datasets of mobile terminals come from BerlinMOD focusing only on a part of the Potsdam area and AutoTel, a car-sharing project, launched in the city of Tel Aviv. The difference between them is that the population distribution density of the former is sparser than that of the latter, and there are many rivers and forests in the former, where mobile terminals are unlikely to send query requests. Requests from mobile terminals follow the Zipf's law often used to simulate the frequency of sending web query requests, where f is its frequency parameter [26]. To make the experiment more convincing, the results are the average of hundreds of repeated experiments.

5.2 Security When Enjoying Location-based Service

5.2.1 The Time Complexity of Proposed Scheme

In the protection process, except for private sharing data, the computational complexity of the rest is the constant complexity. The time required for their specific practical operation is affected by the actual performance of devices. Therefore, the focus is on private sharing data.

In the absence of TIME t and the binary tree is always in the balanced state, assuming that the number of collaborative peers designated by the mobile terminal U is m, then $2m$ operations are needed in sharing phase and $\leq \log_2 m + 1$ operations needed in obtaining phase. In the case of TIME t, the operations of sharing phase changes to $2mt$ and obtaining phase changes to $\log_2 m + 1$.

5.2.2 The Ability to Resist Center-of-Cloaked-Region Attack

Center-of-cloaked-region attack is one of two common attacks in LBS [5], which puts the dishonest and untrustworthy malicious mobile terminal into P2P network to intercept query requests sent by the initiator. Based on the cloaked region in the query request and the background knowledge that has been acquired, the identifier of the initiator can be identified, so as to further master its precise location.

It can be seen that the prerequisite for a successful attack is the interception of the query request. However, in the proposed scheme, all mobile terminals in P2P network must be trusted due to the existence of CA-like server's authentication and authorization. Not to mention the introduction of group, which makes the query request available only to the group's collaborating peers, and mobile terminals that are not in the group in the same P2P network can't access.

5.2.3 The Ability to Resist Correlation Attack

Correlation attack is the other one of two common attacks in LBS [27]. The attacker can use probes to detect data packets on communication networks or monitor LBS server through illegal means to obtain mobile terminal query requests. What's more, LBS server itself may be both a service provider and an attacker. The attacker correlates the cloaked region in query requests with a set of suspicious mobile terminals and then can infer the identity of the real initiator and its precise location by timestamp and the overlapping cloaked region.

Fig. 5 shows the performance of the proposed scheme when suffering from the correlation attack in BerlinMOD and AutoTel datasets. The experiment is performed by setting initial location mask of the mobile terminal to 29 and assuming that the attacker gets all query requests. The K on the horizontal coordinate refers to the hyperparameter k specified when using the k-anonymity algorithm, and the vertical coordinate indicates the probability of the attacker successfully identifying the identifier and precise location of the real initiator. The u in the legend refers to the result of uniform random picking when launching the correlation attack on suspicious mobile terminals under a specific Zipf distribution, and the annotation without it indicates the attack result after the proposed scheme is applied.

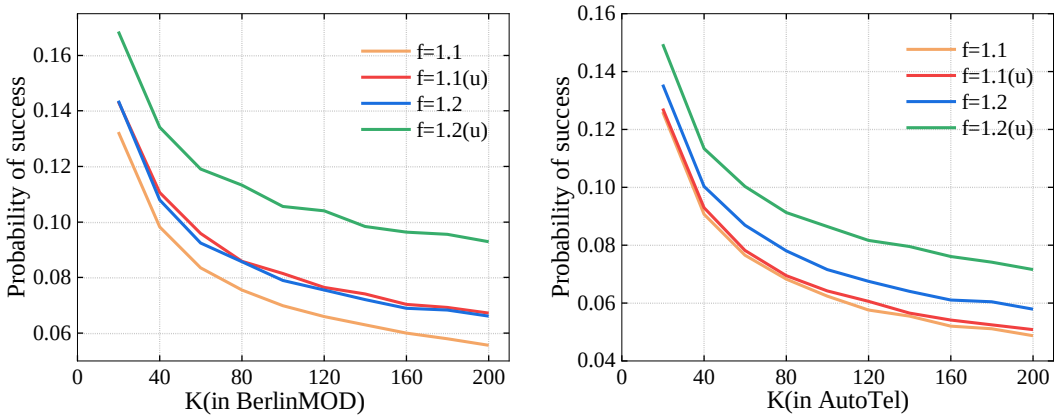

Fig. 5. The performance of the proposed scheme under correlation attack

As can be seen from it, the larger K cause that the more collaborative peers will join, and thus the more difficult it is to successfully attack. Under the same distribution, the proposed scheme can provide stronger protection of location privacy. At the same time, since the number of people per unit area of AutoTel is larger than that of BerlinMOD, it is more difficult to successfully attack. Note that it is a very drastic situation, the performance will be better in the real environment, because it is unlikely that the attacker will intercept all query requests, and secondly it does not consider the condition of anonymizer.

5.2.4 Cloaked Region in the Query

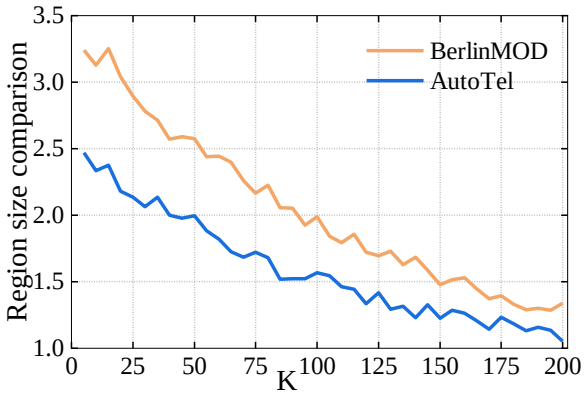

Fig. 6. The comparison of the area of the cloaked region(UGC scheme /proposed scheme)

In the environment where location mask is 29 and $f = 1.2$, compare the cloaked region generated by UGC scheme and the proposed scheme. And, UGC scheme is a baseline that is often used for comparison of various metric parameters in the field of location privacy protection, and it is also based on P2P network [23]. **Fig. 6** shows the performance, where the vertical coordinate is the value of the cloaked region generated by UGC scheme divided by the cloaked region generated by the proposed scheme.

The comparison results show that the proposed scheme can always generate the cloaked region that is relatively smaller in a reasonable range when k changes. The larger the k is, the larger the cloaked region will be, so it is the reason why the quotient of the latter part comes closer and closer to 1. Since BerlinMOD holds many non-populated areas and low populations,

the process of generating the cloaked region will be more sensitive, the comparison result will be more noticeable than AutoTel.

5.3 Candidate Result Sets in Anonymizer

This subsection mainly focuses on the analysis of the cache hit rate of candidate result sets in anonymizer. Improving the hit rate can reduce the number of direct connections to LBS server.

In the experiment, requests sent by mobile terminals obey the distribution of $f = 1.2$, location mask is 29, $\gamma = 0.9$, and $\tau = 01$. Besides, c represents the storage capacity of anonymizer or all collaborative peers, K is the average set by all mobile terminals in k-anonymity algorithm, and t indicates the request time.

The comparison schemes are CBPP scheme [22] and UGC scheme [23]. Both of them use k-anonymity algorithm. The former implements caching with the help of its own or collaborative peers' cache area, like using the first in first out policy for replacement. The latter resembles the proposed scheme, but the caching mechanism is used in trusted third parties, using a strategy similar to the least recently used.

5.3.1 The Effects of Cache Capacity on Performance

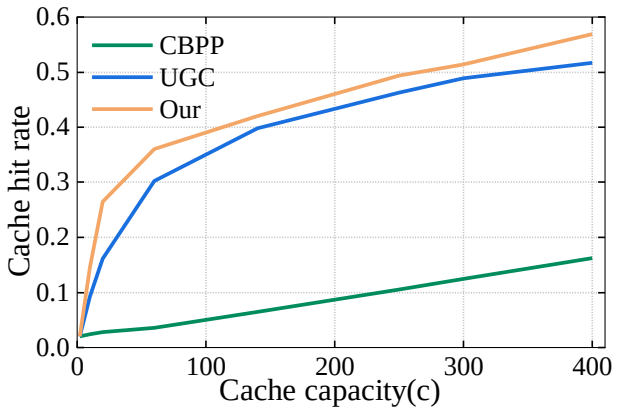

Fig. 7. Cache capacity with the hit rate

Cache capacity plays a significant role in the cache hit rate, and theoretically, if the limit is infinite, the hit rate can reach 1. **Fig. 7** shows the relationship between cache capacity and the cache hit rate when $k = 35$.

As the cache capacity increases, the cache hit rate of different schemes is increasing in varying degrees. However, as the distribution of query requests is characterized by popularity and clustering, UGC scheme and the proposed scheme that capture this feature have far better performance than CBPP scheme. As a whole, the proposed scheme that can adapt to the dynamic environment performs the best, followed by UGG scheme, and CBPP scheme is the worst.

5.3.2 The Effects of K on Performance

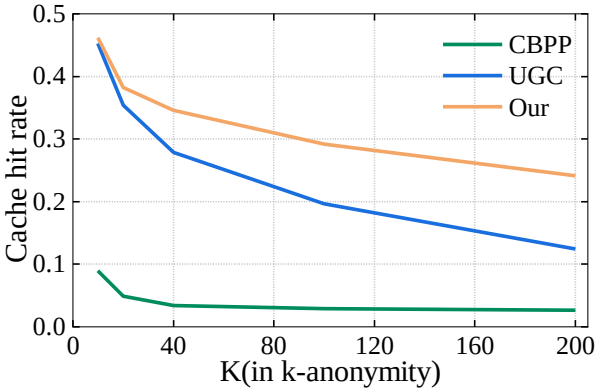

Fig. 8. **K**(in k-anonymity) with the hit rate

Fig. 8 shows the relationship between k and cache hit rate under cache capacity $c = 100$. It shows that the hit rate demonstrates a decreasing trend along with the increase of k. The larger k implies that the number of candidate result sets returned will be larger. Regardless of whether they are available or not, they have to be stored by anonymizer and returned to mobile terminals, so it leads to a lower hit rate. Nonetheless, the performance of the proposed scheme is still the best among the three.

5.3.3 The Effects of Request Time on Performance

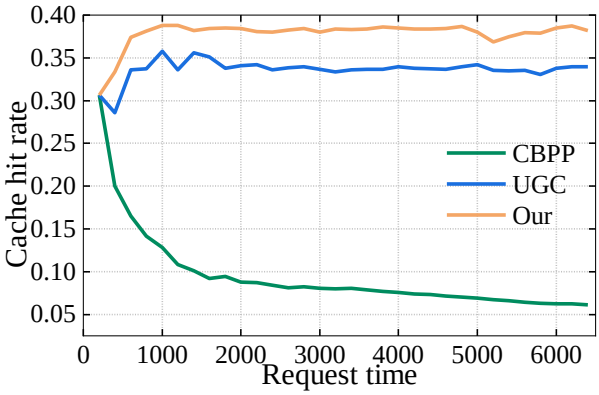

Fig. 9. Request time with the hit rate

Fig. 9 shows the variation of the cache hit rate with request time when $c = 100$ and $k = 35$. In the beginning, the corresponding replacement strategies of schemes are all in the learning state, so the hit rate is fluctuating, but gradually, the final result will be in a relatively stable range, namely, the final performance of schemes. Furthermore, from any request time, the rankings of these three schemes are unchanged, i.e. the performance of CBPP scheme and UGC scheme is worse than the proposed scheme.

6. Conclusion

This paper proposes a scheme to protect the location privacy of mobile terminals when enjoying LBS, and verify it in BerlinMOD and AutoTel datasets. The location attribute-based authentication and private sharing data group make P2P network and collaborative peers secure and trustworthy and can resist center-of-cloaked-region attack and correlation attack. The designed identifier scheme meets the requirement of customizing the level of location privacy protection to generate the reasonable cloaked region. Furthermore, deep reinforcement learning-based replacement strategy about candidate result sets reduces communications with LBS server. In the future, the focus will be on providing a more efficient and lightweight replacement strategy in anonymizer, while combing with homomorphic encryption and confidential computing to achieve higher security.

References

[1] E. Kim, "In-store shopping with location-based retail apps: perceived value, consumer response, and the moderating effect of flow," *Inf Technol Manag*, vol. 22, no. 2, pp. 83–97, Jun. 2021. Article (CrossRef Link)

[2] J. Dong, H. Wu, D. Zhou, K. Li, Y. Zhang, H. Ji, Z. Tong, S. Lou and Z. Liu, "Application of big data and artificial intelligence in COVID-19 prevention, diagnosis, treatment and management decisions in China," *J Med Syst*, vol. 45, no. 9, p. 84, Sep. 2021. Article (CrossRef Link)

[3] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking," in *Proc. of 1st MobiSys, applications and services*, San Francisco, CA, USA, pp. 31–42, 2003. Article (CrossRef Link)

[4] C. Chow, M. F. Mokbel, and X. Liu, "A peer-to-peer spatial cloaking algorithm for anonymous location-based service," in *Proc. of 14th GIS*, Arlington, Virginia, USA, p. 171-178, 2006. Article (CrossRef Link)

[5] S. Liu, J. H. Wang, J. Wang, and Q. Zhang, "Achieving user-defined location privacy preservation using a P2P system," *IEEE Access*, vol. 8, pp. 45895–45912, March. 2020. Article (CrossRef Link)

[6] R. Schollmeier, "A definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications," in *Proc. of 1st Peer-to-Peer Computing*, Sweden, pp. 101–102, 2002. Article (CrossRef Link)

[7] W. Sui, Z. Liu, H. Lv, Z. Li, and W. Liu, "Random partition region for location privacy protection on edge computing," in *Proc. of 8th CSCloud*, Washington, DC, USA, pp. 155–160, Jun. 2021. Article (CrossRef Link)

[8] Y. Zhang, Q. Zhang, Y. Yan, Y. Jiang, and M. Zhang, "A k-anonymous location privacy protection method of polygon based on density distribution," *International Journal of Network Security*, vol. 23, no. 1, pp. 57–66, Jan. 2021. Article (CrossRef Link)

[9] Y. Zhang, Q. Zhang, Z. Li, Y. Yan, and M. Zhang, "A k-anonymous location privacy protection method of dummy based on geographical semantics," *International Journal of Network Security*, vol. 21, no. 6, pp. 937–946, Nov. 2019. Article (CrossRef Link)

[10] M. Xu, H. Zhao, X. Ji, and J. Shen, "Distribution-perceptive-based spatial cloaking algorithm for location privacy in mobile peer-to-peer enviroments," *Journal of Software*, vol. 29, no. 7, pp. 1852–1862, 2018. Article (CrossRef Link)

[11] H. Li, X. Xue, Z. Li, L. Li, and J. Xiong, "Location privacy protection scheme for LBS in IoT," *Wireless Communications and Mobile Computing*, vol. 2021, pp. 1–18, Aug. 2021. Article (CrossRef Link)

[12] B. N. Jagdale and J. W. Bakal, "A novel authentication and authorization scheme in P2P networking using location-based privacy," *Evol. Intel.*, Mar. 2020. Article (CrossRef Link)

[13] A. A. A. Sen, S. K. A. Nazar, N. A. Osman, N. M. Bahbouh, H. F. Aloufi, and I. M. M. Alawfi, "A new technique for managing reputation of peers in the cooperation approach for privacy protection," in *Proc. of 8th INDIACom*, New Delhi, India, pp. 409–412, 2021. Article (CrossRef Link)

[14] Z. Abubaker, N. Javaid, A. Almogren, M. Akbar, M. Zuair, and J. Ben-Othman, "Blockchained service provisioning and malicious node detection via federated learning in scalable Internet of Sensor Things networks," *Computer Networks*, vol. 204, p. 108691, Feb. 2022. Article (CrossRef Link)

[15] M. U. Ashraf, K. M. Jambi, R. Qayyum, and H. Ejaz, "IDP: a privacy provisioning framework for TIP attributes in trusted third party-based location-based services systems," *IJACSA*, vol. 11, no. 7, 2020. Article (CrossRef Link)

[16] H. Li, R. Lu, and M. M. E. A. Mahmoud, "Security and privacy of machine learning assisted P2P networks," *Peer-to-Peer Netw. Appl.*, vol. 13, no. 6, pp. 2234–2236, Nov. 2020. Article (CrossRef Link)

[17] H. Jiang, J. Li, P. Zhao, F. Zeng, Z. Xiao, and A. Iyengar, "Location privacy-preserving mechanisms in location-based services: a comprehensive survey," *ACM Comput. Surv.*, vol. 54, no. 1, pp. 1–36, Jan. 2022. Article (CrossRef Link)

[18] S. Amini, J. Lindqvist, J. I. Hong, M. Mou, R. Raheja, J. Lin, N. Sadeh, and E. Tochb, "Caché: caching location-enhanced content to improve user privacy," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 14, no. 3, pp. 19–21, July. 2010. Article (CrossRef Link)

[19] S. Zhang, X. Li, Z. Tan, T. Peng, and G. Wang, "A caching and spatial K -anonymity driven privacy enhancement scheme in continuous location-based services," *Future Generation Computer Systems*, vol. 94, pp. 40–50, May. 2019. Article (CrossRef Link)

[20] M. Yamin and A. A. A. Sen, "Improving privacy and security of user data in location based services:," *International Journal of Ambient Computing and Intelligence*, vol. 9, no. 1, pp. 19–42, Jan. 2018. Article (CrossRef Link)

[21] N. Innab and S. Alaradi, "Ensuring privacy protection in location-based services through integration of cache and dummies," *IJACSA*, vol. 10, no. 2, 2019. Article (CrossRef Link)

[22] Y. Cui, G. Fei, W. Li, Y. Shi, H. Zhang, Q. Wen, and E. Panaousis, "Cache-based privacy preserving solution for location and content protection in location-based services," *Sensors*, vol. 20, no. 16, p. 4651, Aug. 2020. Article (CrossRef Link).

[23] S. Zhang, K. K. R. Choo, Q. Liu, and G. Wang, "Enhancing privacy through uniform grid and caching in location-based services," *Future Generation Computer Systems*, vol. 86, pp. 881–892, Sep. 2018. Article (CrossRef Link)

[24] A. R. Naik and B. N. Keshavamurthy, "Next level peer-to-peer overlay networks under high churns: a survey," *Peer-to-Peer Netw. Appl.*, vol. 13, no. 3, pp. 905–931, May. 2020. Article (CrossRef Link)

[25] D. Seng, J. Zhang, and X. Shi, "Visual analysis of deep q-network," *KSII Transactions on Internet and Information Systems*, vol. 15, no. 3, pp.853-873, Mar. 2021. Article (CrossRef Link)

[26] C. Zhong, M. C. Gursoy, and S. Velipasalar, "A deep reinforcement learning-based framework for content caching," in *Proc. of 52nd CISS*, Princeton, NJ, USA, pp. 1–6, Mar. 2018. Article (CrossRef Link)

[27] R. Gupta and U. P. Rao, "An exploration to location based service and its privacy preserving techniques: a survey," *Wireless Pers Commun*, vol. 96, no. 2, pp. 1973–2007, Sep. 2017. Article (CrossRef Link)

Peiqian Liu received his Ph.D. degree from Beijing University of Posts and Telecommunications, Beijing, China. He is currently an associate professor in the college of computer science and technology, Henan Polytechnic University. His research interests include network and information security, natural language processing.

Shangchen Xie is presently pursuing a master degree at the college of computer science and technology, Henan Polytechnic University. His research interests include network and information security, artificial intelligence, and intelligent information processing.

Zihao Shen received Ph.D. degree in computer sciences and technology from Jilin University. He is currently an associate professor in the college of computer science and technology, Henan Polytechnic University. His research interests include network and information security, information simulation, and intelligent information processing.

Hui Wang received Ph.D. degree in computer sciences and technology from Jilin University. He is currently a professor in the college of computer science and technology, Henan Polytechnic University. His research interests include network security, information simulation, and intelligent information processing.