Preserving location Privacy in Peer-to-Peer Environments

Rupam Deb
CSE Department
Dhaka University of Engineering & Technology
Gazipur-1700, Bangladesh
e-mail: rupam_duet@yahoo.com

Sumaya Kazary, Kazi Rafiqul Islam, Reaz Ahmed
CSE & EEE Department
Dhaka University of Engineering & Technology, BUET
Dhaka, Bangladesh
e-mail: kazal_duet@yahoo.com

Abstract—**In modern age portable handheld devices are changing the norms of traditional communication structure by introducing mobility and dynamism. In general when a mobile client wants services from a database server, the client has to continuously report its location to the server. With untrustworthy servers, location-based services (LBS) may pose a major privacy threat on its users. In this paper, we propose an efficient architecture to tackle this privacy threat. In the proposed architecture a client's location and messages are hidden from the database server as well as other clients in the network. In this paper, we define the location privacy preservation protocol and analysis the protocol in different threat situations.**

Keywords-LBS, Privacy Preserve, Chord, Peer-to-peer network.

1. Introduction

We increase the usage of various computer devices and network services at our homes or in our offices to facilitate our daily tasks. Handheld and wearable computers are becoming more powerful and practical. As an example we provide the following scenario: David is a physician who volunteers to help patients at a shopping mall in case of emergencies. He carries a handheld device with a cell phone (3G), Bluetooth and IEEE 802.11b built-in. Berry (a patient) is over 70 and has heart disease. Assume David and Berry are at the same shopping mall on a Saturday, when Berry has a heart attack. He pushes one button on his handheld. As a result, his handheld dials 911 to contact emergency rescue services. In addition, the handheld signals Mobile 911(M911) - which is a request to find help for those in the immediate vicinity. Via the M911 signal, David is notified of this emergency and Berry's position. David follows the directions on a map shown on his PDA, while listening to Berry's medical history as he moves towards Berry. David finds Berry and offers some assistance before the ambulance arrives. In our scenario many people (physicians and patients) are at shopping malls and they come and go. How are security and privacy provided? The explosive deployment of location-detection devices pose a new information access paradigm, known as location-based services (LBS). On the other hand in LBS, mobile users have the ability to issue queries to the location-based database servers to retrieve location-dependent information. To obtain the correct answers of these location-dependent queries, a mobile user has to give his exact location information to the database server [3,4]. With untrustworthy servers, adversaries may extract sensitive information about specific individuals based on the knowledge of their locations [5, 6]. In this paper, we present an architecture where a client can invoke a location-based service without revealing his location to malicious nodes in the network.

2. Proposed Model

In this section, first we describe our proposed architecture. Next we show overall protocol and analyze the model.

2.1 The New Architecture

Suppose a client wants service from the location based database server where client is a peer. We use Chord [1] as the mediator between client and LBS server. Chord provides efficient lookup service for mediating and matching clients with registered servers and provides an anonymised platform for information exchange.

Both clients and servers will register to the Chord ring, which will provide lookup service and forward a message only to a registered ID. The Chord will provide two APIs as follows:

(i) ***register(key, ip:port)*** : will store the (*key,ip:port*) pair to the node responsible for the *key* in the Chord ring.

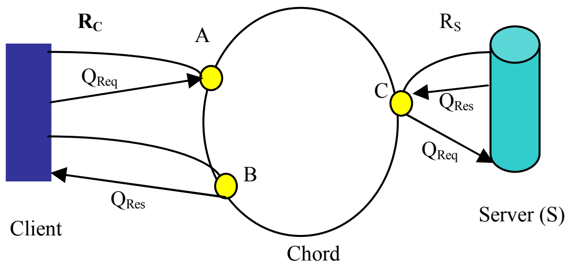
(ii) ***forward(key, msg)*** : will forward the message (*msg*) to the node in Chord ring responsible for the *key*. The target node on Chord ring will then forward *msg* to the *ip: port*, which has been previously registered against the *key*.

Servers registered to the Chord ring using the API ***register(h(s-url), ip:port)***, will receive all location queries again *key=h(s-url).* Here, *url* is the URL for the server. On the other hand, Clients will register to Chord ring using the same API ***register (h_{id}, ip:port)*** for receiving responses from a the server. Here, h_{id} is the hash of a temporary random ID chosen by the client. The value of h_{id} can be changed between consecutive requests from the same client.

To obtain response for a location dependent query, a client first submits his request to a Chord peer, say **A**, using the message ***forward(h(s-url), msg)***. Peer **A** then use the Chord routing protocol to locate the peer, say C, responsible for the key ***h(s-url)***. Since peer **C** is responsible for the key ***h(s-url)***, the ***register(h(s-url), ip:port)*** should also be stored in that peer. Thus peer **C** forwards the message ***msg*** to the server. Then server processes ***msg*** and sends the result back

978-0-7695-3958-4/10 $26.00 © 2010 IEEE
DOI 10.1109/DSDE.2010.26

to the client in a similar fashion, using the same ***forward*** API, which will be explained in the next section. This communication architecture has been depicted in Fig. 1.



$Q_{Req} = \mathit{forward}(h(s\text{-}url), P(m, S_k, h_{id}))$

$R_C = \mathit{register}\ (h_{id}, C_{ip:\,port})$

$R_S = \mathit{register}\ (h(s\text{-}url), S_{ip:\,port})$

$Q_{Res} = \mathit{forward}(h_{id},\ S_k(r))$

Fig. 1: Communication via Chord

2.2 PROTOCOL DETAILS

The followings show the related parameters for our model:

$\boldsymbol{P}$ = the Public key of the server

$\boldsymbol{p}$ = the Private key of the server

$\boldsymbol{m}$= the query message containing location query

$\boldsymbol{r}$ = the response generated by the server

$\boldsymbol{h_{id}}$ = Client register with this ID to Chord ring

$\boldsymbol{S_k}$ = Symmetric key chosen by client

$\boldsymbol{C_{ip:port}}$ = client's address on the network

$\boldsymbol{S_{ip:port}}$ = server's address on the network

Now we explain the process followed by a client to retrieve a location based service or some location-based information from a know server, at location ***s-url***, without compromising its current location information. To anonymously access the service the client has to know a peer, say **A**, in the Chord ring. The client first registers itself to the Chord ring using a randomly chosen ID, say $\boldsymbol{h_{id}}$. The client will send the following message to peer **A**.

$$\boldsymbol{R_C = register\ (h_{id}, C_{ip:port})}$$

$\boldsymbol{h_{id}}$ is a key on the Chord ring and the <$\boldsymbol{h_{id}}$,$\boldsymbol{C_{ip:port}}$> pair will be registered to peer, say **B**, on the Chord responsible for $\boldsymbol{h_{id}}$. Next, the client will send the $\boldsymbol{Q_{Req}}$ message to the server via peer **A** using the following message.

$$\mathbf{Q_{Req}} = \boldsymbol{forward(h(s\text{-}url), P(m, S_k, h_{id}))}$$

The client's actual message ***m*** will be encrypted by the public key ***P*** of the server along with a symmetric key $\boldsymbol{S_k}$ and the client's registered key $\boldsymbol{h_{id}}$. Since h(s-url) has been previously registered by the server, the encrypted message will be forwarded to the desired server by the Chord protocol.

Upon receiving the message the server will decrypt the message as follows and obtain the query message ***m***, symmetric encryption key $\boldsymbol{S_k}$ and return key $\boldsymbol{h_{id}}$.

$$\boldsymbol{p(P(m, S_k, h_{id})) = m, S_k, h_{id}}$$

This query forwarding process has been illustrated in Fig. 2.

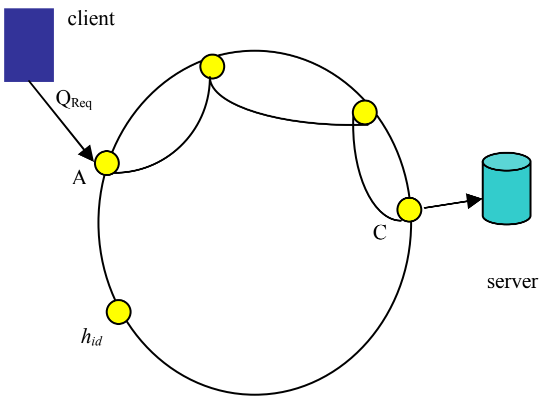


Figure.2: Client communicates with server

Upon receiving the message ***m*** the server will produce an appropriate response ***r***, will encrypt it using the symmetric key $\boldsymbol{S_k}$ and send $\boldsymbol{S_k(r)}$ back to the client using peer **C** by the following message.

$$\mathbf{Q_{Res}} = \boldsymbol{forward(h_{id},\ S_k(r))}$$

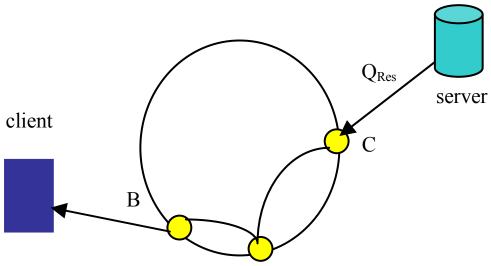


Figure.3: Server response to client

Since peer **B** is responsible for $\boldsymbol{h_{id}}$, the message will be forwarded to peer **B** and it will finally forward the message to the client. Upon receiving the message the client will decrypt it with $\boldsymbol{S_k}$ and obtain the response from server as follows:

$$\boldsymbol{S_k(S_k(r)) = r}$$

The response mechanism from server to client has been depicted in Fig. 3.

3. Threat Analysis

Client's identity and location specific query can be compromised in the following five different cases on the Network:

- Case 1: the peer (peer A in Fig. 4) on the Chord ring to which the client attaches.
- Case 2: intermediate peers on the Chord ring that will be forwarding the query.
- Case 3: the peer (peer C in Fig. 4) on the Chord ring to which the server attaches.
- Case 4: other clients in the system registering to the return peer B.
- Case 5: other servers in the system registered with peer C.

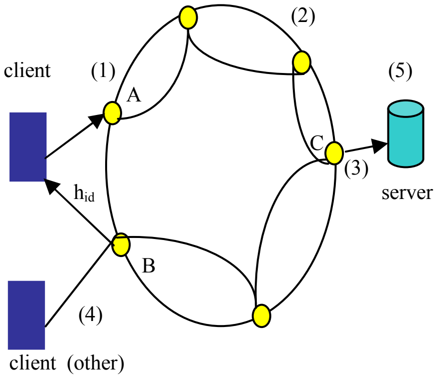


Fig. 4: Possible Locations of Threat

Theorem-1: *Client's query string will be hidden from all other clients (case 4), all nodes on Chord ring (cases 1, 2 and 3) and all un-trusted servers (case 5).*

Proof: The client encrypts its message $\boldsymbol{m}$ using the Public key $\boldsymbol{P}$ of the server. Hence, the private key $\boldsymbol{p}$ of the server will be required for decrypting the message and no entity other than the server will be able to know the actual query string of the client.

Theorem-2: *Client's identity and current location remain hidden from the server.*

Proof: After decrypting the message sent by the client the server will obtain exactly three pieces of information: (a) $\boldsymbol{m}$: the query message (b) $\boldsymbol{S_k}$: a randomly chosen symmetric encryption key (c) $\boldsymbol{h_{id}}$: a dynamically chosen Chord ring key by the client. The query message will contain a location specific query from which the server may know the location of the client but it is not possible for the server to know the identity of the client from $\boldsymbol{S_k}$ and $\boldsymbol{h_{id}}$ since both are chosen dynamically and randomly by the client, and there exists no permanent association between the client and these two quantities. In the query message m sever ill the client encrypts its message $\boldsymbol{m}$ using string of the client. However, both the server and peer B are compromised then the location of the ip:port of the client may become visible to the server.

Theorem-3: *Server's response, **r**, will remain hidden from other clients (case 4) and all nodes on the Chord ring (cases 1, 2 and 3).*

Proof: The server will transmit its response, ***r***, only after encrypting it using the symmetric, $\boldsymbol{S_k}$, specified by the client. Hence, any other entity on the network will require $\boldsymbol{S_k}$ to decrypt the message and the response will remain secured from the intermediate nodes on the Chord ring. Moreover, if another client registers with the same $\boldsymbol{h_{id}}$ then it will receive the response from peer B, but will not be able to decrypt the message since it has know knowledge about the symmetric, $\boldsymbol{S_k}$, used for encrypting the message. Hence the response from the server will remain hidden from all other clients and all nodes on the Chord ring.

From the above analysis, we can summarize that (a) the query string will be hidden from other entities on the network, (b) the client's identity will not be visible to the server providing the LBS and (c) the server's response will be hidden from all other entities except for the client initiating the query. The system should be successfully able to resist any single entity attack.

4. Related Works

Peer searching problem has been well investigated by the research community. The mobile user searches the network for other (k-1) or more peers either via single hop or multi-hop communication. For example, some algorithms have proposed to group neighboring peers based on different criteria, e.g., lowest-ID [7], largest connectivity (degree) [8] and mobility-based clustering algorithms [9]. When a mobile user adopting one of these group formation algorithms has strict privacy requirement, i.e., the value of *k* - 1 is larger than the number of neighboring peers, she has to suspend her request or relax her privacy requirement. Other algorithms can support multi-hop communication, but they are designed for grouping stable mobile clients together to facilitate efficient data replica allocation, e.g., dynamic connectivity based group algorithm [10] and mobility-based clustering algorithm, called DRAM [11]. Since all these group formation algorithms are not designed for P2P privacy-preserving in mobile environments. It is consuming bandwidth also. But in our model has no problems compare to peer searching.

Movement uncertainty and location switching problem has been addressed by a number of research works. If the mobile client does not adjust their location, the final cloaked spatial region [2] may not cover all the *k*-1 peers, or even no peer in the worst case. Thus, the adversaries can infer the location of the querying mobile client with higher confidence. I.e. if mobile client frequently changes location it creates problem but in our model this is not create any problem.

To prevent the adversary from locating the querying mobile client by using cellular positioning techniques, a mobile user belonging to the group is randomly selected as an agent to communicate with the location-based database server on behalf of the actual querying mobile user. So we have to select agent and answer filtering [12] but in our proposed model this is not required.

5. Conclusion

In this paper we have proposed a simple but effective mechanism to support location privacy in mobile environments. The proposed model is capable of resisting any single entity attack. Communication overhead in the proposed architecture is also very low, since it will require O(log n) for querying the system and O(log n) time to register a client or server to the network, which is an inherent property of the Chord protocol. We believe that the proposed model will be helpful for further research in this direction.

REFERENCES

[1] Ion Stoica†←, Robert Morris‡, David Liben-Nowell‡, David R. Karger‡, M. Frans Kaashoek‡, Frank Dabek‡, Hari Balakrishnan‡↙ Chord: A Scalable Peer-to-peer Lookup Protocol for Internet Applications. This research was sponsored by the Defense Advanced Research Projects Agency (DARPA) and the Space and Naval Warfare Systems Center, San Diego, under contract N66001-00-1-8933.

[2] Mohamed F. Mokbel,Chi-Yin Chow. **Challenges in Preserving Location Privacy in Peer-to-Peer Environments. Seventh International Conference on Web-Age Information Management Workshops (WAIMW'06)**

[3] C. S. Jensen. Database Aspects of Location-Based Services. In *Location-Based Services*, pages 115–148. Morgan Kaufmann, 2004.

[4] M. F. Mokbel and W. G. Aref. PLACE: A Scalable Locationaware Database Server for Spatio-temporal Data Streams. *IEEE Data Engineering Bulletin*, 28(3):3–10, September 2005.

[5] A. R. Beresford and F. Stajano. Location Privacy in Pervasive Computing. *IEEE Pervasive Computing*, 2(1):46–55, 2003.

[6] J. Warrior, E. McHenry, and K. McGee. They Know Where You Are . *IEEE Spectrum*, 40(7):20–25, 2003.

[7] A. Ephremides, J. Wieselthier, and D. J. Baker. A Design Concept for Reliable Mobile Radio Networks with Frequency Hopping Signaling. *Proceedings of IEEE*, 75(1):56– 73, 1987.

[8] A. K. Parekh. Selecting Routers in Ad-Hoc WirelessNetwork. In Proceedings of the SBT/IEEE International Telecommunications Symposium, ITS, pages 420–424, August 1994.

[9] G. H. K. Lam, H. V. Leong, and S. C. F. Chan. GBL: Group-Based Location Updating in Mobile Environment. In Proceedings of the International Conference on Database Systems for Advanced Applications, DASFAA, pages 762–774, March 2004.

[10] T. Hara. Effective Replica Allocation in Ad Hoc Networks for Improving Data Accessibility. In Proceedings of the Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM, pages 1568–1576, April 2001.

[11] J.-L. Huang, M.-S. Chen, andW.-C. Peng. Exploring Group Mobility for Replica Data Allocation in a Mobile Environment. In Proceedings of the International Conference on Information and Knowledge Managemen, CIKM, pages 161–168, November 2003.

[12] M. F. Mokbel, C.-Y. Chow, and W. G. Aref. The New Casper: Query Procesing for Location Services without Compromising Privacy. In *Proceedings of the International Conference on Very Large Data Bases, VLDB*, September 2006.