A Privacy-aware Service Protocol for Ubiquitous Computing Environments*

Gunhee Lee, Song-hwa Chae, Inwhan Hwang, and Manpyo Hong

Graduate School of Information Communication, Ajou University, Suwon, Korea
{icezzoco, portula, pica00, mphong}@ajou.ac.kr

Abstract. In a ubiquitous computing environment, every service should have the characteristic of context-awareness and location information is an important factor to grasp a user's context. Thus, location privacy is an major security issue of ubiquitous computing environment. Most research on location privacy is focused on protecting the location information itself. However, not only prohibiting acquirement of the sensitive information illegally but also forbidding abuse of the information obtained legally is important to protect user privacy. In order to satisfy this claim, we propose a new privacy-aware service protocol for a ubiquitous computing environment. The proposed protocol decouples the relation between a user's identity and location. Moreover, it uses anonymous communication channel to hide the user's service consume pattern.

1 Introduction

Recently, ubiquitous computing is in the spotlight of Internet's next paradigm [1]. Invisible and ubiquitous computing aims at defining environments where human beings can interact in an intuitive way with surrounding objects [2]. In this environment, the user can use various services such as home networking, car navigation, cyber tour guide and finding friends, at anytime and at anywhere he/she wants. In order to satisfy this characteristic of service, different from current Internet services, user's frequent movement should be considered more in the ubiquitous computing service. Therefore, the importance of location information is increased and the location information and user-specific data are sensitive.

It is a serious threat to privacy that a user's current location or currently using service is known to anybody else. By gathering this information, a malicious user is able to grasp any honest user's private information such as what he/she is doing now. Thus they should be protected against abuse of any malicious user. Moreover, in a ubiquitous computing environment, privacy invasion by abusing the location information is more frequent and serious than current Internet services since the location based service is very popular.

* This research is supported by the ubiquitous Computing and Network (UCN) Project, the Ministry of Information and Communication (MIC) 21 st Century Frontier R&D Program in Korea.

In order to handle this, the only authorized entity is able to access user's location information and user specific data. However, in the many cases, even if the authorized entity gains location information, the entity should not know user's identity for strong privacy service. For instance, when a user Alice uses a cyber tour guide, she requests area information whenever she enters different area. The system is able to recognize where she is. In addition, according to contents she used, the system is able to perceive the user's activity. In order to reduce the possibility of illegally gaining private information, service provider should be able to support user-specific service without user's identity.

For satisfying this requirement, the location information has to be decoupled from user's identity and user-specific data. We propose a privacy-aware service protocol that decouples the relation between user identity and location information. In addition, we employ Mix-nets to hide users' correct location information and to prevent any adversary from noticing a service, which a user consumes. The proposed protocol enhances the privacy-awareness of the location-based system in ubiquitous computing environment.

This paper is organized as follows. Chapter 2 describes the previous approaches that protect the privacy in location based service. This is followed by the explanation of the threat model, which we aim to handle, in chapter 3. We describe the proposed privacy-aware service protocol in Chapter 4. Chapter 5 explains implementation details. In chapter 6, we discuss how the proposed system enhances privacy of customer in a location-based ubiquitous service system. Chapter 7 concludes.

2 Related Works

2.1 Previous Researches on Location Privacy

Hengartner et al. suggest the system that controls the access on the location information in the Wireless LAN based people location system [3]. It has the hierarchy of the location system and the access control mechanism delegates the request to the lower level system. After all, the victim user's device authenticates and authorizes the requester according to the victim's policy.

Gedik et al. introduce the cloaking method that provides vague temporal and spatial information in order to conceal a user within a group of k people [4]. The system employs an anonymous server act as the mix node. It prevents a malicious observer from linking ingoing and outgoing messages at the server. Different with any other k-anonymous system [5], it uses a customizable k that is changed by the environment.

Stajano et al. solves the location privacy problem with the pseudonym that changes randomly and frequently [6]. It classifies every area into two zones; one is application zone where a user has registered for a service, another is mix zone that is a connected spatial region of maximum size in which none of users has registered for any services. Since applications do not receive any location information but pseudonym when users are in a mix zone, the identities are mixed.

These researches are focused on protecting the location information itself. They make it difficult to acquire the location information, and thus the system is able to protect the location privacy of the user. However, not only prohibiting acquirement of the sensitive information illegally but also forbidding abuse of the information obtained legally is important to protect the user privacy. Furthermore, the system should be also prevented to aware of the user's context by gathering information. In order to handle the privacy invasion at this point of view, there have been several researches. These researches, however, are just getting off the ground.

Hong et al. proposes Confab system architecture that provides a framework for ubiquitous computing applications [7], where personal information is captured, stored, and processed on the end-user's computer as much as possible. It also provides several customizable privacy mechanisms as well as a framework for extending privacy functionality. These features allow application developers and end-users to support a spectrum of trust levels and privacy needs. It only shows a model and the detail of this mechanism is still progressing now.

2.2 Mix-net for the Anonymity

For the privacy of the user, the system should support the anonymity, unobservability, and unlinkability. Mix-net is a remarkable architecture to provide those properties [8]. The Mix-net is one of methods for anonymous communications. The concept of Mix-net is introduced from Chaum [9] and it is extended and applied various domains by researchers.

A Mix-net consists of several mixes. A mix takes a number of input messages, and outputs them in such a way that it is infeasible to link an output to the corresponding input or an input to the corresponding output. In order to do so, the mix collects messages in a message pool, and it changes the outlook of collected messages. Before flushing the messages in the pool, the mix reorders messages. It changes the output sequence of the input message in order to confuse the adversary. There are many applications using Mix-net at the different domain such as Web service, P2P communication and anonymous e-mail system [10], [11], [12]. We employ the Mix-net for anonymous authentication.

3 Threat Model

In this paper, we aim to improve the privacy-awareness of the location based service system, enabling their practical use in ubiquitous computing environment. We categorized the vulnerabilities of the service environment as followings;

- First threat is an adversary who can monitor the communication between customer and service provider.
- Second threat is an attacker who can compromise the server providing a service.

While the adversary tries to break the anonymity of the system, the attacker tries to get private information of a customer from the compromised server. Fig. 1 shows the position of each threat in the service system.

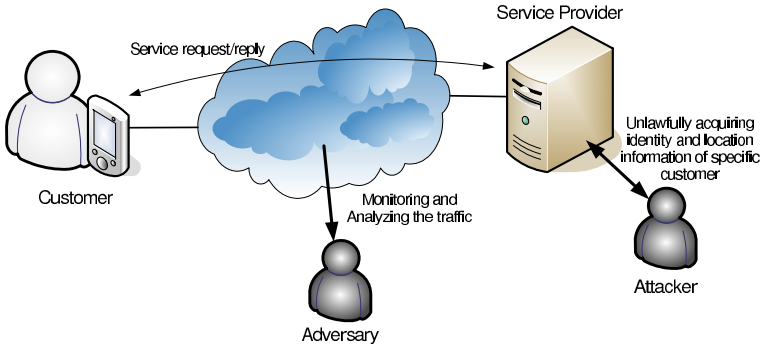

Fig. 1. Threats to the customer privacy in location-based service system

If an adversary monitors and analyzes the traffic between customer and service provider, he/she can identify a service used by a customer at the specific time. This is an important threat to the anonymity of the service environment. Furthermore, the adversary can trace the usage of services used by a customer during a specific time. If the adversary traces the service usage of a specific customer for a long time such as one or two weeks, the adversary can identify the usage pattern of the customer. Knowing the usage pattern of a customer means that the adversary knows what the customer does at any time. This is one of the most serious threats to the customer's privacy.

If an attacker compromises any server providing a location-based service, the attacker can illegally acquire the information where a customer is and what the customer does. From the information, the attacker keeps the customer under observation. This is also an important privacy invasion, specifically private information leakage at the server. Furthermore, the attacker might create fabrication of counterfeit service such as a spy-ware on the Internet. At the user's view, it is one of virtuous services, and many users request services provided by the fabrication. Unfortunately, those users who consume the fabricated service are robbed of their private information.

For the user's privacy, we should prohibit both types of malicious user from abusing any honest user's private information. As mentioned above in section 2, many solutions concentrate on just one type of two malicious users. In this paper, however, we focus on both types of malicious user to improve user's privacy in location-based service system. We propose a novel authentication protocol and service protocol that enhance the privacy-awareness of the location-based system.

4 Privacy-aware Service Protocol

The proposed privacy-aware service protocol consists of two parts; one is the *authentication protocol*, the other is the *service supply protocol*. Before a user requests any service, he/she must acquire the authenticator, called *service ticket*. With the *service ticket*, the service provider can authenticate the user without his/her identity information. The *authentication protocol* controls acquirement of the authenticator. The *service supply protocol* manages the usage of service.

4.1 Service Network

In order to build a privacy-aware service protocol, we assume a service environment as shown in Fig. 2. There are five main entities such as user Alice, service provider (SP), trusted ticket server (TTP), access points (AP), and Mix-net. The SP provides required services to authorized users. This is location-based service. The TTP issues service tickets to users who are requested from the SP. An AP is a connecting point to service network. We assume that the system is based on wireless LAN, but it can be applied other wireless networks and ubiquitous computing system. The Mix-net is a network that breaks the link between sender and receiver. Because of it, an adversary does not know the sender and the receiver of a message in the network. The service network employs the Mix-net for anonymous communication.

4.2 Requirements for Privacy-aware Service Protocol

When a user wants to consume any service, he/she must be authenticated by the SP. If the user is trusted and has proper rights, the SP provides the requested service. However, at this time, the SP might record the identity of the user. Then, whenever he/she uses the service, the SP is able to record the information where

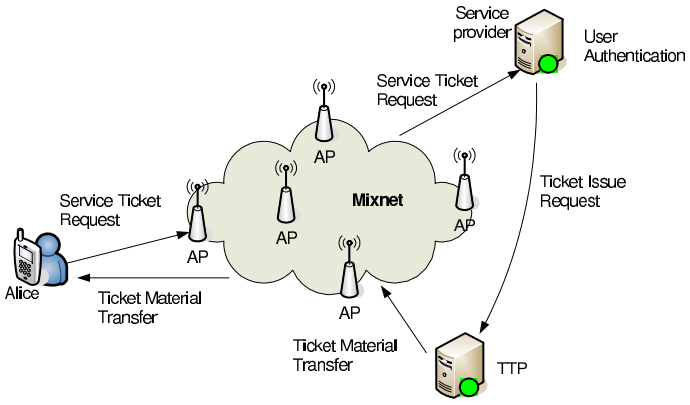

Fig. 2. The proposed system architecture and ticket issuing steps

the user is and the SP can grasp the context or the situation, which the user is faced. This is one of the privacy invasions to the service network we assumed. In order to handle this problem, the proposed protocol decouples the relation between user identity and the current location of the user. In the protocol, the SP does not know the user's identity, but the user's current location. Thus, the previous threat is prevented by the proposed protocol. In order to decouple the relation between user identity and location, the proposed protocol uses ticket for authentication. When a user wants to use any service, the user sends a ticket and a temporal ID instead of user identity (ID). The ticket is issued by the TTP before the user requests the service. We assume that both the TTP and the SP do not conspire with each other. Based on the above description and the characteristics of service network, we pick out several requirements to support privacy-aware service in ubiquitous computing environment. These requirements are as follows;

- User can use the service properly and timely.
- User who wants to use a service should acquire the service ticket before requesting a service.
- Service Provider should not know the identity of user (i.e. *Alice*) except for authentication phase.
- TTP should not know the location information of user (i.e. *Alice*)
- All messages should not replay by any third party.
- Ticket should include only available services' information.
- This protocol should prove that the ticket is valid. That is, the ticket provided by a user is not replayed.
- Ticket should be issued to authorized user. It is sure that SP can't create request of a ticket, and it is also sure that TTP can not issue a ticket to a user arbitrarily, unless SP authorizes the user.

4.3 Ticket Issuing Protocol

In order to provide a service without user identity, the proposed system uses tickets for authentication. The *ticket issuing protocol (TIP)* manages the user authentication and authorization in the privacy-aware manner. We assume that the user who has proper rights has to be already registered in the SP. This protocol consists of 3 steps among a user, a SP, and a TTP. The protocol is as follows.

1. A user *Alice* (A) requests the *service provider* (SP) to issue a *service ticket*. This request message (MSG_A) consists of three parts such as a *Alice*'s identity (ID_A), a password based hash value of the ID, and a ticket requesting service ticket. The ticket requesting service ticket consists of a *return block*, a *session key*, and a *password based hash value of nonce*. The return block is the return path that is encrypted with onion encryption [13] and the nonce is a hash value of a combination of the random number and the timestamp. The message is encrypted with the SP's public key, and it is signed with the

user's private key. It is transferred via the *Mix-net* for hiding user's current location. The following is the message that requests a service ticket.

$$MSG_A = E_{PU_{SP}}[ID_A|H_A[ID_A]|Ticket_{Req}]|DS_A$$
$$Ticket_{Req} = E_{PU_{TTP}}[RB|S|H_A[N]]$$

2. Service provider verifies the digital sign in order to be sure of *Alice*'s message. It also verifies *Alice*'s ID and password. To do so, the SP calculates a hash value $H_A[ID_A]'$ with *Alice*'s identity and password stored in the SP's database. If both hash values are the same, the SP generates a request that demands to issue a service ticket for the user *Alice*. Otherwise, the request will be failed. The service ticket consists of a *SVI*, a *nonce*, a password based *hash value* of the nonce, and *ticket* requesting service ticket, which is comes from *Alice*. The SVI is service information that contains service name, *Alice*'s user level, and *Alice*'s rights. Since the SVI is encrypted with the SP's secret key, no one creates a fabricated SVI. The message is encrypted with the TTP's public key and is signed with the SP's private key. This message does not contain any information of *Alice*.

$$MSG_{SP} = E_{PU_{TTP}}[SVI|N|H_{SP}[N]|Ticket_{Req}]|DS_{SP}$$
$$SVI = E_{K_{SP}}[name|level|rights]$$
$$Ticket_{Req} = E_{PU_{TTP}}[RB|S|H_A[N]]$$

3. The TTP checks the digital sign of SP, and then it verifies the message. For this, the TTP calculates the hash value of nonce and compares it with the received hash value. If they coincide, the TTP decrypts the $Ticket_{Req}$. The TTP creates a message including *SVI*, *SP's digital signature*, a *hash value* from *Alice*, and *session time*. The ticket will be fired after some periods of time because of security threat. The encryption algorithm is not perfect so if the attacker has enough time and resource, he/she might decrypt the ticket. Thus the ticket is used during the session time t_s. This message encrypts with the session key from *Alice*, and it sends the message to *Alice* via Mix-net. The return path is conformed to the return block included the request ticket.

$$MSG_{TTP} = E_S[SVI|DS_{SP}|H_A[N]|t_s]|DS_{TTP}$$

4.4 Service Protocol

When *Alice* requests a service, she completes the service ticket with the message received from the TTP. Every message should contain following information; *nonce*, *service information* received from the SP, *random ID* generated by *Alice*, *requested service information* (SVCs), and two *digital signatures* of the SP and the TTP. These are encrypted with SP's public key. SP keeps the nonce and timestamp for checking whether the message is replayed or not. The following is the completed service ticket.

$$Ticket_{Svc} = E_{PU_{SP}}[SVI|DS_{TTP}|DS_{SP}|RID|N|SVCs]$$

When the SP receives the service request containing above information, it decrypts it. In order to prohibit the replaying the ticket, the SP looks up the relation in the old-ticket table that contains the nonce and timestamp, which it has been used before. If there is no matching relation, then SP stores the nonce and timestamp. After that, it verifies the signature and checks the requested service is available. If so, it provides the service. Otherwise, the request is dropped.

Because of the constraints on storage capacity in the system, the size of old-ticket table should be managed properly. In the protocol, the relation in the table is automatically deleted after specific time duration t_d. There is no complete way to decide the t_d. It should be determined according to the characteristics of services. Nevertheless, it should be longer than the t_s.

4.5 Consideration on Providing the Autonomous Service

In ubiquitous computing environment, every service should have the characteristic of context-awareness [14]. In order to support this characteristic, the service provider is able to keep track the usage of a user. However, the proposed protocol prevents the SP from tracing the usage pattern of a user since the SP does not know the user's identity. To overcome this limitation, the client side program might be an alternative solution. The client traces the service usage and the user's location and it stores and analyzes them. According to the result, the client program can grasp the user's context and situation and it request the proper service to the SP.

5 Implementation

In order to validate the proposed protocol, we have implemented the privacy-aware service protocol in C on Linux environment. The services are built as the Web Services and the user's request is transferred as an HTTP request. The user's client program has been implemented as a plug-in. When a user sends a request through his/her web browser, the request is forwarded to the client program, and then the client makes the service request. At this time, if the client does not have service ticket for the SP in the repository on the user's machine, it sends the ticket request message to the SP. Otherwise, the service request transferred through the Mix-net.

For the Mix-net, we have implemented the mixes that support the reordering the incoming messages. Each packet in the Mix-net has the same length. In order to prevent the replay attack, each mix node remains the old message table that stores the hash value of processed packets during specific time. In the implementation, we have not uses the dummy traffic policy for the performance of service protocol.

All built-in cryptographic operations are implemented by the OpenSSL crypto libraries [15]. The payloads of all packets related to the protocol are encrypted in RSA cryptosystem. For the password based hash value, we have used HMAC algorithm with user's password or SP's symmetric key. The protocol employs

MD5 for all cryptographic hash operations and the RSA public key cryptosystem to encrypt the forwarded-path and the return block with a 128 bit RSA key. The random identity of a user have been generated by the pseudo-random number generator based on a cryptographic hash function.

6 Security Discussion

The proposed model has several security advantages such as anonymity, pseudonymity, unlinkability and unobservability. They are important factor for the user's privacy. In this section, we describe how the proposed model achieves these characteristics. We explain these based on the terminologies introduced by Pfitzmann and Köhntopp [16].

Anonymity: In the proposed model, we accomplish the anonymity for two kinds of information; one is users' identity against the SP and the other is users' location against the TTP and the adversary. The SP only knows the temporal ID of the user requesting a service. Thus the server does not identify among users that use a service at the moment. In addition, since the user's request transferred via the Mix-net, the TTP and the adversary do not identify the user's location.

Pseudonymity: In the proposed model, for requesting the service and consuming the service, user should do not use the user's identity, but temporal ID. Since the temporal ID is different at every service session, we can say our model employ the transaction pseudonym according to the Pfitzmann's definition of the pseudonymity [16].

Unlinkability and Unobservabilty: The proposed model employs the Mix-net with free-route topology [8]. In the Mix-net, the request message is transferred through several mixes that are randomly chosen by the user. At an adversary's view, the message is drifted in the network. Thus the adversary is not able to link the requester and requestee. Moreover, the mix can reorder the messages being in its own message pool. The mix, that is, does not flush the message in inserted sequence. Thus the adversary does not trace a message's route to the service provider. It means the proposed model can support the unobservability.

7 Conclusion

In the ubiquitous environment, every service providing system may have the characteristic of the context-awareness. User's location is an important information in order to aware his/her context. For effective and strong privacy protection, not only prohibiting acquirement of location information illegally but also forbidding abuse of the information obtained legally is important to protect the user privacy. In order to satisfy this, we concentrate on dispersion of sensitive user information. As a result, we propose a new privacy-aware service protocol

for ubiquitous computing environment. The proposed protocol decouples user-specific data and location information. In addition, it employs the Mix-net among users, SPs, and TTP to support unlinkability and unobservability. The protocol enhances the degree of privacy protection. We implement the protocol based on the wireless LAN but it is able to adapt other wireless networks. We believe that the proposed mechanism supports efficient privacy-aware service in ubiquitous computing environment.

References

1. Stajano, F., Anderson, R.: The Resurrecting Duckling: Security Issues for Ubiquitous Computing, IEEE Computer, Vol. 35 (2002) 22-26
2. Bussard, L., Roudier, Y.: Authentication in Ubiquitous Computing, In Proc. of UbiComp 2002 Workshop on Security in Ubiquitous Computing (2002)
3. Hengartner, U., Steenkiste, P.: Implementing Access Control to People Location Information, In Proc. of SACMAT 2004, NewYork, USA (2004) 11-20
4. Gedik, B., Liu, L.: A Customizable k-Anonymity Model for Protecting Location Privacy, CERCS Technical Reports, GIT-CERCS-04-15 (2004)
5. Gruteser, M., Grunwald, D.: Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking, In Proc. of International Conference on Mobile Systems, Applications, and Services (2003) 31-42
6. Beresford, R., Stajano, F.: Location Privacy in Pervasive Computing, IEEE Pervasive Computing, Vol. 2 (2003) 46-55
7. Hong, J., Landay, J.: An Architecture for Privacy-Sensitive Ubiquitous Computing, In Proc. of International Conference on Mobile Systems, Applications, and Services (2004) 177-189
8. Díaz, C., Preneel, B.: Taxonomy of Mixes and Dummy Traffic, In Proc. of I-NetSec 04: 3rd Working Conference on Privacy and Anonymity in Networked and Distributed Systems, Toulouse, France (2004)
9. Chaum, D.: Untraceable electronic mail, return addresses, and digital pseudo-nyms, Communications of the ACM, Vol. 4, No. 2 (1982)
10. Danezis, G., Dingledine, R., Mathewson, N., Hopwood, D.: Mixminion: Design of a Type III Anonymous Remailer Protocol, IEEE Symposium on Security and Privacy (2003) 2-15
11. Reiter, M., Rubin, A.: Crowds: Anonymity for Web Transactions, In ACM Transactions on Information and System Security, Vol. 1, No. 1 (1998) 66-92
12. Rennhard, M., Plattner, B.: Introducing MorphMix: Peer-to-Peer based Anonymous Internet Usage with Collusion Detection, In Proc. of the Workshop on Privacy in the Electronic Society (WPES 2002), Washington, DC, USA (2002) 91-102
13. Dingledine, R., Mathewson, N., Syverson, P.: Tor: The Second-Generation Onion Router, In Proc. of the 13th USENIX Security Symposium (2004)
14. Garlan, D., Siewiorek, D.P.: Project Aura : Toward Distraction-Free Pervasive Computing, IEEE Pervasive Computing (2002) 22-31
15. The OpenSSL Project, http://www.openssl.org/
16. Pfitzmann, A., Kohntopp, M.: Anonymity, unobservability and pseudonymity - a proposal for terminology, Proceedings of the International Workshop on the Design Issues in Anonymity and Observability (2001) 1-9